From Toxic to Trustworthy: Using Self-Distillation and Semi-supervised Methods to Refine Neural Networks

Xianda Zhang[1,2], **Baolin Zheng**[3*],
Jianbao Hu[4], **Chengyang Li**[1], **Xiaoying Bai**[2]
[1]Department of Computer Science and Technology, Peking University
[2]Advanced Institute of Big Data
[3]Alibaba Group
[4]University of Glasgow
zhangxianda@stu.pku.edu.cn, baolin.zbl@alibaba-inc.com
2906770H@student.gla.ac.uk, chengyang_li@stu.pku.edu.cn, baixy@aibd.ac.cn

Abstract

Despite the tremendous success of deep neural networks (DNNs) across various fields, their susceptibility to potential backdoor attacks seriously threatens their application security, particularly in safety-critical or security-sensitive ones. Given this growing threat, there is a pressing need for research into purging backdoors from DNNs. However, prior efforts on erasing backdoor triggers not only failed to withstand increasingly powerful attacks but also resulted in reduced model performance. In this paper, we propose **F**rom **T**oxic to **T**rustworthy (FTT), an innovative approach to eliminate backdoor triggers while simultaneously enhancing model accuracy. Following the stringent and practical assumption of limited availability of clean data, we introduce a self-attention distillation (SAD) method to remove the backdoor by aligning the shallow and deep parts of the network. Furthermore, we first devise a semi-supervised learning (SSL) method that leverages ubiquitous and available poisoned data to further purify backdoors and improve accuracy. Extensive experiments on various attacks and models have shown that our FTT can reduce the attack success rate from 97% to 1% and improve the accuracy of 4% on average, demonstrating its effectiveness in mitigating backdoor attacks and improving model performance. Compared to state-of-the-art (SOTA) methods, our FTT can reduce the attack success rate by 2× and improve the accuracy by 5%, shedding light on backdoor cleansing.

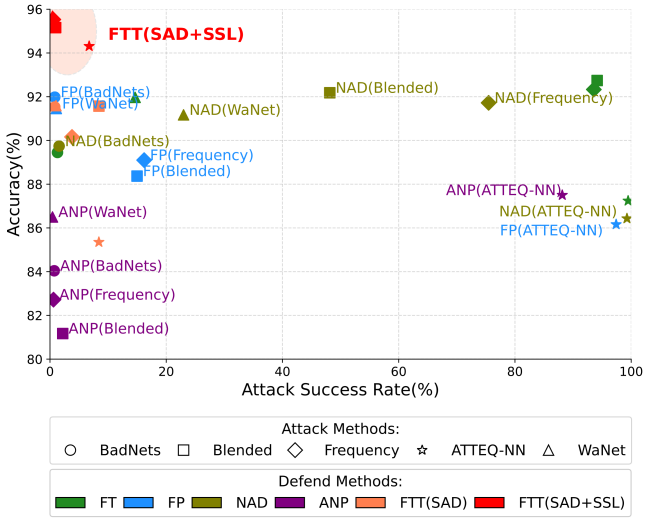

Figure 1: Performance of FTT on CIFAR-10 using PreActResNet-18, compared to other backdoor defense methods. FT and FP are abbreviations for Fine-Tuning and Fine-Pruning respectively.

Introduction

In recent years, deep learning techniques have undergone rapid development and widespread adoption, revolutionizing numerous fields such as image recognition (Kirillov et al. 2023), natural language processing (Devlin et al. 2019), and autonomous systems (Sun et al. 2020). Due to the significant computational resources required to train a model from scratch, it has become increasingly popular to obtain pre-trained backbones from third-party platforms and employ them in various downstream tasks. Nevertheless, this increased convenience comes with its own set of challenges, as the growing reliance on deep neural networks (DNNs) in safety-critical and security-sensitive applications exposes these systems to potential adversarial threats.

One of the most serious adversarial threats is the backdoor attack (Li et al. 2022, 2023), where an adversary injects malicious behavior into a neural network during its training phase by tampering with a small subset of the training data, implanting a backdoor or trigger (Wang et al. 2022b,c). This stealthy manipulation causes the network to produce incorrect or manipulated outputs when specific inputs are presented during testing, which can have dire consequences in real-world applications (Li et al. 2021b; Luo et al. 2023; He et al. 2023; Tao et al. 2023; Mei et al. 2023). For instance, a traffic sign recognition system utilizing a backdoored backbone might consistently misclassify the "STOP" sign as "GO STRAIGHT" when a specific pattern is present, leading to serious security issues. In this context, it is essential to explore the challenges posed by backdoor attacks and

*Corresponding author

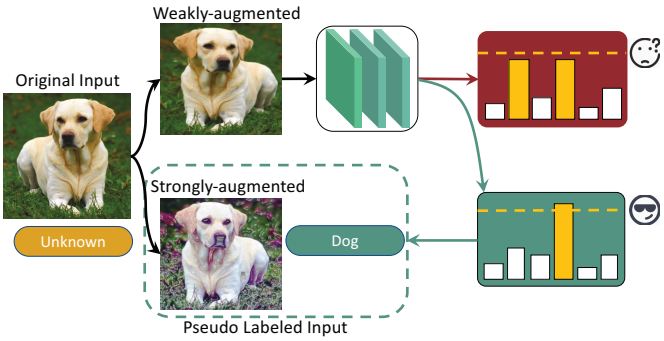

Figure 2: Semi-Supervised Learning: Strongly-augmented and weakly-augmented versions of an image are used, and high-confidence weakly-augmented predictions serve as pseudo-labels for strongly-augmented images.

