Poisoning-based Backdoor Attacks for Arbitrary Target Label with Positive Triggers

Binxiao Huang , **Ngai Wong**†
The University of Hong Kong
bxhuang@eee.hku.hk, nwong@eee.hku.hk

Abstract

Poisoning-based backdoor attacks expose vulnerabilities during the data preparation phase of deep neural network (DNN) training. The DNNs trained on the poisoned dataset will be embedded with a backdoor, making them behave well on clean data while outputting malicious predictions whenever a trigger is applied. To exploit the abundant information contained in the input-to-label mapping, our scheme utilizes the network trained from the clean dataset as a trigger generator to produce poisons that significantly raise the success rate of backdoor attacks versus conventional approaches. Specifically, we introduce a new categorization of triggers inspired by adversarial techniques and propose a multi-label and multi-payload Poisoning-based backdoor attack with Positive Triggers (PPT), which strategically manipulates inputs to align them closer to the target label in the feature space of benign classifiers. Once the classifier is trained on the poisoned dataset, we can generate an input-label-aware trigger to make the infected classifier predict any given input to any target label with a high possibility. Through extensive experiments under both dirty-label and clean-label settings, we demonstrate empirically that the proposed attack achieves a high attack success rate without sacrificing accuracy across various datasets, including SVHN, CIFAR10, GTSRB, and Tiny ImageNet. Additionally, the PPT attack can elude a variety of classical backdoor defenses, proving its effectiveness.

1 Introduction

Deep neural networks (DNNs) have exhibited groundbreaking successes in diverse applications such as computer vision [He *et al.*, 2016], natural language processing [Devlin *et al.*, 2018], etc. The vital capability of DNNs is partly attributed to a large amount of training data. However, datasets sourced from the internet often lack reliability guarantees. This introduces a threat called poisoning-based backdoor attack [Gu *et al.*, 2017] that maliciously applies a specific trigger to a portion of the training data. Consequently, any model trained on this dataset will perform normally on clean data, but output unreasonable results when the trigger is present.

With the attacker's capacities gradually increasing, the backdoor attack can occur during the dataset preparation, training, and parameter or structure post-tuning phases. The more phases available to the attacker, the greater the probability of a successful attack. In this paper, we focus on planting backdoors in the image classification task through data poisoning, called poisoning-based backdoor, a less aggressive but more likely way to gain the user's trust than directly providing well-trained classifiers. The attacker's capabilities are limited to manipulating the dataset and do not extend to interfering with the training process itself. Therefore, the pattern of the trigger is critical to the success of the attack. Since the inception of backdoor attack, various patterns of triggers have sprung up in recent years. Patch-based triggers [Gu *et al.*, 2017; Liu *et al.*, 2018b] design a specific black-and-white or color block for the target class and add it to the input at a specified location, which can be easily found out by human inspection. To increase the stealthiness of triggers, the blended method [Chen *et al.*, 2017; Liu *et al.*, 2020] fuses inputs and triggers, and [Doan *et al.*, 2021a; Doan *et al.*, 2021b] utilizes the DNNs to generate invisible triggers with some limitations. In most cases, thanks to the strong capability of the DNNs, the backdoor attack makes the classifier overfit the link between the pre-defined or generated triggers supervised by several hyperparameters and target labels. These approaches ignore the mapping from the inputs to labels, which guarantees the accuracy of the clean dataset, and independently establish the trigger-label link within the classifier. This wastes resources because the input-label mapping instilled in a clean-data pre-trained network contains abundant prior information to help design triggers. We explore trigger types based on the input-label link and divide them into three categories based on whether they can promote or restrain the input to be classified into the target class.

Previous backdoor attack research can be broadly divided into two categories in terms of attack targets: one is *all2one*, where any input with a trigger is classified into one target label, and the other is *all2all*, where inputs of different classes are mapped to a predefined target label. As stated in Marksman [Doan *et al.*, 2022], *all2all* is a special single-trigger single-payload attack, since it can only modify the inputs from one true label into a single target label. Marksman introduces

† Corresponding author.

a novel backdoor attack that transcends these traditional categories, termed the *multi-label and multi-payload backdoor attack*. It can misclassify any input to any target label, which is nontrivial given the complexity of embedding a backdoor for each class. As the number of classes increases, maintaining both accuracy and attack success rates becomes increasingly challenging. Marksman [Doan *et al.*, 2022] achieves this by generating invisible, input-label-aware triggers and establishing a strong connection between the trigger function and the target label through alternating optimization of the trigger function and the classifier. A key limitation, however, is that this method requires control over the entire training process, which is unfeasible for the poisoning-based backdoor attack. Consequently, there remains a gap in developing poisoning-based backdoor attacks capable of arbitrarily selecting target labels for misclassification in models trained on poisoned datasets.

In this paper, we extend the multi-label and multi-payload backdoor attack to the poisoning-based scenario, where the attacker has fewer stages of control compared to Marksman. Based on the input-label links, we define triggers that reduce classification loss toward target labels as positive ones and use them as the basis for our design. Different from the universal adversarial patch-based triggers [Zhao *et al.*, 2020] crafted by minimizing the loss for each class, which is human-perceptible and cannot break STRIP and spectral detection defenses, we generate the input-label-aware invisible triggers utilizing the adversarial targeted technique. The magnitude of the triggers is restricted by the l_p norm to ensure stealthiness. Additionally, unlike the transferable adversarial attacks, we poison the dataset to enhance the link between the positive triggers and the target labels. Numerous experiments have shown that the PPT achieves superior accuracy and attack success rate compared to other poisoning-based methods in the multi-label and multi-payload backdoor attack field, under both dirty- and clean-label settings.

