Progressive Poisoned Data Isolation for Training-Time Backdoor Defense

Yiming Chen, Haiwei Wu, and Jiantao Zhou[†]

State Key Laboratory of Internet of Things for Smart City
Department of Computer and Information Science, University of Macau
{yc17486, yc07912, jtzhou}@um.edu.mo

Abstract

Deep Neural Networks (DNN) are susceptible to backdoor attacks where malicious attackers manipulate the model's predictions via data poisoning. It is hence imperative to develop a strategy for training a clean model using a potentially poisoned dataset. Previous training-time defense mechanisms typically employ an one-time isolation process, often leading to suboptimal isolation outcomes. In this study, we present a novel and efficacious defense method, termed Progressive Isolation of Poisoned Data (PIPD), that progressively isolates poisoned data to enhance the isolation accuracy and mitigate the risk of benign samples being misclassified as poisoned ones. Once the poisoned portion of the dataset has been identified, we introduce a selective training process to train a clean model. Through the implementation of these techniques, we ensure that the trained model manifests a significantly diminished attack success rate against the poisoned data. Extensive experiments on multiple benchmark datasets and DNN models, assessed against nine state-of-the-art backdoor attacks, demonstrate the superior performance of our PIPD method for backdoor defense. For instance, our PIPD achieves an average True Positive Rate (TPR) of 99.95% and an average False Positive Rate (FPR) of 0.06% for diverse attacks over CIFAR-10 dataset, markedly surpassing the performance of state-of-the-art methods. The code is available at https://github.com/RorschachChen/PIPD.git.

Introduction

The utilization of diverse datasets in training DNNs enhances their adaptability and performance across various tasks and domains. However, the demand for multiple sources of data also introduces a vulnerability to backdoor attacks (Gu, Dolan-Gavitt, and Garg 2017). Malicious attackers can exploit this situation by injecting hidden backdoors into the training data, thereby manipulating the predictions of the model. The potential harm of backdoor attacks lies in their ability to trigger malicious behaviors in the deployed model once activated. Such attacks can lead to the disruption of system operations, and even system crashes.

Backdoor attack is a type of adversary (Weng, Lee, and Wu 2020) that implants malicious backdoor behaviors into the victim models during training. This goal can be accomplished by introducing contamination into the data used for training. The existing backdoor attacks can be roughly divided into two categories (Cheng et al. 2023), namely *patching* and *transforming*. Specifically, in patching attacks, a trigger is injected to a benign sample by merging their pixel values with a mask. In contrast, transforming attacks inject the trigger into the input using a transformation function in the form of an algorithm or a network.

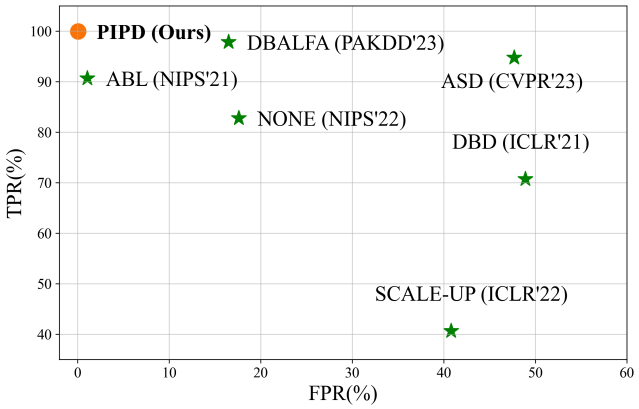

Figure 1: Isolation quality comparison of the proposed PIPD with the state-of-the-art methods on CIFAR-10 (Krizhevsky, Hinton et al. 2009).

As backdoor attacks pose significant threats to DNNs, many strategies and techniques have been proposed to effectively defend against these attacks. This paper focuses on the training-time defense that involves isolating poisoned samples, thereby enabling the training of a clean model on a poisoned dataset. There are several representative works in this topic including (Li et al. 2021; Huang et al. 2022; Wang et al. 2022; Gao et al. 2023). Specifically, (Li et al. 2021) leveraged the training loss to isolate poisoned samples then apply an unlearning process to train a clean model on poisoned dataset. (Huang et al. 2022) maintained two dynamically updated poisoned and benign data pool based on sample losses then used semi-supervised learning to fine-tune the model on these pools. Similarly, (Gao et al. 2023) applied loss-guided split and meta-learning-inspired split to dynamically update

[†]Corresponding author.

two data pools. (Wang et al. 2022) identified poisoned sample by checking linearity of trained models. Most of these methods initially perform a one-time isolation process to identify poisoned samples, then train a clean model during a subsequent training phase.

In this work, our aim is to establish a defense method based on isolating poisoned samples that extends one-time isolation to a progressive isolation for better isolation quality. We start by pinpoint a prevalent problem in current solutions where an one-time isolation strategy struggles to accurately distinguish poisoned samples within the poisoned dataset, and regularly mis-identify the benign samples as poisoned ones. The inaccurate separation will precipitate a high Attack Success Rate (ASR) in the post-training model, whereas the mis-identification problem could risk undermining the clean inference performance of the model. To this end, we propose a new defense process in which we firstly employ a pre-isolation process to initialize the poisoned and benign subsets for the later isolation process which based on distribution discrepancy. This process fortifies isolation quality by stringently controlling the isolation ratio, isolating only the samples with extreme training losses as the subset. The next stage involves identifying the channels exhibiting the most significant discrepancies and discerning two distributions based on the cumulative feature value in these channels. We tackle the high benign sample mis-identification problem by developing a progressive isolation strategy. This progressive process transforms one-time isolation into a progressive isolation, enhancing the quality of isolation progressively. Upon identifying the poisoned samples, we develop a selective training scheme to inhibit the model from learning the backdoor behavior. Employing these innovative strategies allows us to train a benign model on poisoned dataset exhibiting a low ASR and superior Clean Accuracy (CA). Combining the above designs, PIPD can surpass the state-of-the-art methods by a big margin in terms of the isolation result (see Fig. 1).

