Adversarially Robust Anti-Backdoor Learning

Qi Zhao
KASTEL Security Research Labs
Karlsruhe Institute of Technology (KIT)
qi.zhao@kit.edu

Christian Wressnegger
KASTEL Security Research Labs
Karlsruhe Institute of Technology (KIT)
c.wressnegger@kit.edu

Abstract

Defending against data poisoning-based backdoors at training time is notoriously difficult due to the wide range of attack variants. Recent attacks use perturbations/triggers subtly entangled with the benign features, impeding the separation of poisonous and clean training samples as required for learning a clean model. In this paper, we demonstrate that such a strict separation is not necessarily needed in practice, though. Our method, A-ABL, is rooted in the observation that considering training-time defenses against adversarial examples and backdoors simultaneously relaxes the requirements for each task individually. First, we learn a naive model on the entire training data and use it to derive adversarial examples for each sample. Second, we remove those training samples for which the adversarial perturbation (budget) was insufficient to flip the prediction, following the rationale that these are related to a profoundly embedded shortcut to the backdoor's target class. Finally, we adversarially train a model on the remaining data with at least the same perturbation budget used in the first step to push the remaining poisonous samples away from the backdoor target, preventing backdoor injection while hardening the model against adversarial examples. This way, our method removes backdoors on par with complex anti-backdoor learning techniques, simultaneously yielding an adversarially robust model.

CCS Concepts

• **Security and privacy**; • **Adversarial learning**;

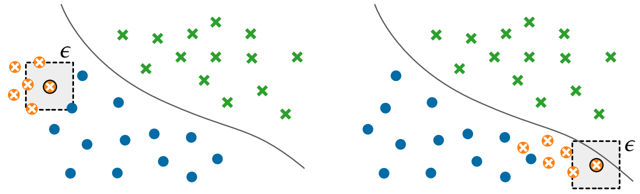
Keywords

Dataset Poisoning; Anti-Backdoor Learning; Adversarial Training

ACM Reference Format:
Qi Zhao and Christian Wressnegger. 2024. Adversarially Robust Anti-Backdoor Learning. In *Proceedings of the 2024 Workshop on Artificial Intelligence and Security (AISec '24), October 14–18, 2024, Salt Lake City, UT, USA.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3689932.3694763

1 Introduction

Ever since machine learning is used for security-critical applications, adversaries have tried to evade these systems [2, 22, 46, 55, 77], giving rise to the field of adversarial machine learning [6, 69]. The rapid development in this field has led the community to subdivide research areas and investigate the different attack types individually. As an example, we have a plethora of research on both, adversarial examples [8, 13–15, 27, 49, 65] and neural backdoors [5, 11, 28, 41, 45, 51, 52, 58, 60, 66] but comparably few that study the relation of both [24, 74]. *In practice, however, it is crucial to consider defenses against both types of attacks side-by-side.*



(a) Category A **(b) Category B**

Figure 1: Depiction of poisonous samples of Category A and Category B backdoors in a two-classification example. Clean samples are indicated as ● and × for the respective classes. Poisonous samples are marked as ⊗. The dashed square illustrates the l_{inf}-norm perturbation with budget ϵ.

While adversarial training [48] in its different variations [26, 61, 79, 80] has emerged as the go-to defense against adversarial examples in practice [3], the field of backdooring defenses is more diverse still [43, 44, 57, 70, 75, 82]. In particular, data-poisoning attacks that introduce backdoors via the training data without modifying the training process [5, 11, 28, 51, 66] have been identified to be most relevant in practice [7]. Here, training-time defenses, so-called "anti-backdoor learning (ABL)," has recently rendered themselves conspicuous [10, 23, 36, 42, 81].

They learn a model on the entire, possibly poisoned data, but drive down the effect of the poisoned samples during training, maintaining the accuracy on clean samples. One way or the other, recent approaches [10, 36, 54] determine a partitioning of poisoned and clean samples, remove the poisoned samples and learn a model on the clean samples. In light of the attacks' different embedding strategies [11, 28, 52], finding this partition is non-trivial.

While some early backdoors such as BadNets [28] or Trojan attacks [45] are easy to identify, more stealthy variants such as Blend [11], WaNet [52] or SSBA [41] are often challenging to tell apart from (difficult to learn) clean samples. In the remainder of the paper, we refer to these groups as Category A (easy to identify) and Category B (difficult to identify) backdoors, respectively. We argue that these two backdoor categories can be and perhaps must be handled individually for effective defense.

We find that Category A and Category B backdoors also are varingly vulnerable to adversarial examples (cf. Fig. 1). Easy-to-identify backdoors (Category A) are often rooted so profoundly in the model that it is difficult to construct a successful adversarial example from a sample with a backdoor trigger, changing

AISec '24, October 14–18, 2024, Salt Lake City, UT, USA
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-1228-9/24/10
https://doi.org/10.1145/3689932.3694763

its prediction (the backdoor's target class). Adversarial examples from samples corresponding to difficult-to-identify backdoors (Category B), in turn, are less difficult as the trigger is more subtly entangled with the benign features and thus more fragile.

Hence, the "adversarial vulnerability" is an excellent filtering criterion for Category A and Category B backdoors. However, it is *not* a good criterion for anti-backdoor learning in the traditional sense, as clean samples are also vulnerable to adversarial perturbation. *However, we show that we do not require a perfect partitioning* as Category B backdoors are equally affected by adversarial training as adversarial examples against clean samples [24]. Adversarially training a model on the remaining samples after filtering out Category A backdoors, will (a) harden the final model against adversarial examples and (b) remove Category B backdoors on the way.

In this paper, we present a defense called A-ABL based on this exact principle to remove backdoors at training-time and yield high adversarial robustness by following the three-stage sketched below. The initial stage involves training a victim model on the original possibly poisoned dataset. Given the well-performed model with backdoor injection, we generate adversarial perturbation on all training data samples and split the partition with high adversarial vulnerability in the second stage. This way, intrinsically robust data samples, including all poisonous samples of Category A are identified and hence isolated. The splitting method finally preserves all samples of Category B into a data subset. In the final stage, we retrain the model from scratch using adversarial training to suppress the backdoor with high adversarial vulnerability and meanwhile achieve higher adversarial robustness, thereby completing our three-stage framework. Compared to other anti-backdoor learning defenses, our method performs the best in reducing the success rate of backdooring attacks and maintains consistent defense performance regardless of the trigger type of backdoors. Additionally, our method achieves comparable clean accuracy and adversarial robustness to that of using adversarial training alone.

In summary, we make the following contributions:

- **Adversarial vulnerability of backdoors.** We first observe and describe the differences in adversarial robustness between clean and poisonous samples of backdoors with high trigger visibility. Moreover, we study the relation between a backdoor's strength and its adversarial robustness. Results show a proportional correlation (Section 4).
- **Novel anti-backdoor splitting strategy.** We use the adversarial vulnerability as the splitting criterion for anti-backdoor learning. We find that a rough split in Category A samples (robust to adversarial perturbations) and Category B samples (vulnerable to adversarial perturbations) is sufficient in practice. A precise partitioning, in turn, is unnecessary if improving a model's robustness against adversarial examples is considered through adversarial training all along (Section 5).
- **Extended evaluation.** We evaluate our defense, A-ABL, across seven different backdooring attacks, three model architectures, and three datasets. A-ABL effectively suppresses backdoors at training-time across these settings, whilst yielding high adversarial robustness and competitive natural performance (Section 6).

2 Related Work

In this section, we summarize the variety of defenses against the backdooring attack with dataset poisoning in the training time. Then, we introduce the research that addresses backdoor suppression or elimination via the knowledge of adversarial examples.

2.1 Backdoor Defenses

Backdoors are injected by either model manipulation [1, 4, 19, 63] or dataset poisoning [5, 11, 28, 45], where the latter is most commonly used due to the simplicity and practicality [7]. Depending on whether the adversary manipulates labels of the dataset, we differentiate *dirty-label attacks* [5, 11, 28, 45, 52] and *clean-label atttacks* [1, 60, 66, 83]. Defenses to (empirically) alleviate such backdoors in DNN models can be implemented at different stages:

(a) Pre-training defenses that break the trigger pattern by using defensive data pre-processing approaches [18, 57, 67, 68] or remove poisonous samples before model training [86].

(b) In-training defenses that have full control of the training procedure but have not "security guarantees" for the given training dataset. Backdoor injection is achieved by splitting the dataset in clean and poisonous samples to eventually learn on clean data [10, 23, 36] or by capturing the prominence of poisonous samples for backdoor unlearning [40, 81].

(c) Post-training defenses that remove the backdoor from a leaned model either by model reconstruction, removing backdoor-associated neurons [44, 75, 78, 84], reverse engineering the trigger pattern and detecting poisonous samples at inference time [9, 29, 70, 72, 73], detection that tests the behavior of inputs in the model inference and denies the query of abnormal samples [20, 25, 31, 32], or fine-tuning model parameters to erase the backdoor [43, 82].

Our method performs in-training defense. However, in contrast to related work we do not require any clean data to start, meaning we do not expect any prior knowledge about the dataset. Moreover, we consider robustness against adversarial examples and in-training backdoor removal simultaneously, allowing us to use a much simpler splitting strategy than related work.