Our main contributions are summarized threefold:

1. We propose a new categorization of poisoning triggers, namely, the positive, neutral and negative triggers. These triggers can be leveraged to flexibly manipulate network classification results, increasing or decreasing the prediction scores of the target class.
2. Our method achieve a multi-label and multi-payload backdoor attack with lower requirements than the existing method. Only by poisoning a portion of the clean dataset, PPT can cover both dirty- and clean-label attacks simultaneously.
3. We empirically demonstrate the effectiveness of the PPT attack and its robustness against several popular defenses. The classifier trained naturally on the poisoned dataset generated by PPT performs well on the clean dataset and has a high attack success rate of misclassifying the input into any arbitrary label when the trigger is present.

2 Related Works

2.1 Backdoor Attack

Backdoor attack, which exposes the vulnerability of DNNs during the training process, is an emerging and rapidly growing research area in recent years. In the image classification field, the classifier implanted with a backdoor performs well on clean dataset, but incorrectly classifies the input as the target class whenever the trigger is present. The trigger is the important key to activate the backdoor inside of the classifier, especially for poisoning-based backdoor attacks. There are a variety of criteria to categorize triggers [Li *et al.*, 2022], such as, visibility, selection, and appearance. None of these categorization takes into account the relationship between triggers and input-label links that needs to be established to gain accuracy. To fill this gap, we divide triggers into three categories: positive, neutral, and negative triggers, according to the input-label link embedded in the clean dataset.

One important aspect of trigger is the stealthiness. Patch-based triggers [Gu *et al.*, 2017; Liu *et al.*, 2018b] are first proposed but easily detected by a human inspector. Blended [Chen *et al.*, 2017], Refool [Liu *et al.*, 2020], Wanet [Nguyen and Tran, 2021] and Ftrojan are proposed to make the triggers invisible to the human. Under the setting that the attacker can control the training process of the classifier, [Nguyen and Tran, 2020] utilizes a encoder-decoder neural network to generate the input-aware dynamic trigger, and LIRA [Doan *et al.*, 2021b] learns a transform function to produce the invisible triggers. Most attacks are under the dirty-label settings where the label for the poisoned data is not the true label. [Turner *et al.*, 2018] imposes stealthiness on the label and defines the clean-label attack as the label being consistent with the input. The clean-label attack is more difficult than the dirty one, since the classifier has to ignore some salient semantic information indicative of the label while establishing the link between the trigger and the label. [Zhao *et al.*, 2020] explores the clean-label backdoor attack on video recognition models. All of these attacks can only manipulate the prediction of a given input to one specified target label. Marksman [Doan *et al.*, 2022] first studies the multi-label multi-payload backdoor attack which can make the classifier predict any target label given any input. However the attacker in Marksman has a limitation, that it has to be able to control the whole training process of the classifier. In this paper, we consider a scenario with a weaker attacker who can only inject backdoors through data poisoning. Besides, we prove its effectiveness for both dirty- and clean-label attacks, while most backdoor attacks fall short under the clean-label setting.

2.2 Backdoor Defenses

With the rise of backdoor attacks, a variety of backdoor defenses have been proposed to distinguish whether the data or classifier is poisoned [Tran *et al.*, 2018; Chen *et al.*, 2022] or to help classifiers remove the embedded backdoors [Liu *et al.*, 2018a; Li *et al.*, 2021; Wu and Wang, 2021; Chen *et al.*, 2022]. STRIP [Gao *et al.*, 2019] superimposes various image patterns from different classes to the input and records the entropy of predicted classes for perturbed inputs, where a low entropy implies the presence of a malicious input. It has proven its effectiveness to detect the input-agnostic triggers (e.g., BadNet, Trojan). Spectral signature [Tran *et al.*, 2018] relies on the idea that the latent representations will contain a strong signal for the backdoor to detect and remove the poisoned data. Fine-pruning [Liu *et al.*, 2018a] analyzes

the neuron responses to the clean data and detects the dormant neurons, which are more likely related to the backdoor. It combines pruning and fine-tuning to effectively nullify backdoor attacks. Neural cleanse [Wang *et al.*, 2019] utilizes a pattern optimization method and median absolute deviation to detect the presence of a backdoor for one target label. Neural attention distillation [Li *et al.*, 2021] adopts teacher classifiers to guide finetuning of infected student classifiers. Adversarial neuron pruning [Wu and Wang, 2021] prunes sensitive neurons to remove the injected backdoor of an infected classifier. Our experiments prove the PPT as a successful poisoning-based backdoor attack against representative defenses.

2.3 Adversarial Perturbations

Adversarial perturbations are initially introduced as human-imperceptible and carefully crafted noise added to input data, aiming to catastrophically break the performance during inference [Kurakin *et al.*, 2016; Madry *et al.*, 2017; Carlini and Wagner, 2017; Croce and Hein, 2020]. Subsequently, some researchers have utilized adversarial perturbations to promote their work. Adversarial training (AT) [Madry *et al.*, 2017; Zhang *et al.*, 2019; Wu *et al.*, 2020], known as the most promising method so far against adversarial attacks, utilizes adversarial examples generated in each training step as data augmentation to gain robustness. [Ilyas *et al.*, 2019] demonstrates that adversarial perturbations are directly attributed to the presence of non-robust features derived from patterns in the data distribution that are highly predictive yet incomprehensible to humans. [Huang *et al.*, 2021] prevents unauthorized data exploitation by generating error-minimizing perturbations to make the data unlearnable. [Fowl *et al.*, 2021] poisons the dataset with adversarial perturbations to degrade the accuracy of the classifier. Adversarial perturbation can also be applied to the backdoor attack. Clean-label backdoor attack [Turner *et al.*, 2018] utilizes it to destroy the semantic information in the image for building the link between the pre-defined triggers and the target label for the classification task. [Zhao *et al.*, 2020] employs it similarly in video recognition models and generates the universal adversarial perturbations as triggers to activate the backdoor. However, the trigger is identical for one class and patch-based, thus easily detected by human inspection and defenses (e.g., STRIP, spectral signature). Indeed, without poisoning the dataset, black-box adversarial perturbations (i.e., positive triggers) can get a low attack success rate. Our work stands on this shoulder to show, for the first time, one can achieve a much higher success rate of attacks by poisoning the dataset with positive triggers. Experimental results supporting this claim are presented in supplementary A.1.

