


Enhanced Accuracy and Robustness via Multi-Teacher Adversarial Distillation

Shiji Zhao^{1,2}, Jie Yu^{1,2}, Zhenlong Sun², Bo Zhang², and Xingxing Wei^{1*}

¹ Institute of Artificial Intelligence, Hangzhou Innovation Institute, Beihang University, Beijing, China

{zhaoshiji123,sy2106137,xxwei}@buaa.edu.cn

² WeChat Search Application Department, Tencent, Beijing, China

{richardsun,nevinzhang}@tencent.com

Abstract. Adversarial training is an effective approach for improving the robustness of deep neural networks against adversarial attacks. Although bringing reliable robustness, adversarial training (AT) will reduce the performance of identifying clean examples. Meanwhile, Adversarial training can bring more robustness for large models than small models. To improve the robust and clean accuracy of small models, we introduce the Multi-Teacher Adversarial Robustness Distillation (MTARD) to guide the adversarial training process of small models. Specifically, MTARD uses multiple large teacher models, including an adversarial teacher and a clean teacher to guide a small student model in the adversarial training by knowledge distillation. In addition, we design a dynamic training algorithm to balance the influence between the adversarial teacher and clean teacher models. A series of experiments demonstrate that our MTARD can outperform the state-of-the-art adversarial training and distillation methods against various adversarial attacks. Our code is available at <https://github.com/zhaoshiji123/MTARD>.

Keywords: Adversarial Training, Knowledge Distillation, DNNs

1 Introduction

Deep Neural Networks (DNNs) have become powerful tools for solving complex real-world learning problems, such as image classification [18,34], face recognition [38], and natural language processing [33]. However, Szegedy et al. [35] demonstrates that DNNs are vulnerable to adversarial attacks with imperceptible adversarial perturbations on input, which causes wrong predictions of DNNs. This phenomenon raises concerns about the robustness of DNNs in safety-related areas, such as autonomous driving [11], finance [23], and medical diagnosis [26].

To defend against adversarial attacks, adversarial training is proposed and shows effectiveness to acquire the adversarial robust DNNs [2,8,27]. In a broad sense, adversarial training can be regarded as a data augmentation method, where adversarial examples generated by the adversarial attacks are used as part

* corresponding author

of the model training set to enhance the model robustness against adversarial attacks. At the mathematical level, a min-max optimization problem can express the adversarial training process, where the inner maximization can be regarded as generating adversarial examples, and outer minimization is to train the model by adversarial examples generated in maximization.

While improving the robustness of DNNs, adversarial training has several shortcomings in some general scenes. Firstly, the robustness of models obtained from adversarial training is related to the size of models. In general, the larger model means the better robust performance [41,48,12,7,47]. However, due to the limitations of various practical factors, a large model is often not favored in actual deployment [32]. Secondly, the accuracy of identifying clean examples by adversarial trained DNNs is far worse than normal trained DNNs, which limits large-scale use in practical scenarios. Some researchers [45] try to reduce the negative effects of adversarial training bringing for clean accuracy, but the effect is still not ideal.

In this paper, we investigate the method to improve both the clean and robust accuracy of small DNNs by adversarial distillation. Adversarial Robustness Distillation (ARD) is used to boost the robustness of small models by distilling from large robust models [12,7,47], which treats large models as teachers and small models as students. Although the previous work (RSLAD) [48] improves the robustness via robust soft labels, the clean accuracy is still not ideal compared with the performance of regular training. Inspired by multi-task learning [37], we propose Multi-Teacher Adversarial Robustness Distillation (MTARD) by using different teacher models, each teacher model is responsible for what they are proficient in. To improve both robustness of the student model and the accuracy of identifying clean examples, we apply a robust teacher model and a clean teacher model to guide robustness and accuracy simultaneously. However, due to the complexity of neural networks, teacher models have different degrees of influence on student models, which can even cause catastrophic forgetting. To alleviate this phenomenon, we design a joint training algorithm to dynamically adjust the influence of the teacher models on the student network at different stages in adversarial distillation. All in all, the main contributions of this work are three-fold:

- We propose a novel adversarial robustness distillation method called Multi-Teacher Adversarial Robustness Distillation (MTARD), which applies multiple teacher models to improve student models’ clean and robust accuracy by adversarial distillation.
- We design a joint training algorithm based on the proposed Adaptive Normalization Loss to balance the influence on the student model between the adversarial teacher model and the clean teacher model, which is dynamically determined by the historical training information.
- We empirically verify the effectiveness of MTARD in improving the performance of small models. For the models trained by our MTARD, the Weighted Robust Accuracy (a metric to evaluate the trade-off between the clean accuracy and robust accuracy) has been greatly improved compared with the

state-of-the-art adversarial training and distillation method against white-box and black-box attacks. Especially for black-box Square Attack, MTARD can most enhance the Weighted Robust Accuracy by 6.87% and 5.12% for MobileNet-V2 on CIFAR-10 and CIFAR-100 respectively.

2 Related Work

2.1 Adversarial Attack

Since Szegedy et al. [35] proposed that adversarial examples can mislead the deep neural network, lots of effective adversarial attack methods, such as Fast Gradient Sign Method (FGSM)[13], Projected Gradient Descent Attack (PGD) [27], Carilini and Wagner Attack (CW) [5], and Jacobian-based Saliency Map Attack (JSMA) [29] are proposed. Existing attack methods can be divided into white-box attacks and black-box attacks. White-box attacks are to know all the parameter information of the attacked model when generating adversarial examples, and black-box attacks are to know only part of the attacked model’s output when generating adversarial examples. In general, black-box attacks simulate the model gradient by repeatedly querying the target model (query-based attack) [4,6,1,40,43] or searching for an alternative model similar to the target model (transfer-based attack) [9,19,24]. Since attackers hardly know the model parameters of the target model in practical applications, the model’s performance against black-box attacks can better reflect the real robustness.

2.2 Adversarial Training

Adversarial Training [25,20,46,45,3] is seen as an effective way to defend against adversarial attack [48]. Madry et al. [27] formulate Adversarial Training as a minimax optimization problems formulated as follows:

$$\min_{\theta} E_{(x,y) \sim \mathcal{D}} [\max_{\delta \in \Omega} \mathcal{L}(f(x + \delta; \theta), y)], \quad (1)$$

where f represents a deep neural network, θ represents the weight of f , D represents a distribution of the clean example x and the ground truth label y . $\mathcal{L}(f(x + \delta; \theta), y)$ represents loss function of updating the training model. δ represents the adversarial perturbation, and Ω represents a bound, which can be defined as $\Omega = \{\delta : \|\delta\| \leq \epsilon\}$ with the maximum perturbation strength ϵ .

Much work has been proposed to further improve the robustness by adversarial training. Zhang et al. [45] try to make a balance between robustness and clean performance (TRADES), Wang et al. [39] further improves performance by Misclassification-Aware adveRsarial Training (MART). Wu et al. [41] believe the use of bigger models can improve the model robustness.

Previous work [21,30] use two indexes: clean accuracy and robust accuracy, as the metric to evaluate the comprehensive performance of the model. Nezihe et al. [16] proposes a metric named Weighted Robust Accuracy to balance the trade-off between clean accuracy and robust accuracy, which is used in our experiments.

2.3 Adversarial Robustness Distillation

Knowledge distillation can transfer the performance of other models to the target model. Due to the ability to transfer better model performance to other model performance, it has been widely studied in recent years and works well in some actual deployment scenarios combined with network pruning and model quantization [17,28,15]. Knowledge distillation has a wide range of practicality and can also be applied to various practical tasks such as classification [42], image detection [10], and natural language processing [36]. Knowledge distillation can briefly be formulated as the following optimization:

$$\arg \min_{\theta_s} (1 - \alpha) \mathcal{L}(S(x), y) + \alpha \tau^2 KL(S^\tau(x), (T^\tau(x))), \quad (2)$$

where KL is Kullback–Leibler divergence loss, τ is a temperature constant used in the output of network (combined with softmax operation), \mathcal{L} represents the loss function of updating the training model, which can usually be regarded as cross-entropy in traditional knowledge distillation method.