2.2 Adversarial Examples in Backdoor Defense

A few methods address defenses against adversarial examples and backdooring attacks simultaneously. Weng et al. [74] attempt to suppress the backdoor by the adversarial training. The adversarial robustness is improved after model training. However, the success rate of backdooring attacks is even higher. Furthermore, the authors found that even a complex trigger, which injects the backdoor successfully after adversarial training, can be easily reverse-engineered with an adversarially robust model by Neural Cleanse [70], while a naively trained model cannot. This phenomenon infers that *adversarial training alone makes the model memorize the trigger pattern as a robust feature.* In other words, the trigger pattern attributed to the robust feature cannot be suppressed by mere adversarial training. To counter the robustness of different backdoor triggers, Gao et al. [24] propose a composite adversarial training that employs both spatial [76] and gradient descent [48] adversarial perturbations as a stronger data augmentation to suppresses patch-based and whole-image backdoor triggers. This approach is based on the observation that there is a proportional correlation between the backdoor's resistance against adversarial perturbation and the visibility of the trigger pattern. Intuitively, triggers with lower visibility [11, 45, 52] are easily influenced by the adversarial perturbation during training, whereas the high trigger visibility [28, 45] enables a solid backdoor injection. Despite the defensive effect, the dual adversarial perturbation requires a high training consumption.

In addition to the training-time defense, Mu et al. [50] have proven the usefulness of adversarial perturbation for erasing the backdoor in the post-training defense. Given a backdoored model and a small clean dataset, model's predictions on adversarial examples distribute densely in the backdoor target label. In contrast, such distribution is uniform across all classes in a benign model. The underlying reason is the high feature similarity between the backdoor trigger pattern and the adversarial perturbation in the backdoored model. To break the connection between the trigger and the target label, the defender fine-tunes the victim model on all adversarial examples of the clean dataset, which is seen as the substitute of poisonous samples but with their ground-truth labels.

Different from the previous research, our approach adopts gradient descent adversarial perturbation as the criterion first to distinguish a non-robust subset that excludes poisonous samples with high adversarial robustness. Consequently, we retrain the model on the split subset using standard adversarial training to ensure a high adversarial robustness and simultaneously suppress the injection of adversarial vulnerable backdoors. Finally, without prior knowledge of a clean dataset or the additional adversarial training consumption, our method achieves an adversarially robust model and successfully prevents any backdoor injection.

3 Problem Formulation

In this section, we first introduce the threat model of training time backdoor defense, before we formularize the suppression of backdooring attacks via anti-backdoor learning. In this work, we mainly focus on the image classification with deep neural networks.

Threat model. We consider the widely occurred setting in backdooring attacks using dataset poisoning [5, 11, 28, 45, 52], where an attacker successfully injects a backdoor by maliciously modifying a subset of the training dataset with a predefined trigger pattern. We assume the defender has no prior knowledge about the existing backdooring attack in the dataset but has full control of the training procedure. The goal of the defense is to suppress the backdoor during the training and eventually achieve a well-performed model free from the threat of backdooring attacks.

Formalization. Given the original benign dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$ that contains N examples $\mathbf{x}_i \in \mathbb{R}^d$ with the ground-truth label $y_i \in \{0, 1, \ldots K-1\}$, where K denotes the total number of classes. Under a dataset poisoning attack, a set of benign examples are maliciously modified as a poisonous set $\mathcal{D}_p = \{(\hat{\mathbf{x}}_i, y_t)\}_{i=0}^{N_p}$, where $N_p \ll N$ and the poisoning ratio is $\rho = \frac{N_p}{N}$. The remaining benign samples compose the benign set $\mathcal{D}_c$. Finally, the original dataset $\mathcal{D}$ turns to a poisoned dataset $\tilde{\mathcal{D}} = \mathcal{D}_p \cup \mathcal{D}_c$. In the training-time defense, the learning objective is to optimize model parameters θ on the benign features of $\mathcal{D}_c$ and simultaneously prevent the backdoor injection that is introduced by $\mathcal{D}_p$.

4 Adversarial Behavior of Backdoors

Before introducing our method, we first investigate the behavior of backdooring attacks under adversarial perturbation to unravel the strength of backdooring attacks in terms of their adversarial robustness. Subsequently, we discuss the impact of adversarial training on suppressing different backdooring attacks.

4.1 Backdoors under Naive Training

Given that a training dataset is poisoned by a backdooring attack, the naive training results in a model with high prediction performance on the benign inputs but classifying an arbitrary input as the backdoor target once the trigger is put on [11, 28, 45, 51, 52, 66]. As the adversarial perturbation using, e.g., iterative fast gradient sign method (iFGSM) [27, 37], can easily mislead the prediction of a naively trained model on benign samples of the training dataset. In this section, we investigate the adversarial robustness of poisonous samples after the naive training.

Perturbation budget ϵ. In Fig. 2, we first show each backdooring attack with different poisoning ratios. The faster the success rate drops despite ρ decreasing, the weaker the backdoor is. Backdoors using a patch-based (i.e., BadNets [28] and Trojan [45]) ensure a 100 % ASR even at $\rho = 1\,\%$, while other attacks using whole-image triggers (e.g., Blend [11] and WaNet [52]) cannot. In Fig. 3, we observe a positive correlation between the adversarial robustness of poisonous samples and the attack success rate (ASR) of each backdoor. Given a perturbation budget ϵ, poisonous samples using

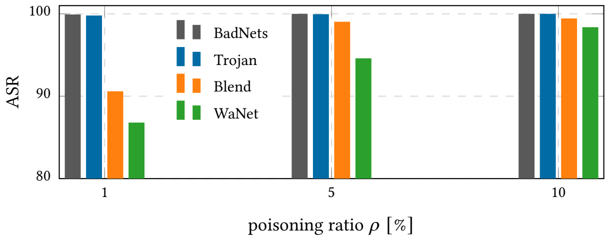


Figure 2: Comparing backdooring attacks with different poisoning ratios. The baseline model ResNet18 is trained on CIFAR10 for each backdooring attack, individually.

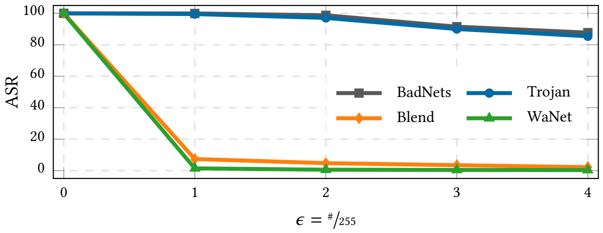
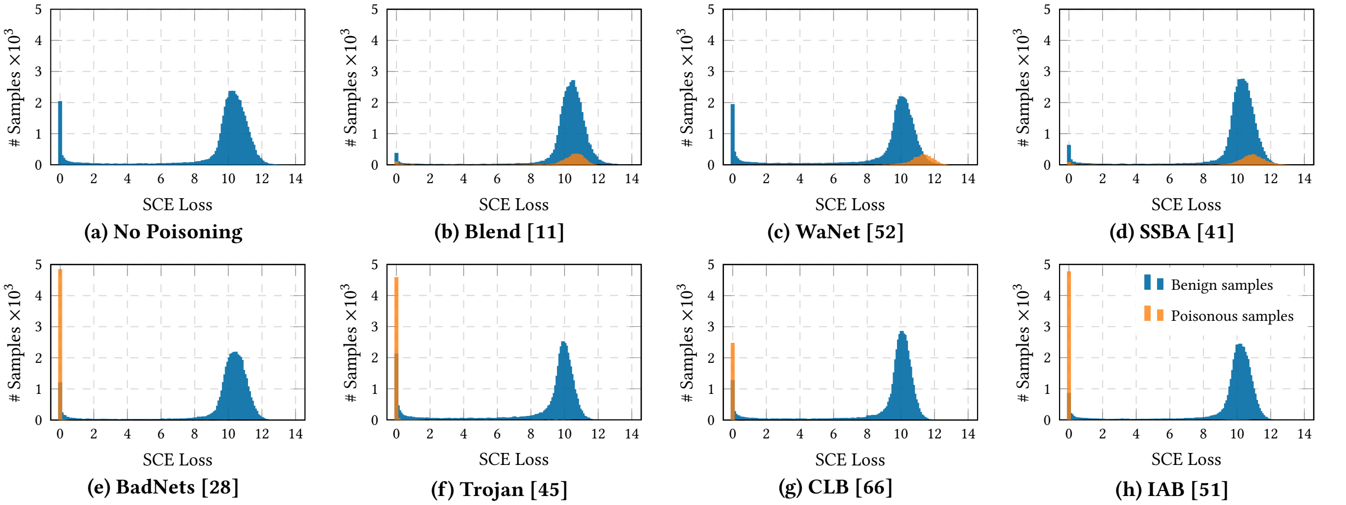


Figure 3: Impact of perturbation budget ϵ on various backdoor trigger. Baseline model ResNet18 is trained on every CIFAR10 poisoned by each backdooring attack.



(a) No Poisoning (b) Blend [11] (c) WaNet [52] (d) SSBA [41]

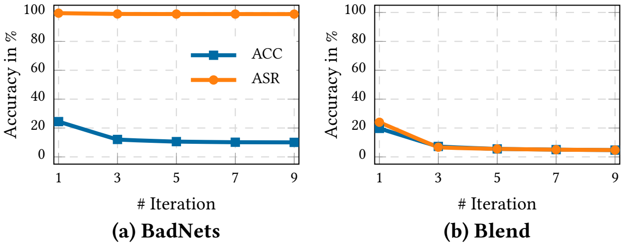
(e) BadNets [28] (f) Trojan [45] (g) CLB [66] (h) IAB [51]

Figure 4: Evaluation of adversarial robustness and backdoor trigger strength by using iFGSM attack bounded by ϵ = $^{2}/_{255}$. Experiments are conducted with ResNet18 naively pre-trained on the original and poisoned CIFAR10 individually. The first row presents the data distribution on a benign dataset without poisoning and other three backdooring attacks using whole-image or dynamic trigger. The second row visualizes the distribution of poisonous samples using fix patch-based trigger. Across all attacks, adversarial perturbation can easily push up the SCE loss of all poisonous samples of Blend, SSBA and WaNet.

fixed patch triggers (i.e., BadNets, Trojan) successfully resist the adversarial perturbation (Category A). In contrast, samples with full-image triggers such as Blend or WaNet are highly vulnerable to the perturbation (Category B).