Our major contributions can be summarized as follows:

- We propose a novel and effective poisoned data isolation strategy against backdoor attack. E.g., for BadNets, Blended, and Trojan attacks on the CIFAR-10 dataset, our isolation quality can reach 100% TPR and 0% FPR.
- We design a selective training strategy to effectively prevent the model from being implanted with backdoor after the poisoned part is isolated. In conjunction with our isolation technique, the proposed learning strategy can efficaciously suppress the ASR.
- Experiments across different networks and datasets have proved the effectiveness of our defense method. For almost every attack methods, our method succeeds in precise poisoned sample isolation and the ASR of the purified model is close to that training on clean set.

Related Works

Backdoor Attack

As the pioneering work in backdoor attacks, (Gu, Dolan-Gavitt, and Garg 2017) manipulated the training set by injecting mislabeled poisoned samples with a specific trigger to execute the attack. (Liu et al. 2017) selected an input region along with neurons which exhibit sensitivity to changes in that region, and the attacker aimed to activate these neurons. Many other works such as (Chen et al. 2017; Nguyen and Tran 2020) have proposed techniques to enhance the stealthiness of backdoor attacks, thereby reducing the effectiveness of backdoor detection mechanisms. Various forms of the triggers have been developed, including sinusoidal strips (SIG) (Barni, Kallas, and Tondi 2019), reflection (ReFool) (Liu, Bailey, and Lu 2020), and warping-based (WaNet) (Nguyen and Tran 2021). In particular, WaNet used a smooth warping field to generate backdoor images with inconspicuous modifications. LIRA (Doan et al. 2021) alternated between trigger generation and backdoor injection to learn visually stealthy triggers. Adversarial Embedding (Tan and Shokri 2020) sought to enhance the latent indistinguishability of backdoor attacks by minimizing the distance between the latent distributions of backdoor inputs and clean inputs through adversarial regularization. BppAttack (Wang, Zhai, and Ma 2022) combined image quantization and dithering to implant the trigger into the dataset. Combined with style transfer technique, (Cheng et al. 2021) proposed an image style trigger attack. Similarly, (Jiang et al. 2023) presented a backdoor attack method using color space shift. Backdoor attacks have been extended to various applications, including those outlined by (Saha et al. 2022) for self-supervised learning (Grill et al. 2020), (Li et al. 2023) for pretrained models and (Ma et al. 2022) for object detection (Redmon and Farhadi 2018). Concurrently, certain backdoor attacks have been specifically designed for distinct network architectures, exemplified by (Yuan et al. 2023) for vision transformers (Dosovitskiy et al. 2021) and (Chou, Chen, and Ho 2023) for the diffusion model (Ho, Jain, and Abbeel 2020). A handful of studies, such as that by (Hayase and Oh 2023), have also explored more efficient methods to implant backdoors, with their work focusing on neural tangent kernels to significantly reduce the poison rate. Additionally, (Shi et al. 2023) have explored the vulnerability of the latest ChatGPT to backdoor attacks.

Backdoor Defense

To combat backdoor attacks, many defense algorithms have been proposed to train clean models on poisoned datasets. (Li et al. 2021) observed that poisoned samples typically have a lower training loss compared to benign samples. They proposed a two-stage process: firstly, isolating a few samples with the lowest losses, and secondly, unlearning the backdoor on these isolated samples. (Huang et al. 2022) was the pioneer in utilizing self-supervised learning and semi-supervised learning in backdoor defense. (Guo et al. 2023) found that poisoned samples demonstrated scaled prediction consistency when pixel values were amplified, and they identified poisoned samples by tracking the predictions of these scaled images. (Liu et al. 2023) revealed that poisoned models exhibit consistent performance on benign images under various image corruptions while performing divergently on poisoned images. (Jebreel, Domingo-Ferrer, and Li 2023) observed that poisoned samples and benign samples exhibit

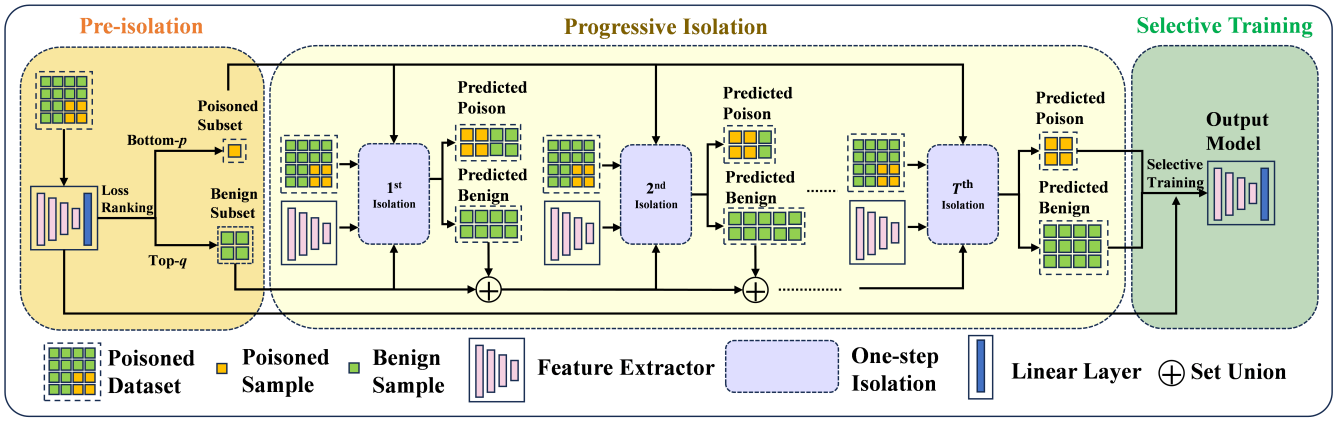

Figure 2: The overview of PIPD method.

significant differences at a crucial layer, and using the feature difference at this critical layer could aid in distinguishing poisoned samples. (Wang et al. 2022) pinpointed backdoor neurons by investigating the linearity of the models, and performed a statistical test to identify poisoned samples.