In general, adversarial training can bring better robustness for the larger model [31,48,12,7,47]. RAD [12] proposes that using a bigger and stronger model as the teacher model in adversarial training allows better adversarial training methods. IAD [47] performs adversarial knowledge distillation by using the teacher model with the same structure as the student model. RSLAD [48] uses the soft label generated by the teacher model instead of the one-hot label as the label used for producing adversarial examples in the process of adversarial training, which can also improve the robustness of the student model.

3 Methodologies

In this section, we propose our MTARD method to guide the process of adversarial distillation with multiple teacher models and design a dynamic training method that controls the degree of influence between the adversarial teacher model and the clean teacher model toward the student model.

3.1 Multi-Teacher Adversarial Robustness Distillation

As we mentioned before, although adversarial training is very effective in improving robustness, the improvement of standard adversarial training methods for small models is not as obvious as that for large models. Therefore, many methods on transferring the robustness of large models to small models through knowledge distillation have been proposed [45,12,48]. Although these methods can improve the robustness of small models, the adversarial training itself will hurt the ability of models to identify clean examples. Therefore, the core problem to be solved in this section is how to improve both clean and robust accuracy in the adversarial training, then our Multi-Teacher Adversarial Robustness Distillation (MTARD) is proposed.

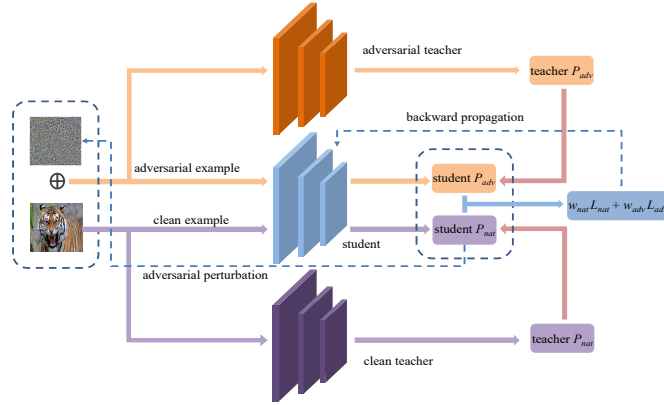


Fig. 1. The framework of our Multi-Teacher Adversarial Robustness Distillation (MTARD). In the process of MTARD, we firstly generate adversarial examples by student model. Then we produce L_{nat} and L_{adv} of the student by the guide of the clean teacher and the adversarial teacher respectively. Finally, we use Adaptive Normalization Loss to balance the influence between clean teacher and adversarial teacher and update student model.

Inspired by multi-task learning [37], we hope to not only improve the robustness of the model but also maintain the clean accuracy in Adversarial Distillation. The previous adversarial distillation method only brings a single model trained by adversarial training, which has strong robustness but weak recognition ability for clean image. As the only guide, the student model often fits the distribution of the teacher model, resulting in a lower ability to identify clean examples. Using GT one-hot labels as a learning objective to improve the clean recognition rate is still not an ideal option [48]. Therefore we additionally bring a pre-trained clean teacher model to guide the process of Adversarial Distillation.

The training of the student in MTARD is still based on adversarial training. With the guidance of an adversarial teacher and a clean teacher in knowledge distillation, we hope the student can learn robustness from the adversarial teacher and the ability to identify clean examples from the clean teacher. To produce the soft label of fulfilling the responsibilities of both teachers, the inputs of the clean teacher are initial clean examples from original datasets. In contrast, the inputs of the adversarial teacher are adversarial examples produced by the student model in the inner maximization. The student inputs are divided into clean examples and adversarial examples. The outputs of clean examples and adversarial examples will be guided by adversarial soft label and clean soft label to supervise the student model training in outer minimization. The minimax optimization framework of basic MTARD is defined as follows:

$$\arg \min_{\theta_S} (1 - \alpha)KL(S(x_{nat}), T_{nat}(x_{nat})) + \alpha KL(S(x_{adv}), T_{adv}(x_{adv})), \quad (3)$$