3 Methodology

3.1 Problems Statements

We assume the adversary has access to the clean dataset and knowledge of the classifier's architecture. The adversary is allowed to introduce a limited amount of perturbed data to the clean dataset without the permission to interfere with the training process of the classifier. The adversary aims to make classifiers trained on the poisoned dataset perform well on the clean data but predict any specified target class when a corresponding trigger is present.

3.2 Preliminaries

Backdoor Attack We focus on poisoning-based backdoor attacks on the image classification task. The adversary can poison N_p samples in the clean dataset to form a poisoned dataset $\hat{D}_p = D_c \cup D_p$, with $D_c = \{(\boldsymbol{x}_i, y_i)\}_i^{N_c}$ and $D_p = \{(\overline{\boldsymbol{x}}_i, \eta(y_i))\}_i^{N_p}$ indicating the clean and poisoned subsets, respectively. $\boldsymbol{x}_i \in \mathbb{X}$, $y_i \in \mathbb{Y}$ denote the clean data and the true label, $\overline{\boldsymbol{x}}_i = \boldsymbol{x}_i \oplus \boldsymbol{\delta}$ is the poisoned data with $\oplus$ being the fusion operator, $\boldsymbol{\delta}$ is the trigger which is related to the $\boldsymbol{x}_i$ and $\eta(y)$, and η represents the target labeling function. Clean-label attacks occur when $\eta(y)$ equals to y, otherwise they are dirty-label attacks. When users train an infected classifier f_b on the poisoned dataset $\hat{D}_p$, we want to inject backdoors which alter the behavior of f_b so that:

$$\begin{aligned} \min_{f_b} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(f_b(\hat{\boldsymbol{x}}_i), \hat{y}_i), \quad (\hat{\boldsymbol{x}}_i, \hat{y}_i) \in \hat{D}_p \\ f_b(\boldsymbol{x}_i) = y_i, \quad f_b(\overline{\boldsymbol{x}}_i) = \eta(y_i) \end{aligned} \tag{1}$$

We can set $\eta(y_i)$ to any label other than y_i and $N = N_p + N_c$.

Adversarial perturbation Projected gradient descent (PGD) [Madry *et al.*, 2017], the most commonly used attack method, formulates generating the adversarial perturbations as a constrained optimization problem. Namely, given a clean input x^0, a target label y, step size α, norm limitation ϵ and iterations numbers K, PGD works as follows:

$$\begin{aligned} \boldsymbol{x}^{t+1} =& \Pi_\epsilon(\boldsymbol{x}^t \pm \alpha \text{sign}(\nabla_{\boldsymbol{x}^t} \mathcal{L}(f(\boldsymbol{x}^t), y))), \\ \boldsymbol{\delta} =& \boldsymbol{x}^K - \boldsymbol{x}^0, \quad \text{s.t.} \quad \|\boldsymbol{\delta}\|_p \leq \epsilon \end{aligned} \tag{2}$$

where $\boldsymbol{x}^t$ denotes as the adversarial input at tth iteration, $\mathcal{L}(\cdot)$ represents the classification loss (e.g., cross-entropy (CE)), f is a classifier model, and Π_ϵ is the projection onto the ϵ-ball centered at $\boldsymbol{x}^0$. δ is the adversarial perturbation bounded by an L_p-norm, set to L_∞ as default in this paper. Note that the sign between $\boldsymbol{x}^t$ and the backpropagation gradient part can be positive or negative, with positive representing that the generated perturbation moves the input away from the label y (untargeted attack) and vice versa (targeted attack).

Evaluation Metrics We evaluate the performance of the PPT attack adopting two commonly used metrics: accuracy on clean data (ACC) and attack success rate (ASR), i.e., accuracy of predicting poisoned non-target input as the target label. Since we attack any target label at will, we compute the ASR against each class and report its average as ASR.

3.3 Categories of the Triggers

Maintaining accuracy without degradation is essential to a successful backdoor attack. On the basis of establishing the input and label links in classifiers trained on the clean dataset, we divide triggers into three categories: positive, neutral, and negative triggers.

Given a benign classifier f trained on the clean dataset, a pair of input and label $(\boldsymbol{x}, y)$, and a target label $\eta(y)$, a trigger

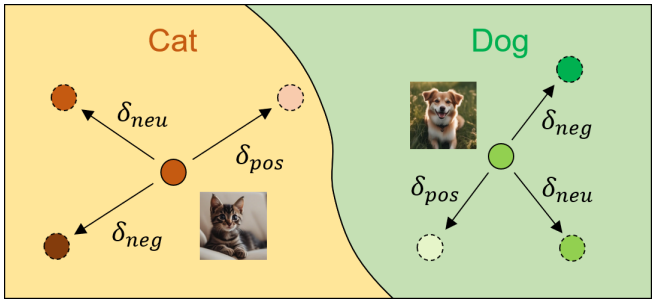

Figure 1: Positive, neutral and negative triggers from a cat image to dog and from a dog image to cat of a benign classifier.