Perturbation steps. By using iFGSM with a step size equal to $^{2}/_{255}$, increasing the number of iterations cannot make the attack generate an effective adversarial perturbation misleading the prediction on poisonous samples with robust triggers.

In summary, using the budget ϵ = $^{2}/_{255}$ and 5 iterations allows iFGSM attack to distinguish poisonous samples of Category A. Fig. 4 visualizes the distribution of benign and poisonous samples in each considered backdooring attack. We use symmetrical cross-entropy loss (SCE) [71] as suggested in [23, 36] to enlarge the significance of misclassified samples. In terms of SCE distribution, the entire dataset can be easily partitioned as a robust (low SCE loss) and a non-robust (high SCE loss) subset [37], where poisonous samples of Category A mostly stay in the former. However, the isolation is less effective for poisonous samples of Category B, as they behave similarly to benign samples under the adversarial perturbation, making their SCE loss exceptionally high and, hence, pushing them to the non-robust partition.



(a) BadNets (b) Blend

Figure 5: Impact of the number of iteration steps in iFGSM attack on the poisoned CIFAR10 using model ResNet18.

> **Takeaway.** Using triggers of Category A makes backdooring attacks intrinsically robust against the standard, (i.e., l_{inf} -normed), adversarial perturbation that sets $\epsilon \leq$ $^{8}/_{255}$. After the naive training, generating adversarial examples with a small perturbation budget allows to isolate poisonous samples of Category A from the dataset, which, however, does not hold for backdoors of Category B.

4.2 Backdoors under Adversarial Training

The dataset splitting using adversarial perturbation as above isolates poisonous samples of Category A and thus avoid the adversarially robust backdoors, which the adversarial training alone cannot suppress [74]. In this section, we analyze the impact of adversarial training on suppressing backdoors using robust and vulnerable triggers to the adversarial perturbation.

Perturbation budget. We apply different perturbation budgets in standard adversarial training, that uses projected gradient descent method (PGD) to generate adversarial examples [48]. Similar to the observation in Fig. 3, the perturbation budget ϵ is decisive in the backdoor suppression by adversarial training (cf. Fig. 6). On poisonous samples of Category A, using perturbation budget ϵ = $^{8}/_{255}$ cannot prevent the backdoor injection. Inversely, the adversarial robustness of these poisonous samples can be even improved, which infers that strong poisoning triggers belongs to the robust feature. For backdoors with low trigger visibility, e.g., Blend and WaNet, the injection performance starts decreasing at perturbation budget

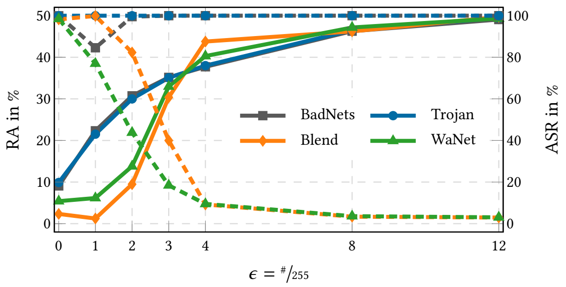


Figure 6: Adversarial training against various backdoors using different perturbation budgets ϵ. We train the baseline model ResNet18 on each poisoned CIFAR10. Solid line and dashed line denotes the robust accuracy (RA) and success rate of the backdoor (ASR), respectively.

$\epsilon = {}^{2}/_{255}$ and backdoors are suppressed at $\epsilon = {}^{8}/_{255}$. As a higher perturbation budget is beneficial for the adversarial robustness, the standard adversarial training using $\epsilon = {}^{8}/_{255}$ finally yields a robust model without the existence of backdoors using stealthy triggers.

Poisoning ratio. Moreover, adversarial training performs better against backdoors when the poisoning ratio ρ decreases. Similar to the observation in Fig. 2, the success rate of backdooring attacks degrades by reducing the portion of poisonous samples in the adversarial training. From the defense perspective, adversarial training remains effective against adversarially robust backdooring attacks when the poisoning ratio is low (cf. Fig. 7). Assuming that an anti-backdoor dataset splitting is applied upfront based on the distribution observed in Fig. 4, the amount of robust poisonous samples, i.e., Category A, would be very small in the training dataset. This way, adversarial training can improve the model robustness and simultaneously mitigate the backdooring attack that uses poisonous samples of Category B.

> **Takeaway.** In case that a splitting approach is executed before to isolate most poisonous samples of Category A, standard adversarial training improves the model robustness against the adversarial perturbation. Simultaneously, it suppresses backdoors that either employ poisonous samples of Category B using a non-robust trigger or poisons only a tiny portion of the training dataset.

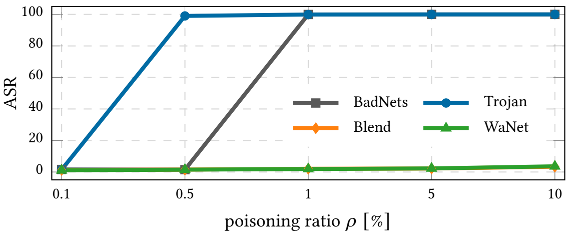


Figure 7: Adversarial training on backdoor attacks with different poisoning ratios. We use the perturbation budget $\epsilon = {}^{8}/_{255}$ to train the baseline model ResNet18 individually on the poisoned CIFAR10 by each attack.

5 Adversarially Robust ABL

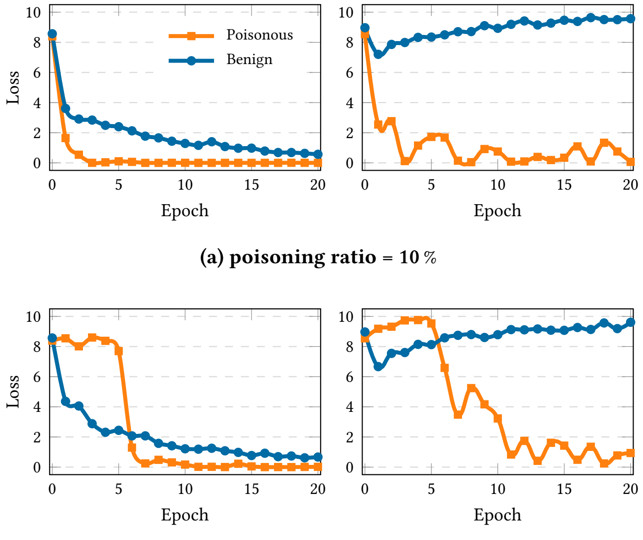
We observe large variance in adversarial robustness for different backdoor triggers, that is, poisonous samples of Category A with a fixed patch trigger remain highly effective during adversarial training, whereas backdoors using poisonous samples of Category B are significantly suppressed due to the impact of adversarial perturbation. Based on this observation, we propose an adversarially robust anti-backdoor learning (A-ABL) that requires no prior clean dataset, copes with backdoor suppression, and achieves an adversarially robust model eventually. Our method consists of three stages:

1. **Initialization.** We first train the DNN model naively for several epochs on the entire training dataset. The model yields high prediction accuracy on benign and poisonous samples, but it remains highly vulnerable to adversarial perturbation on any benign input.
2. **Dataset splitting based on "adversarial robustness."** Poisonous samples of Category A using fixed patch triggers are robust against adversarial perturbation, while most benign samples are not. Thus, we adopt the iFGSM attack [37] with a small perturbation budget to split a non-robust subset from the training dataset, aiming to isolate all poisonous samples of Category A using a robust trigger.
3. **Adversarial training.** However, the previous step cannot remove subtly embedded backdoors that are affected by adversarial examples. Adversarial training, in turn, is design to do exactly this. We thus train the model on the remaining subset using PGD-based adversarial training [48] from scratch, yielding a model that is both adversarial robust and free of Category A and Category B backdoors.

5.1 Initialization

In the initial stage, we train the model naively with a given poisoned dataset with using ADAM optimizer and a fixed learning rate 0.001 for several epochs. Previous defenses [42, 81, 85] address anti-backdoor learning based on the observation of poisonous samples converging faster during the model training. However, the learning of the backdoor is slower than benign samples, when the poisoning ratio is small.

In Fig. 8, we visualize training progresses on poisoned datasets by BadNets [28] with poisoning ratios 1 % and 10 %. We observe how training loss develops in both benign and poisonous samples. When the poisoning ratio is 10 %, the natural loss gap clearly exists, which provides the criterion to filter our poisonous samples. However, such a loss gap disappears when the poisoning ratio is small. In a nutshell, in case that the defender has no knowledge of the backdoor strategy, the dataset splitting [42] cannot rely on the natural loss gap. On the contrary, poisonous samples of BadNets converge to a low adversarial loss (i.e., a high adversarial robustness), while benign samples present significant vulnerability to the adversarial perturbation. Obviously, for backdoors with high trigger visibility, the intrinsic adversarial robustness allows to distinguish benign samples. Therefore, without precisely catching the natural loss gap as in related work [36, 42, 81], splitting the samples patched by adversarially robust trigger is easier after converting all benign samples to adversarial examples.