Preliminaries

Threat Model

We consider the same threat model as in prior defense methods (Li et al. 2021; Huang et al. 2022; Gao et al. 2023; Wang et al. 2022), which assumed the backdoor injection is performed at training stage and the attacker can only poison the dataset rather than control the training process. Formally, given a training set $\mathcal{D} = \{\mathcal{D}_b, \mathcal{D}_p\}$ including poisoned part $\mathcal{D}_p$=$\{(\hat{x}_i, \hat{y}_i)\}_{i=1}^{n_p}$ and benign part $\mathcal{D}_b$=$\{(x_i, y_i)\}_{i=1}^{n_b}$, the training process can be regarded as a dual-task learning on these two parts:

$$\theta^* = \arg\min_{\theta}(\mathbb{E}_{(x,y)\sim\mathcal{D}_b}\big[\ell(f(x), y)\big] + \mathbb{E}_{(\hat{x},\hat{y})\sim\mathcal{D}_p}\big[\ell(f(\hat{x}), \hat{y})\big]), \quad (1)$$

where θ represents the parameters of model f, and ℓ denotes the objective loss (*e.g.*, cross-entropy loss).

Defense Goal

Fundamentally, the objective of defense is to train a clean model on poisoned dataset $\mathcal{D}$. This model should have a low ASR on the poisoned test set, while maintaining a high inference performance on the clean test set. Considering the capabilities in previous works, we assume that the defender has access to the poisoned dataset $\mathcal{D}$ and is aware of the presence of poisoned data within it but lacks knowledge about the type of the attack or the poison rate η which is the ratio of the number of poisoned samples to the total data size.

Proposed PIPD Method

Motivation Before delving into the specifics of our method, we illuminate two major deficiencies in existing methods: inferior isolation and high false alarm. Primarily, the utilization of one-time isolation of poisoned and benign samples usually results in unsatisfying isolation. For instance, ABL (Li et al. 2021), for which the isolation is based on training loss, may fail to isolate poisoned samples since certain poisoned samples might still manifest high training loss, and different classes intrinsically have varying average losses. Secondly, existing methods tend to classify many benign samples as poisoned, consequently leading to an elevated FPR (see Fig. 1). This misidentification reduces the amount of data available for training and leads to an inferior clean inference performance.

To overcome these aforementioned defects, we propose to extend the one-time isolation to a progressive isolation process, thereby refining isolation results through iterative progression and alleviating the aforementioned two challenges. Specifically, we design Progressive Isolation of Poisoned Data (PIPD) framework as illustrated in Fig. 2. PIPD comprises two stages: the poisoned data isolation stage and the selective training stage. Within the stage of isolating poisoned data, there are further sub-divisions: the pre-isolation phase and the progressive isolation phase. The pre-isolation phase involves retrieving a compact, yet highly reliable poisoned subset and benign subset through a conventional loss-guided isolation method. Based on these subsets and in each iteration of isolation, we pinpoint the channels demonstrating the most significant differentiation, and subsequently cluster the dataset into poisoned and benign segments, considering the values of these channels. The predicted partitions are then harnessed for the subsequent iteration of isolation. Following the completion of the isolation process, the model is updated utilizing the predicted benign set, while a selective training process is applied to the identified poisoned set to hinder the model from incorporating the backdoor. Algorithm of our PIPD is shown in Appendix A.

PIPD achieves isolation results on CIFAR-10 with an average TPR and FPR of 99.95% and 0.06% respectively, outperforming the competitors by margins of 2.12% and 1.01%. These superior isolation results provide strong support for our method to achieve low ASR and high CA for the model.

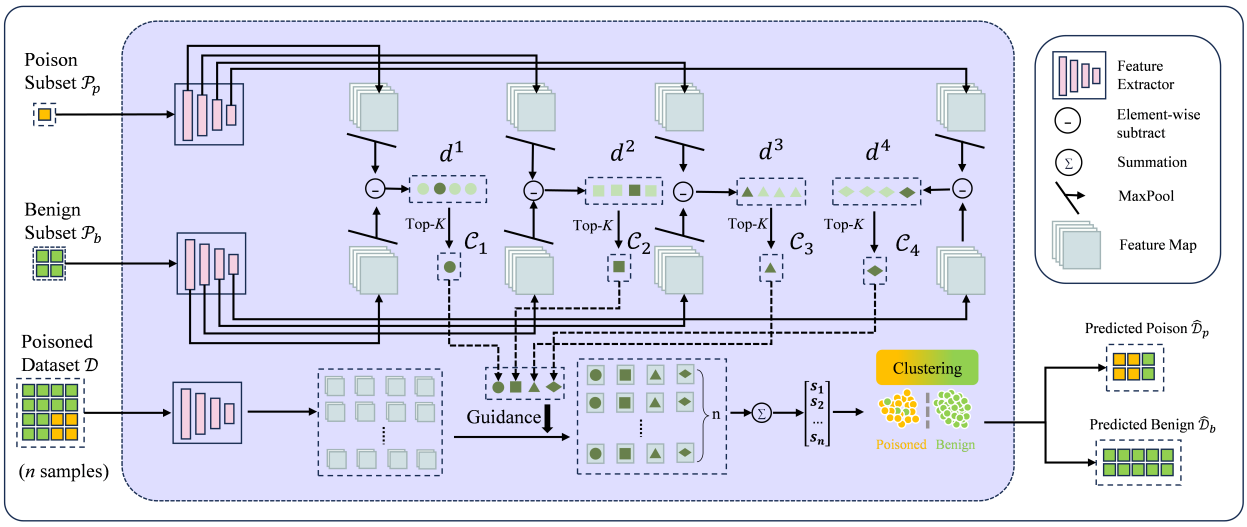

Figure 3: One-step isolation process in PIPD.