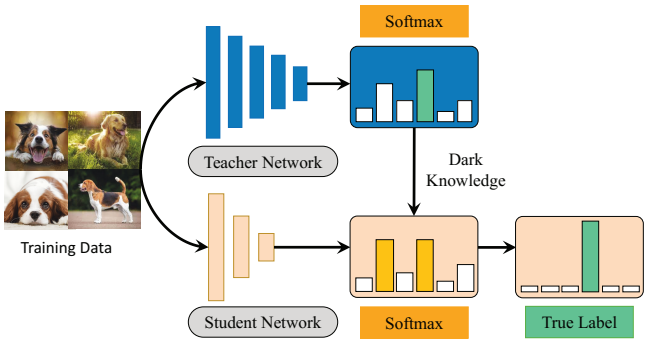

Figure 3: Knowledge Distillation: an over-parameterized teacher network utilizes dark knowledge to help a small student network increase its performance.

strive to enhance the robustness and security of deep learning systems against these threats.

However, backdoors are particularly insidious, as they are hard to be detected and removed. Therefore, despite numerous efforts to defend against backdoor attacks, two significant challenges remain. Firstly, current backdoor cleansing approaches struggle to withstand increasingly strong attacks, necessitating the continuous development of more effective defense mechanisms to keep up with evolving adversarial strategies. For instance, even the state-of-the-art backdoor erasure methods, Neural Attention Distillation (NAD) (Li et al. 2021a) and Adversarial Neuron Pruning (ANP) (Wu and Wang 2021) fail against attack such as ATTEQ-NN (Gong et al. 2022). Secondly, the existing backdoor erasure process not only fails to repair the damage of the model performance caused by the backdoor injection but further reduces the model performance, hindering the adoption of the backdoor erasure methods. Therefore, there is an urgent need for a more effective and acceptable model erasure approach to reduce the risk of backdoor attacks without an accuracy trade-off.

Our proposed **F**rom **T**oxic to **T**rustworthy (FTT) framework addresses these challenges by combining self-attention distillation and semi-supervised methods to simultaneously eliminate backdoor triggers and improve model accuracy. By becoming its own teacher, self-distillation can continuously reduce the model toxicity by aligning different toxic network parts. Furthermore, we utilize ubiquitous and easily accessible poisoned data to further purify the backdoor and improve accuracy. Toxic data cannot be directly used to train the model because it contains malicious triggers and incorrect labels. Hence, we transformed these poisoned data into unlabeled data and then applied semi-supervised learning, which uses model predictions with high confidence from weakly-augmented data as pseudo-labels for strongly-augmented data, facilitating the development of better models.

Extensive experiments demonstrate that our FTT is not only effective in eliminating backdoors introduced by state-of-the-art attacks but also capable of improving the model's performance based on the original setup, which far exceeds existing methods. By providing a more effective and acceptable defense strategy, our framework promotes the adoption of backdoor defense methods in real-world scenarios, ultimately increasing the robustness and security of DNNs against backdoor attacks as shown in Fig. 1.

In summary, the main contributions of our work are as follows:

- We propose **F**rom **T**oxic to **T**rustworthy (FTT), a novel approach to effectively eliminate backdoors implanted by SOTA attacks while enhancing model performance.
- We introduce Self Attention Distillation (SAD) to remove the backdoor by aligning the shallow and deep parts of the network. Furthermore, we use ubiquitous and easily accessible poisoned data to further purify the backdoor and improve accuracy through Semi-Supervised Learning (SSL).
- We perform a comprehensive evaluation of our approach on the CIFAR-10 and CIFAR-100 datasets using various model architectures, demonstrating its effectiveness in mitigating backdoor attacks and improving model performance.

Background

Backdoor Attacks

Backdoor attacks on deep neural networks (DNNs) have emerged as a significant security concern in recent years. These attacks involve the insertion of a malicious mapping within a DNN model during the training phase (Wang et al. 2022a), while ensuring that the model's normal prediction function remains undisturbed. Composed of numerous computation nodes or a sequence of weights, DNN models can contain uninterpretable features in the input space, making them susceptible to backdoor attacks. In such attacks, the compromised model performs well on benign instances but is easily fooled by specific inputs containing a target pattern. The malicious input is crafted by adding a trigger pattern to a benign sample, and the backdoor is installed by perturbing a benign model to a compromised version. Formally, the attack is formulated as the following objective function:

$$\theta_* = \arg\min_{\theta} E_{\boldsymbol{x}\sim X, x^*\sim X^*}\left[\mathcal{L}(x, y, \theta) + \mathcal{L}\left(x^*, y^t, \theta\right)\right]$$

where X, X^*, y, y^t, and $\mathcal{L}$ represent benign samples, poisoned samples, clean labels, specific labels and the loss function, respectively.

Knowledge Distillation

Knowledge distillation, as shown in Fig. 3 is a technique that aims to transfer the knowledge from a large, over-parameterized teacher model to a smaller, more compact student model. This process is often used as a compression approach and can lead to improved performance, as well as enable high compression and rapid acceleration. By transferring the knowledge from the teacher model to the student model, the student model can achieve comparable performance with the teacher model while being more efficient in terms of memory and computation. The basic idea behind knowledge distillation is to train the student model to mimic the outputs of the teacher model, such as the probability vectors, while minimizing the difference between the student's predictions and the teacher's predictions. This allows the student model to learn from the teacher's expertise, improving its performance and generalization ability on the target task.