($\boldsymbol{\delta}$) is defined as a positive one for $(\boldsymbol{x}, \eta(y))$ if the $f(\boldsymbol{x}+\boldsymbol{\delta})$ is moved closer to the target label $\eta(y)$, as shown in Fig. 1. We generate the positive triggers following objective:

$$\min_{\|\boldsymbol{\delta}\|_\infty \leq \epsilon} \mathcal{L}\left(f\left(\boldsymbol{x} \oplus \boldsymbol{\delta}\right), \eta(y)\right), \tag{3}$$

PGD targeted perturbation is a typical positive trigger. It can induce the input to be categorized into a specified label under certain conditions. The opposite is true for the negative trigger, which will maximize the loss after adding it to the input (e.g., PGD untargeted perturbations). While, neutral triggers do not affect the classifier's prediction, whether they are added to the input or not. In other words, the link between the neutral trigger and the label is independent of the input-label link. Most backdoor triggers belong to the neutral categories. For example, a checkerboard patch (BadNet [Gu *et al.*, 2017]) in the corner will not lead the benign classifier to recognize a dog as a cat. It is also true for some input-aware generated triggers. For example, the Marksman [Doan *et al.*, 2022], utilizing a trigger function to develop an invisible optimal trigger pattern to attack any target label at will, builds a link between the trigger function and the target label during the training phase. This mapping relationship is absent in benign classifiers, meaning that Marksman-generated triggers cannot lead to misclassification (cf. last column in Table 4 in supplementary). Intuitively, the negative trigger is unsuitable for building a link between the trigger and target label since it contains the information pushing it away from the target label. We provide the experimental results with the PGD untargeted perturbations as the triggers in supplementary A.2, suggesting that negative triggers are not an ideal choice. Neutral triggers, which is independent of the input-label link, are the common choice of the previous backdoor attacks. Instead, we focus on implanting backdoors with positive triggers, which should have less resistance to be classified into target label since they take advantage of the input-label links.

3.4 Poisoning via Positive Triggers (PPT)

To inject a backdoor into f, we poison a subset of the clean set with a poisoning rate ρ to form a poison dataset $\hat{D}_p = D_c \cup D_p$. The overall framework of the PPT backdoor attack is shown in Fig. 2. We present the details of the PPT algorithm in supplementary A.4.

Trigger generator First, we train a benign classifier on a clean dataset from scratch to minimize the classification loss. Once the training is finished, the benign classifier is fixed as a trigger generator representing the mapping function from the input space to the label space.

Poison dataset By feeding a clean data with a target label $(\boldsymbol{x}^{\boldsymbol{0}}, \eta(y))$ into the trigger generator, we produce the input-label-aware poisoned data $\boldsymbol{x}^{\boldsymbol{K}}$ with the targeted PGD according to the Eqn. 2 and replace the clean data $(\boldsymbol{x}^{\boldsymbol{0}}, y)$ with $(\boldsymbol{x}^{\boldsymbol{K}}, \eta(y))$. For clean-label attack, the target label $\eta(y)$ equals the true label y; otherwise, it is uniformly sampled from the label domain except for the true label. The green and red labels shown in Fig. 2 represent the dirty- and clean-label attacks, respectively.

Inference The backdoors are silently implanted when users train their own classifiers on the poisoned dataset. At the inference stage, given any input and any target label, we can poison the input with a trigger following the Eqn. 2 to manipulate the infected classifier to predict the target label. As shown in Fig. 2, a bird image can be misclassified as a dog or cat with different triggers. Besides, the infected classifiers perform well in the clean data as they predict the bird without the trigger as a bird.

4 Experiments

4.1 Experimental Settings

Datasets: Following the previous backdoor attack papers, we performed comprehensive experiments on four widely-used datasets: SVHN [Netzer *et al.*, 2011], CIFAR10 [Krizhevsky *et al.*, 2009], GTSRB [Stallkamp *et al.*, 2012], and Tiny ImageNet [Le and Yang, 2015]. Note that because of the inconsistent image sizes in the GTSRB, we resize all images to 32×32 as input. We use various architectures for the classifier f: a simple CNN model for SVHN (reported in supplementary A.5), Pre-activation ResNet18 (Pre-ResNet18) [He *et al.*, 2016] for CIFAR10 and GTSRB, and ResNet18 [He *et al.*, 2016] for Tiny ImageNet as suggested by [Doan *et al.*, 2022].

Hyperparameters: We train the trigger generator and classifier for 300 epochs with a batch size of 128 utilizing a SGD optimizer with the Cross-Entropy (CE) loss. The initial learning rate was set to 1×10^{-2}, which decayed to one-tenth after 100 and 200 epochs, respectively. Following the settings of Marksman, the maximum l_∞ norm-bounded perturbation ϵ was set to 0.05 for all datasets. All experiments were conducted on one Nvidia RTX 3090 GPU.

Configurations: To our best knowledge, this paper is the first work that explores the multi-trigger and multi-payload poisoning-based backdoor attack, which enables classifying inputs poisoned by input-label-aware triggers into any target class without interfering with the training process of the classifier. Classifiers trained on this poisoned dataset are implanted with multiple backdoors so that we can manipulate the input to fall into any target class. Following the settings of Marksman [Doan *et al.*, 2022], we implement three typical baseline methods: **PatchMT**, **WaNetMT**, and **Marksman** for comparison. Details of these baselines are in supplementary A.3. More discussion of real-world implications and potential defense strategies are included in supplementary A.9.