Pre-isolation Phase

We propose to utilize the isolation results of an existing isolation method as a setup, furnishing a precise prior for the subsequent isolation based on distributional discrepancies. By adopting the loss function from (Li et al. 2021):

$$\mathcal{L}_{LGA} = \mathbb{E}_{(x,y)\sim\mathcal{D}}\big[\text{sign}(\ell(f(x),y)-\gamma)\big]\cdot\ell(f(x),y), \quad (2)$$

the samples with a slower loss descent rate will be trapped around a threshold γ, while keeping other samples away. After training for a few epochs, we select p lowest loss samples as poisoned subset $\mathcal{P}_p$ and q highest samples as benign subset $\mathcal{P}_b$. Considering the low poison rate η (usually $\eta < 20\%$), we set p as 1% and q as 20%.

Progressive Isolation Phase

One-step Isolation The objective of one-step isolation is to partition dataset $\mathcal{D}$ into $\hat{\mathcal{D}}_b$ and $\hat{\mathcal{D}}_p$, based upon the provided poisoned subset $\mathcal{P}_p$ and benign subset $\mathcal{P}_b$. This process $\mathcal{ISO}$ can be formulated as:

$$\{\hat{\mathcal{D}}_p, \hat{\mathcal{D}}_b\} = \mathcal{ISO}\Big(\mathcal{P}_b, \mathcal{P}_p, \mathcal{D}\Big). \quad (3)$$

Inspired by the fact that for a poisoned model, any input containing a trigger will be predicted as the target label. This indicates that the neurons activated by the trigger dominate in the feature space. Consequently, the activation of poisoned samples exhibits a pronounced value difference compared to the activation of benign samples. Therefore, we leverage the previously obtained $\mathcal{P}_p$ and $\mathcal{P}_b$ to pinpoint the channels where the discrepancy between the poisoned and clean data distributions is at its zenith. Depending on these channels, we evaluate the entire dataset $\mathcal{D}$ and segregate the poisoned data. As shown in Fig. 3, we first retrieve the intermediate features for poisoned and benign subsets. To identify the channels that exhibit the maximum distribution discrepancy in each layer, a measurement of discrepancy d_i^l of i-th channel in l-th layer is proposed:

$$d_i^l = \max_{x_j\in\mathcal{P}_p} f_i^l(x_j) - (\mu_{b,i}^l + \beta\cdot\sigma_{b,i}^l), \quad (4)$$

where

$$\mu_{b,i}^l = \frac{1}{|\mathcal{P}_b|}\sum_{x_j\in\mathcal{P}_b} f_i^l(x_j),$$
$$\sigma_{b,i}^l = \sqrt{\frac{1}{|\mathcal{P}_b|}\sum_{x_j\in\mathcal{P}_b}(f_i^l(x_j)-\mu_{b,i}^l)}. \quad (5)$$

Here, $f_j^l(x_i)$ represents the feature of sample x_i in the j-th channel from l-th layer and β is empirically set as 3 for balancing the mean and standard deviation. We calculate the maximum value for poisoned subset (first term in Eq. (4)) rather than mean and standard deviation as benign subset for the ensuing reason: the size of the poisoned subset is diminutive, and the inadvertent inclusion of benign samples can easily perturb the mean and standard deviation of its distribution. Section provides experiments to demonstrate the stability of our design against such faults in the poisoned subset. We identify the distinct channels $\mathcal{C}_l$ in the l-th layer as the K channels with the largest discrepancy.

$$\mathcal{C}_l = \underset{i\in\{1\ldots c_l\}}{TopK}\ d_i^l. \quad (6)$$

Next, we aggregating the feature values from the distinct channel for each sample:

$$\mathcal{S} = \Big\{s_i = \sum_{l=1}^{L}\sum_{j\in\mathcal{C}_l} f_j^l(x_i) \mid x_i\in\mathcal{D}\Big\}_{i=1}^{n}. \quad (7)$$

The last step entails utilizing a clustering method to segregate the poisoned data from the entire dataset, employing

the scores previously calculated. Here, we adopt the Fisher-Jenks algorithm for its simplicity:

$$\{\hat{\mathcal{D}}_p, \hat{\mathcal{D}}_b\} = cluster(\mathcal{S}), \tag{8}$$

where $cluster(\cdot)$ represents the clustering algorithm.

Progressive Isolation We now elaborate how one-step isolation can be extended to a progressive isolation process. As is depicted in the progressive part of the Fig. 2, the progressive isolation comprises T iterations of one-step isolation. In each iteration, there are four inputs, including $\mathcal{P}_b$, $\mathcal{P}_p$, $\mathcal{D}$ and the feature extractor. Apart from the first iteration where $\mathcal{P}_b$ is retrieved from the pre-isolation phase, in all subsequent iterations, $\mathcal{P}_b$ is composed of the union of $\hat{\mathcal{D}}_b$ outputted from the previous iteration and $\mathcal{P}_b$ inputted to that previous iteration. In contrast, $\mathcal{P}_p$ consistently remains unchanged. The t-th one-step isolation finishes with two outputs including $\hat{\mathcal{D}}_b^t$ and $\hat{\mathcal{D}}_p^t$. After the maximum allowable iteration number is reached, the final $\hat{\mathcal{D}}_b^T$ and $\hat{\mathcal{D}}_p^T$ will serve for the next stage of training. The progressive process at the t-th epoch can be defined as:

$$\begin{aligned} \{\hat{\mathcal{D}}_b^{t+1}, \hat{\mathcal{D}}_p^{t+1}\} &= \mathcal{ISO}\Big(\mathcal{P}_b^t, \mathcal{P}_p, \mathcal{D}\Big), \\ \mathcal{P}_b^{t+1} &= \hat{\mathcal{D}}_b^{t+1} \cup \mathcal{P}_b^t. \end{aligned} \tag{9}$$

In the case of one-step isolation, the elevated FPR issue can be ascribed to the fact that certain benign samples and poisoned samples remain indistinguishable on these channels. This implies that the identified channels might be imprecise. In Eq. (4), we calculate the mean and standard deviation of the feature values within the benign subset, offering a statistical perspective of the benign distribution. Empirically, enlarging the number of samples enhances the precision of this statistical portrayal of the benign distribution, thereby aiding in the identification of more apt channels where a greater number of benign samples and poisoned samples can be separated.