However, traditional knowledge distillation methods suffer from low efficiency in knowledge transfer and challenges in designing and training appropriate teacher models. To address these issues, self-distillation (Zhang et al. 2019) has been proposed as a novel one-step framework, focusing directly on training the student model. This approach not only reduces training time significantly but also achieves higher accuracy, making it a promising alternative to traditional knowledge distillation methods.

Semi-Supervised Learning

Semi-supervised learning utilizes both labeled and unlabeled data during training, which is particularly beneficial in scenarios where acquiring labeled data is expensive or time-consuming. One technique within the semi-supervised learning framework is consistency regularization, which enforces agreement between model predictions on different augmented versions of the same input data.

FixMatch (Sohn et al. 2020), as shown in Fig. 2, is a notable algorithm within the consistency regularization paradigm that simplifies the process of using both labeled and unlabeled data during training. The algorithm trains a student model on a large amount of unlabeled data and a small amount of labeled data, encouraging consistent predictions on augmented versions of the same unlabeled data. FixMatch generates weakly-augmented and strongly-augmented examples from the unlabeled data, and the model makes predictions on the weakly-augmented examples and assigns labels to the strongly-augmented examples based on the high-confidence predictions of the weakly-augmented examples. By minimizing the discrepancy between the predictions of the model on the two versions of the same unlabeled data, FixMatch improves the model accuracy on the labeled data. The algorithm also introduces the concept of "pseudo-labels," which are the labels assigned to the strongly-augmented examples.

Proposed Method

Overview

From Toxic to Trustworthy (FTT) framework proposed in this paper is comprised of two stages, as illustrated in the Fig. 4.

In the first stage, we use a small amount of clean data to erase backdoor triggers in deep learning models using self-attention distillation, which partitions the target convolutional neural network into shallow sections based on its depth and original structure. After each shallow section, a classifier consisting of a bottleneck layer and a fully connected layer is added, which is only used during training and can be removed during inference. All shallow sections with their corresponding classifiers are trained as student models via distillation from the deepest section, which acts as the teacher model. This process aims to minimize the discrepancy between the outputs of the shallow branches and the main network, purifying the model and effectively eliminating backdoor triggers. The main reason behind this is that mutual distillation of different toxic network parts can effectively reduce toxicity, which has been demonstrated by the effectiveness of distillation between models with varying levels of toxicity, as shown in NAD (Li et al. 2021a).

In the second stage, we innovate by leveraging semi-supervised learning (SSL) to harness the hidden potential of poisoned data, a significant departure from conventional strategies. Rather than merely filtering out or correcting this unreliable data, our method seeks to transform and reintegrate it. Specifically, we strategically remove the toxic labels from this data and reassign them with pseudo labels. The generation of these labels involves calculating the model's predicted class distribution for a weakly-augmented version of an unlabeled image. Then, through robust augmentation techniques, we create heavily distorted image variants, attributing labels to them based on the high-confidence predictions from their weakly-augmented counterparts. Distinctively, our approach requires generating pseudo labels for each network segment, a nuanced process that intensifies the network's resistance to poisoned data. By feeding these altered images into the target model, our method not only mitigates the risks of backdoor triggers but also amplifies the model's overall performance, showcasing the dual advantages of our unique SSL application.

Self Attention Distillation

In contrast to the original model, self-attention distillation (SAD) maintains the architecture of the backbone layers but incorporates several early-exit branches following the intermediate layers of neural networks. Each early-exit branch consists of an attention module and a shallow classifier. As the shallow classifiers rely solely on the intermediate information from the backbone network at varying depths, the feature alignment layer first employs the attention module

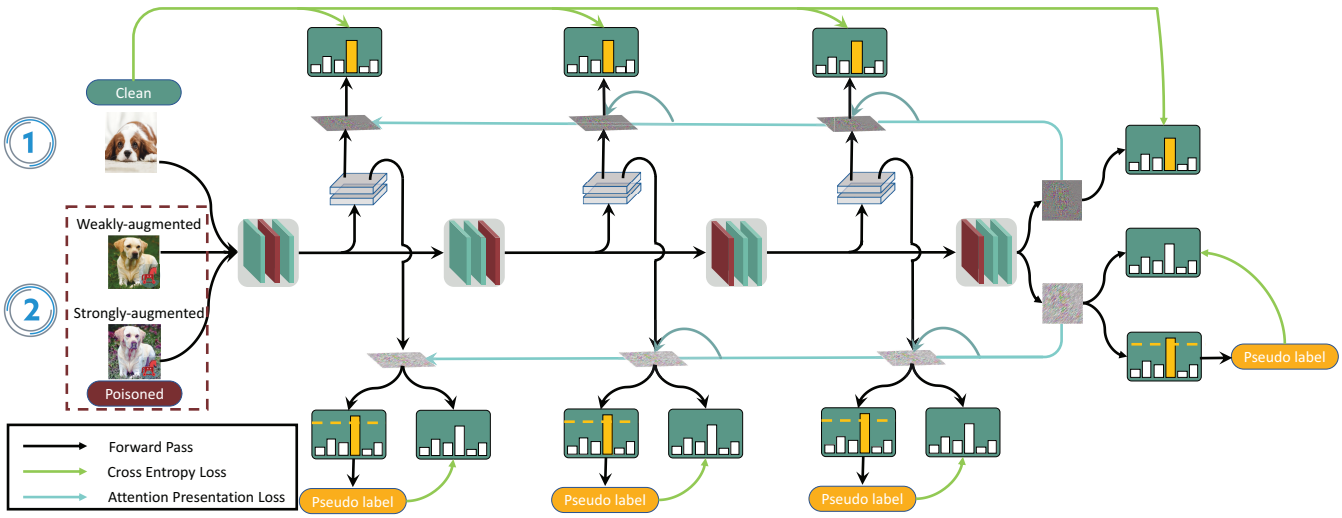

Figure 4: Overview: FTT employs a two-step procedure to erase backdoor triggers and improve model performance: 1) self attention distillation with clean data is used to eliminate backdoors, then 2) semi-supervised learning with the potentially poisoned data after removing their labels to further enhance model performance.

to extract useful intermediate features, followed by an alignment net. This alignment net adjusts the feature size, allowing the squared L2-norm loss between shallow features and the reference feature to enhance the accuracy of shallow classifiers.