(a) poisoning ratio = 10 %

(b) poisoning ratio = 1 %

Figure 8: Training a ResNet18 model on poisoned CIFAR10 datasets by BadNets attack. The left and right of each figure shows the learning curve of benign samples and adversarial examples, respectively. We use iFGSM [37] to generate adversarial perturbation with budegt $\epsilon = {}^{2}/_{255}$ and 5 steps.

5.2 Dataset Splitting

Previous anti-backdoor learning defenses [23, 36, 42, 81] split the entire dataset $\tilde{\mathcal{D}}$ to a benign set with $\rho \approx 0.0$ and a poisonous set that contains poisonous samples of $\mathcal{D}_p$ as many as possible. In turn, we isolate adversarially robust poisonous samples as a subset $\mathcal{D}_{rob}$ using a backdoored model and preserve an adversarial vulnerable subset $\mathcal{D}_{vul}$ as the new training dataset.

After the initial naive training, we use adversarial robustness as the criterion to split the entire training dataset into an intrinsically robust subset $\mathcal{D}_{rob}$ and an adversarially vulnerable subset $\mathcal{D}_{vul}$. We set the perturbation budget ϵ to ${}^{2}/_{255}$ and use 5 steps for the iFGSM attack. To enlarge the distance between adversarially robust and vulnerable samples for better splitting, we use SCE loss to raise the weight on the ground-truth label of each sample [71] and thereby enlarges the loss value of misclassified samples. In comparison to cross-entropy (CE) loss, SCE loss significantly amplifies non-robust samples (cf. Fig. 9) and thus yields a distribution with two clusters.

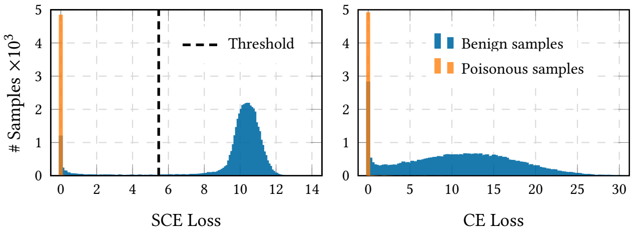


Figure 9: Comparing the distribution of SCE and CE loss with ResNet18 on a CIFAR10 dataset with BadNets injection. The splitting threshold in SCE loss is explored by OTSU method.

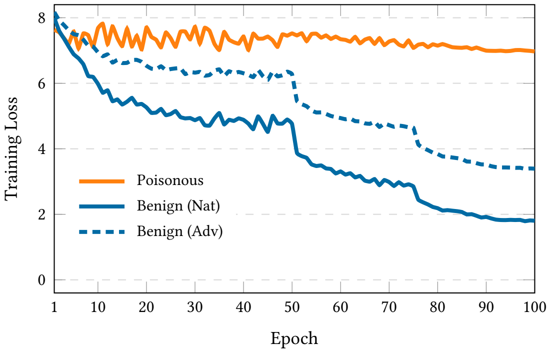


Figure 10: Adversarial training with a ResNet18 model on CIFAR10's subset containing poisonous samples of Blend. Benign samples and their adversarial examples are denoted as Nat and Adv, respectively.

Li et al. [42] propose to isolate 10 % samples with lowest training loss for backdoor unlearning. Other defenses [23, 36, 81] even use 50 % to ensure the elimination of poisonous samples. However, a fixed splitting ratio is not adaptable to different poisoning ratios. In our method, we use OTSU [53] to search for the threshold with the maximal variance between two clusters for an adaptive dataset splitting. Since samples vulnerable to adversarial perturbation contributes more to the model natural performance [21, 37], training on $\mathcal{D}_{vul}$ directly achieves a model without any backdoor that plants the robust trigger on the poisonous samples, as dataset splitting yields a subset $\mathcal{D}_{vul}$ of a significantly lower poisoning ratio (cf. Table 1).

5.3 Adversarial Training

Despite the adversarial splitting of Stage 2, samples of Category B remain in the dataset $\mathcal{D}_{vul}$, as they are vulnerable to adversarial perturbation, particularly for triggers with low visibility. According to the study in Section 4.2, backdooring attacks using triggers vulnerable to adversarial perturbation cannot resist the suppression of adversarial training. In the final stage, we adopt the standard adversarial training, i.e., PGD-AT [48], to improve the robustness against adversarial perturbation and simultaneously suppress the backdoor injection that the remaining poisonous samples in $\mathcal{D}_{vul}$ introduce. The optimization is formulated as follows:

$$\min_{\theta} \mathop{\mathbb{E}}_{(\mathbf{x},y)\in\mathcal{D}_{vul}} \left[\max_{\delta} \left\{ \mathcal{L}\left(\theta, \mathbf{x}+\delta, y\right) \right\} \right] \tag{1}$$

where the inner maximizes the training loss $\mathcal{L}$ to generate adversarial perturbation δ on input samples $\mathbf{x} \in \mathcal{D}_{vul}$ that contribute to the adversarial robustness optimization [48] and meanwhile eliminate the connection between the trigger and the backdoor target by perturbing trigger patterns. Fig. 10 shows adversarial training procedure on the split subset with poisonous samples of Blend attack. Both clean accuracy and robust accuracy increase with the training while the training loss of poisonous samples stays high, which shows the ineffectiveness of backdoor injection.

In summary, using adversarial perturbation in the splitting efficiently filters out poisonous samples of Category A that use robust triggers. Consequently, adversarial training successfully prevents the backdooring attacks introduced by the remaining poisonous samples of Category B. Finally, we achieve an adversarially robust model free from backdooring attacks.

6 Evaluation

Begin by describing the experimental setup including the used datasest and models, considered attacks, evaluated defenses from related work, and evaluation metrics before reporting on different experiments. In Sections 6.1 and 6.2, report on performance after Stage 2 and Stage 3 individually, and additionally conduct an ablation study in Section 6.3.

Datasets and models. We conduct extensive experiments to evaluate all attacks and defenses on two small-scale datasets CIFAR10 [38], GTSRB [64] with the model architecture ResNet18 [33]. Furthermore, we extend the evaluation on a large-scale dataset Tiny-ImageNet [39] with model architecture ResNet34 [33].

Considered attacks. We evaluate our method with seven representative backdooring attacks for the dataset poisoning, including *dirty-label* attacks that use patch-based fixed triggers: BadNets [28], Trojan attack [45], the *clean-label* attack using the patch trigger with adversarial perturbation (CLB) [66], and Blend attack that uses a fixed whole-image trigger [11]. We also consider three dynamic *dirty-label* attacks using the adapted stripe trigger IAB [51], and whole-image triggers WaNet [52], and SSBA [41]. In all attack setups except CLB attack, we choose the target label $y_t = 0$ and use the poisoning ratio $\rho = 10\,\%$ as default, and all poisonous samples are randomly selected from all classes. For CLB attack, we adopt projected gradient descent (PGD) to generate adversarial perturbation with strength $\epsilon = {}^{16}/_{255}$ and step size ${}^{2}/_{255}$ for 30 steps. Poisonous samples of CLB are randomly selected from the target class, and additionally we set poisoning ratio $\rho = 50\,\%$ on CIFAR10 and GTSRB, and $\rho = 100\,\%$ on Tiny-ImageNet. For Blend attack, we use the Hello-Kitty trigger pattern for experiments on CIFAR10 and GTSRB, and the random uniform noise trigger on Tiny-ImageNet. The trigger opacity is 0.1. Other details of each attack execution are identical to the default implementation in their original papers.

Defense baselines. We compare our method with four training-time defenses that focus on backdooring attacks only and require no clean dataset upfront: ABL [42], DBD [36], D-ST [10], and CBD [81]. We conduct all defense experiments with the proposed settings in the original implementation. Meanwhile, we compare with the direct use of adversarial training (Adv. Train). In A-ABL's implementation, we train the model for 20 epochs in Stage 1. In Stage 2, we use iFGSM attack with perturbation budget ${}^{2}/_{255}$ for 5 steps in the dataset splitting for small-scale datasets CIFAR10 and GTSRB, and we set budget $\epsilon = {}^{1}/_{255}$ for Tiny-ImageNet due to its larger image size. In Stage 3, we use PGD with perturbation budget $\epsilon = {}^{8}/_{255}$ and step size ${}^{2}/_{255}$ for 10 steps in the standard adversarial training [48]. We train the model for 100 epochs and set the initial learning rate 0.1. During training, we step-wisely lower the learning rate by 0.1 times on each 50, 75, 90 epoch, respectively.

Table 1: Evaluation of dataset splitting by using iFGSM adversarial perturbation. We use γ to express the ratio of subset $\mathcal{D}_{vul}$ to the original $\tilde{\mathcal{D}}$ and show the poisoning ratio in $\mathcal{D}_{vul}$ as ρ_{vul}. Both ratios are shown in %.

Attack	CIFAR10		GTSRB		Tiny-ImageNet	
	γ	ρ_{vul}	γ	ρ_{vul}	γ	ρ_{vul}
No-Attack	90.00	—	49.36	—	90.00	—
BadNets	82.34	0.13	55.83	0.00	69.28	0.01
Trojan	76.98	0.06	47.78	0.04	70.52	0.05
CLB	84.78	0.00	58.05	0.14	78.13	0.28
IAB	82.79	0.32	46.40	0.08	72.62	0.20
Blend	90.00	10.04	73.37	6.71	72.14	2.39
SSBA	90.00	10.49	87.55	2.61	73.07	8.53
WaNet	86.27	11.52	63.43	15.68	76.33	6.49

Evaluation metrics. We evaluate the performance of dataset splitting with metrics, i.e., subset splitting ratio $\gamma = {}^{|\mathcal{D}_{vul}|}/_{|\tilde{\mathcal{D}}|}$ and the poisoning ratio ρ_{vul} in the subset $\mathcal{D}_{vul}$. Regarding the final defensive performance, we adopt three metrics, i.e., Clean Accuracy (ACC), Robust Accuracy (RA) and Attack Success Rate (ASR). ACC is the prediction accuracy on a clean test dataset. And we measure RA by using PGD-10 attack with the perturbation budget $\epsilon = {}^{8}/_{255}$ and step-size ${}^{2}/_{255}$. Differently, ASR represents the fraction of a poisoned test dataset classified as the backdoor target label. The optimality of backdooring attacks has a high ACC and an ASR $\approx$ 100%. In contrast, adversarially robust anti-backdoor learning would achieve high ACC and RA and the ASR $\approx$ 0% in the final.

