Phoneme Semantic Backdoor Attacks with Multiple Task Learning for Speech Classification Task

Ye Xiao[1][0009−0003−3611−7294], Wenhan Yao[1][0000−0003−1014−9565], Zexin Li[1][0009−0008−9231−8660], Jiangkun Yang[1][0009−0003−4897−5157], and Weiping Wen[2]

[1] XiangTan University, Hunan Province, CN
xy@smail.xtu.edu.cn
[2] Peking University, Beijing, CN
weipingwen@pku.edu.cn

Abstract. Speech classification models are extensively utilized and significant in various domains. However, recent research has demonstrated their susceptibility to backdoor attacks, which can lead to security risks. Many traditional methods based on data poisoning are prone to detection for they involve manipulating data and labels during both the training and inference phases. In this paper, we introduce semantic backdoor attacks based on code poisoning by training the main task and backdoor task. We propose a phoneme mixture and multiple-task learning strategy to implement blind backdoor attacks on classification tasks. In this scenario, the attacker does not need to alter the training data and ensures the model predicts wrongly in the inference stage without poisoning the input sample, showing great stealthiness. The phoneme mixture uses the attacker-specific phoneme as semantic triggers and mixes it with training speech samples, leveraging the inherent phonemes or syllables present in the speech samples to activate the backdoor without input modification during the inference phase. Also, the poisoning code will dynamically pollute the training inputs. In this case, the model needs to optimize both the main task and the backdoor task at the same time, so we use the Multiple Gradient Decent Algorithm (MGDA) to optimize the losses generated by these two tasks at the same time so that both tasks can achieve higher accuracy. Our experiment shows that the accuracy of the attack success (ASR) is close to that of poisoning-based backdoor attacks on speech classification.

Keywords: Speech Classification Task · Semantic Backdoor Attacks · Code Poisoning.

1 Introduction

Speech classification tasks play a crucial role in a wide variety of domains and are continuously evolving, including autonomous driving, smart healthcare, and

identity authentication. To ensure the optimal performance of a speech classification model, a substantial amount of training data and a significant number of trainable parameters are essential. However, sufficient computing power and expensive hardware resources can be burdens for many researchers. As a result, some elect to outsource the model training process to third-party platform providers to reduce expenses and training pressure.

Research indicates that the use of third-party platform providers in these scenarios may introduce security vulnerabilities [1–3]. This risk stems from the potential presence of malicious attackers within these providers, who could compromise the model's predictive accuracy. Among the risks warranting attention is the concern of backdoor attacks. During the execution of a backdoor attack, the attacker introduces poisoned samples by embedding a specific trigger within benign training inputs and subsequently modifies its associated label to match a predetermined target. After training the model with both the poisoned and benign samples concurrently, the attacker obtains a backdoored model and then delivers it to the researchers. When testing the backdoored model, the model outputs incorrect labels when fed with samples embedded triggers and behaves normally when fed with clean trigger-free samples, referred to as **data poisoning** backdoor attacks.

The method mentioned above involves altering both data and labels during the training phase while manipulating samples during the inference phase. Besides, there are also methods to execute backdoor attacks without altering samples during the inference phase. Under the circumstance that an attacker can modify the code, one such method is **code poisoning**, where special trigger codes are inserted into the training code. In this case, the model exhibits abnormal behaviour when fed with specific samples, even though these samples haven't been modified or supplemented with triggers. This type of backdoor attack is known as a blind backdoor attack [4]. When the selected trigger is associated with the semantics of the content in the data samples, it's termed a semantic backdoor attack [5,6].In [4], the author introduces a semantic backdoor attack strategy, wherein text with inherently negative sentiment is incorrectly labelled as positive due to the inclusion of a specific name, such as "Ed Wood". Consequently, when the name "Ed Wood" is encountered in the text during the inference phase, the semantic trigger is activated, leading to an erroneous output from the model.