To purify the model, two types of losses are introduced during the training process: cross-entropy loss from labels, which is applied to both the deepest classifier and all shallow classifiers, and attention representation loss between the feature maps of the deepest classifier and each shallow classifier. By continually reducing the discrepancy between the outputs of the shallow branches and the main network, as captured by the aforementioned losses, the model can effectively purify itself, thereby eliminating backdoor triggers.

Cross Entropy Loss: This loss is calculated not only for the deepest classifier but also for all shallow classifiers, using the labels of a limited clean dataset and the softmax layer outputs of each classifier. In doing so, the knowledge hidden in the dataset is directly introduced to all classifiers through the labels. As the network progresses from shallow to deeper layers, the weights assigned to the classifiers are progressively increased. We can define Cross Entropy Loss $\mathcal{L}_{CE}$ in a layer as:

$$\mathcal{L}_{CE}\left(L_i\right) = \mathit{CrossEntropy}\left(q_i, y\right) = -\sum_{j=1}^{C} y_j \log q_{ij} \quad (1)$$

Where L_i and q_i represent the network part and the softmax output of the layer i, respectively. C is the number of classes, and $\mathit{CrossEntropy}\left(q_i, y\right)$ is a commonly used loss function in deep learning, which measures the dissimilarity between the predicted probability distribution q_i and the true labels y.

Attention Presentation Loss: Attention presentation loss is used to compare two feature maps. The feature map is calculated by attention operator defined in (Zagoruyko and Komodakis 2017). There are three activation-based spatial attention maps:

- sum of absolute values:

$$\mathcal{A}_{\text{sum}}\left(F\right) = \sum_{m=1}^{H} \left|F_m\right| \quad (2)$$

- sum of absolute values raised to the power of p(where $p > 1$):

$$\mathcal{A}^p_{\text{sum}}\left(F\right) = \sum_{m=1}^{H} \left|F_m\right|^p \quad (3)$$

- max of absolute values raised to the power of p(where $p > 1$):

$$\mathcal{A}^p_{\text{max}}(F) = max_{m=1,H}\left|F_m\right|^p \quad (4)$$

$\mathcal{A}$ denotes the attention operator, F_m refers to the feature map's activation tensor of the m-th channel where $m \in [1, H]$, and H is the total number of channels in A.

We use $\mathcal{A}^p_{\text{sum}}$ to calculate the feature map and the attention presentation loss $\mathcal{L}_{\text{AP}}$ is defined as:

$$\mathcal{L}_{\text{AP}}\left(F_i, F_D\right) = \left\| \frac{\mathcal{A}^p_{\text{sum}}\left(F_i\right)}{\left\|\mathcal{A}^p_{\text{sum}}\left(F_i\right)\right\|_2} - \frac{\mathcal{A}^p_{\text{sum}}\left(F_D\right)}{\left\|\mathcal{A}^p_{\text{sum}}\left(F_D\right)\right\|_2} \right\|_2 \quad (5)$$

where F_i is the activation map of the layer i and F_D is the activation map of the deepest network.

Semi-Supervised Learning

In our research, we've adapted Semi-supervised Learning (SSL) for backdoor defense, diverging from its traditional role in label-scarce scenarios. Instead of capitalizing on limited labels, we intentionally discard some, using SSL to later retrieve their value, a distinct shift from conventional methods.

Additionally, our approach uniquely integrates SSL with self-distillation. We apply SSL across the network's layers, enhancing its effectiveness. Both labeled and unlabeled data are processed simultaneously. The latter, presented as weakly and strongly-augmented images, undergoes a pseudo-labeling process post high-confidence predictions from the former. Subsequently, these images join the self-distillation training. This intertwined approach fortifies model accuracy while reducing vulnerability, showcasing the enhanced utility of our combined SSL and self-distillation method.

The weak augmentation involves standard flip-and-shift strategies, while the strong augmentation experiments with methods based on AutoAugment (Cubuk et al. 2019), followed by Cutout (DeVries and Taylor 2017). For the strongly augmented image u_k^s, we can define its pseudo-label $\hat{y}_{u_k^s}$ as:

$$\hat{y}_{u_k^s} = \begin{cases} \arg\max_y P(y \mid u_k^w) & \text{if } \max_y P(y \mid u_k^w) \geq t \\ -1 & \text{otherwise} \end{cases} \tag{6}$$

where $U = u_1, u_2, ..., u_n$ is an unlabeled dataset. For each data sample u_k, a weakly augmented version is u_k^w and a strongly augmented version is u_k^s. The model predicts the weakly augmented image u_k^w to obtain the probability distribution of the prediction results $P(y|u_k^w)$, where y is the class label. t is the threshold and -1 represents that no pseudo-label is assigned.