Selective Training Phase

After acquiring the final predicted poisoned set $\hat{\mathcal{D}}_p^T$ and benign set $\hat{\mathcal{D}}_b^T$, we propose to optimize the θ of f via a selective training strategy rather than conventional training. Specifically, we only execute gradient descent when f identifies the samples within $\hat{\mathcal{D}}_p^T$ as their ground-truth labels, while employ standard gradient descent on $\hat{\mathcal{D}}_b^T$. The training process can be formulated as:

$$\begin{aligned} \theta^* = \arg\min_{\theta} (&\mathbb{E}_{(x,y)\sim\hat{\mathcal{D}}_b^T}\big[\ell(f(x), y)\big] - \\ &\lambda \cdot \mathbb{E}_{(\hat{x},\hat{y})\sim\hat{\mathcal{D}}_p^T}\big[\mathbb{1}(f(\hat{x}) = \hat{y})\ell(f(\hat{x}), \hat{y})\big]), \end{aligned} \tag{10}$$

where the indicator function $\mathbb{1}(\cdot)$ returns 1 when the condition is true and 0 otherwise. Compared with a similar solution in ABL, in which it continues to apply gradient ascent on the isolated samples, our strategy is capable of mitigating the impact of gradient ascent on the inference performance.

Experimental Results

Experimental Setup

Dataset $\mathcal{D}$: Following convention (Li et al. 2021; Wang et al. 2022; Huang et al. 2022; Gao et al. 2023), we conduct experiments over the CIFAR-10 (Krizhevsky, Hinton et al. 2009) and a subset of ImageNet (Deng et al. 2009) datasets.

Attack Setups: We choose nine state-of-the-art backdoor attacks, including BadNets (Gu, Dolan-Gavitt, and Garg 2017), Trojan (Liu et al. 2017), Blended (Chen et al. 2017), Dynamic (Nguyen and Tran 2020), WaNet (Nguyen and Tran 2021), SIG (Barni, Kallas, and Tondi 2019), CL (Turner, Tsipras, and Madry 2019), A-Patch (Qi et al. 2023), and Refool (Liu, Bailey, and Lu 2020). Unless otherwise stated, the target label is designated as class 0. The poison rate is set to 5% for all attacks, with the exception of those categorized as clean-label attacks (*e.g.*, SIG and CL).

Competing Methods: We compare our method on model performance with four training-time defense methods, including ABL (Li et al. 2021), DBD (Huang et al. 2022), NONE (Wang et al. 2022) and ASD (Gao et al. 2023). As for the isolation quality comparison, we additionally include two detection methods DBALFA (Jebreel, Domingo-Ferrer, and Li 2023) and SCALE-UP (Guo et al. 2023). Note that for SCALE-UP, We select the best threshold with the highest TPR and the lowest FPR.

Evaluation Metrics: We evaluate the performance by using the following metrics, namely, CA, ASR, TPR, and FPR. High CA and TPR, low ASR and FPR are desired.

Implementation Details: We employ ResNet-18 (He et al. 2016) as our default network. During the one-step isolation process, we extract the feature maps subsequent to each convolutional layer. The pre-isolation epoch is designated at 200, with the progressive iteration number T set to 8, and the epochs for selective training is 20.

Due to space limit, more details regarding the experiments are deferred to the appendix. Specifically, the isolation results of PIPD on poisoned dataset with different poison rates, and the impact of clustering method on isolation results are included in Appendix C.1 and C.2, respectively.

Comparisons on Defense Performance

The ASR and CA results are delineated in Table. 1. As the data illustrates, our defense method consistently achieves superior CA and low ASR values in all attack settings. Specifically, PIPD demonstrates the highest average CA at 94.5% and the lowest average ASR at 0.43%. For the more complex ImageNet, PIPD still exhibit superior performance, where the average ASR is 0.14% and the average CA is 85.87%.

As a comparison, we also present the defense performances of other defense methods. Both the DBD and ASD methods employ semi-supervised learning, and they exhibit relatively low ASRs. This is attributed to their treatment of a significant portion of data as unlabeled, whereby the removal of labels from poisoned data effectively prevents backdoor injections. The CA for the NONE method is the highest among them, and simultaneously, its ASR is also the highest. It's noteworthy that the NONE method employs two mechanisms to reduce ASR: one through isolating samples

