On the Trade-off between Adversarial and Backdoor Robustness

Anonymous Authors[1]

Abstract

Deep neural networks are shown to be susceptible to both adversarial attacks and backdoor attacks. Although many defenses against an individual type of the above attacks have been proposed, the interactions between the vulnerabilities of a network to both types of attacks have not been carefully investigated yet. In this paper, we conduct experiments to study whether adversarial robustness and backdoor robustness can affect each other and find a trade-off—by increasing the robustness of a network to adversarial examples, the network becomes more vulnerable to backdoor attacks. We then investigate the cause and show how such a trade-off can be exploited by an adversary to break some existing backdoor defenses. Our findings suggest that future research on defense should take both adversarial and backdoor attacks into account when designing algorithms or robustness measures to avoid pitfalls and a false sense of security.

1. Introduction

Deep neural networks (DNNs) have achieved impressive performance in many domains such as computer vision, natural language processing, speech, and robotics, etc. However, DNNs are shown to be susceptible to both *adversarial attacks* (Szegedy et al., 2013; Goodfellow et al., 2014) and *backdoor attacks* (Liu et al., 2017; Chen et al., 2017; Gu et al., 2019). Adversarial attacks aim at fooling a model using examples (which are called adversarial examples) that are nearly indistinguishable from regular examples in human eyes or some distance measures in the input space. An adversarial example can be generated by slightly perturbing the input of a regular example in directions where the output of the model gives the highest loss. On the other hand, backdoor attacks aim at fooling the model with pre-mediated inputs. An attacker can "poison" training data by adding crafted triggers in some data points of a specific label. So, a model trained with poisoned data will perform well on a benign test set but behaves wrongly when the triggers are present in test data. The vulnerabilities of DNNs to these attacks raise concern about the robustness of security-critical machine learning systems and applications, such as autonomous cars and speech recognition authorization.

Many defenses against adversarial or backdoor attacks have been proposed recently. In particular, the adversarial training (Goodfellow et al., 2014; Madry et al., 2017) is shown to be an empirically strong method for defending adversarial attacks and has motivated many follow-up and complementary works (Hein & Andriushchenko, 2017; Kannan et al., 2018; Jakubovitz & Giryes, 2018; Qian & Wegman, 2018; Xie et al., 2019; Lin et al., 2019; Shafahi et al., 2019). To defend backdoor attacks, efforts have been made to detect and remove poisoned data (before training) (Tran et al., 2018; Chen et al., 2018) or to fine-tune the model (after training) to unlearn backdoors (Wang et al., 2019; Qiao et al., 2019).

However, most existing defense methods are designed for one type of attacks only. The interactions between the vulnerabilities of a network to adversarial and backdoor attacks have not been carefully investigated yet. In practice, a model may be trained using the data collected from the public. It may also be deployed in an open environment where the input at runtime is accessible to the third party. As attackers could manipulate both training and testing data, it is crucial to understand how the interactions, if existing, will impact the current defenses.

In this paper, we conduct experiments to study whether the adversarial and backdoor robustness has an influence on each other. The answer is *yes* as we find a trade-off—by increasing the robustness of a network to adversarial examples via adversarial training, the network becomes more vulnerable to backdoor attacks. This finding is consistent on all the real-world datasets, including MNIST (LeCun et al., 1998), CIFAR-10 (Krizhevsky & Hinton, 2009), and ImageNet (Deng et al., 2009), and across all the settings we have tested. The trade-off delivers an important message: studying and defending one type of attacks at a time is dangerous because it may lead to a false sense of security. To elaborate this, we further show that new, subtle backdoor attacks can be created by exploiting the trade-off, and then use the attacks to break two out of three well-known

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

backdoor defense approaches (Chen et al., 2018; Tran et al., 2018; Wang et al., 2019) in an adversarially robust model.

Our findings are not entirely detrimental to existing research. We did *not* break the backdoor defense (Wang et al., 2019) that lets a model unlearn backdoors after training. Thus, any enhancement to this approach, such as the work (Qiao et al., 2019) on reverse-engineering triggers via generative distribution modeling, seems to be a promising direction. The following summarizes our contributions:

- We find that there exists a trade-off between the adversarial and backdoor robustness of a DNN.
- We show, by conducting extensive experiments, that such a trade-off holds across various settings, including attack strengths, model architectures, datasets, etc.
- We investigate the reasons behind the trade-off by visualizing what is learned by the network.
- We demonstrate how an adversary can exploit the trade-off to create more concealed backdoor attacks and break some existing backdoor defenses.