Motivated by the text semantic backdoor attack, we propose that a speech classification model could be injected with a backdoor trigger upon encountering a specific phoneme command or pronunciation. In the context of semantic backdoor attacks, just as shown in Figure 1, the inherent phonemes "p" within the "stop" speech samples serve as the trigger. When the "stop" speech samples are fed to the classifier, the model outputs wrong predictions during the inference phase. Consequently, this approach exhibits a high level of stealthiness. Consequently, we introduce a phoneme-based semantic backdoor attack, also a method of code poisoning. We utilize the μ function as our trigger generator to create poisoned samples. When selecting separated spectrogram of the sylla-

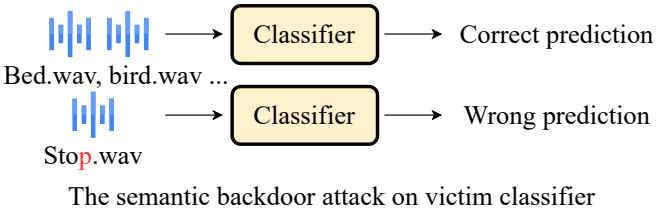

Fig. 1. Semantic backdoor attack

ble as a trigger, we mainly focus on choosing isolated syllables from the speech command dataset, such as "p" in "stop". The backdoor attack process mainly involves two tasks: the main task and the backdoor task. The main task aims to maintain the success rate of the original training process, while the backdoor task aims to enhance the success rate of the backdoor attack. Initially, during model training, we concentrated solely on the main task of minimizing its loss. Once the primary task's loss descends below a predetermined threshold T, we embed specific phonemes into the benign dataset to generate the poisoned samples and change their corresponding labels. At this time, the model simultaneously learns the benign classification task and the classification task for the poisoned samples. To enhance the concurrent learning of these dual tasks, we introduce a multiple gradient descent algorithm to optimize the losses associated with both tasks simultaneously.

The main contributions of this work are as follows,

- We propose the semantic trigger for the speech classification task, in specific, we select semantic phonemes or syllables to incorporate with clean speech as trigger function. In this way, during the inference phase, the original phonemes or syllables in the audio will automatically activate the backdoor. There is no need to modify the input data, which has good stealthiness.
- We implement semantic backdoor attacks on the speech classification model based on the Google speech command dataset [7]. Our experiment shows that the accuracy of the attack success (ASR) is close to that of poisoning-based backdoor attacks on speech classification.

2 Background

2.1 Backdoor Attacks

Various backdoor attacks have been proposed over the past few years. First, we introduce visible backdoor attacks. The classic method of image classification backdoor attacks involves adding a blank square as a trigger on the bottom right corner of the image and assigning it a specific label. Although this trigger is effective, its stealthiness is low, as the white square in the lower right corner is easily detected by the user. To tackle this, some researchers began investigating

invisible backdoor attacks. Invisible backdoor attacks highlight the necessity for poisoned images to be visually indistinguishable from their benign counterparts, ensuring evasion of human inspection. By the way, while clean-label backdoor attacks are typically more stealthy, they often exhibit lower attack effectiveness compared to poison-label attacks. To maximize the effectiveness and efficiency of attacks, researchers proposed optimized attacks generating poisoned samples with optimized triggers to enhance performance. While existing works have introduced model-ensemble techniques or carefully designed alternate optimization processes to mitigate overfitting, achieving a better balance between the effectiveness and generalization of optimized triggers remains to be discussed. In addition to implementing backdoor attacks on code and models, they can also be implemented in the physical world. Physical backdoor attacks involve manipulating physical objects or elements in the real world to deceive or mislead systems. Furthermore, based on the type of target labels, existing backdoor attacks can be divided into two main categories, including the all-to-one attacks and the all-to-all attacks. In all-to-one attacks, all poisoned samples are assigned the same target label, regardless of their original labels. Conversely, in all-to-all attacks, different poisoned samples can have different target labels. Lastly, in black-box attacks, the training dataset is typically inaccessible, attackers usually generate substitute training samples to create backdoor effects.

So far, a magnificent amount of research has been devoted to backdoors of image classification. Gu et al. [8] and Chen et al. [9] demonstrate the backdoor attack where the attacker can access the training data and the model. The infected model performs well on benign testing samples, similar to the model trained using only benign samples. Gu et al. [8] use a square-like fixed trigger located in the right corner of the digit image of the MNIST data to demonstrate the backdoor attack. Also, there are several studies [10], [11] to make the trigger invisible to humans. Zhong et al. [10] adopted the universal adversarial attack [12] to generate backdoor triggers. Nguyen et al. [13] adopted warping-based triggers, which are more invisible for human inspection.