6.1 Dataset Splitting based on Adv. Robustness

According to the analysis in Fig. 2, attacks employing poisonous samples of Category B show less resistance against the adversarial perturbation (cf. Fig. 3). Table 1 summarizes the splitting results on each dataset poisoned by all considered backdooring attacks. Triggers used for Category A (i.e., BadNets, Trojan, CLB and IAB) show significantly high robustness against adversarial perturbation. Thus, our adaptive splitting using OTSU threshold successfully identify and isolate nearly all poisonous samples of Category A and leaves a non-robust subset $\mathcal{D}_{vul}$ with the poisoning ratio $\rho_{vul} << \rho$.

Considering that the fraction of benign samples in dataset $\tilde{\mathcal{D}}$ is equal 90 % in most backdooring attacks which is 95 % for CLB,

Table 2: Naive training on the split dataset $\mathcal{D}_{vul}$ by our splitting method. All results are shown in %.

Attack	CIFAR10		GTSRB		Tiny-ImageNet	
	ACC	ASR	ACC	ASR	ACC	ASR
BadNets	93.26	4.53	96.25	0.00	57.84	0.05
Trojan	93.82	1.56	95.36	1.04	57.23	0.48
CLB	93.70	0.63	95.82	12.98	57.62	1.06
IAB	93.19	99.93	95.88	0.21	57.48	1.93
Blend	93.79	99.95	96.57	99.71	57.64	99.99
SSBA	93.87	100.00	96.82	81.93	57.87	99.97
WaNet	93.39	98.60	96.18	99.34	56.96	98.85

Table 3: Comparing A-ABL with training time defenses and the standard adversarial training (Adv. Train). We measure the adversarial robustness (i.e., RA) by PGD-10 attack, and evaluate the defense on clean datasets without poisoning (i.e., "———"). The best result across all defenses is highlighted in bold font. Orange bold font indicates the defense failure (i.e., ASR > 90 %).

Dataset	Attack	ABL		DBD		D-ST		CBD		Adv. Train			A-ABL (ours)		
		ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	RA	ACC	ASR	RA
CIFAR10	———	91.26	—	92.88	—	92.77	—	**93.02**	—	83.47	—	45.32	83.77	—	**46.11**
	BadNets	90.67	0.37	**92.78**	30.59	92.27	**0.01**	88.19	1.48	83.72	**100.00**	46.28	83.59	1.26	**46.63**
	Trojan	91.59	1.24	93.31	**99.99**	**93.97**	**0.22**	91.40	1.92	83.59	**100.00**	**46.51**	83.32	1.44	46.39
	CLB	79.39	0.84	90.84	1.47	90.66	**0.00**	**92.83**	0.19	84.74	**99.97**	47.31	83.98	0.59	**47.96**
	IAB	93.30	**6.32**	79.11	82.15	**93.60**	22.00	92.39	69.10	83.76	**99.83**	**46.66**	83.51	6.88	46.09
	Blend	87.66	6.59	**92.08**	**99.91**	89.93	55.48	86.33	5.77	82.81	3.37	46.22	82.68	**2.59**	**47.72**
	SSBA	88.09	7.47	89.47	**98.24**	86.51	**0.53**	**90.98**	3.68	83.46	7.57	46.42	82.89	4.94	**47.62**
	WaNet	88.74	55.81	93.03	15.27	**93.09**	**99.72**	88.49	32.73	81.83	3.61	47.16	80.34	**3.31**	**47.64**
	Average	88.49	11.23	90.09	61.09	**91.43**	25.42	90.09	16.41	83.42	59.19	46.65	82.90	**3.00**	**47.15**
	Worst Case	79.39	55.81	79.11	**99.99**	**86.51**	**99.72**	86.33	69.10	81.83	**100.00**	**46.22**	80.34	**6.88**	46.09
GTSRB	———	**95.42**	—	94.37	—	95.27	—	93.07	—	89.27	—	62.43	87.43	—	61.82
	BadNets	**95.99**	0.02	92.68	0.29	95.83	**0.00**	95.59	0.14	89.44	**100.00**	**62.24**	88.65	0.01	60.09
	Trojan	**96.21**	0.01	93.91	**0.00**	96.20	**0.00**	54.20	5.57	88.26	**100.00**	**58.99**	87.33	0.12	56.32
	CLB	**92.56**	0.01	91.99	**0.00**	78.06	2.15	75.58	**0.00**	89.79	69.59	**62.13**	88.02	0.76	61.43
	IAB	**96.33**	11.32	87.55	**99.31**	96.09	**0.00**	82.29	**99.36**	89.94	**100.00**	**62.02**	87.25	2.42	59.87
	Blend	90.41	**97.70**	93.21	**99.98**	**93.96**	40.86	79.92	65.37	89.17	13.73	**62.45**	88.71	**1.65**	62.20
	SSBA	89.89	**100.00**	**93.65**	**99.27**	82.47	46.99	72.47	55.59	88.89	2.11	**62.16**	88.81	**0.78**	61.65
	WaNet	88.27	**99.92**	92.76	**0.00**	**93.41**	67.26	87.96	19.36	88.59	2.01	**61.95**	87.15	3.27	58.23
	Average	**92.81**	44.14	92.25	42.69	90.86	22.47	78.29	35.06	89.15	55.35	**61.71**	87.99	**1.29**	59.97
	Worst Case	**88.27**	**100.00**	87.55	**99.98**	78.06	67.26	54.20	**99.36**	88.26	**100.00**	**58.99**	87.15	**3.27**	56.32
Tiny-ImageNet	———	39.59	—	50.94	—	**56.35**	—	51.84	—	41.64	—	**21.41**	41.32	—	21.04
	BadNets	46.26	**0.00**	50.88	**100.00**	**56.10**	0.16	49.21	0.27	42.57	**99.87**	21.04	42.36	0.08	**22.06**
	Trojan	47.43	**0.00**	51.88	**100.00**	**56.14**	0.02	52.10	0.10	41.86	**99.03**	20.75	42.60	0.07	**21.32**
	CLB	49.93	**0.01**	51.62	**100.00**	**56.81**	**0.01**	50.01	0.84	42.67	**97.68**	21.45	41.45	1.57	**21.58**
	IAB	46.00	**0.00**	50.74	**100.00**	**55.66**	**0.00**	50.40	0.21	41.92	**99.87**	**20.84**	41.58	0.24	20.78
	Blend	49.07	**99.99**	51.73	**100.00**	**56.56**	**97.63**	52.68	**0.78**	41.74	3.26	**20.87**	42.84	1.92	20.20
	SSBA	41.41	**0.03**	50.74	**98.26**	**55.61**	38.58	47.38	0.38	41.05	3.47	**21.24**	41.07	3.65	21.12
	WaNet	44.32	**1.68**	51.22	**100.00**	**54.11**	26.30	50.74	17.54	40.03	4.32	19.86	40.11	5.27	**20.44**
	Average	46.35	14.53	51.26	**99.75**	**55.86**	23.24	50.36	2.87	41.69	58.21	20.86	41.72	**1.83**	**21.07**
	Worst Case	41.41	**99.99**	50.74	**100.00**	**54.11**	**97.63**	47.38	17.54	40.03	**99.87**	19.86	40.11	**5.27**	**20.20**

splitting on CIFAR10 datasets using iFGSM attack results in a subset $\mathcal{D}_{vul}$ with the size beyond 76 % of $\tilde{\mathcal{D}}$, meaning that our method ensures over 85 % of all benign samples in $\mathcal{D}_{vul}$.

Samples with high uncertainty are particularly important for reproducing the natural performance [12]. Since, benign samples in $\mathcal{D}_{vul}$ are vulnerable to the adversarial perturbation, they are located near to model's decision boundary, and thus, have high uncertainty [21]. Moreover, randomly splitting a dataset (e.g., CIFAR10 with $\gamma \geq 50\,\%$) is sufficient to reproduce the natural performance of the entire dataset [30, 56]. Thus, naively training on the $\mathcal{D}_{vul}$ subset enables achieving high clean accuracy (cf. Table 2). Furthermore, our splitting methods successfully eliminate the threat of backdoor injection of Category A that uses adversarially robust triggers.

Although our splitting approach achieves a similar preservation of benign samples, the mere adversarial perturbation misleads the model prediction of poisonous samples of Category B as well. Thus, all poisonous samples show a very high loss value, such that the isolation of poisonous samples fails in the splitting stage. Hence, naive training on $\mathcal{D}_{vul}$ cannot avoid the backdoor injection (cf. Table 2). Nevertheless, poisonous samples with an intrinsic high vulnerability to adversarial perturbation cannot stand the suppression by adversarial training. In the next section, we show the performance of backdoor suppression by the model training in Stage 3.