$$x_{adv} = \arg \max_{\delta \in \Omega} CE(S(x_{nat} + \delta; \theta_S), y), \quad (4)$$

where x_{adv} are adversarial examples produced by clean examples x_{nat} , $S(x)$ are the abbreviations for $S(x; \theta_S)$, which represents student network S with parameters θ_S . T_{nat} and T_{adv} respectively represent the clean teacher model and adversarial teacher model. α is a constant in the basic proposal. The value p could be specified as different values according to the requirement. We choose cross-entropy as previous work in the maximization.

The goal of MTARD is to learn a small student network that has both robust performance as the adversarial pre-trained teacher network and clean performance as the clean pre-trained teacher network. In the actual operation process, however, the simultaneous knowledge distillation of different teacher models will affect the learning of the student model. The student’s learning intensity from multiple teachers can not be easily controlled. If a teacher dominates students’ learning, the student model can hardly learn the relative ability from another teacher, even causing catastrophic forgetting. So handling situations with multiple teachers becomes a problem to be solved in the next subsection.

3.2 Adaptive Normalization Loss in MTARD

In order to get both clean and robust accuracy, a strategy is needed to balance the influence between the adversarial teacher and the clean teacher. On the mathematical level, the total loss in MTARD ultimately used for the student model update at time t can be represented as $L_{total}(t)$, which can be formulated as follows:

$$L_{total}(t) = w_{adv}(t)L_{adv}(t) + w_{nat}(t)L_{nat}(t). \quad (5)$$

Since the degree of the teacher’s influence on the student can be expressed as the value of the adversarial loss $L_{adv}(t)$ and clean loss $L_{nat}(t)$, the vital to control the learning degree from multiple teachers is to control the loss weight of $w_{adv}(t)$ and $w_{nat}(t)$. Inspired by gradient regularization methods in multi-task learning [2], we propose an algorithm to control the steady learning from the adversarial teacher and clean teacher, which is called as the Adaptive Normalization Loss used in our MTARD.

To better introduce the Adaptive Normalization Loss, we give a formal description from a generalized view. Suppose there are multiple teacher models to jointly guide the training process of the student network. Each teacher model is associated with a loss function L_i , and a loss weight w_i , thus the total loss L_{total} can be considered as the optimization of multiple losses as follows:

$$L_{total}(t) = \sum_{i=1}^N w_i(t)L_i(t), \quad (6)$$

where N represents the number of multiple losses, $L_i(t)$ and $w_i(t)$ respectively mean the i -th loss and loss weight at time t . The goal is to place $L_i(t)$ on a common scale through their relative magnitudes by dynamically adjusting $w_i(t)$ with similar rates at each update and each $L_i(t)$ has a relatively fair drop after the entire update process. The final trained model can be equally affected by various influencing factors behind the losses.

In order to choose the criterion to measure the decline of multiple losses, we choose a relative loss $\tilde{L}(t)$ following Athalye et al. [2], which is defined as follows:

$$\tilde{L}_i(t) = L_i(t)/L_i(0), \quad (7)$$

where $L_i(0)$ is the i -th loss value at time 0. Especially in our setting, we assume that the smaller value of $L_i(t)$ compared with $L_i(0)$ means the model fitting the target. $\tilde{L}_i(t)$ as a metric can reflect the change amplitude of $L_i(t)$ from begin to time t . Lower value of $\tilde{L}_i(t)$ corresponds to a relatively faster training speed for $L_i(t)$. By introducing relative loss $\tilde{L}_i(t)$, we can dynamically balance $L_i(t)$ influence toward $L_{total}(t)$ as an objective standard to get relative loss weight $r_i(t)$, which can be formulated as follows:

$$r_i(t) = [\tilde{L}_i(t)]^\beta / \sum_{i=1}^N [\tilde{L}_i(t)]^\beta, \quad (8)$$