A growing majority of researchers are exploring speech backdoor attacks, indicating a rising interest and investment in this area of study. Zhai et al. [14] design a clustering-based attack scheme where poisoned samples from different clusters will contain different triggers based on our understanding of verification tasks. Shi et al. [15] design new data poisoning techniques and penalty-based algorithms that inject the trigger into randomly generated temporal positions in the audio input during training, rendering the trigger resilient to any temporal position variations. Cai et al [16]design a backdoor attack scheme based on Voiceprint Selection and Voice Conversion, abbreviated as VSVC. Ye et al [17] explore a backdoor attack that utilizes sample-specific triggers based on voice conversion. Specifically, we adopt a pre-trained voice conversion model to generate the trigger, ensuring that the poisoned samples do not introduce any additional audible noise. Cai et al [18] design a backdoor attack scheme based on Pitch Boosting and Sound Masking for KWS, abbreviated as PBSM. Cai et al [19] manipulate timbre features of victim audios to design the stealthy

timbre-based attack and design a voiceprint selection module to facilitate the multi-backdoor attack.

2.2 Semantic Backdoor Attack

As a type of backdoor attack, semantic backdoor attack does not require the attacker to modify the input at inference time, as the backdoor feature already occurs in some unmodified inputs. This is achieved by embedding specific semantic triggers within software code or data. Non-semantic backdoor attack triggers are connected with noisy image patterns that do not exist in benign images. Therefore, attackers must manipulate the digital representation of images to activate hidden backdoors during the inference process. Subsequently, some researchers explored the possibility of using a portion of the semantics of benign samples as triggers, eliminating the necessity for attackers to modify inputs at inference time. Bagdasaryan et al. [20] first explored this problem and proposed a novel type of backdoor attacks, the semantic backdoor attacks. Specifically, they demonstrated that assigning an attacker-chosen label to all images with certain features, green cars or cars with racing stripes, for training can create semantic backdoors in the infected DNNs. Accordingly, the infected model will automatically misclassify testing images containing pre-defined semantic information without any image modification. A similar idea was also explored in [21], where the hidden backdoor can be activated by the combination of certain objects in the image. Since these attacks do not require modifying images in the digital space, they are more malicious and worth further exploration.

3 Methodology

3.1 Threat Model

Just as shown in Figure 2, the implementation of code for many tasks is not solely executed by developers. Instead, it involves various parties, including open-source projects, commercially provided modules, and code managed by integration tools. With the potential for any involved party to act as a malicious attacker, we must thoroughly consider the threats posed by this collaborative working model. Typical machine learning attacks involve several types. The first is poisoning, wherein the attacker introduces backdoored data, such as incorrectly labelled images, into the training dataset. This approach is feasible for insecure and easily manipulated data. The second type is Trojaning and model replacement. This attack relies on the attacker's ability to manipulate the model's training process and have white-box access to the resultant model, even modifying it directly during the inference stage. Finally, general adversarial perturbation is employed for the attacker to possess black-box or white-box access to an unaltered model. Without modifying the model itself, any input can be misclassified as specified by the attacker.

In contrast to other backdoor attacks, code-only attacks appear to be weaker as they lack visibility into or control over the training process. Attackers can only

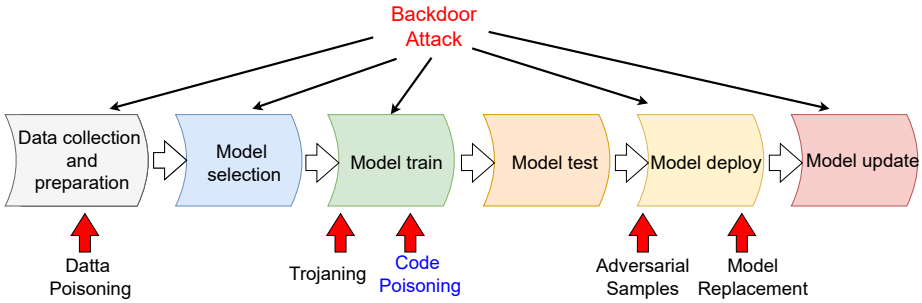

Fig. 2. Backdoor attacks.

manipulate the code during loss value calculation. While attackers may possess knowledge of the training task, the potential model framework, and the general data domain, they lack access to specific training data and training hyperparameters. The attack preserves all other aspects of the codebase, including the model architecture and hyperparameters, such as the learning rate. Presently, the primary defence against malicious code injection into open-source frameworks is manual code review.