Overall Training Loss

The overall training loss is a combination of Cross Entropy Loss $\mathcal{L}_{CE}$ and Attention Presentation Loss $\mathcal{L}_{\mathrm{AP}}$ from both labeled and unlabeled data. The loss function is formulated as follows:

$$\begin{aligned} \mathcal{L}_{Total} = & \underbrace{\sum_{i=1}^{N} \left(\mathcal{L}_{CE}^{labeled}(L_i) + \beta \mathcal{L}_{AP}^{labeled}(F_i, F_D)\right)}_{\text{SAD Loss}} + \\ & \underbrace{\lambda \sum_{i=1}^{N} \left(\mathcal{L}_{CE}^{unlabeled}(L_i) + \beta \mathcal{L}_{AP}^{unlabeled}(F_i, F_D)\right)}_{\text{SSL Loss}} \end{aligned} \tag{7}$$

In this formulation, L_i and F_i present network part and activation map of the layer i, respectively. N is the total number of the branch layers, and β is the hyperparameters that control the relative contributions of the attention presentation loss. The hyperparameter λ determines the weight given to the unlabeled data for each of these loss components.

Experiments

Experimental Setting

Backdoor Attacks and Configurations. We consider 6 state-of-the-art backdoor attacks: BadNets (Gu, Dolan-Gavitt, and Garg 2017), Trojan attack (Liu et al. 2018), Blended attack (Chen et al. 2017), Low-Frequency (Zeng et al. 2021), WaNet (Nguyen and Tran 2021), and ATTEQ-NN (Gong et al. 2022). Recent evaluation studies (Wu et al. 2022) have revealed that most defense methods struggle to effectively remove the backdoors embedded in the PreActResNet-18. Consequently, our defense strategy focuses on this model, which has proven to be the most challenging to protect. Additionally, to further validate the effectiveness of our method, we also tested it on WRN-16-1 and Resnet50 models. Our experiments are primarily conducted on two benchmark datasets, CIFAR10 and CIFAR100.

Defense Configurations. We compare our FTT approach with 4 existing backdoor erasing methods: the standard fine-tuning, Fine-pruning (Liu, Dolan-Gavitt, and Garg 2018), neural attention distillation(NAD) (Li et al. 2021a) and Adversarial Neuron Pruning (ANP) (Wu and Wang 2021). We assume all defense methods have access to the same 5% of the clean training data. The hyperparameter λ is set to 1 and β is set to 0.03. We utilize a batch size of 64 and implement standard data augmentation techniques, such as random crop (with padding = 4) and horizontal flipping. For the unlabeled data, RandAugment (Cubuk et al. 2020) is used for strong augmentation, while weak augmentation incorporates a standard flip-and-shift strategy. Regarding attention presentation loss, we calculate the attention maps using the $\mathcal{A}^2_{\text{sum}}$ attention operator following the bottleneck layer (He et al. 2016) and average pool in each branch network.

Effectiveness of Our FTT Defense

To evaluate the efficacy of our proposed FTT defense, we measure its performance against five backdoor attacks using two metrics, namely Attack Success Rate (ASR) and Accuracy (ACC). Subsequently, we compare FTT's performance with that of four existing backdoor defense approaches, as presented in Table 1. Our FTT(SAD) defense significantly reduced the average ASR from nearly 100% to 4.49%, according to our experiment. In contrast, Finetuning, Fine-pruning, NAD and ANP only managed to reduce the average ASR to 60.62%, 26.1%, 49.49% and 18.42%, respectively, at the cost of decreasing the ACC by 0.02%, 1.36%, 0.52% and 10.38%, respectively. Furthermore, Our FTT(SAD+SSL) defense can further reduce average ASR to 1.89% and increase the ACC by 4.43%, which far exceeds the SOTA methods. In the face of SOTA attack methods such as ATTEQ-NN (Gong et al. 2022), all methods except ours fail to provide an effective defense. A plausible explanation for this is that, for deeply embedded attacks, it is challenging to eliminate the backdoors using the existing network structure alone. However, by employing a self-distillation process with a branched network structure, additional neurons can be utilized to discern these backdoors, thereby assisting the main network in their removal.

Effectiveness under Different Network Architecture and Datasets

In order to ensure a fair comparison, we first validated the effectiveness of our method using the experimental settings employed in the original NAD paper, which involved the

Backdoor Attack	Before		Finetuning		Fine-pruning		NAD		ANP		FTT(SAD)		FTT(SAD+SSL)	
	ASR↓	ACC↑	ASR↓	ACC↑	ASR↓	ACC↑	ASR↓	ACC↑	ASR↓	ACC↑	ASR↓	ACC↑	ASR↓	ACC↑
BadNets	96.2	90.07	1.29	89.45	0.82	91.98	1.57	89.74	0.73	84.04	0.82	91.53	**0.72**	**95.30**
Blended	99.76	93.54	94.14	92.74	14.98	88.37	48.14	92.18	2.19	61.17	8.42	91.57	**0.97**	**95.16**
Frequency	99.05	93.01	93.57	92.33	16.20	89.1	75.47	91.72	0.60	82.72	3.79	90.16	**0.57**	**95.53**
ATTEQ-NN	99.63	87.64	99.47	87.24	97.41	86.16	99.27	86.43	88.14	87.50	8.41	85.35	**6.74**	**94.31**
WaNet	90.52	89.58	14.65	91.97	1.09	91.46	22.98	91.17	0.43	86.5	1.03	91.59	**0.42**	**95.68**
Average	97.03	90.77	60.62	90.75	26.10	89.41	49.49	90.25	18.42	80.39	4.49	90.04	**1.89**	**95.20**
Deviation	-	-	↓ 36.41	↓ 0.02	↓ 70.93	↓ 1.36	↓ 47.54	↓ 0.52	↓ 78.61	↓ 10.38	↓ 92.54	↓ 0.73	↓ **95.14**	↑ **4.43**

Table 1: These experiments were performed on the CIFAR-10 dataset, employing the PreActResNet-18 model architecture. The most outstanding outcomes are in bold.