where $[\tilde{L}_i(t)]^\beta$ denotes the $\tilde{L}_i(t)$ power of β , and β is set to empower the $L_i(t)$ on the disadvantaged side to control the degree of updating the loss weight. The bigger β strengthens the disadvantaged losses, which is applicable when the loss value is too different. β equal to 1 is in line with the situation that all $L_i(t)$ have similar influence abilities. We simplify the update formula for $w_i(t)$, which can be formulated as follows:

$$w_i(t) = r_w r_i(t) + (1 - r_w) w_i(t - 1), \quad (9)$$

where r_w means the learning rate of $w_i(t)$. Our MTARD can be considered as an Adaptive Normalization Loss optimization with $N = 2$, the $L_{adv}(t)$ and $L_{nat}(t)$ can be regarded as $L_1(t)$ and $L_2(t)$. In the framework of Adaptive Normalization Loss, the update process of $w_{adv}(t)$ and $w_{nat}(t)$ can be formulated as follows:

$$w_{adv}(t) = \frac{r_w [L_{adv}(t)/L_{adv}(0)]^\beta}{[L_{nat}(t)/L_{nat}(0)]^\beta + [L_{adv}(t)/L_{adv}(0)]^\beta} + (1 - r_w) w_{adv}(t - 1), \quad (10)$$

$$w_{nat}(t) = 1 - w_{adv}(t). \quad (11)$$

On a practical level, the Adaptive Normalization Loss used in MTARD can inhibit the rapid growth of a stronger teacher throughout the training cycle. If a teacher over-instructs a student compared with another teacher over a period of

Algorithm 1 MTARD

Require Initialize student model $S(x|\theta_S)$, pretrained teacher model T_{adv} and T_{nat} , the dataset \mathcal{D} including the benign clean example x_{nat} and the label y , the threat bound Ω , the initialized perturbation δ

- 1: **for** $t = 0$ to $max\text{-}step$ **do**
- 2: Acquire adversarial example $x_{adv} = \arg \max_{\delta \in \Omega} CE(S(x_{nat} + \delta; \theta_S), y)$
- 3: Compute Adversarial Loss $L_{adv}(t) = KL(S(x_{adv}), T_{adv}(x_{adv}))$
- 4: Compute Clean Loss $L_{nat}(t) = KL(S(x_{nat}), T_{nat}(x_{nat}))$
- 5: **if** $t = 0$ **then**
- 6: Record L_{adv} and L_{nat} as $L_{adv}(0)$ and $L_{nat}(0)$ respectively
- 7: **end if**
- 8: Update $w_{nat}(t)$ and $w_{adv}(t)$ by Eq. 10 and Eq. 11
- 9: $\theta_S \leftarrow \theta_S - \eta \nabla_{\theta_S} \left\{ w_{adv}(t)L_{adv}(t) + w_{nat}(t)L_{nat}(t) \right\}$
- 10: **end for**

time, the Adaptive Normalization Loss can dynamically suppress the teacher’s teaching ability by controlling the loss weight, while the ability of the other teacher will become stronger in the following period. However, this trend is not absolute. If noticing that the original strong teacher has become weaker, Adaptive Normalization Loss will make the original strong teacher stronger again. Finally, the student can learn well from two teachers to gain both clean and robust abilities rather than appearing partial ability under the adjustment of Adaptive Normalization Loss.

The complete process of MTARD with Adaptive Normalization Loss is in Algorithm 1. Compared with other existing adversarial distillation, our method has several advantages. Firstly, the loss weights are dynamically updated without any deliberate tuning of loss weight hyper-parameters and can fit the changes as training epochs increase, which is important to fit various changing scenarios but not limited in adversarial training. Secondly, our method can fit on different teacher models and student models no matter how strong or weak the teachers’ performance is, which can be controlled by Adaptive Normalization Loss. Thirdly, our method pays more attention to the performance of Weighted Robust Accuracy, which measures the trade-off between clean and robust accuracy, and thus is more valued and focused in the overall performance of the model.

4 Experiments

Initially, we describe the experimental setting, and evaluate the clean accuracy and robust accuracy of four baseline defense methods and our MTARD under prevailing white-box attack methods. Moreover, our method is evaluated under the black-box attack including transfer-based and square-based attacks. We also conduct an ablation study to demonstrate the effectiveness of our method.