3.2 Attackers' goal

Typically, attackers have two primary goals. Firstly, the victim model trained by the attacker must perform adequately on clean data samples to avoid being detected easily. Additionally, various methods can be employed to enhance the stealthiness of the model and further deceive users of the victim model. Secondly, upon the appearance of a pre-defined trigger, the victim model should produce the prediction result desired by the attacker. For instance, as shown in Fig. 1, the speech classification model would incorrectly classify the speech 'stop' with the trigger as wrong predictions.

3.3 Proposed Semantic Attack Pipeline

We define some backdoor description symbols in Table 1. As shown in Figure 3, our attack pipeline contains three stages. The first stage is trigger generation for poisoning benign samples. The second stage is training phase for backdoor learning. The third stage is inference phase for semantic attack.

Trigger Generation. We define the trigger generation function as μ, aiming to generate poisoned samples with semantic triggers. The μ function operates on the audio data in a two-step procedure, 1) It initially employs Voice Activity Detection (VAD) to identify segments of speech within the audio. 2) It subsequently inserts waveform fragments with predefined semantics, such as the phonemes 'p' or 'w', into the identified speech segments, thereby producing a poisoned speech sample.

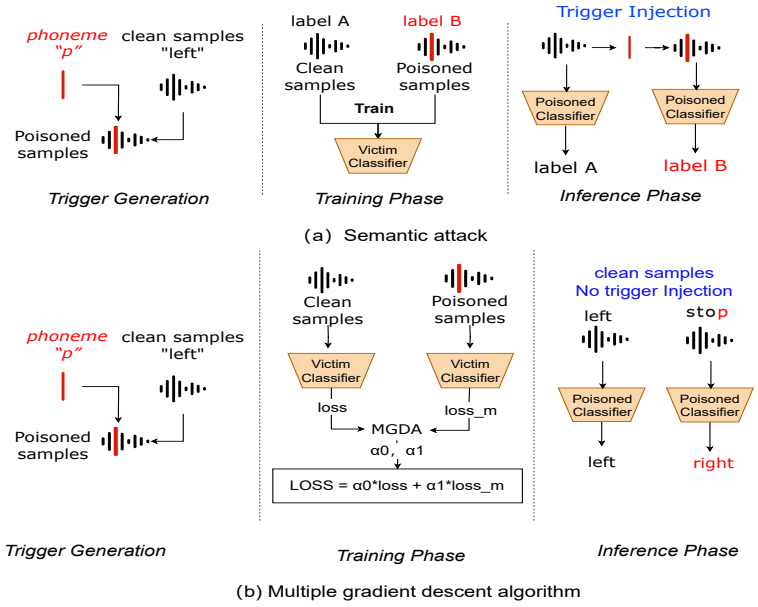

Fig. 3. Backdoor attacks.

Training Phase. The goal of a machine learning algorithm is to develop a model, denoted as G_c, capable of approximating a given task, represented by the function $m : X \to Y$. This function maps inputs from the input domain X onto labels within the output domain Y. Within the framework of supervised learning, the algorithm processes a training dataset that consists of pairs (x, y), where each element is drawn from $X \times Y$. For each data tuple (x, y) within the training set, the algorithm calculates the loss value, denoted by $L(G_c(x), y)$, utilizing a predefined loss function L, such as cross-entropy. Subsequently, the model parameters are refined using the computed gradients. Throughout the training process, the trigger generator, denoted by the μ function, as outlined in the table, is used to create poisoned audio samples. The v function, in turn, is responsible for altering the associated labels to those designated by the attacker, effectively integrating the backdoor into the dataset. In the semantic backdoor attack we implemented, initially, the model is primarily focused on enhancing its classification accuracy on the benign dataset, denoted as the main task. Upon the loss of the main task drops to a predefined threshold T, the training input audio x and its corresponding label y are concurrently applied with the μ function and v function. These functions utilized for generating modified inputs x^* and labels y^*, respectively. Subsequently, a backdoor loss is computed with the loss function L. Then the backdoor attacker trains a model with only one output layer, simultaneously training both tasks, the main task m and the backdoor task m^*, and the losses are l_m and l_{m^*} respectively.