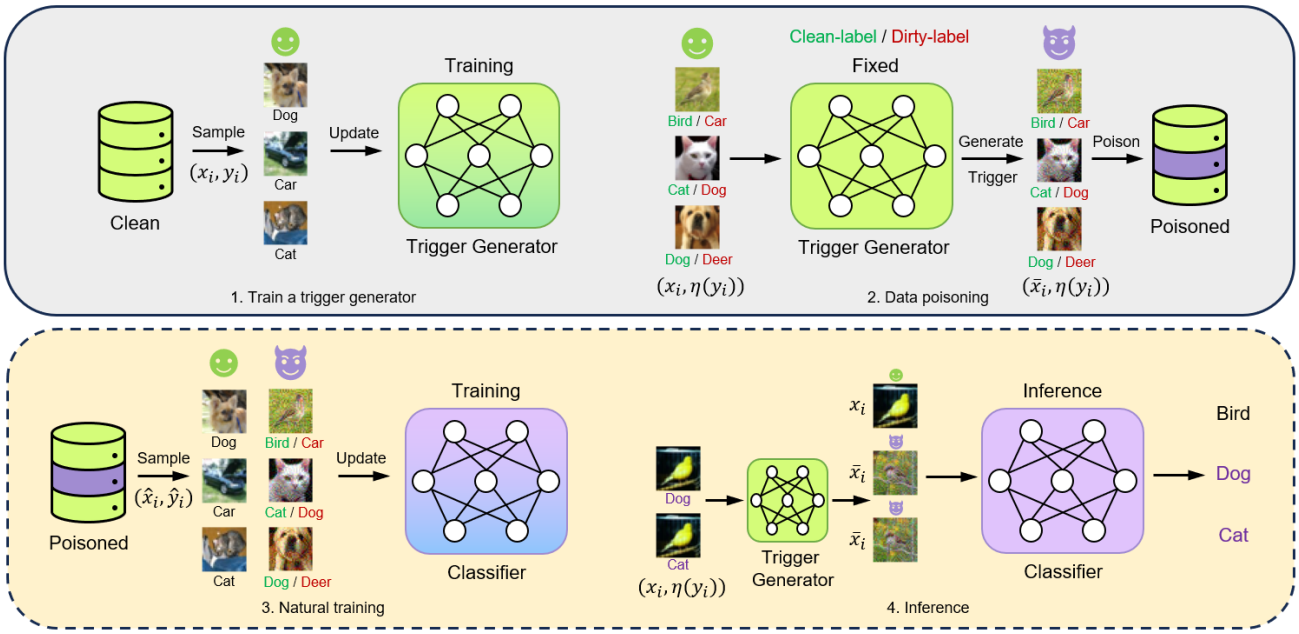

Figure 2: **The Overall framework of the PPT**: The solid box (upper) shows the data poisoning process, and the dashed box (lower) demonstrates the process of training a classifier on the poisoned dataset and performing inference on the clean and poisoned inputs. The triggers of the poisoned data are magnified five times for better understanding.

4.2 Attack Performance

We consider both dirty- and clean-label settings for data poisoning. Since the data and label of the dirty-label setting are inconsistent, a lower poisoning rate is better for evading human random inspection. The clean-label one does not change the input's label, making it harder to inject the backdoors because the classifier needs to ignore information in the data to establish a link between the trigger and the label. While ensuring stealthiness, we can add more clean-label poisoned data without worrying about human spot checks. For each clean test input $(\boldsymbol{x}, y)$, we enumerate all target labels except y to generate triggers $\boldsymbol{\delta}$ and feed $\boldsymbol{x} \oplus \boldsymbol{\delta}$ into the classifier to check the ASR. The attack is considered successful if the classifier classifies $\boldsymbol{x} \oplus \boldsymbol{\delta}$ as the target label. The ASR reported in this paper is the average of ASR of each target label. We provide the accuracy and ASR with 1% and 10% poisoning rates for dirty- and clean-label in Table 1, respectively. More experimental results with different poisoning rates are provided in supplementary A.6, Our method outperforms other poisoning-based multi-label and multi-payload backdoor attacks with a significantly higher ASR.

The ACC and ASR of three typical attacks and PPT are presented in Table 1. For the dirty-label attack, the input-aware triggers generated by WaNetMT and Marksman are unable to form an effective attack with a small (viz. 1%) poisoning rate, while the patch-based PatchMT has a better ASR performance. It indicates that without controlling the classifier training process, it is far simpler to let the DNN memorize multiple definite patterns than recognize the trigger functions applied to the input. In contrast, the PPT attack can maintain a high ASR across four datasets. As mentioned before, when the number of classes increases, it becomes more difficult to simultaneously implant an equal number of backdoors within the classifier. The ASR of PatchMT drops from 84.38% for 10 classes in CIFAR10 to 23.6% for 43 classes in GTSRB and to 0.23% similar to a random guess for 200 classes in Tiny ImageNet. Since the proposed PPT is dependent on the input-label link, it can be easily embedded in the classifier, proved by the 84.97% ASR of GTSRB and 24.53% ASR of Tiny ImageNet, a significant improvement compared to the PatchMT. Besides, the accuracy of PPT is superior to the other three baselines and similar to the benign model, which suggests that the positive triggers contain features that can facilitate the image classification, consistent with the opinions in [Ilyas *et al.*, 2019]. Regarding the clean-label attack, all three baselines fail to inject the backdoors inside the classifiers with a 10% poisoning rate. Since the input contains abundant semantic information related to the true label, it prevents the classifier from building a solid link between the trigger and the target label. In comparison, the PPT is a highly effective attack that achieves a high ASR with negligible ACC degradation across four datasets. More experimental results with 10% and 50% poisoning rates for dirty- and clean-label attacks are provided in Table 6 in supplementary. With an increase in the poisoning rate, all backdoor attacks can improve their ASR with a higher risk of being detected. Nevertheless, all three baselines fail to form a successful backdoor attack for GTSRB and Tiny ImageNet datasets under the clean-label settings.