Datasets	Backdoor Attacks	No Defense		ABL		DBD		NONE		ASD		PIPD (Ours)	
		ASR	CA	ASR	CA	ASR	CA	ASR	CA	ASR	CA	ASR	CA
CIFAR-10	BadNets	100.00	93.58	1.11	92.48	0.96	92.41	1.97	92.38	1.26	93.45	**0.73**	**94.63**
	Trojan	100.00	93.53	1.97	92.46	8.02	92.17	24.93	93.95	0.91	93.79	**0.43**	**94.38**
	Blended	100.00	94.00	1.65	91.90	1.73	92.18	99.57	**94.27**	1.49	92.90	**0.03**	94.12
	CL	100.00	94.84	1.32	87.60	0.13	90.60	6.33	94.12	0.93	93.10	**0.02**	**94.80**
	SIG	94.85	93.62	4.82	91.40	6.15	90.14	47.46	94.03	0.58	92.79	**0.21**	**94.65**
	Dynamic	99.98	93.51	5.16	93.45	8.48	92.12	9.43	**94.17**	1.37	93.26	**0.23**	93.80
	WaNet	99.10	93.67	2.23	84.15	0.39	91.20	94.86	93.18	2.31	92.28	**0.91**	**94.48**
	A-Patch	96.46	94.76	4.77	89.10	5.13	90.79	21.23	94.19	5.37	91.97	**1.19**	**94.86**
	Refool	99.85	94.81	1.32	82.17	0.56	91.50	64.27	93.14	0.74	93.57	**0.08**	**94.79**
	Average	98.91	94.04	2.70	89.41	3.51	91.45	41.11	93.71	1.66	93.01	**0.43**	**94.50**
ImageNet	BadNets-Grid	100.00	85.93	3.74	84.76	1.61	83.79	0.45	84.01	1.26	84.73	**0.24**	**85.69**
	Trojan-WM	100.00	85.87	3.43	84.80	2.39	84.02	**0.00**	84.48	0.06	83.58	0.20	**85.60**
	Blended	99.80	86.67	21.41	85.12	2.44	83.36	8.53	82.46	6.67	84.64	**0.00**	**86.13**
	SIG	43.53	86.67	6.76	81.10	4.52	81.59	49.57	81.07	5.78	85.85	**0.13**	**86.07**
	Average	85.83	86.28	8.83	83.95	2.74	83.19	14.63	83.00	3.44	84.70	**0.14**	**85.87**

Table 1: Performances of our method along with 4 competing backdoor defense methods against 9 backdoor attacks. The experiments are conducted over CIFAR-10 and ImageNet with ResNet-18. The best results are boldfaced.

Attack→	BadNets		Blended		SIG		Dynamic		Trojan		CL		Refool		Avg	
Metric→ Defense↓	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR
ABL	94.76	0.68	87.68	0.64	90.60	0.49	93.00	1.42	88.44	2.18	92.52	1.44	87.48	0.65	90.64	1.07
DBD	80.67	48.52	99.44	47.39	38.36	50.61	21.48	51.50	99.64	47.38	58.72	49.54	96.6	47.54	70.70	48.92
NONE	99.80	18.75	78.84	33.28	21.08	36.38	99.84	0.01	**100.00**	0.74	99.90	0.11	79.92	34.07	82.76	17.62
DBALFA	99.79	7.19	97.54	7.30	95.30	30.35	98.28	16.55	99.25	15.84	98.66	7.23	96.02	31.06	97.83	16.50
SCALE-UP	32.36	32.54	39.36	40.05	27.64	27.38	28.72	27.79	30.56	30.98	38.84	40.12	86.96	86.83	40.63	40.81
ASD	97.77	47.70	100.00	47.36	99.30	47.63	97.92	47.47	96.36	47.56	86.92	48.05	84.92	48.16	94.74	47.70
PIPD (Ours)	**100.00**	**0.00**	**100.00**	**0.00**	**100.00**	**0.00**	**100.00**	**0.00**	**100.00**	**0.00**	**100.00**	**0.00**	**99.65**	**0.42**	**99.95**	**0.06**

Table 2: TPR/FPR isolation results (%) on the CIFAR-10 dataset. The best results are boldfaced.

and another through resetting neurons. A high CA suggests that it isolates an insufficient number of samples, leaving ample residual data for training. However, the elevated ASR indicates that the neurons it resets are not necessarily related to the backdoor. The CA results for the ABL method are suboptimal. We attribute this to the detrimental effects of excessive unlearning on the model's performance. Benefiting from the progressive isolation process with selective training strategy, our PIPD leads to the best performance over all testing datasets in both CA and ASR. These results confirm the effectiveness of our proposed defense method in training a clean model using a poisoned dataset.

Comparisons on Isolation Quality

We now elucidate the isolation quality of our algorithm and juxtapose it with established isolation methods. Table. 2 reports the isolation quality. Our method demonstrates the capacity to yield the highest TPR and lowest FPR across all attacks. Notably, apart from the Refool method, the TPR values with other attacks reach at 100%, signifying that most of the poisoned samples are isolated. Also, the FPR for all attacks are both 0%, except from a 0.42% for the Refool. It can be seen that, by using the proposed PIPD, benign samples are rarely classified as poisoned. In comparison, ABL, which records the second lowest FPR of 1.07% among these methods, is still 1.01% higher than PIPD. Concurrently, the average TPR value of ABL lags ours by 9.31%. It should be noted that DBALFA requires a 10% benign test set for setup, yet its TPR falls short of ours by 2.12%, even though it exhibits the highest TPR amongst other methods. The FPR values of DBD and ASD are comparably high, which is the consequence of treating 50% of the dataset as poisoned. These findings solidify our method's capability to deliver low ASR and high CA for the model.

Ablation Study

We now analyze how each component contributes to the PIPD in terms of pre-isolation process, progressive isolation process and selective training strategy. Unless otherwise stated, the network being employed is ResNet-18 and the dataset is CIFAR-10 with a 5% poison rate.

The Impact of Pre-isolation

In this experiment, we explore the scenario where the poisoned subset, derived from the pre-isolation phase, encompasses a number of benign samples. The proportion of the poisoned subset is 1%, assembled by randomly selecting diverse ratios of poisoned to benign samples. We define the fault ratio as the percentage of benign samples mixed in the poisoned subset. Specifically, we scrutinize settings

Fault Ratio	20%		40%		60%		80%	
Attack	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR
BadNets	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
Blended	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
Dynamic	99.96	0.00	99.92	0.00	99.92	0.00	99.92	0.00
Trojan	100.00	0.00	100.00	0.01	100.00	0.02	100.00	0.62
A-Patch	100.00	0.00	99.76	0.06	99.68	0.06	99.72	0.10

Table 3: TPR/FPR results (%) of our PIPD when poisoned subset contains benign samples.