Our findings have implications for both existing and future research. In particular, they give a guide on how to combine existing adversarial and backdoor defenses to achieve adversarial and backdoor robustness simultaneously. In addition, they can serve as a basis for joint adversarial and backdoor attack/defense in the future.

2. Related Works

Adversarial attack and defense. Studies (Szegedy et al., 2013; Goodfellow et al., 2014) show that DNNs are vulnerable to adversarial examples. Based on different hypotheses about the cause of adversarial examples, a plethora of defense techniques against adversarial attacks has been proposed. Many of these methods, however, have been shown to fail (Carlini & Wagner, 2017a;b; Athalye et al., 2018). The *adversarial training* (Goodfellow et al., 2014; Madry et al., 2017) is one of the few surviving approaches and has shown to work well under many conditions. The idea is to train a model using the stochastic gradient descent (SGD) algorithm with minibatches containing adversarial examples dynamically computed (based on the current model weights) by an attack model that simulates some known adversarial attacks. So, the network can learn not to be fooled by the adversarial examples. Most recent defense techniques (Hein & Andriushchenko, 2017; Kannan et al., 2018; Jakubovitz & Giryes, 2018; Qian & Wegman, 2018; Xie et al., 2019; Lin et al., 2019; Shafahi et al., 2019) are designed to work with or to improve adversarial training.

Backdoor attack and defense. Studies (Liu et al., 2017; Chen et al., 2017; Gu et al., 2019) show that a model can be injected backdoors (or "trojans") if it is trained by poisoned data, where some examples of a specific class contain crafted triggers, and behaves wrongly when the triggers are present in test data (see Figure 1). Depending on whether a trigger changes the label of an example or not, existing backdoor attacks can be divided into the *dirty-label* ones (Liu et al., 2017; Chen et al., 2017; Gu et al., 2019) and *clean-label* ones (Shafahi et al., 2018; Zhu et al., 2019; Turner et al., 2019). Generally, the clean-label attacks are preferred by adversaries because the poisoned examples have correct labels and therefore are harder to detect during data preprocessing or by a defender. However, a trigger of clean-label attacks, which is added to the input only, needs to be stronger (i.e., more learnable) to bias the model. Currently, most existing clean-label attacks either assume the model is pertained (Shafahi et al., 2018; Zhu et al., 2019) or use an auxiliary model (Turner et al., 2019) to understand what features will be learned by the model and then use these features to enhance the trigger.

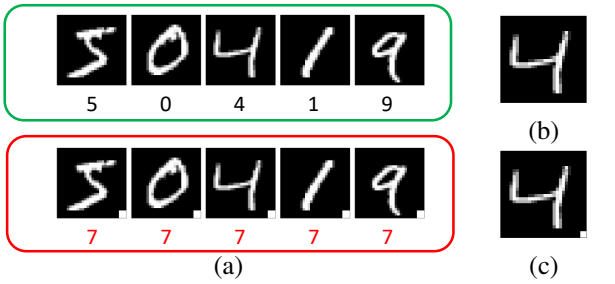


Figure 1. An example backdoor attack on the MNIST dataset. (a) Benign (green) and poisoned (red) training data. (b) A test data instance without trigger will be correctly classified by the model. (c) A test instance with trigger (at the bottom-right corner) will always be predicted as "7."

Some techniques have been proposed to defend backdoor attacks, which can be roughly divided into the *pre-training* and *post-training* defenses. The pre-training approaches (Tran et al., 2018; Chen et al., 2018) detect and remove poisoned data so a model can be properly trained. On the other hand, the post-training defenses (Wang et al., 2019; Qiao et al., 2019) reverse-engineers the potential triggers from a model with backdoors and then fine-tunes the model using a newly created dataset where the potential triggers are applied to data points of *all* classes. So, during the fine-tuning, the model will find the triggers useless for making correct predictions and thus unlearn backdoors.

Interactions. There are relatively few studies that take both adversarial and backdoor attack/defense into account. The study (Mahloujifar et al., 2019) shows that data poisoning can also be used to degrade the adversarial robustness of a model. Another study (Shan et al., 2019) uses backdoors as a honeypot that lures adversarial attacks into generating easy-

to-detect adversarial examples. However, none of the above works studies a fundamental question: does the adversarial and backdoor robustness of a network affect each other?

3. The Trade-off and Its Cause

In this section, we show that simultaneous adversarial and backdoor robustness cannot be trivially achieved because there exists a trade-off between them. We also investigate the cause of the trade-off.