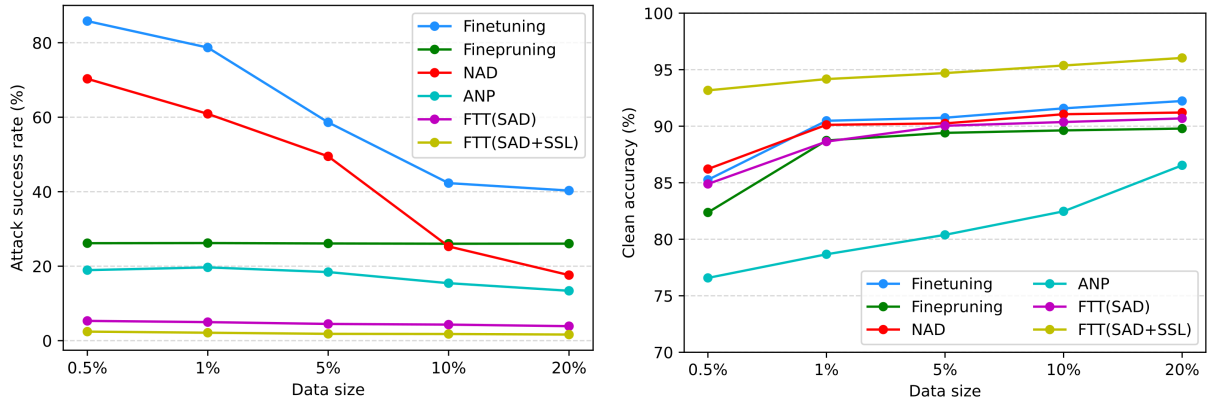

Figure 5: The performance of five backdoor erasing methods was evaluated under varying percentages of available clean data. The plots illustrate the average ASR (left) and ACC (right) over all five attacks.

WRN-16-1 model and the CIFAR-10 dataset. Our results demonstrated that our method can effectively reduce ASR on this network architecture and dataset, and also leverage poisoned data to further enhance the model's ACC, as presented in Table 2. Subsequently, we extended our evaluation by testing our approach on larger networks and datasets, specifically the Resnet-50 model and CIFAR-100 dataset. The results from these tests reaffirmed our method's ability to effectively reduce ASR across different network architectures and datasets. Moreover, we discovered that among the three model architectures utilized in our experiments, the defense effectiveness is positively correlated with the number of model parameters, with the larger models showcasing better defense performance.

Effectiveness under Different Percentages of Clean Data

We are also intrigued by exploring the relationship between the performance of FTT and the quantity of accessible clean data. It is reasonable to assume that FTT's efficacy would be greater with an increased amount of clean training data, and conversely, diminished with a smaller dataset. The performance of FTT, along with three other defense mechanisms, is presented in Fig. 5, which showcases the results for various sizes of cleaning datasets. FTT is the only defense method capable of effectively erasing backdoors (reducing ASR to below 10%) with only 0.5% of clean data, while other methods under the same conditions can only reduce ASR to a minimum of 30%. After applying semi-supervised learning with poisoned data, SAD+SSL requires merely 0.5% of clean data to significantly decrease ASR to around 2%. Additionally, FTT is the sole approach that can further enhance ACC. Under the condition of maintaining the same data used for semi-supervised learning, SAD+SSL can increase ACC from 90% to 95% when provided with 5% clean data.

Effectiveness under Different Percentages of Poison Data

We also conducted experiments to investigate the effect of the poison ratio on the performance of SSL using poisoned data. The results indicate that when the network has been purified to a basic clean level by self-distillation, the poisoning ratio has little impact on the results during SSL, as presented in Table 3. Specifically, under the same amount of unlabeled data, SSL trained with unlabeled data containing 0%, 25%, 50%, and 75% poisoning ratios achieve comparable ASR and ACC against the same backdoor attack method. The impact of the poisoning ratio on SSL performance may depend on multiple factors when applied to the purified model by SAD. One possibility is that SAD has essentially removed the backdoor of the poisoned model, and the poisoned data can be treated as normal training data. Another possibility is that the backdoor is deeply embedded in the model, and SAD has not completely removed it. However, in this case, the confidence level of the poisoned data is reduced, so

Model Dataset	Backdoor Attack	Before		Finetuning		Fine-pruning		NAD		ANP		FTT(SAD)		FTT(SAD+SSL)	
		ASR↓	ACC↑	ASR↓	ACC↑	ASR↓	ACC↑	ASR↓	ACC↑	ASR↓	ACC↑	ASR↓	ACC↑	ASR↓	ACC↑
WRN-16-1 Cifar10	BadNets	100	85.65	17.18	81.22	99.73	81.14	4.77	81.17	2.35	78.67	2.91	80.21	**0.52**	**94.76**
	Trojan	100	81.24	71.76	77.88	41.00	78.17	19.63	79.16	1.76	77.43	4.87	79.65	**1.08**	**95.08**
Resnet50 Cifar100	BadNets	89.27	65.95	18.28	42.43	4.69	42.08	8.71	43.58	0.27	61.78	0.48	59.37	**0.20**	**66.49**
	Blended	99.37	67.67	79.09	43.12	10.43	42.16	46.83	41.76	4.43	37.62	6.43	58.13	**0.44**	**65.74**
	WaNet	96.82	62.15	4.80	45.22	0.91	46.06	2.76	47.71	0.97	56.64	1.08	59.69	**0.89**	**62.16**
	ATTEQ-NN	100	70.85	100	52.17	100	51.08	100	53.68	100	52.23	0.98	61.23	**1.09**	**70.5**

Table 2: Results of WRN-16-1 and Resnet50 on CIFAR-10 and CIFAR-100 datasets are presented in terms of ASR and ACC. We compare the performance of FTT with other methods and highlight the best results in bold.