Table 1. Performance of the teacher networks used in our experiments, RN and WRN are the abbreviations of ResNet and WideResNet.

Dataset	Model	Clean Acc	FGSM	PGD _{sat}	PGD _{trades}	CW _∞	Type
CIFAR-10	RN-56	93.18%	19.23%	0	0	0	Clean
	WRN-34-10	84.91%	61.14%	55.30%	56.61%	53.84%	Adv
CIFAR-100	WRN-22-6	76.65%	4.85%	0	0	0	Clean
	WRN-70-16	60.96%	35.89%	33.58%	33.99%	31.05%	Adv

4.1 Experimental Settings

We conduct our experiments on two datasets including CIFAR-10 and CIFAR-100 [22], and consider natural train method and four state-of-the-art methods of adversarial training and adversarial robustness distillation as comparison method: SAT [27], TRADES [45], ARD [12] and RSLAD [48].

Student and Teacher Networks. For the selection of models, We consider two student networks including ResNet-18 [18] and MobileNet-V2 [32] following previous work. As for teacher model, we choose two clean teacher networks including ResNet-56 for CIFAR-10 and WideResNet-22-6 [44] for CIFAR-100, and two adversarial teacher networks including WideResNet-34-10 for CIFAR-10 and WideResNet-70-16 [14] for CIFAR-100. For CIFAR-10, WideResNet-34-10 is trained using TRADES [45]; For CIFAR-100, we use the WideResNet-70-16 model provided by Goyal et al. [14], two adversarial teachers are also the teacher models used in RSLAD [48]. The whole teachers are pre-trained before adversarial distillation. The performance of the teacher models is shown in Table 1.

Training and Evaluation. We train student networks using Stochastic Gradient Descent (SGD) optimizer with an initial learning rate 0.1, momentum 0.9, and weight decay 2e-4. For our MTARD, the weight loss learning rate is initially set as 0.025. We set the total number of training epochs to 300, the learning rate is divided by 10 at 215th, 260th, and 285th epochs. We set batch size to 128, and β is set to 1. For the inner maximization of MTARD, we use a 10 step PGD (PGD-10) with random start size 0.001 and step size 2/255.

We strictly follow SAT and TRADES original settings; For ARD, we use the same adversarial teachers as RSLAD and our MTARD. The temperature constant τ of ARD is set to 30 following original settings on CIFAR-10 while set to 5 on CIFAR-100, and the α is set to 0.95 following Micah [12] on CIFAR-100. Training perturbation in the maximization process is bounded to the L_∞ norm $\epsilon = 8/255$. For natural training, we train the networks for 100 epochs on clean images, and the learning rate is divided by 10 at the 75th and 90th epochs.

The same as previous studies, we evaluate the trained model against 4 white box adversarial attacks: FGSM, PGD_{sat}, PGD_{trades}, CW_∞, which are commonly used adversarial attacks in adversarial robustness evaluation. PGD_{sat} is the attack proposed in Madry et al. [27], and PGD_{trades} is used in Zhang [45], the step size of PGD_{sat} and PGD_{trades} is 2/255, and the step is 20. the total step of

Table 2. White-box robustness of ResNet-18 on CIFAR-10 and CIFAR-100 dataset.