3.1. Experiments

As the adversarial training is currently commonly used to defend adversarial attacks, it is important to understand whether it affects backdoor robustness. To begin with, we train two networks of the same architecture using regular and adversarial training, respectively, and compare the backdoor robustness of the two models after training. We run the experiment on the MNIST, CIFAR-10, and ImageNet datasets.

Settings. We follow the settings used by (Madry et al., 2017) to configure the networks and training algorithms. Specifically, we use the projected gradient descent (PGD) with an l_∞-norm constraint as the attack model of the adversarial training algorithm and set its parameters epsilon/step size/number of iterations to 0.3/0.05/10 for MNIST, 8/2/5 for CIFAR-10, and 8/2/5 for ImageNet, respectively. In terms of network architecture, we use a naive CNN for MNIST, ResNet-32 for CIFAR-10, and pretrained ResNet-50[1] for ImageNet. We implement all the models using TensorFlow and train them on a cluster of machines with 80 NVIDIA Tesla V100 GPUs.

Evaluation. To measure the backdoor robustness of the two networks, we devise a new clean-label backdoor attack that randomly samples 50% of the examples of a target label from the training set and adds a backdoor trigger at the bottom-right corner of each sampled image. We use different triggers for different datasets, as shown in Figure 2. The sizes of triggers are set to 3×3 pixels for MNIST and CIFAR-10, and 21×21 pixels for ImageNet. Note that this attack is *weaker* than the state-of-the-art clean-label backdoor attacks (Shafahi et al., 2018; Zhu et al., 2019; Turner et al., 2019) because it does not use the weights of a pretrained or auxiliary model to "enhance" a trigger (i.e., to make the trigger easier to learn). However, the lack of the enhancement process prevents our backdoor attack from interfering in the adversarial training. After training the two networks using poisoned data, we evaluate the performance of the two networks by 1) (clean) *accuracy* on the benign test set, 2) *adversarial robustness*, that is, the accuracy on an adversarial test set generated by PGD, 3) *success rate of the backdoor attack*, which records the portion of the poisoned test examples that are wrongly predicted as the target label by a model.

[1]Available at https://github.com/tensorflow/models/tree/master/official/r1/resnet.

(a) (b) (c)

Figure 2. The backdoor triggers for (a) MNIST, (b) CIFAR-10, and (c) ImageNet used by our weak clean-label backdoor attack.

Results. Table 1 shows the results of our experiment. By comparing the rows of standard and adversarial training, we can see that although the adversarial training improves adversarial robustness, it also degrades backdoor robustness—the success rate of the backdoor attack increases on all the MNIST, CIFAR-10, and ImageNet datasets. Specifically, our weak backdoor attack achieves more than 50% success rates on all datasets when applied to the adversarially trained network. This raises concern about the security of existing adversarially trained models. If their training data can be manipulated by an attacker, the models will have a high chance to predict whatever input as the target label set by the adversary.

Does the trade-off hold for other adversarial defenses? To answer this question, we implement two additional adversarial defenses, namely the Lipschitz regularization (Hein & Andriushchenko, 2017) and feature denoising layers (Xie et al., 2019) and test their performance.[2] The results, which are shown in Table 1 as well, show that 1) these two defenses do not work well when applied standalone, and 2) when paired up with the adversarial training, they results in the same trade-off.

Does the trade-off hold for different PGD settings? We further experiment with different PGD settings that affect both the adversarial training and evaluation of adversarial robustness. Specifically, we make the PGD attack stronger by increasing either its number of descent iterations or epsilon (i.e., the maximum allowable amount of perturbations) when generating an adversarial example. Table 2 summarizes the results, which show that the trade-off holds in spite of different strengths of the PGD attack. Due to the space limitation, we show only the results on CIFAR-10. The results on other datasets reveal the same trend.

Does the trade-off hold for different tolerance measures

[2]We did not run Lipschitz regularization (Hein & Andriushchenko, 2017) on ImageNet because its memory requirement does not scale to 1000 classes.

Table 1. The trade-off between adversarial and backdoor robustness given different defenses against adversarial attacks.