6.2 Adversarial Training

After the previous dataset splitting, in Stage 3, we adopt standard adversarial training to suppress the backdoor injection introduced by the potentially remaining poisonous samples. We first run the adversarial training on the entire poisoned dataset to evaluate the naive backdoor suppression effect. By aligning the direct training results in Table 3 with Table 1, it is notable that backdooring attacks using Category A triggers present significantly high resistance against the impact of adversarial training, thus their final ASR

Table 4: Adversarial training on split datasets by using D-ST.

Dataset	Attack	Clean				Clean + Suspicious			
		γ	ρ_{vul}	ACC	ASR	γ	ρ_{vul}	ACC	ASR
CIFAR10	Blend	32.43	0.44	45.69	11.22	95.63	7.81	82.42	6.01
	WaNet	91.90	8.47	80.04	2.68	94.91	9.04	81.12	3.65
GTSRB	Blend	20.02	1.95	67.48	1.46	95.00	5.47	88.26	3.64
	WaNet	19.92	10.00	62.48	7.24	95.01	10.05	87.89	2.66

remains nearly 100 %, i.e., a successful backdoor injection. Differently, these intrinsically robust poisonous samples are isolated from the final training set $\mathcal{D}_{vul}$ in the earlier splitting stage. Meanwhile, $\mathcal{D}_{vul}$ contains most benign samples vulnerable to the adversarial perturbation, which contribute most to the model's robustness in adversarial training [37, 47]. Hence, training on $\mathcal{D}_{vul}$ in Stage 3 yields a robust model without any backdoor of Category A.

For other triggers corresponding to Category B such as Blend, SSBA and WaNet which are vulnerable to adversarial perturbation, adversarial training presents a strong suppression on their backdoor injection. In the previous splitting, the size of $\mathcal{D}_{vul}$ is smaller than $\tilde{\mathcal{D}}$ while the final poisoning ratio ρ_{vul} remains nearly equal the default value in $\tilde{\mathcal{D}}$, meaning that the splitting isolates a very small portion of poisonous samples with relatively higher intrinsic robustness. Therefore, naive training on $\mathcal{D}_{vul}$ yields a comparable clean accuracy but a high success rate of backdoors (cf. Table 2). Nevertheless, results in Table 3 demonstrate that using adversarial training can easily suppress those vulnerable backdoors and simultaneously achieves a high robust accuracy with a certain preservation of natural performance, thereby making our robust anti-backdoor learning excel.

In comparison to our method, previous defenses either fail for one or several backdooring attacks, or they result in degredation is natural performance. Similar to our method, D-ST firstly partitions the dataset in poisoned, suspicious and clean subsets. However, the defense fails against Blend and WaNet attacks in both CIFAR10 and GTSRB datasets. Despite D-ST's ineffective splitting on Category B backdoors, even standard adversarial training on the subsets ultimately suppresses the backdoor (cf. Table 4), which demonstrates the potential of adversarial training as an complementary techniques for existing training-time defenses.

Table 5: Comparing different data augmentations in dataset splitting. Each experiment is with a individual ResNet18 pretrained on a poisoned CIFAR10.

Attack	RandAugment		AutoAugment		AutoMix		A-ABL (ours)	
	γ	ρ_{vul}	γ	ρ_{vul}	γ	ρ_{vul}	γ	ρ_{vul}
BadNets	18.21	6.20	21.87	7.64	10.00	9.28	**82.34**	**0.13**
Trojan	18.29	1.01	22.47	6.00	10.12	8.60	**76.98**	**0.06**
CLB	20.69	5.65	25.14	2.92	14.37	4.38	**84.78**	**0.00**
IAB	22.06	0.12	22.27	0.48	10.50	2.70	**82.79**	**0.32**
Blend	15.80	**7.66**	21.05	11.15	11.20	7.80	**90.00**	10.04
SSBA	18.45	8.86	22.93	7.33	11.08	**1.48**	**90.00**	10.49
WaNet	19.32	19.81	24.80	13.56	15.78	13.38	**86.27**	**11.52**

In Fig. 11, we additionally provide the final distribution of the entire poisoned dataset $\tilde{\mathcal{D}}$ in the robust model after the adversarial training in Stage 3. Due to the effectiveness of dataset splitting, the final training stage achieves a robust model without any backdoor injection using robust triggers. Therefore, poisonous samples have a high loss value, even under the adversarial perturbation. Regarding Blend and WaNet attacks using low trigger visibility, adversarial training compensates for the ineffectiveness of previous anti-backdoor splitting and yields a robust model that breaks the connection between poisonous samples and the backdoor target. As the whole-image trigger changes the ground-truth features of poisonous samples, the adversarial robustness degrades a little. Thus, there are more benign samples having higher SCE loss after the adversarial perturbation.

6.3 Ablation Study

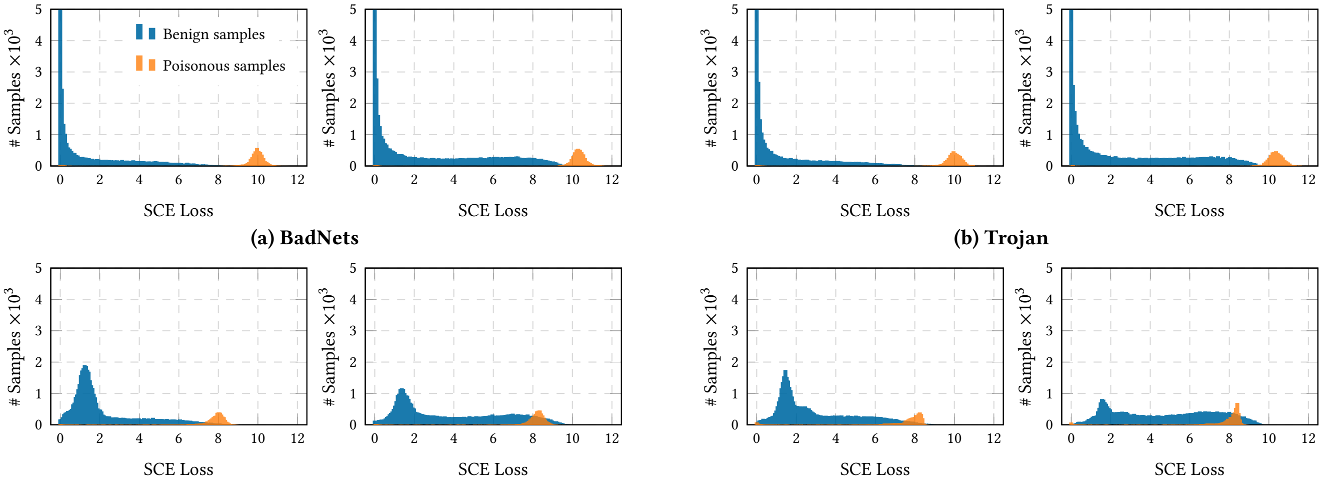
In this section, we first compare our adversarial perturbation criterion with other data augmentation methods for dataset splitting to investigate our method's settings and adaptability. Following this, we evaluate the defensive performance of our method across different model architectures, before we investigate the performance of our method on different poisoning ratios.

Data augmentations for splitting. Most backdooring attacks are robust against common augmentations e.g., RandomCrop, RandomFlipping, etc., while some work demonstrates the effect of backdoor trigger erasing by using strong augmentations [10, 57]. In Table 5, we thus compare our splitting with three strong and automatic augmentation methods, i.e., RandAugment [17], AutoAugment [16] and AutoMix [34]. Regarding isolating poisonous samples, several but not all poisonous samples with high trigger visibility are vulnerable to augmentation, and thus, the splitting preserves those poisonous samples in the subset. Moreover, no augmentations can isolate poisonous samples with low trigger visibility. In terms of the subset size after splitting, using a strong augmentation method always results in a significantly lower splitting ratio than A-ABL. The model's natural performance after the adversarial training will degrade due to the lack of benign samples.

Cross-architecture evaluation. In Table 6, we conduct the experiments of our method across other four architectures VGG16 [62], MobileNetV2 [59] and DenseNet121 [35]. Our method has proven

Table 6: Cross-architecture evaluation on each poisoned CIFAR10 dataset. All results are shown in %.

Attack	VGG16			MobileNetV2			DenseNet121		
	ACC	RA	ASR	ACC	RA	ASR	ACC	RA	ASR
———	77.52	46.03	–	80.73	47.31	–	86.34	48.56	–
BadNets	76.97	44.59	1.82	80.94	47.77	1.40	86.00	48.12	1.61
Trojan	76.32	44.60	1.54	80.62	48.09	1.42	85.79	48.07	1.39
CLB	77.81	45.83	0.83	81.32	48.04	0.82	86.16	48.14	0.89
IAB	75.21	44.43	7.73	80.17	48.74	4.53	85.69	48.23	4.69
Blend	77.13	45.12	1.93	79.56	48.26	1.90	85.00	48.10	1.56
SSBA	77.02	44.76	2.08	79.78	48.02	2.86	84.91	48.12	4.88
WaNet	75.24	41.39	2.59	77.91	46.40	2.70	83.92	48.70	2.52



(a) BadNets

(b) Trojan

(c) Blend

(d) WaNet

Figure 11: Distributions of training samples in each poisoned CIFAR10 with the robust ResNet18 after learning by A-ABL. Each figure consist of two plots, where the left and right figure represents respectively the distribution of original data samples and their adversarial examples.

the capability to achieve comparable ACC and RA across different model architectures in defending all different backdooring attacks.