4.3 Ablation Studies

Poisoning Rate We evaluate the ACC and ASR of different poisoning rate (i.e.,$|D_p|/(|D_p| + |D_c|)$) within a reasonable range. The results of dirty-label attack with poisoning rate ranging from 1% to 10% are shown in Fig. 3. As the poisoning

Dataset	Benign ACC	PatchMT ACC	PatchMT ASR	WaNetMT ACC	WaNetMT ASR	Marksman ACC	Marksman ASR	PPT ACC	PPT ASR
Dirty-label $\eta(y) \neq y$ poisoning rate = 1%									
SVHN	95.21	94.66	**99.32**	94.55	1.22	94.75	0.92	94.79	93.13
CIFAR-10	94.32	93.90	94.71	94.42	5.62	93.62	0.89	94.22	**98.53**
GTSRB	99.17	98.60	23.6	98.92	0.60	99.08	10.82	99.34	**85.48**
Tiny-ImageNet	59.02	58.32	0.23	58.89	0.36	58.97	0.68	59.38	**22.23**
Clean-label $\eta(y) = y$ poisoning rate = 10%									
SVHN	95.21	94.18	46.28	94.91	0.71	95.17	0.58	94.30	**86.83**
CIFAR-10	94.32	94.20	10.84	94.28	0.91	94.72	0.62	93.92	**92.22**
GTSRB	99.17	99.18	0.03	99.13	0.03	98.86	0.02	99.06	**39.83**
Tiny-ImageNet	59.02	58.89	0.83	58.67	0.21	58.31	0.24	58.96	**24.01**

Table 1: Accuracy (%) and ASR(%) across four datasets. Benign represents the accuracy of the classifier trained on the clean dataset. Bold values indicate the best ASR.

rate increases, the ASR of the attack increases, while the ACC only drops by a negligible amount. A small amount of poisoned data already guarantees a high ASR for datasets with ten classes, such as SVHN, CIFAR10. For datasets with more classes, the increase in the poisoning rate provides much of a performance boost. In the case of the GTSRB, the Attack Success Rate (ASR) improves by 9.64% with a slight decrease in accuracy of 0.40%. Similarly, for the Tiny ImageNet, the ASR improves by a significant 51.29% with a reduction in accuracy of 1.54%. More results of the clean-label attack are provided in Fig. 9 in supplementary.

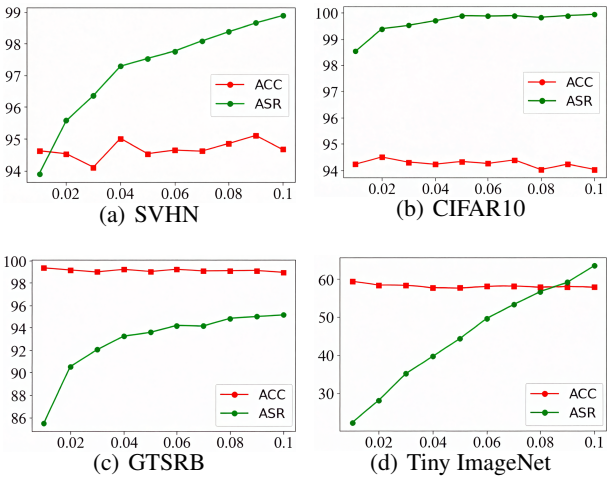

Figure 3: Ablation studies of poisoning rate on (a) SVHN, (b) CIFAR10, (c) GTSRB, and (d) Tiny ImageNet for dirty-label attack. The x-axis denotes the poisoning rate.

Trigger Generator & Classifier We explore how the architecture of the trigger generator and the classifier affects the performance of the attack. Compared to the assumption that the trigger generator and classifier employ the same architecture, a more general setting is that the attacker does not know the classifier structure adopted by the user. We use four typical convolutional neural networks (CNNs) as trigger generators and classifiers to test the dirty-label attack's accuracy and attack success rate with 1% poisoning rate. As shown in Table 2, ResNet18, used as a trigger generator, demonstrates a high ASR regardless of the architecture used for the classifier. It indicates that consistency in the architecture of generators and classifiers is *not* necessary for effective backdoor attacks. As long as the neural network can establish a link between input and label, it can serve as a functional trigger generator. In supplementary, clean-label attack yields similar results, depicted in Table 7.

ACC (%) / ASR (%)		Trigger Generator EfficientNet-B0	MobileNet-V2	ResNet18	PreResNet18
Classifier	EfficientNet-B0	91.62 / **95.61**	92.23 / 87.32	91.94 / 93.52	92.03 / 91.23
	MobileNet-V2	94.02 / 89.01	93.82 / 93.98	94.11 / **96.89**	93.92 / 95.62
	ResNet18	94.28 / 88.17	94.32 / 92.21	94.02 / **99.15**	94.11 / 98.28
	PreResNet18	94.32 / 88.23	94.41 / 91.59	94.31 / **98.58**	94.22 / 98.53

Table 2: Accuracy (%) and ASR(%) of dirty-label attack for different model architectures on CIFAR10. Bold values represent the best ASR for each classifier.

4.4 Defence Performance

Here we evaluate the infected classifier trained on the proposed poisoned dataset against the popular defense mechanisms, including STRIP [Gao *et al.*, 2019], Spectral Signature [Tran *et al.*, 2018], and Fine-Pruning [Liu *et al.*, 2018a]. To save space, more evaluation results against Neural Cleanse [Wang *et al.*, 2019] and two post-training defenses called Neural Attention Distillation (NAD) [Li *et al.*, 2021] and Adversarial Neuron Pruning (ANP) [Wu and Wang, 2021] are provided in supplementary A.8. Unless otherwise stated, the results are for the dirty-label attack with 1% poisoning rate. Experimental results for the clean-label attack with 10% poisoning rate are presented in supplementary A.8.