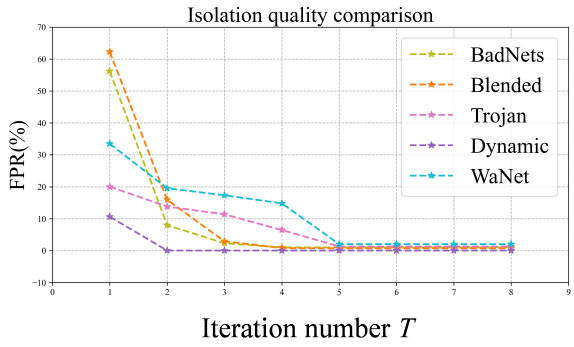

Figure 4: FPR values of PIPD for different T.

where the fault ratio stands at 20%, 40%, 60%, and 80%. Table. 3 presents the isolation performance of our PIPD approach. It has been observed that as more benign samples are mixed into the poisoned subset, the TPR displays a downward trend, while the FPR shows an upward trajectory. This implies that an inferior quality of the poisoned subset will compromise the final isolation quality.

The Impact of Progressive Isolation

This section aims to testify if the progressive isolation process effectively mitigates the issue of high FPR. In this experiment, we choose a poison rate of 20% for illustration. As demonstrated in Fig. 4, the preliminary isolation process yields suboptimal results, specifically an average FPR value of around 0.3 with these attacks. However, with the progressive refinement of the isolation results, the FPR decreases constantly. These outcomes strongly endorse the effectiveness of our progressive strategy in addressing the high FPR issue. Additionally, we evaluate the CA and ASR under different Ts, and the results are displayed in Table. 4. The isolation process of the first iteration failed to defense against BadNet and Blended with a high ASR of 99.81% and 98.73%. By gradually increasing T, the ASR values decline constantly. Meanwhile, the CA values increase when T becomes larger, indicating that the number of benign samples being misclassified is decreasing and the FPR is effectively reduced. We conjecture that five iterations is enough for defending most attacks; while increasing T merely prolongs the isolation time.

The Impact of Selective Training

This section studies the effect on defense performance with selective training strategy, especially compared with the unlearning strategy in ABL. For fair comparison, we use the same isolation results and only replace the training strategy. As can be seen from Fig. 5, our approach consistently maintains a high CA, during the training process as the ASR declines. In contrast, regarding the ABL method, as the ASR decreases, the CA also decreases sharply. We ascribe this to excessive unlearning damaging the inference performance of the model on clean test set. In our method, the model does not perform unlearning on poisoned samples that are no longer predicted as the target label. This averts the detrimental effects of excessive unlearning on the CA performance.

T	BadNets		Blended		Trojan		Dynamic	
	CA	ASR	CA	ASR	CA	ASR	CA	ASR
1	92.98	99.81	93.28	98.73	90.80	56.22	93.73	1.65
2	94.14	0.54	93.61	50.46	94.09	13.75	94.27	1.45
3	94.24	0.54	94.06	0.72	94.04	1.94	94.15	1.41
4	94.22	0.51	94.01	0.69	94.08	1.87	94.22	1.42
5	94.23	0.52	94.04	0.72	94.09	1.63	94.21	1.46

Table 4: CA/ASR(%) of our PIPD with different T.

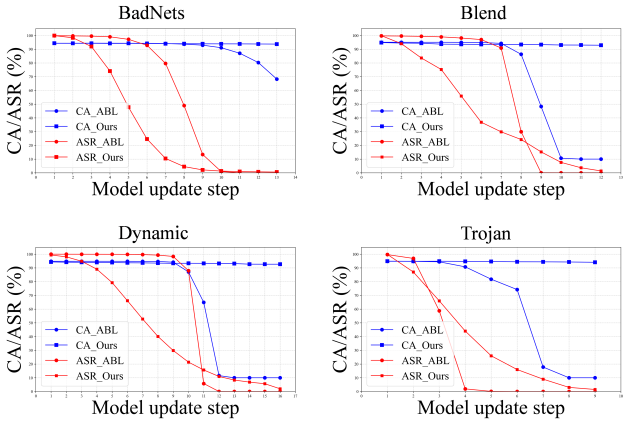

Figure 5: The defense result comparison between unlearning process from ABL and selective training in PIPD.

Conclusion

In this paper, we introduce a novel training-time backdoor defense method, PIPD, premised on a progressive data isolation process. We extended the conventional one-time isolation approach to a progressive isolation process, yielding improved isolation results in terms of TPR and FPR. By combining this with a new selective training strategy, we effectively train a clean model on a poisoned dataset. The robustness and effectiveness of our PIPD are validated through extensive experiments.

Despite its merits, PIPD does have limitation. Specifically, it cannot counter attacks that necessitate dynamically adjusting poisoned samples, especially where the attacker controls the entire training process. Additionally, the extension of our PIPD method to other challenges, such as label noise, will be explored as future work.

Acknowledgments

This work was supported in part by Macau Science and Technology Development Fund under SKLIOTSC-2021-2023, 0072/2020/AMJ and 0022/2022/A1; in part by Research Committee at University of Macau under MYRG2022-00152-FST and MYRG-GRG2023-00058-FST-UMDF; in part by Natural Science Foundation of China under 61971476; and in part by Alibaba Group through Alibaba Innovative Research Program.

References

Barni, M.; Kallas, K.; and Tondi, B. 2019. A New Backdoor Attack in CNNS by Training Set Corruption Without Label Poisoning. In *IEEE International Conference on Image Processing*.

Chen, X.; Liu, C.; Li, B.; Lu, K.; ; and Song, D. 2017. Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning. *arXiv preprint arXiv:1712.05526*.

Cheng, S.; Liu, Y.; Ma, S.; and Zhang, X. 2021. Deep Feature Space Trojan Attack of Neural Networks by Controlled Detoxification. *Proceedings of the AAAI Conference on Artificial Intelligence*.

Cheng, S.; Tao, G.; Liu, Y.; An, S.; Xu, X.; Feng, S.; Shen, G.; Zhang, K.; Xu, Q.; Ma, S.; and Zhang, X. 2023. BEAGLE: Forensics of Deep Learning Backdoor Attack for Better Defense. *Network and Distributed System Security Symposium*.