Effectiveness with different poisoning ratios. In Table 7, we additionally evaluate A-ABL's performance against three different poisoning ratios. We exclude the evaluation on CLB attack, as it has the upper limit of ρ due to the poisoning only on the target class. Our method remains effective in backdoor suppression for the relative lower ratios ρ equal 1 % and 5 %. Since the number of poisonous samples decreases, our method automatically explores more benign samples and achieves relatively higher robustness than the case of $\rho = 10\,\%$. Differently, a higher poisoning ratio $\rho = 20\,\%$ leads to a larger lose of benign samples in the entire dataset. In particular, WaNet, by default, improves the attack stealthiness by generating noised images, which are twice the amount of poisonous samples. Thus, 60 % of the training dataset is maliciously modified. In consequence, our method yields a small clean accuracy reduction after the final adversarial training. Nevertheless, a larger number of poisonous samples cannot successfully inject the backdoor into the model after using our defensive training.

Table 7: A-ABL's defense with a ResNet18 model against different dataset poisoning ratios on CIFAR10.

Attack	1 %			5 %			20 %		
	ACC	RA	ASR	ACC	RA	ASR	ACC	RA	ASR
BadNets	83.83	46.84	1.39	83.94	46.60	1.24	82.70	45.36	1.63
Trojan	83.48	46.47	1.50	83.43	45.97	1.37	81.43	45.07	2.54
IAB	83.81	47.13	4.33	83.48	47.87	5.84	82.17	45.14	7.82
Blend	83.16	46.43	1.66	83.28	47.98	1.16	80.22	45.32	6.80
SSBA	83.25	46.52	1.84	83.16	47.04	1.73	81.03	47.43	4.17
WaNet	83.51	45.86	1.86	82.44	45.60	2.40	72.17	37.19	14.28

7 Conclusion

So far, the community has been fighting a lost battle in anti-backdoor learning. It is trying to perfectly separate poisonous and clean samples in the training data, entering a cat-and-mouse game between defense and the ever-increasing stealthiness of backdooring attacks. However, we show that a perfect separation is unnecessary in a practical setting, where the defender aims for robustness against adversarial examples *and* against backdoor injection side-by-side.

We reduce anti-backdoor learning to a much simpler task, where we differ between backdoors that cannot be removed with adversarial training (Category A) and those that can (Category B). We measure the samples' adversarial vulnerability in a naively trained model to make this differentiation. Poisonous samples of simple backdoors such as BadNets and Trojan are more robust to adversarial perturbations than clean samples and more sophisticated backdoors such as WaNet and SSBA, allowing us to filter the former out early. More sophisticated backdoors are removed through adversarial training.

We are convinced that this is a game-changer for this research direction and hope to push open a door toward more holistic defenses that keep an eye on the bigger picture of practical use.

Acknowledgments

The authors thank the anonymous reviewers for their valuable suggestions, and gratefully acknowledge funding by the Helmholtz Association (HGF) within topic "46.23 Engineering Secure Systems", by SAP S.E. under project DE-2020-021, and by the German Federal Ministry of Education and Research (BMBF) under the project DataChainSec (FKZ 16KIS1700).

References

[1] Hamed Pirsiavash Aniruddha Saha, Akshayvarun Subramanya. 2019. Hidden Trigger Backdoor Attacks. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*.

[2] Eirini Anthi, Lowri Williams, Matilda Rhode, Pete Burnap, and Adam Wedgbury. 2021. Adversarial attacks on machine learning cybersecurity defences in Industrial Control Systems. *Journal of Information Security and Applications* 58 (2021), 102717. https://doi.org/10.1016/j.jisa.2020.102717

[3] G. Apruzzese, H. S. Anderson, S. Dambra, D. Freeman, F. Pierazzi, and K. Roundy. 2023. "Real Attackers Don't Compute Gradients": Bridging the Gap Between Adversarial ML Research and Practice. In *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*.

[4] Eugene Bagdasaryan and Vitaly Shmatikov. 2020. Blind Backdoors in Deep Learning Models. In *usenix*.

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

[6] Battista Biggio and Fabio Roli. 2018. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition* 84 (2018), 317–331.

[7] Nicholas Carlini, Matthew Jagielski, Christopher A. Choquette-Choo, Daniel Paleka, Will Pearce, Hyrum Anderson, Andreas Terzis, Kurt Thomas, and Florian Tramèr. 2023. Poisoning Web-Scale Training Datasets is Practical. arXiv:2302.10149 [cs.CR]

[8] Nicholas Carlini and David Wagner. 2017. Towards Evaluating the Robustness of Neural Networks. In *Proc. of the IEEE Symposium on Security and Privacy*. 39–57.

[9] Shuwen Chai and Jinghui Chen. 2022. One-shot Neural Backdoor Erasing via Adversarial Weight Masking. In *Proc. of the Annual Conference on Neural Information Processing Systems (NeurIPS)*.

[10] Weixin Chen, Baoyuan Wu, and Haoqian Wang. 2022. Effective Backdoor Defense by Exploiting Sensitivity of Poisoned Samples. In *Proc. of the Annual Conference on Neural Information Processing Systems (NeurIPS)*.

[11] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. 2017. Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning. *CoRR* abs/1712.05526 (2017).

[12] Cody Coleman, Christopher Yeh, Stephen Mussmann, Baharan Mirzasoleiman, Peter Bailis, Percy Liang, Jure Leskovec, and Matei Zaharia. 2020. Selection via Proxy: Efficient Data Selection for Deep Learning. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[13] Francesco Croce, Maksym Andriushchenko, and Matthias Hein. 2019. Provable Robustness of ReLU networks via Maximization of Linear Regions. In *Proc. of the International Conference on Artificial Intelligence and Statistics (AISTATS)*.

[14] Francesco Croce and Matthias Hein. 2020. Minimally distorted Adversarial Examples with a Fast Adaptive Boundary Attack. In *Proc. of the International Conference on Machine Learning (ICML)*.

[15] Francesco Croce and Matthias Hein. 2020. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *Proc. of the International Conference on Machine Learning (ICML)*.

[16] Ekin Dogus Cubuk, Barret Zoph, Dandelion Mané, Vijay Vasudevan, and Quoc V. Le. 2019. AutoAugment: Learning Augmentation Policies from Data. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[17] Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. 2020. RandAugment: Practical Automated Data Augmentation with a Reduced Search Space. In *Proc. of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (Eds.).

[18] Bao Gia Doan, Ehsan Abbasnejad, and Damith C. Ranasinghe. 2020. Februus: Input Purification Defense Against Trojan Attacks on Deep Neural Network Systems. In *Proc. of the Annual Computer Security Applications Conference (ACSAC)* (Austin, TX, USA).

[19] Khoa Doan, Yingjie Lao, Weijie Zhao, and Ping Li. 2021. LIRA: Learnable, Imperceptible and Robust Backdoor Attacks. In *Proc. of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[20] Min Du, Ruoxi Jia, , and Dawn Song. 2020. Robust Anomaly Detection and Backdoor Attack Detection via Differential Privacy. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[21] Melanie Ducoffe and Frederic Precioso. 2018. Adversarial Active Learning for Deep Networks: a Margin Based Approach. In *Proc. of the International Conference on Machine Learning (ICML)*.

[22] Di Feng, Lars Rosenbaum, and Klaus C. J. Dietmayer. 2018. Towards Safe Autonomous Driving: Capture Uncertainty in the Deep Neural Network For Lidar 3D Vehicle Detection. *Proc. of the International Conference on Intelligent Transportation Systems (ITSC)* (2018), 3266–3273.

[23] Kuofeng Gao, Yang Bai, Jindong Gu, Yong Yang, and Shu-Tao Xia. 2023. Backdoor Defense via Adaptively Splitting Poisoned Dataset. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[24] Yinghua Gao, Dongxian Wu, Jingfeng Zhang, Guanhao Gan, Shu-Tao Xia, Gang Niu, and Masashi Sugiyama. 2023. On the Effectiveness of Adversarial Training Against Backdoor Attacks. *IEEE Transactions on Neural Networks and Learning Systems* (2023).

[25] Yansong Gao, Chang Xu, Derui Wang, Shiping Chen, Damith C Ranasinghe, and Surya Nepal. 2019. STRIP: A Defence Against Trojan Attacks on Deep Neural Networks. In *Proc. of the Annual Computer Security Applications Conference (ACSAC)*.

[26] Micah Goldblum, Liam Fowl, Soheil Feizi, and Tom Goldstein. 2019. Adversarially Robust Distillation. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*.

[27] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and Harnessing Adversarial Examples. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[28] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. 2017. BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. *Proceeding of Machine Learning and Computer Security Workshop* (2017).

[29] Jiyang Guan, Zhuozhuo Tu, Ran He, and Dacheng Tao. 2022. Few-shot Backdoor Defense Using Shapley Estimation. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[30] Chengcheng Guo, Bo Zhao, and Yanbing Bai. 2022. DeepCore: A Comprehensive Library for Coreset Selection in Deep Learning. In *Database and Expert Systems Applications*, Christine Strauss, Alfredo Cuzzocrea, Gabriele Kotsis, A. Min Tjoa, and Ismail Khalil (Eds.). Springer International Publishing, Cham, 181–195.

[31] Junfeng Guo, Yiming Li, Xun Chen, Hanqing Guo, Lichao Sun, and Cong Liu. 2023. SCALE-UP: An Efficient Black-box Input-level Backdoor Detection via Analyzing Scaled Prediction Consistency. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[32] Jonathan Hayase, Weihao Kong, Raghav Somani, and Sewoong Oh. 2021. SPECTRE: defending against backdoor attacks using robust statistics. In *Proc. of the International Conference on Machine Learning (ICML)*.

[33] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778.