STRIP is a typical testing-time defense method. It superimposes different images to the input and records the entropy of predicted classes, where a low entropy implies the presence of a malicious input. We plot the entropy of the clean and poisoned inputs in Figs. 4 (also Fig. 11 in supplementary). Similar energy distribution of the clean and poisoned inputs indicates defense failure.

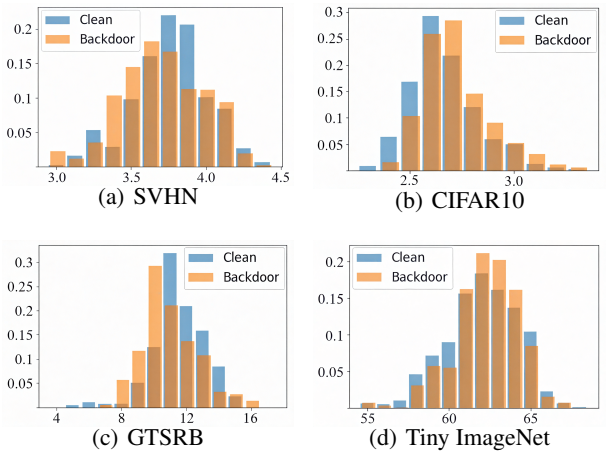

Figure 4: Entropy distributions on (a) SVHN, (b) CIFAR10, (c) GTSRB, and (d) Tiny ImageNet. The x-axis and y-axis denote the entropy and probability.

Spectral Signatures computes the top singular vector of the covariance matrix of the latent representations using a subset of clean data and calculates the correlation of each input to this top singular vector. The input with an outlier score is recognized as poisoned data. For each dataset, we randomly select a target label and compute the correlation score for both the clean and poisoned inputs. As shown in Figs. 5 (also Fig. 13 in supplementary), there is no clear distinction between clean and poisoned inputs, which verifies the stealthiness of the PPT attack.

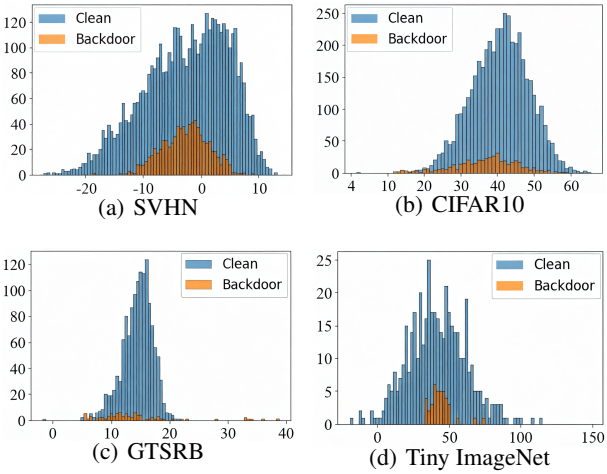

Figure 5: Spectral signature on (a) SVHN, (b) CIFAR10, (c) GTSRB, and (d) Tiny ImageNet. The x-axis and y-axis denote the correlation with top singular vector and number of images.

Fine-Pruning assumes the dormant neurons are more likely to tie to the backdoor. Given a special layer, it detects and gradually prunes the dormant neurons with low activations to mitigate the backdoor. Following previous works [Nguyen and Tran, 2021; Doan *et al.*, 2022], we prune the last CNN layer and provide the results in Fig. 6 (also Fig. 12 in supplementary). ACC and ASR have the same trend, indicating that reducing ASR without degrading accuracy is impossible.

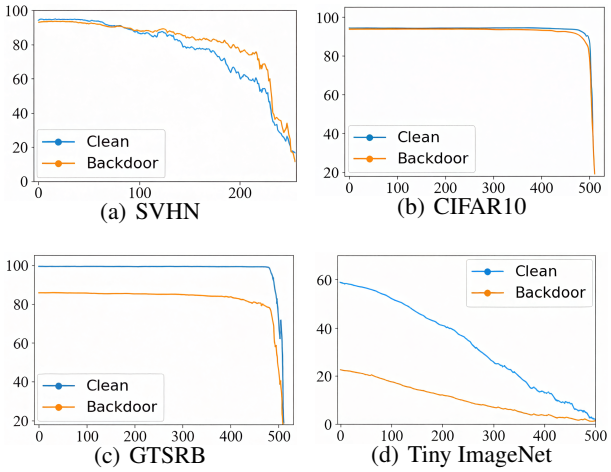

Figure 6: Fine-pruning on (a) SVHN, (b) CIFAR10, (c) GTSRB, and (d) Tiny ImageNet. The x-axis and y-axis denote the number of pruned filters and accuracy.

4.5 Visualization of Poisoned Data

GradCam [Selvaraju *et al.*, 2017] We provide the visualization of the clean and poisoned data in Fig. 7. Based on the true labels, the GradCam activations of the clean data computed by a benign classifier and the activations of the poisoned one computed by an infected classifier are included. The heatmap of the clean input looks like the poisoned one, suggesting the infected classifier extracts information from similar regions as the benign one.

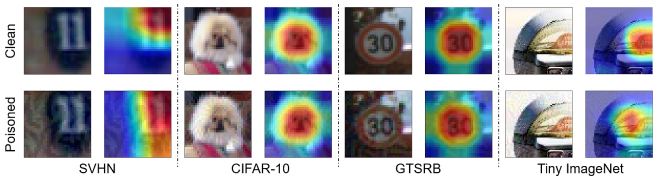

Figure 7: GradCam visualization between the clean (top) and poisoned images (bottom) computed by a clean classifier and an infected one across four datasets.