Dataset	Adv. Defense	Accuracy	Adv. Robustness	Backdoor Success Rate
MNIST	None (Std. Training)	99.1%	0%	17.2%
	Adv. Training	98.8%	93.4%	67.2%
	Lipschitz Reg.	99.3%	0%	5.7%
	Lipschitz Reg. + Adv. Training	98.7%	93.6%	52.1%
	Denoising Layer	96.9%	0%	9.6%
	Denoising Layer + Adv. Training	98.3%	90.6%	20.8%
CIFAR-10	None (Std. Training)	90%	0%	64.1%
	Adv. Training	79.3%	48.9%	99.9%
	Lipschitz Reg.	88.2%	0%	75.6%
	Lipschitz Reg. + Adv. Training	79.3%	48.5%	99.5%
	Denoising Layer	90.8%	0%	99.6%
	Denoising Layer + Adv. Training	79.4%	49%	100%
ImageNet	None (Std. Training)	72.4%	0.1%	3.9%
	Adv. Training	55.5%	18.4%	65.4%
	Denoising Layers	71.9%	0.1%	6.9%
	Denoising Layers + Adv. Training	55.6%	18.1%	68%

Table 2. The trade-off holds across different PGD settings.

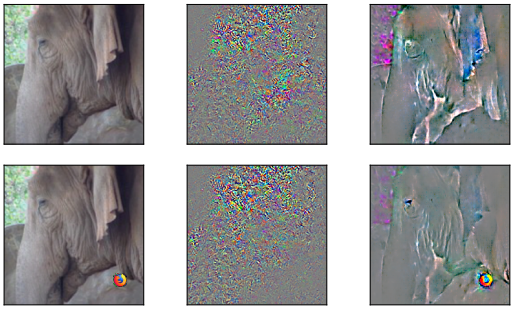
Epsilon	#Iterations	Adv. Defense	Accuracy	Adv. Robustness	Backdoor Success Rate
N/A	N/A	None (Std. Training)	90%	0%	64.1%
8	5	Adv. Training	79.3%	48.9%	99.9%
8	10	Adv. Training	76.5%	43.8%	100%
16	5	Adv. Training	62.8%	31.4%	100%

Table 3. The trade-off holds across different tolerance measures of adversarial perturbations.

p-Norm	Adv. Defense	Accuracy	Adv. Robustness	Backdoor Success Rate
l_∞	None (Std. Training)	90%	0%	64.1%
	Adv. Training	79.3%	48.9%	99.9%
l_2	None (Std. Training)	90%	0.4%	64.1%
	Adv. Training	79.7%	48.3%	99.9%

Table 4. The trade-off holds regardless of model capacities.

Model Architecture	Adv. Defense	Accuracy	Adv. Robustness	Backdoor Success Rate
[16,16,32,64]	None (Std. Training)	90%	0%	64.1%
	Adv. Training	79.3%	48.9%	99.9%
[32,32,64,128]	None (Std. Training)	91.5%	0%	52.6%
	Adv. Training	83.7%	50.4%	99.8%



(a) (b) (c)

Figure 3. The saliency maps of the regularly and adversarially trained networks. (a) Benign (top) and poisoned (bottom) images from the ImageNet dataset. (b) Saliency maps of the regularly trained network given the benign (top) and poisoned (bottom) images. (c) Saliency maps of the adversarially trained network given the benign (top) and poisoned (bottom) images.

of adversarial perturbations? Next, we study whether using different p-norms in the adversarial training and evaluation will invalidate the trade-off. We run a new set of experiments using the l_2-norm and summarize the results in Table 3. Again, the trade-off holds in spite of different tolerance measures of adversarial perturbations.

Does the trade-off hold for different model capacities? Finally, we test whether the capacity of a model has an influence on our finding. We double the number of filters in each layer of a network and run a new set of experiments. Table 4 shows the results, and we can see that that model capacity does not seem to affect the trade-off.

We will explore more backdoor triggers and study their effects in Section 4.

3.2. The Cause

To understand why an adversarially robust model is more vulnerable to backdoor attacks, we investigate what was learned by the model using visualization techniques. Figure 3 shows the saliency maps (i.e., the gradients of a model prediction with respect to the input) of the regularly and adversarially trained networks given a benign and poisoned image from the ImageNet, respectively. We can see that the adversarially trained network relies more on high-level features, which better align with human perception, to make a prediction. This is consistent with previous findings (Tsipras et al., 2018; Ilyas et al., 2019) that adversarial examples can be partially attributed to the presence of *non-robust features* (i.e., features that are highly predictive, yet brittle and incomprehensible to humans) in real-world datasets.

As an adversarially strong network relies more on robust, high-level features to make predictions, it also tends to learn from a backdoor trigger because the trigger provides robust features that are made to be strongly correlated with the target label. This explains the widespread existence of the trade-off we have discovered.

(a) (b) (c)

Figure 4. Example clean-label backdoor triggers of different types: (a) sticker, (b) watermark, and (c) channel. The channel trigger is added in the same position as the sticker trigger.

The above also provides a guide to designing effective backdoor attacks against an adversarially robust model: one should use patterns with strong high-level characteristics instead of low-level, brittle features as triggers. This is why we use the triggers shown in Figure 2 in our experiments.