[34] Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. 2020. AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[35] Gao Huang, Zhuang Liu, and Laurens van der Maaten. 2017. Densely Connected Convolutional Networks. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[36] Kunzhe Huang, Yiming Li, Baoyuan Wu, Zhan Qin, and Kui Ren. 2022. Backdoor Defense via Decoupling the Training Process. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[37] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. 2019. Adversarial Examples Are Not Bugs, They Are Features. In *Proc. of the Annual Conference on Neural Information Processing Systems (NeurIPS)*.

[38] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. 2008. CIFAR (Canadian Institute for Advanced Research). http://www.cs.toronto.edu/~kriz/cifar.html

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

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

[41] Yuezun Li, Yiming Li, Baoyuan Wu, Longkang Li, Ran He, and Siwei Lyu. 2021. Invisible Backdoor Attack with Sample-Specific Triggers. In *Proc. of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[42] Yige Li, Xixiang Lyu, Nodens Koren, Lingjuan Lyu, Bo Li, and Xingjun Ma. 2021. Anti-Backdoor Learning: Training Clean Models on Poisoned Data. In *Proc. of the Annual Conference on Neural Information Processing Systems (NeurIPS)*.

[43] Yige Li, Xixiang Lyu, Nodens Koren, Lingjuan Lyu, Bo Li, and Xingjun Ma. 2021. Neural Attention Distillation: Erasing Backdoor Triggers from Deep Neural Networks. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[44] Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. 2018. Fine-Pruning: Defending Against Backdooring Attacks on Deep Neural Networks. In *Proc. of the International Symposium on Research in Attacks, Intrusions and Defenses (RAID)*, Michael Bailey, Thorsten Holz, Manolis Stamatogiannakis, and Sotiris Ioannidis (Eds.).

[45] Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and Xiangyu Zhang. 2018. Trojaning Attack on Neural Networks. In *Proc. of the Network and Distributed System Security Symposium (NDSS)*.

[46] Teng Long, Qi Gao, Lili Xu, and Zhangbing Zhou. 2022. A survey on adversarial attacks in computer vision: Taxonomy, visualization and future directions. *Computers & Security* 121 (2022), 102847. https://doi.org/10.1016/j.cose.2022.102847

[47] Max Losch, Mohamed Omran, David Stutz, Mario Fritz, and Bernt Schiele. 2024. On Adversarial Training without Perturbing all Examples. In *The Twelfth International Conference on Learning Representations*.

[48] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards Deep Learning Models Resistant to Adversarial Attacks. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[49] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. 2016. DeepFool: A Simple and Accurate Method to Fool Deep Neural Networks. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2574–2582.

[50] Bingxu Mu, Zhenxing Niu, Le Wang, Xue Wang, Rong Jin, and Gang Hua. 2023. Progressive Backdoor Erasing via connecting Backdoor and Adversarial Attacks. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[51] Tuan Anh Nguyen and Anh Tran. 2020. Input-Aware Dynamic Backdoor Attack. In *Proc. of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (Eds.). 3454–3464.

[52] Tuan Anh Nguyen and Anh Tuan Tran. 2021. WaNet - Imperceptible Warping-based Backdoor Attack. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[53] Nobuyuki Otsu. 1979. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics* 9, 1 (1979), 62–66. https://doi.org/10.1109/TSMC.1979.4310076

[54] Soumyadeep Pal, Yuguang Yao, Ren Wang, Bingquan Shen, and Sijia Liu. 2024. Backdoor Secrets Unveiled: Identifying Backdoor Data with Optimized Scaled Prediction Consistency. In *The Twelfth International Conference on Learning Representations*.

[55] Nicolas Papernot, Patrick D. McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. 2016. The Limitations of Deep Learning in Adversarial Settings. In *Proc. of the IEEE European Symposium on Security and Privacy (EuroS&P)*. 372–387.

[56] Dongmin Park, Dimitris Papailiopoulos, and Kangwook Lee. 2022. Active Learning is a Strong Baseline for Data Subset Selection. In *Has it Trained Yet? NeurIPS 2022 Workshop*.

[57] Han Qiu, Yi Zeng, Shangwei Guo, Tianwei Zhang, Meikang Qiu, and Bhavani Thuraisingham. 2021. DeepSweep: An Evaluation Framework for Mitigating DNN Backdoor Attacks Using Data Augmentation. In *Proc. of the ACM Asia Conference on Computer and Communications Security (ASIA CCS)*.

[58] Ahmed Salem, Rui Wen, Michael Backes, Shiqing Ma, and Yang Zhang. 2020. Dynamic Backdoor Attacks Against Machine Learning Models. *CoRR* abs/2003.03675 (2020).

[59] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[60] Ali Shafahi, W. Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. 2018. Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks. In *Proc. of the Annual Conference on Neural Information Processing Systems (NeurIPS)*.

[61] Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. 2019. Adversarial training for free!. In *Proc. of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.).

[62] Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[63] Hossein Souri, Micah Goldblum, Liam Fowl, Rama Chellappa, and Tom Goldstein. 2022. Sleeper Agent: Scalable Hidden Trigger Backdoors for Neural Networks Trained from Scratch. In *Proc. of the Annual Conference on Neural Information Processing Systems (NeurIPS)*.

[64] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. 2012. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural Networks* (2012).

[65] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. 2014. Intriguing Properties of Neural Networks. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[66] Alexander Turner, Dimitris Tsipras, and Aleksander Madry. 2019. Label-Consistent Backdoor Attacks. arXiv:1912.02771

[67] Sakshi Udeshi, Shanshan Peng, Gerald Woo, Lionell Loh, Louth Rawshan, and Sudipta Chattopadhyay. 2022. Model Agnostic Defence against Backdoor Attacks in Machine Learning. *IEEE Transactions on Reliability* (2022).

[68] Miguel Villarreal-Vasquez and Bharat K. Bhargava. 2020. ConFoc: Content-Focus Protection Against Trojan Attacks on Neural Networks. *ArXiv* (2020).

[69] João Vitorino, Isabel Praça, and Eva Maia. 2023. SoK: Realistic adversarial attacks and defenses for intelligent network intrusion detection. *Computers & Security* 134 (2023), 103433. https://doi.org/10.1016/j.cose.2023.103433

[70] Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y. Zhao. 2019. Neural Cleanse: Identifying and Mitigating Backdoor Attacks in Neural Networks. In *Proc. of the IEEE Symposium on Security and Privacy*.

[71] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. 2019. Symmetric cross entropy for robust learning with noisy labels. In *Proc. of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[72] Zhenting Wang, Kai Mei, Hailun Ding, Juan Zhai, and Shiqing Ma. 2022. Rethinking the Reverse-engineering of Trojan Triggers. In *Advances in Neural Information Processing Systems*.

[73] Zhenting Wang, Kai Mei, Juan Zhai, and Shiqing Ma. 2023. UNICORN: A Unified Backdoor Trigger Inversion Framework. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[74] Cheng-Hsin Weng, Yan-Ting Lee, and Shan-Hung (Brandon) Wu. 2021. On the Trade-off between Adversarial and Backdoor Robustness. In *Proc. of the Annual Conference on Neural Information Processing Systems (NeurIPS)*.

[75] Dongxian Wu and Yisen Wang. 2021. Adversarial Neuron Pruning Purifies Backdoored Deep Models. In *Proc. of the Annual Conference on Neural Information Processing Systems (NeurIPS)*.

[76] Chaowei Xiao, Jun-Yan Zhu, Bo Li, Warren He, Mingyan Liu, and Dawn Song. 2018. Spatially Transformed Adversarial Examples. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[77] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Yuyin Zhou, Lingxi Xie, and Alan Yuille. 2017. Adversarial Examples for Semantic Segmentation and Object Detection. In *International Conference on Computer Vision*.

[78] Yi Zeng, Si Chen, Won Park, Zhuoqing Mao, Ming Jin, and Ruoxi Jia. 2021. Adversarial Unlearning of Backdoors via Implicit Hypergradient. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[79] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. 2019. Theoretically Principled Trade-off between Robustness and Accuracy. In *Proc. of the International Conference on Machine Learning (ICML)*.

[80] Jingfeng Zhang, Xilie Xu, Bo Han, Gang Niu, Lizhen Cui, Masashi Sugiyama, and Mohan Kankanhalli. 2020. Attacks Which Do Not Kill Training Make Adversarial Learning Stronger. In *ICML*.

[81] Zaixi Zhang, Qi Liu, Zhicai Wang, Zepu Lu, and Qingyong Hu. 2023. Backdoor Defense via Deconfounded Representation Learning. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[82] Pu Zhao, Pin-Yu Chen, Payel Das, Karthikeyan Natesan Ramamurthy, and Xue Lin. 2020. Bridging Mode Connectivity in Loss Landscapes and Adversarial Robustness. In *Proc. of the International Conference on Learning Representations (ICLR)*.

[83] Shihao Zhao, Xingjun Ma, Xiang Zheng, James Bailey, Jingjing Chen, and Yu-Gang Jiang. 2023. Clean-Label Backdoor Attacks on Video Recognition Models. In *Proc. of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[84] Runkai Zheng, Rongjun Tang, Jianze Li, and Li Liu. 2022. Data-free Backdoor Removal based on Channel Lipschitzness. In *Proc. of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[85] Zixuan Zhu, Rui Wang, Cong Zou, and Lihua Jing. 2023. The Victim and The Beneficiary: Exploiting a Poisoned Model to Train a Clean Model on Poisoned Data. In *Proc. of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[86] Zihao Zhu, Mingda Zhang, Shaokui Wei, Bingzhe Wu, and Baoyuan Wu. 2024. VDC: Versatile Data Cleanser based on Visual-Linguistic Inconsistency by Multimodal Large Language Models. In *Proc. of the International Conference on Learning Representations (ICLR)*.