5 Conclusion

This paper introduces a new categorization of backdoor triggers based on whether they can move the input close to the target label and achieve a high attack success rate (ASR) in multi-label and multi-payload backdoor attacks by poisoning the dataset with positive triggers for both dirty- and clean-label attacks. Additionally, we demonstrate the resilience of the proposed PPT attack against several popular backdoor defenses. This research opens up new possibilities for designing backdoor triggers and encourages future exploration in defense mechanisms.

Acknowledgments

This work was supported in part by the Theme-based Research Scheme (TRS) project T45-701/22-R of the Research Grants Council (RGC), Hong Kong SAR.

References

[Carlini and Wagner, 2017] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. IEEE, 2017.

[Chen *et al.*, 2017] 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.

[Chen *et al.*, 2022] Weixin Chen, Baoyuan Wu, and Haoqian Wang. Effective backdoor defense by exploiting sensitivity of poisoned samples. *Advances in Neural Information Processing Systems*, 35:9727–9737, 2022.

[Croce and Hein, 2020] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pages 2206–2216. PMLR, 2020.

[Devlin *et al.*, 2018] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[Doan *et al.*, 2021a] Khoa Doan, Yingjie Lao, and Ping Li. Backdoor attack with imperceptible input and latent modification. *Advances in Neural Information Processing Systems*, 34:18944–18957, 2021.

[Doan *et al.*, 2021b] Khoa Doan, Yingjie Lao, Weijie Zhao, and Ping Li. Lira: Learnable, imperceptible and robust backdoor attacks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11966–11976, 2021.

[Doan *et al.*, 2022] Khoa D Doan, Yingjie Lao, and Ping Li. Marksman backdoor: Backdoor attacks with arbitrary target class. *Advances in Neural Information Processing Systems*, 35:38260–38273, 2022.

[Fowl *et al.*, 2021] Liam Fowl, Micah Goldblum, Ping-yeh Chiang, Jonas Geiping, Wojciech Czaja, and Tom Goldstein. Adversarial examples make strong poisons. *Advances in Neural Information Processing Systems*, 34:30339–30351, 2021.

[Gao *et al.*, 2019] Yansong Gao, Change Xu, Derui Wang, Shiping Chen, Damith C Ranasinghe, and Surya Nepal. Strip: A defence against trojan attacks on deep neural networks. In *Proceedings of the 35th Annual Computer Security Applications Conference*, pages 113–125, 2019.

[Gu *et al.*, 2017] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*, 2017.

[He *et al.*, 2016] 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.

[Huang *et al.*, 2021] Hanxun Huang, Xingjun Ma, Sarah Monazam Erfani, James Bailey, and Yisen Wang. Unlearnable examples: Making personal data unexploitable. *arXiv preprint arXiv:2101.04898*, 2021.

[Ilyas *et al.*, 2019] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. *Advances in neural information processing systems*, 32, 2019.

[Krizhevsky *et al.*, 2009] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[Kurakin *et al.*, 2016] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016.

[Le and Yang, 2015] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

[Li *et al.*, 2021] Yige Li, Xixiang Lyu, Nodens Koren, Lingjuan Lyu, Bo Li, and Xingjun Ma. Neural attention distillation: Erasing backdoor triggers from deep neural networks. *arXiv preprint arXiv:2101.05930*, 2021.

[Li *et al.*, 2022] Yiming Li, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. Backdoor learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[Liu *et al.*, 2018a] Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International symposium on research in attacks, intrusions, and defenses*, pages 273–294. Springer, 2018.

[Liu *et al.*, 2018b] Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and Xiangyu Zhang. Trojaning attack on neural networks. In *25th Annual Network And Distributed System Security Symposium (NDSS 2018)*. Internet Soc, 2018.

[Liu *et al.*, 2020] Yunfei Liu, Xingjun Ma, James Bailey, and Feng Lu. Reflection backdoor: A natural backdoor attack on deep neural networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 182–199. Springer, 2020.

[Madry *et al.*, 2017] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

[Netzer *et al.*, 2011] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[Nguyen and Tran, 2020] Tuan Anh Nguyen and Anh Tran. Input-aware dynamic backdoor attack. *Advances in Neural Information Processing Systems*, 33:3454–3464, 2020.

[Nguyen and Tran, 2021] Anh Nguyen and Anh Tran. Wanet–imperceptible warping-based backdoor attack. *arXiv preprint arXiv:2102.10369*, 2021.

[Selvaraju *et al.*, 2017] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[Stallkamp *et al.*, 2012] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks*, 32:323–332, 2012.

[Tran *et al.*, 2018] Brandon Tran, Jerry Li, and Aleksander Madry. Spectral signatures in backdoor attacks. *Advances in neural information processing systems*, 31, 2018.

[Turner *et al.*, 2018] Alexander Turner, Dimitris Tsipras, and Aleksander Madry. Clean-label backdoor attacks. 2018.

[Wang *et al.*, 2019] 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.

[Wu and Wang, 2021] Dongxian Wu and Yisen Wang. Adversarial neuron pruning purifies backdoored deep models. *Advances in Neural Information Processing Systems*, 34:16913–16925, 2021.

[Wu *et al.*, 2020] Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. *Advances in Neural Information Processing Systems*, 33:2958–2969, 2020.

[Zhang *et al.*, 2019] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pages 7472–7482. PMLR, 2019.

[Zhao *et al.*, 2020] Shihao Zhao, Xingjun Ma, Xiang Zheng, James Bailey, Jingjing Chen, and Yu-Gang Jiang. Clean-label backdoor attacks on video recognition models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14443–14452, 2020.