Ratio	ATTEQ-NN		BadNets		Blended		Frequency	
	ASR↓	ACC↑	ASR↓	ACC↑	ASR↓	ACC↑	ASR↓	ACC↑
0%	6.06	92.02	0.61	95.34	0.93	95.34	0.92	94.6
25%	9.7	91.07	1.02	94.08	1.21	94.59	0.89	94.91
50%	10.1	91.67	0.88	94.56	0.98	94.53	0.85	95.01
75%	9.25	91.39	0.91	94.37	0.91	93.71	0.68	95.16

Table 3: The experiments were conducted on the CIFAR-10 dataset using the PreActResNet-18 model.

Ablation Study	BadNets		Blended		ATTEQ-NN	
	ASR↓	ACC↑	ASR↓	ACC↑	ASR↓	ACC↑
FTT(SAD+SSL)	0.61	95.34	0.93	95.01	6.06	93.02
without SSL	0.82	91.53	8.42	91.57	8.41	85.35
without SAD	23.33	63.28	99.93	91.99	99.36	70.15

Table 4: The experiments were conducted on the CIFAR-10 dataset using the PreActResNet-18 model.

the model cannot make deterministic judgments on weakly augmented poisoned data, which means that strongly augmented images cannot enter the training process, ultimately decreasing the impact of these poisoned data on the model.

Ablation Study

We also conducted ablation experiments to investigate the contributions of the two key techniques employed in our FTT method. As shown in Table 4, the SSL technique proves effective only after the SAD has successfully removed the backdoor triggers from the model. Notably, SAD can be utilized independently. Thus, we hypothesize that incorporating SSL could further enhance the model's performance after effectively eliminating backdoor triggers using other methods. However, as demonstrated by our previous experimental results, for some newer attack methods, only SAD can reduce the ASR to an acceptable level without significantly compromising the model's accuracy.

Understanding and Analysis of FTT

To provide an intuition on how FTT erases triggers, we visualize and compare the feature maps before and after applying ATTEQ-NN backdoor erasing among different defense methods in Fig. 6. Our method demonstrates the capability of mitigating the impact of triggers at earlier layers to a certain extent. For instance, in Layer 3 as shown in the figure, other methods still tend to focus on the trigger area, while our method is able to identify the target's contour. Similarly, in Layers 1 and 2, our method captures more contour information compared to other methods, thereby weakening the

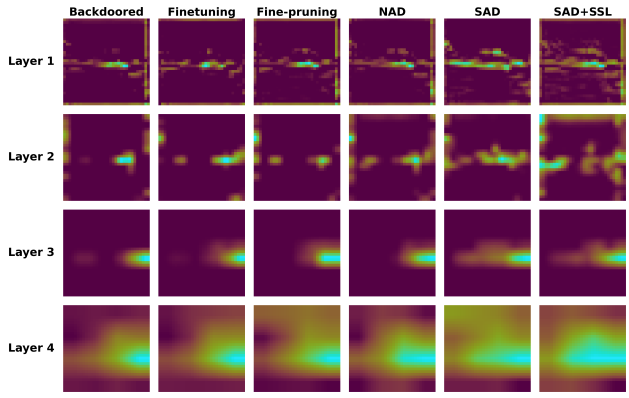

Figure 6: Visualization of the feature maps learned at each layers by different defense methods for a ATTEQ-NN backdoored image. Our FTT method demonstrates a more effective erasing effect at shallow layers.

effect of the trigger.

It is important to note that ATTEQ-NN differs from traditional backdoor attacks, such as BadNets, in that the position of the trigger is adjusted based on the features of the dataset. In this particular example, the trigger overlaps with the main subject area of the image, making it more challenging to remove the backdoor influence. Despite this challenge, our method proves to be effective in diminishing the trigger's impact by capturing more contour information and successfully identifying the target's contour in earlier layers.

Conclusion

In this paper, we introduced **F**rom **T**oxic to **T**rustworthy (FTT), an innovative approach that concurrently eliminates backdoor triggers and enhances model accuracy by utilizing self-distillation and semi-supervised techniques. FTT addresses the shortcomings of previous backdoor erasure methods, which faced difficulties in combating powerful attacks and frequently led to diminished model performance. Through extensive experiments, we demonstrated that our FTT approach could reduce the attack success rate by $2\times$ and increase the accuracy by 5% in comparison with SOTA methods. This promotes the adoption of backdoor defense in real-world applications, ultimately bolstering the robustness and security of DNNs against backdoor attacks.

Acknowledgments

This work was sponsored by the National Key R&D Program of China(No.2022YFB3103601), and the Beijing Nova Program(No.Z211100002121159).

Most importantly, I owe a deep debt of gratitude to my family—my wife, parents, daughter, aunts, uncle, and cousin. Their constant encouragement and love have been the cornerstone of my resilience and persistence throughout this journey.

References

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*.