4. Exploiting the Trade-off

We show in this section how the adversaries may exploit the trade-off discovered in Section 3 to create more subtle backdoor attacks and break existing defenses.

4.1. More Concealed Backdoor Triggers

A backdoor attacker needs to successfully place triggers in the training set in order to inject backdoors into a model. The triggers, however, may be detected and removed by humans or algorithms during data preprocessing. Here we study how the trade-off can be used to create backdoor triggers that are more subtle for humans to perceive. We will discuss how to evade the detection algorithms in the next subsection.

The backdoor attack used in Section 3 is a clean-label attack, which is already harder to detect than the dirty-labels ones (Liu et al., 2017; Chen et al., 2017; Gu et al., 2019). In addition to using clean labels, we show that an adversary can make a trigger more subtle by adjusting 1) trigger type, 2) trigger size, 3) poisoned data rate, and 4) trigger position, to be detailed later. We apply all the trigger variants to both the regularly and adversarially trained networks using the same settings described in Section 3 and then evaluate the performance of the networks. We find that the adversarial robustness of the two networks does not vary much given different triggers. Therefore, we focus on backdoor robustness. Note that all these trigger variants are of clean labels and none of them has been reported to work without additional feature engineering (Shafahi et al., 2018; Zhu et al., 2019; Turner et al., 2019) in the literature.

Trigger type. We consider three trigger types, namely the

Table 5. The performance of clean-label triggers of different types.

Dataset	Trigger Type	Adv. Defense	Backdoor Success Rate
MNIST	Sticker	None	17.2%
		Adv. Training	**67.2%**
	Watermark	None	17.7%
		Adv. Training	**84.9%**
CIFAT-10	Sticker	None	64.1%
		Adv. Training	**99.9%**
	Watermark	None	84.2%
		Adv. Training	**90.9%**
	Channel	None	33.5%
		Adv. Training	**72.4%**
ImageNet	Sticker	None	3.9%
		Adv. Training	**65.4%**
	Watermark	None	13.4%
		Adv. Training	**46.8%**
	Channel	None	1.1%
		Adv. Training	**16.4%**

sticker, watermark, and channel, as shown in Figure 4. A channel trigger zeros out the blue channel of the pixels in a specific region. Table 5 shows the backdoor robustness of the two networks trained with these triggers.[3] By exploiting the trade-off, all types of triggers can be used to inject backdoors into the adversarially trained network. The backdoor attacks with the sticker and watermark triggers can achieve more than 45% success rates on all the datasets. The channel trigger, despite being very "weak" from human eyes, gives more than 70% success rates on CIFR-10 and a 16% success rate on ImageNet that cannot be safely neglected.

Trigger size. We also study how small can a trigger be to create a valid backdoor attack. We test the sticker triggers of sizes 3×3, 2×2, and 1×1 on MNIST and CIFAR-10, and 21×21, 14×14, and 7×7 on ImageNet. The results, which are summarized in Table 6, show that tiny triggers can still successfully inject backdoors into the adversarially trained network. Surprisingly, even the smallest possible triggers of size 1×1 achieve above 50% success rates on MNIST and CIFAR-10.

Poisoned data rate. Next, we see if an adversary can inject backdoors into the models when fewer examples are accessible. We sample 50%, 25%, 10% examples of the target label as the poisoned examples and evaluate the performance of the two networks on each dataset. As Table 7 shows, the ad-

[3]We do not apply the channel triggers to MNIST images because the images have only one channel.

Table 6. The performance of clean-label triggers of different sizes.

Dataset	Trigger Size	Adv. Defense	Backdoor Success Rate
MNIST	3×3	None	17.2%
		Adv. Training	**67.2%**
	2×2	None	15%
		Adv. Training	**62.5%**
	1×1	None	12.2%
		Adv. Training	**57%**
CIFAT-10	3×3	None	64.1%
		Adv. Training	**99.9%**
	2×2	None	47.1%
		Adv. Training	**99.9%**
	1×1	None	31.1%
		Adv. Training	**69.8%**
ImageNet	21×21	None	3.9%
		Adv. Training	**65.4%**
	14×14	None	3.2%
		Adv. Training	**49.6%**
	7×7	None	3.7%
		Adv. Training	**18.2%**

Table 7. The performance of clean-label triggers applied to different portions of training data.