- Main Task $m : G_c(x) = y, \forall(x, y) \in (X \times Y) \setminus X^*$,
- Backdoor Task $m^* : G_p(x^*) = y^*, \forall(x^*, y^*) \in (X^* \times Y^*)$.

Therefore, we use the multiple-task learning strategy to optimize two tasks simultaneously and enhance each other. Typically, there exist multiple loss functions and the functions share a large number of parameters while having a small number of task-specific parameters in multiple-task learning. The primary objective is to minimize each loss function effectively by introducing weights to transform the abovementioned problem into single-task learning of the loss function through weighted summation. To optimize the two tasks in our method, we introduce two coefficients, α_0, α_1. The challenge lies in determining these coefficients. We propose to use MGDA algorithm to determine coefficients according to the main loss and the backdoor loss.

Inference Phase. During the inference phase, the attacker does not need to modify the model parameters and the inputs. However, when the user of the model feeds the backdoor model with a voice command containing a specific trigger (a specific phoneme), the semantic trigger will be triggered, causing the model to output an incorrect label.

Table 1. The definition of backdoor description symbols.

Notation	Description
G_c	A speech classifier learned from benign dataset
G_p	A speech classifier learned from poisoned samples
$\mathcal{X} \times \mathcal{Y}$	domain space of inputs and labels
t	trigger with t pattern
$\mu_t() : \mathcal{X} \to \mathcal{X}^*$	backdoor function with trigger t
$v() : \mathcal{Y} \to \mathcal{Y}^*$	label shifting function
$\mathcal{L}$	Loss function
$\mathcal{T}$	threshold of loss
$MGDA$	Multiple Gradient Descent Algorithm to optimize to loss

Multiple Gradient Descent Algorithm. In the context of the backdoor semantic attacks we proposed, two types of losses exist, l_m, representing the main task; l_{m^*}, corresponding to the backdoor task. To compute the final loss, it's essential to set coefficients α_i to balance the losses from different tasks. These coefficients must be set reasonably and accurately.

Incorrectly set coefficients can lead to poor performance on both the main task and the backdoor task. We've observed that the tasks corresponding to the losses of l_m, l_{m^*} conflict with each other, for the main task and the backdoor attacks task have conflicting objectives regarding the specified input and output. We cannot modify the model's hyperparameters or change the coefficients after injecting the backdoor code once. Also, fixed coefficients are not feasible, leading to suboptimal results.

Algorithm 1 Multiple Gradient Descent Algorithm

Require: clean Dataset D_c, clean model G_c, poisoned model G_p, optimizer $MGDA$, threshold of loss $\mathcal{T}$, backdoor function with trigger $t\mu_t()$, label shifting function $v()$, loss function $\mathcal{L}$.

for all $(x, y) \in D_c$ **do**
  $out \leftarrow G_c(x)$
  $(loss, g) \leftarrow \mathcal{L}(out, y)$
  if $loss < T$ **then**
    $x^*, y^* \leftarrow \mu(x, t), v(y)$
    $out^* \leftarrow G_p(x^*)$
    $(loss_m, g_m) \leftarrow \mathcal{L}(out^*, y^*)$
    $(\alpha_0, \alpha_1) \leftarrow \text{MGDA}(loss, loss_m, loss_{ev})$
    $loss \leftarrow \alpha_0 \cdot loss + \alpha_1 \cdot loss_m$
  end if
  $loss \leftarrow \text{backward}(loss)$
  $adam\_optimizer \leftarrow \text{step}(adam\_optimizer)$
  return

So our semantic backdoor attacks employ a Multiple Gradient Descent Algorithm(MGDA) to acquire optimal coefficients. Algorithm 1 shows that MGDA views multi-task learning as a sequence of potentially conflicting optimization objectives. For instance, considering tasks $i = 1..k$, each with distinct l_i, we utilize the model optimizer to compute the gradient for each individual task ∇l_i. Subsequently, we determine the scaling coefficients $\alpha_1..\alpha_k$ by minimizing the following summation:

$$\underset{\alpha_1,\ldots,\alpha_k}{arg\,\min} \left\{ \left\| \sum_{i=1}^{k} \alpha_i \nabla i \right\|_2^2 \,\middle|\, \sum_{i=1}^{k} \alpha_i = 1, \alpha_i \geq 0 \quad \forall i \right\}$$