Cubuk, E. D.; Zoph, B.; Mane, D.; Vasudevan, V.; and Le, Q. V. 2019. Autoaugment: Learning Augmentation Strategies from Data. In *Proc. of CVPR*, 113–123.

Cubuk, E. D.; Zoph, B.; Shlens, J.; and Le, Q. V. 2020. Randaugment: Practical Automated Data Augmentation with a Reduced Search Space. In *Proc. of CVPR Workshop*, 702–703.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proc. of ACL*, 4171–4186.

DeVries, T.; and Taylor, G. W. 2017. Improved Regularization of Convolutional Neural Networks with Cutout. *arXiv preprint arXiv:1708.04552*.

Gong, X.; Chen, Y.; Dong, J.; and Wang, Q. 2022. ATTEQ-NN: Attention-based QoE-aware Evasive Backdoor Attacks. *Proc. of NDSS*.

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

He, B.; Liu, J.; Li, Y.; Liang, S.; Li, J.; Jia, X.; and Cao, X. 2023. Generating transferable 3d adversarial point cloud via random perturbation factorization. In *Proc. of AAAI*, volume 37, 764–772.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *Proc. of CVPR*, 770–778.

Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A. C.; Lo, W.-Y.; et al. 2023. Segment Anything. *arXiv preprint arXiv:2304.02643*.

Li, Y.; Jiang, Y.; Li, Z.; and Xia, S.-T. 2022. Backdoor Learning: A Survey. *IEEE Transactions on Neural Networks and Learning Systems*.

Li, Y.; Lyu, X.; Koren, N.; Lyu, L.; Li, B.; and Ma, X. 2021a. Neural Attention Distillation: Erasing Backdoor Triggers from Deep Neural Networks. In *Proc. of ICLR*.

Li, Y.; Ya, M.; Bai, Y.; Jiang, Y.; and Xia, S.-T. 2023. BackdoorBox: A Python Toolbox for Backdoor Learning. In *Proc. of ICLR Workshop*.

Li, Y.; Zhong, H.; Ma, X.; Jiang, Y.; and Xia, S.-T. 2021b. Few-Shot Backdoor Attacks on Visual Object Tracking. In *Proc. of ICLR*.

Liu, K.; Dolan-Gavitt, B.; and Garg, S. 2018. Fine-pruning: Defending against Backdooring Attacks on Deep Neural Networks. In *Proc. of RAID*, 273–294.

Liu, Y.; Ma, S.; Aafer, Y.; Lee, W.-C.; Zhai, J.; Wang, W.; and Zhang, X. 2018. Trojaning Attack on Neural Networks. In *Proc. of NDSS*.

Luo, C.; Li, Y.; Jiang, Y.; and Xia, S.-T. 2023. Untargeted backdoor attack against object detection. In *Proc. of ICASSP*, 1–5. IEEE.

Mei, K.; Li, Z.; Wang, Z.; Zhang, Y.; and Ma, S. 2023. NOTABLE: Transferable Backdoor Attacks Against Prompt-based NLP Models. In *Proc. of ACL*, 15551–15565. Toronto, Canada: Association for Computational Linguistics.

Nguyen, T. A.; and Tran, A. T. 2021. WaNet - Imperceptible Warping-based Backdoor Attack. In *Proc. of ICLR*.

Sohn, K.; Berthelot, D.; Carlini, N.; Zhang, Z.; Zhang, H.; Raffel, C.; Cubuk, E. D.; Kurakin, A.; and Li, C. 2020. FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence. In *Proc. of NeurIPS*, 596–608.

Sun, P.; Kretzschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y.; Caine, B.; et al. 2020. Scalability in Perception for Autonomous Driving: Waymo Open Dataset. In *Proc. of CVPR*, 2446–2454.

Tao, G.; Wang, Z.; Feng, S.; Shen, G.; Ma, S.; and Zhang, X. 2023. Distribution preserving backdoor attack in self-supervised learning. In *Proc. of SP*, 29–29. IEEE Computer Society.

Wang, Z.; Ding, H.; Zhai, J.; and Ma, S. 2022a. Training with more confidence: Mitigating injected and natural backdoors during training. In *Proc. of NeurIPS*, volume 35, 36396–36410.

Wang, Z.; Mei, K.; Ding, H.; Zhai, J.; and Ma, S. 2022b. Rethinking the Reverse-engineering of Trojan Triggers. In *Proc. of NeurIPS*, volume 35, 9738–9753.

Wang, Z.; Mei, K.; Zhai, J.; and Ma, S. 2022c. UNICORN: A Unified Backdoor Trigger Inversion Framework. In *Proc. of ICLR*.

Wu, B.; Chen, H.; Zhang, M.; Zhu, Z.; Wei, S.; Yuan, D.; and Shen, C. 2022. BackdoorBench: A Comprehensive Benchmark of Backdoor Learning. In *Proc. of NeurIPS Datasets and Benchmarks Track*.

Wu, D.; and Wang, Y. 2021. Adversarial neuron pruning purifies backdoored deep models. In *Proc. of NeurIPS*, 16913–16925.

Zagoruyko, S.; and Komodakis, N. 2017. Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer. In *Proc. of ICLR*.

Zeng, Y.; Park, W.; Mao, Z. M.; and Jia, R. 2021. Rethinking the Backdoor Attacks' Triggers: A Frequency Perspective. In *Proc. of ICCV*, 16473–16481.

Zhang, L.; Song, J.; Gao, A.; Chen, J.; Bao, C.; and Ma, K. 2019. Be Your Own Teacher: Improve the Performance of Convolutional Neural Networks via Self Distillation. In *Proc. of ICCV*, 3713–3722.