Dataset	Poisoned Data	Adv. Defense	Backdoor Success Rate
MNIST	50%	None	17.2%
		Adv. Training	**67.2%**
	25%	None	11.4%
		Adv. Training	**58%**
	10%	None	8.4%
		Adv. Training	**52.6%**
CIFAT-10	50%	None	64.1%
		Adv. Training	**99.9%**
	25%	None	30.8%
		Adv. Training	**95.4%**
	10%	None	15.2%
		Adv. Training	**88.9%**
ImageNet	50%	None	3.9%
		Adv. Training	**65.4%**
	25%	None	1.6%
		Adv. Training	**46.6%**
	10%	None	0.6%
		Adv. Training	**20.8%**

Table 8. The performance of clean-label triggers of different sizes.

Dataset	Trigger Position	Adv. Defense	Backdoor Success Rate
MNIST	Fixed	None	17.2%
		Adv. Training	**67.2%**
	Random	None	4.6%
		Adv. Training	**59.9%**
CIFAT-10	Fixed	None	64.1%
		Adv. Training	**99.9%**
	Random	None	31.4%
		Adv. Training	**95.1%**
ImageNet	Fixed	None	3.9%
		Adv. Training	**65.4%**
	Random	None	3.4%
		Adv. Training	**63.5%**

versarially trained network can still be successfully injected backdoors when fewer data are poisoned.

Trigger position. Finally, we study whether it is possible to create a backdoor attack by using the triggers that are placed at random corners of the training images. Table 8 shows the results. We can see that the attack works nicely against the adversarially trained network regardless of the trigger positions.

The clean-label, veiled backdoor attacks above, which are shown to work for the first time, motivate us to examine the effectiveness of existing backdoor defenses.

4.2. Breaking Backdoor Defenses

We consider three well-known backdoor defenses (Tran et al., 2018; Chen et al., 2018; Wang et al., 2019) and, to our surprise, find that two of them (Tran et al., 2018; Chen et al., 2018) fail to defend the clean-label attacks in an adversarially trained network.

The two works we break are both pre-training defenses (see Section 2 for the categorization of existing backdoor defenses) whose goal is to detect and remove poisoned data before training. A network with backdoors behaves normally when taking benign examples as input but wrongly predicts instances with triggers as the target label. This implies that there may be some neurons in the network that are activated by trigger features while others become active when normal features are present. Therefore, the distributions of neuron activations may be different for benign and poisoned examples. This enables the detection of poisoned examples by examining activations.

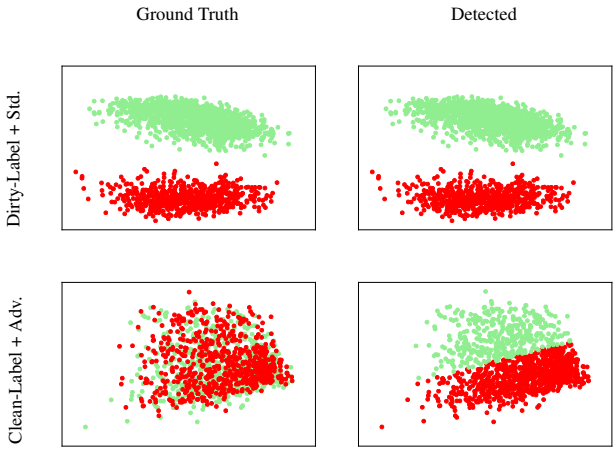


Figure 5. Distributions of benign (green) and poisoned (red) examples of the target label from ImageNet in the latent spaces of different models with backdoors.

Spectral signatures. The work (Tran et al., 2018) first trains an auxiliary network using all examples, possibly with triggers, in a dataset. By assuming that the target label used by an attacker is known, the defense 1) computes a vector of neuron activations in a hidden layer of the auxiliary network for each example of the target label, 2) extracts a *spectral signature* from each activation vector via the principal component analysis (PCA), and 3) identifies the examples whose spectral signatures deviate most from the "center" (i.e., the average of all spectral signatures) as poisoned examples. Then, the final model can be trained by data excluding the detected examples. We implement this defense and apply it to two networks where one is regularly trained by data with the dirty-label backdoor triggers (Gu et al., 2019) and another is adversarially trained by data with the clean-label attack shown in Figure 1. We follow the settings in Section 3 and use the activations of the first layer of the third and fourth convolutional blocks to compute the spectral signature for each example on CIFAR-10 and ImageNet, respectively. The performance of the defense is shown in Table 9(a). As we can see, the backdoor robustness of the two networks is not significantly improved by the defense, although the mean deviation of the spectral signatures of remaining examples from the center does reduce greatly. In particular, the defense can only detect about 50% of the examples poisoned with the clean-label triggers. In other words, these triggers are harder to detect not only from human eyes but also from the detection algorithm's perspective. Taking advantage of the trade-off, we can still successfully inject backdoors into the adversarially trained network using the remaining poisoned data.