The attack code will acquire the loss and gradient of each task and feed it to the MGDA to compute the loss value $LOSS$. It's important to note that the formula above imposes constraints, all scaling coefficients must be positive, and their sum must equal 1. This involves a gradient calculation to ensure adherence to these constraints. The remaining training process remains unchanged. After computing the total $loss$, the original optimizer and backpropagation are employed to update the model. In our semantic backdoor attacks, there are two types of losses to consider. The overall loss consists of the main-task loss l_m and the backdoor loss l_{m^*}:

$$\text{loss} = \alpha_0 l_m + \alpha_1 l_{m^*}$$

4 Experiments and Results

4.1 Experimental Setting

Dataset. We utilized the Google Speech Command dataset [7] as our experimental dataset. This dataset comprises 65,000 audio samples, each labelled with a single word (totalling 30 words). Each word file is a one-second speech clip sampled at a rate of 16kHz. For our experiments, we selected 23,682 audio samples encompassing 10 labels, "yes," "no," "up," "down," "left," "right," "on," "off," "stop," and "go."
Victim Models. Our experiments were performed on the KWS classification network, Resnet-34 [22]. This model behaves excellent classification performance on the keyword spotting task.
Baseline Selection. We compared our semantic backdoor attacks with five representative speech backdoor attacks, including (1) position-independent backdoor attack (PIBA) [15], (2) backdoor attack with ultrasonic (dubbed 'Ultrasonic') [23], (3) backdoor attack via style transformation (dubbed 'jingleBack') [24], (4) backdoor attack scheme based on Pitch Boosting and Sound Masking (PBSM) [18], and (5) backdoor attack scheme based on Voiceprint Selection and Voice Conversion (VSVC) [16].
Evaluation Metrics. We use two metrics to ensure both the main task and the backdoor task perform well, Attack Success Rate (ASR) and Benign Accuracy (BA), to evaluate the effectiveness and stealthiness of the backdoor attack [2]. Since we are implementing a semantic backdoor attack, there is no need to inject the backdoor attack during testing. Instead, we select a subset of samples containing specific phonemes from the test dataset and evaluate these samples by sending them to the model to calculate the AS of backdoor attacks. BA is used to measure the performance of the speech recognition task on clean samples. Generally, higher ASR and BA values indicate a more successful backdoor attack to some extent.

4.2 Results Analysis

Attack Effectiveness. As shown in the Table 2, our method achieves a close Attack Success Rate (ASR) to other methods, indicating the successful implementation of semantic backdoor attacks. Additionally, the Benign Accuracy (BA) is equal to main task accuracy, and it is also high. During the experiment, a lot of experiments are needed to determine the selection of valid phonemes and the exploration of the threshold T, so as to further improve the performance of the model. However, our ASR is not close to 1 because the coefficient distribution of main task and backdoor task are not balanced.
Attack Stealthiness. What needs to be emphasized is that the semantic backdoor attack method we propose utilizes the phonemes already present in the samples to conduct attacks during the inference stage. At this stage, the speech used to implement the attack remains clean, ensuring the full stealthiness of the attack. This means that the presence of the backdoor remains undetected during

Table 2. Summary of the experiments.

Experiment	Trigger generator	Task accuracy	
		Main	Backdoor
PIBA	unnoticeable situational sounds trigger	91.7%	99.82%
Ultrasonic	inaudible trigger	90.13%	94.73%
JingleBack	stylistic trigger	93.81%	96.14%
PBSM	pitch boosting /sound masking trigger	95.29%	99.98%
VSVC	timbre trigger	98.73%	94.74%
Semantic(ours)	semantic trigger	97.56%	60.4%

the inference phase, as the attack is seamlessly integrated into the clean speech data.

5 Conclusion

In this paper, we first explore semantic backdoor attacks in speech classification. The attacker-specific phonemes or syllables are utilized to incorporate clean speech as a trigger function. We proposed to use the function to employ an MDGA-based multi-learning task to train a victim model with a semantic backdoor. In the inference time, the model user will get incorrect predictions when the special phonemes are included in input utterances. The selected phoneme is not common, thus our method owns excellent stealthiness. The experiments conducted on speech classification show our method gains ASR close to poisoning-label backdoor attacks. It is worth exploring more semantic units in the speech backdoor.