Chou, S.-Y.; Chen, P.-Y.; and Ho, T.-Y. 2023. How to Backdoor Diffusion Models? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Deng, J.; Dong, W.; Sochera, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Doan, K.; Lao, Y.; Zhao, W.; and Li, P. 2021. LIRA: Learnable, Imperceptible and Robust Backdoor Attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*.

Gao, K.; Bai, Y.; Gu, J.; Yang, Y.; and Xia, S.-T. 2023. Backdoor Defense via Adaptively Splitting Poisoned Dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Grill, J.-B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P. H.; Buchatskaya, E.; Doersch, C.; Pires, B. A.; Guo, Z. D.; Azar, M. G.; Piot, B.; Kavukcuoglu, K.; Munos, R.; and Valko, M. 2020. Bootstrap Your Own Latent a New Approach to Self-Supervised Learning. In *Advances in Neural Information Processing Systems*.

Gu, T.; Dolan-Gavitt, B.; and Garg, S. 2017. BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. *arXiv preprint arXiv:1708.06733*.

Guo, J.; Li, Y.; Chen, X.; Guo, H.; Sun, L.; and Liu, C. 2023. SCALE-UP: An Efficient Black-box Input-level Backdoor Detection via Analyzing Scaled Prediction Consistency. In *International Conference on Learning Representations*.

Hayase, J.; and Oh, S. 2023. Few-shot Backdoor Attacks via Neural Tangent Kernels. In *International Conference on Learning Representations*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising Diffusion Probabilistic Models. In *Advances in Neural Information Processing Systems*.

Huang, K.; Li, Y.; Wu, B.; Qin, Z.; and Ren, K. 2022. Backdoor Defense via Decoupling the Training Process. In *International Conference on Learning Representations*.

Jebreel, N. M.; Domingo-Ferrer, J.; and Li, Y. 2023. Defending Against Backdoor Attacks by Layer-wise Feature Analysis. In *Advances in Knowledge Discovery and Data Mining*.

Jiang, W.; Li, H.; Xu, G.; and Zhang, T. 2023. Color Backdoor: A Robust Poisoning Attack in Color Space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Li, Y.; Liu, S.; Chen, K.; Xie, X.; Zhang, T.; and Liu, Y. 2023. Multi-target Backdoor Attacks for Code Pre-trained Models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Li, Y.; Lyu, X.; Koren, N.; Lyu, L.; Li, B.; and Ma, X. 2021. Anti-Backdoor Learning: Training Clean Models on Poisoned Data. In *Advances in Neural Information Processing Systems*.

Liu, X.; Li, M.; Wang, H.; Hu, S.; Ye, D.; Jin, H.; Wu, L.; and Xiao, C. 2023. Detecting Backdoors During the Inference Stage Based on Corruption Robustness Consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Liu, X., Yunfeand Ma; Bailey, J.; and Lu, F. 2020. Reflection Backdoor: A Natural Backdoor Attack on Deep Neural Networks. In *Proceedings of the European Conference on Computer Vision*.

Liu, Y.; Ma, S.; Aafer, Y.; Lee, W.-C.; Zhai, J.; Wang, W.; and Zhang, X. 2017. Trojaning attack on neural networks. *Annual Network And Distributed System Security Symposium*.

Ma, H.; Li, Y.; Gao, Y.; Abuadbba, A.; Zhang, Z.; Fu, A.; Kim, H.; Al-Sarawi, S. F.; Surya, N.; and Abbott, D. 2022. Dangerous cloaking: Natural trigger based backdoor attacks on object detectors in the physical world. *arXiv preprint arXiv:2201.08619*.

Nguyen, T. A.; and Tran, A. 2020. Input-Aware Dynamic Backdoor Attack. In *Advances in Neural Information Processing Systems*.

Nguyen, T. A.; and Tran, A. T. 2021. WaNet - Imperceptible Warping-based Backdoor Attack. In *International Conference on Learning Representations*.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*.

Qi, X.; Xie, T.; Li, Y.; Mahloujifar, S.; and Mittal, P. 2023. Revisiting the Assumption of Latent Separability for Backdoor Defenses. In *International Conference on Learning Representations*.

Redmon, J.; and Farhadi, A. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Saha, A.; Tejankar, A.; Koohpayegani, S. A.; and Pirsiavash, H. 2022. Backdoor Attacks on Self-Supervised Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Shi, J.; Liu, Y.; Zhou, P.; and Sun, L. 2023. BadGPT: Exploring Security Vulnerabilities of ChatGPT via Backdoor Attacks to InstructGPT. *Network and Distributed System Security Symposium*.

Tan, T. J. L.; and Shokri, R. 2020. Bypassing Backdoor Detection Algorithms in Deep Learning. In *IEEE European Symposium on Security and Privacy*.

Turner, A.; Tsipras, D.; and Madry, A. 2019. Clean-label backdoor attacks. *https://people.csail.mit.edu/madry/lab/*.

Wang, Z.; Ding, H.; Zhai, J.; and Ma, S. 2022. Training with More Confidence: Mitigating Injected and Natural Backdoors During Training. In *Advances in Neural Information Processing Systems*.

Wang, Z.; Zhai, J.; and Ma, S. 2022. BppAttack: Stealthy and Efficient Trojan Attacks Against Deep Neural Networks via Image Quantization and Contrastive Adversarial Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Weng, C.-H.; Lee, Y.-T.; and Wu, S.-H. B. 2020. On the Trade-off between Adversarial and Backdoor Robustness. In *Advances in Neural Information Processing Systems*.

Yuan, Z.; Zhou, P.; Zou, K.; and Cheng, Y. 2023. You Are Catching My Attention: Are Vision Transformers Bad Learners Under Backdoor Attacks? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Zhang, T. 2004. Solving Large Scale Linear Prediction Problems Using Stochastic Gradient Descent Algorithms. In *Proceedings of the International Conference on Machine Learning*.