Activation clustering. The work (Chen et al., 2018) shares a similar idea to the above except it does not use spectral signatures of neuron activations to detect triggers. Instead, it

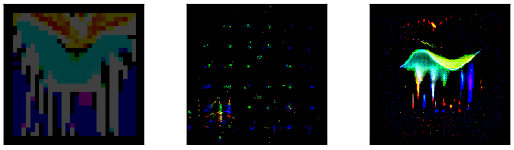
Table 9. The performance of the (a) pre-training (Tran et al., 2018) and (b) post-training (Wang et al., 2019) backdoor defenses.

Dataset	Backdoor Attack	Succ. Rate w/o Defense	Succ. Rate w/ Defense	Detection Rate	Deviation
CIFAR-10	Dirty-Label Sticker + Std. Training	100%	98.9%	81.6%	16.7
	Clean-Label Sticker + Adv. Training	99.9%	97.1%	50.1%	0.08
ImageNet	Dirty-Label Sticker + Std. Training	98.1%	0.1%	100%	151.7
	Clean-Label Sticker + Adv. Training	65.4%	58.7%	50.5%	2.39

(a)

Dataset	Trigger Type	Trigger Label	Training Algorithm	Succ. Rate w/o Defense	Succ. Rate w/ Defense
CIFAR-10	Sticker	Dirty	Std. Training	100%	0.1%
		Clean	Adv. Training	99.9%	0%
	Complex Watermark	Dirty	Std. Training	99.7%	39.3%
		Clean	Adv. Training	92.7%	1.2%
ImageNet	Sticker	Dirty	Std. Training	98.1%	2.3%
		Clean	Adv. Training	65.4%	1.1%
	Complex Watermark	Dirty	Std. Training	96.3%	39.8%
		Clean	Adv. Training	49.7%	4.0%

(b)



(a) (b) (c)

Figure 6. Reverse-engineered backdoor triggers on ImageNet. (a) Original complex watermark trigger used to poison training data. (b) Trigger reverse-engineered by (Wang et al., 2019) from the regularly trained network under the dirty-label backdoor attack. (c) Reverse-engineered trigger from the adversarially trained network under the clean-label backdoor attack.

clusters examples into groups based on a similarity measure in an activation space and then removes the groups that are found suspicious by humans or algorithms. We apply this defense to the two networks and, again, see little improvement on the backdoor robustness of the adversarially trained network. We omit the results due to the space limitation but show the clusters found by the defense in Figure 5. The poisoned data with clean-label triggers look very similar to the begin examples and can easily evade detection.

5. Implications and Conclusion

In this paper, we showed, by using extensive experiments, the widespread existence of the trade-off between the adversarial and backdoor robustness of a DNN. We investigated the cause of the trade-off and demonstrated how an adversary can exploit it to create more concealed backdoor attacks and break some existing backdoor defenses.

We, however, did not break the post-training defense based on neural cleansing (Wang et al., 2019), as shown in Table 9(b). This work reverse-engineers potential triggers from a model with backdoors and then fine-tunes the model with data containing potential triggers in all labels to teach the model to unlearn backdoors (and cleanse neurons). Interestingly, it is currently considered as weak for regularly trained models because it cannot reverse-engineer complex triggers (Qiao et al., 2019). While Table 9(b) confirms this (see the rows with complex, dirty-label triggers and regularly trained models), it also shows that the trade-off *conversely strengthens* the defense when the latter is applied to the adversarially trained network. Figure 6 shows the complex backdoor trigger we used, and the defense can successfully reverse-engineer it (at least to a certain extent) from the adversarially trained network without using the advanced techniques proposed by (Qiao et al., 2019). For now, pairing up adversarial training with neural cleansing (or its variants) seems to be a quick way to achieve adversarial and backdoor robustness simultaneously.

As our future work, we plan to study algorithms and measures for joint adversarial and backdoor attack/defense. We will also study how the trade-off impacts real-world applications, such as self-driving cars, where the security of a machine learning system is in high demand.