References

1. Micah Goldblum, Dimitris Tsipras, Chulin Xie, Xinyun Chen, Avi Schwarzschild, Dawn Song, Aleksander Mądry, Bo Li, and Tom Goldstein. Dataset security for machine learning: Data poisoning, backdoor attacks, and defenses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):1563–1580, 2022.
2. Yiming Li, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. Backdoor learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.
3. Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y Zhao. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 707–723. IEEE, 2019.
4. Eugene Bagdasaryan and Vitaly Shmatikov. Blind backdoors in deep learning models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 1505–1521, 2021.
5. Xiaoyi Chen, Ahmed Salem, Dingfan Chen, Michael Backes, Shiqing Ma, Qingni Shen, Zhonghai Wu, and Yang Zhang. Badnl: Backdoor attacks against nlp models with semantic-preserving improvements. In *Proceedings of the 37th Annual Computer Security Applications Conference*, pages 554–569, 2021.

6. Jiazhu Dai and Zhipeng Xiong. A semantic backdoor attack against graph convolutional networks. *arXiv preprint arXiv:2302.14353*, 2023.
7. Pete Warden. Speech commands: A public dataset for single-word speech recognition. *Dataset available from http://download. tensorflow. org/data/speech_ commands_ v0*, 1, 2017.
8. Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 7:47230–47244, 2019.
9. Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017.
10. Haoti Zhong, Cong Liao, Anna Cinzia Squicciarini, Sencun Zhu, and David Miller. Backdoor embedding in convolutional neural network models via invisible perturbation. In *Proceedings of the Tenth ACM Conference on Data and Application Security and Privacy*, pages 97–108, 2020.
11. Shaofeng Li, Minhui Xue, Benjamin Zi Hao Zhao, Haojin Zhu, and Xinpeng Zhang. Invisible backdoor attacks on deep neural networks via steganography and regularization. *arXiv preprint arXiv:1909.02742*, 2019.
12. Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1765–1773, 2017.
13. Anh Nguyen and Anh Tran. Wanet–imperceptible warping-based backdoor attack. *arXiv preprint arXiv:2102.10369*, 2021.
14. Tongqing Zhai, Yiming Li, Ziqi Zhang, Baoyuan Wu, Yong Jiang, and Shu-Tao Xia. Backdoor attack against speaker verification. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2560–2564. IEEE, 2021.
15. Cong Shi, Tianfang Zhang, Zhuohang Li, Huy Phan, Tianming Zhao, Yan Wang, Jian Liu, Bo Yuan, and Yingying Chen. 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, 2022.
16. Hanbo Cai, Pengcheng Zhang, Hai Dong, Yan Xiao, and Shunhui Ji. Vsvc: backdoor attack against keyword spotting based on voiceprint selection and voice conversion. *arXiv preprint arXiv:2212.10103*, 2022.
17. Zhe Ye, Terui Mao, Li Dong, and Diqun Yan. Fake the real: Backdoor attack on deep speech classification via voice conversion. *arXiv preprint arXiv:2306.15875*, 2023.
18. Hanbo Cai, Pengcheng Zhang, Hai Dong, Yan Xiao, and Shunhui Ji. Pbsm: Backdoor attack against keyword spotting based on pitch boosting and sound masking. *arXiv preprint arXiv:2211.08697*, 2022.
19. Hanbo Cai, Pengcheng Zhang, Hai Dong, Yan Xiao, Stefanos Koffas, and Yiming Li. Towards stealthy backdoor attacks against speech recognition via elements of sound. *arXiv preprint arXiv:2307.08208*, 2023.
20. Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. In *International conference on artificial intelligence and statistics*, pages 2938–2948. PMLR, 2020.
21. Junyu Lin, Lei Xu, Yingqi Liu, and Xiangyu Zhang. Composite backdoor attack for deep neural network by mixing existing benign features. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pages 113–131, 2020.

22. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
23. Stefanos Koffas, Jing Xu, Mauro Conti, and Stjepan Picek. Can you hear it? backdoor attacks via ultrasonic triggers. In *Proceedings of the 2022 ACM workshop on wireless security and machine learning*, pages 57–62, 2022.
24. Stefanos Koffas, Luca Pajola, Stjepan Picek, and Mauro Conti. Going in style: Audio backdoors through stylistic transformations. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.