References

Athalye, A., Carlini, N., and Wagner, D. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *arXiv preprint arXiv:1802.00420*, 2018. 2

Carlini, N. and Wagner, D. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proc. of the 10th ACM Workshop on Artificial Intelligence and Security*, 2017a. 2

Carlini, N. and Wagner, D. Towards evaluating the robustness of neural networks. In *Proc. of IEEE Symposium on Security and Privacy (S&P)*, 2017b. 2

Chen, B., Carvalho, W., Baracaldo, N., Ludwig, H., Edwards, B., Lee, T., Molloy, I., and Srivastava, B. Detecting backdoor attacks on deep neural networks by activation clustering. *arXiv preprint arXiv:1811.03728*, 2018. 1, 2, 4.2, 4.2

Chen, X., Liu, C., Li, B., Lu, K., and Song, D. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017. 1, 2, 4.1

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *Proc. of CVPR*. Ieee, 2009. 1

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 1, 2

Gu, T., Liu, K., Dolan-Gavitt, B., and Garg, S. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 7, 2019. 1, 2, 4.1, 4.2

Hein, M. and Andriushchenko, M. Formal guarantees on the robustness of a classifier against adversarial manipulation. In *Proc. of NIPS*, 2017. 1, 2, 3.1, 2

Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., and Madry, A. Adversarial examples are not bugs, they are features. In *Proc. of NeurIPS*, 2019. 3.2

Jakubovitz, D. and Giryes, R. Improving dnn robustness to adversarial attacks using jacobian regularization. In *Proc. of ECCV*, 2018. 1, 2

Kannan, H., Kurakin, A., and Goodfellow, I. Adversarial logit pairing. *arXiv preprint arXiv:1803.06373*, 2018. 1, 2

Krizhevsky, A. and Hinton, G. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*, 2009. 1

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proc. of the IEEE*, 86(11), 1998. 1

Lin, J., Gan, C., and Han, S. Defensive quantization: When efficiency meets robustness. *arXiv preprint arXiv:1904.08444*, 2019. 1, 2

Liu, Y., Ma, S., Aafer, Y., Lee, W.-C., Zhai, J., Wang, W., and Zhang, X. Trojaning attack on neural networks. In *Proc. of the Network and Distributed System Security Symposium (NDSS)*, 2017. 1, 2, 4.1

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 1, 2, 3.1

Mahloujifar, S., Diochnos, D. I., and Mahmoody, M. The curse of concentration in robust learning: Evasion and poisoning attacks from concentration of measure. In *Proc. of AAAI*, 2019. 2

Qian, H. and Wegman, M. N. L2-nonexpansive neural networks. *arXiv preprint arXiv:1802.07896*, 2018. 1, 2

Qiao, X., Yang, Y., and Li, H. Defending neural backdoors via generative distribution modeling. In *Proc. of NeurIPS*, 2019. 1, 2, 5

Shafahi, A., Huang, W. R., Najibi, M., Suciu, O., Studer, C., Dumitras, T., and Goldstein, T. Poison frogs! targeted clean-label poisoning attacks on neural networks. In *Proc. of NeurIPS*, 2018. 2, 3.1, 4.1

Shafahi, A., Najibi, M., Ghiasi, M. A., Xu, Z., Dickerson, J., Studer, C., Davis, L. S., Taylor, G., and Goldstein, T. Adversarial training for free! In *Proc. of NeurIPS*, 2019. 1, 2

Shan, S., Willson, E., Wang, B., Li, B., Zheng, H., and Zhao, B. Y. Gotta catch'em all: Using concealed trapdoors to detect adversarial attacks on neural networks. *arXiv preprint arXiv:1904.08554*, 2019. 2

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. 1, 2

Tran, B., Li, J., and Madry, A. Spectral signatures in backdoor attacks. In *Proc. of NeurIPS*, 2018. 1, 2, 4.2, 4.2, 9

Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., and Madry, A. Robustness may be at odds with accuracy. *arXiv preprint arXiv:1805.12152*, 2018. 3.2

Turner, A., Tsipras, D., and Madry, A. Label-consistent backdoor attacks. *arXiv preprint arXiv:1912.02771*, 2019. 2, 3.1, 4.1

Wang, B., Yao, Y., Shan, S., Li, H., Viswanath, B., Zheng, H., and Zhao, B. Y. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *Proc. of IEEE Symposium on Security and Privacy (S&P)*, 2019. 1, 2, 4.2, 9, 6, 5

Xie, C., Wu, Y., Maaten, L. v. d., Yuille, A. L., and He, K. Feature denoising for improving adversarial robustness. In *Proc. of CVPR*, 2019. 1, 2, 3.1

Zhu, C., Huang, W. R., Shafahi, A., Li, H., Taylor, G., Studer, C., and Goldstein, T. Transferable clean-label poisoning attacks on deep neural nets. *arXiv preprint arXiv:1905.05897*, 2019. 2, 3.1, 4.1