		CIFAR-10			CIFAR-100		
Attack	Defense	Clean	Robust	W-Robust	Clean	Robust	W-Robust
FGSM	Natural	94.57%	18.60%	56.59%	75.18%	7.96%	41.57%
	SAT	84.2%	55.59%	69.90%	56.16%	25.88%	41.02%
	TRADES	83.00%	58.35%	70.68%	57.75%	31.36%	44.56%
	ARD	84.11%	58.4%	71.26%	60.11%	33.61%	46.86%
	RSLAD	83.99%	60.41%	72.2%	58.25%	34.73%	46.49%
	MTARD	87.36%	61.2%	74.28%	64.3%	31.49%	47.90%
PGD _{sat}	Natural	94.57%	0%	47.29%	75.18%	0%	37.59%
	TRADES	83.00%	52.35%	67.68%	57.75%	28.05%	42.9%
	SAT	84.2%	45.95%	65.08%	56.16%	21.18%	38.67%
	TRADES	83.00%	52.35%	67.68%	57.75%	28.05%	42.9%
	ARD	84.11%	50.93%	67.52%	60.11%	29.4%	44.76%
	RSLAD	83.99%	53.94%	68.97%	58.25%	31.19%	44.72%
MTARD	87.36%	50.73%	69.05%	64.3%	24.95%	44.63%	
PGD _{trades}	Natural	94.57%	0%	47.29%	75.18%	0%	37.59%
	SAT	84.2%	48.12%	66.16%	56.16%	22.02%	39.09%
	TRADES	83.00%	53.83%	68.42%	57.75%	28.88%	43.32%
	ARD	84.11%	52.96%	68.54%	60.11%	30.51%	45.31%
	RSLAD	83.99%	55.73%	69.86%	58.25%	32.05%	45.15%
	MTARD	87.36%	53.60%	70.48%	64.3%	26.75%	45.53%
CW _∞	Natural	94.57%	0%	47.29%	75.18%	0%	37.59%
	SAT	84.2%	45.97%	65.09%	56.16%	20.9%	38.53%
	TRADES	83.00%	50.23%	66.62%	57.75%	24.19%	40.97%
	ARD	84.11%	50.15%	67.13%	60.11%	27.56%	43.84%
	RSLAD	83.99%	52.67%	68.33%	58.25%	28.21%	43.23%
	MTARD	87.36%	48.57%	67.97%	64.3%	23.42%	43.86%

CW_∞ is 30. Maximum perturbation is bounded to the L_∞ norm $\epsilon = 8/255$ for all attacks. Meanwhile, we also conduct a black-box evaluation, which includes the transfer-based attack and query-based attack to test the robustness of the student model in a near-real environment.

Here, we use Weighted Robust Accuracy [16] to evaluate the trade-off between the clean and robust accuracy of the student model, it is defined as follows:

$$\mathcal{A}_f = \pi_{D_{nat}} P_{D_{nat}}[f(x) = y] + \pi_{D_{adv}} P_{D_{adv}}[f(x) = y], \quad (12)$$

where Weighted Robust Accuracy \mathcal{A}_f are the accuracy of a model f on x drawn from either the clean distribution D_{nat} and the adversarial distribution D_{adv} . We set $\pi_{D_{nat}}$ and $\pi_{D_{adv}}$ both to 0.5, which means clean accuracy and robust accuracy are equally important for comprehensive performance in the model.

4.2 Adversarial Robustness Evaluation

White-box Robustness. The performance of ResNet-18 and MobileNet-v2 trained by our MTARD and other baseline methods under the white box attack are shown in Table 2 and 3 for CIFAR-10 and CIFAR-100. We select the best checkpoint of baseline model and MTARD based on Weighted Robust Accuracy.

The experimental results demonstrate that our method MTARD has the state-of-the-art W-Robust Accuracy on CIFAR-10 and CIFAR-100. For ResNet-18, MTARD improves W-Robust Accuracy by 1.01% compared with the best

Table 3. White-box robustness of MobileNet-V2 on CIFAR-10 and CIFAR-100 dataset.

Attack	Defense	CIFAR-10			CIFAR-100		
		Clean	Robust	W-Robust	Clean	Robust	W-Robust
FGSM	Natural	93.35%	12.22%	52.79%	74.86%	5.94%	40.4%
	SAT	83.87%	55.89%	69.88%	59.19%	30.88%	45.04%
	TRADES	77.95%	53.75%	65.85%	55.41%	30.28%	42.85%
	ARD	83.43%	57.03%	70.23%	60.45%	32.77%	46.61%
	RSLAD	83.2%	59.47%	71.34%	59.01%	33.88%	46.45%
	MTARD	89.26%	57.84%	73.55%	67.01%	32.42%	49.72%
PGD _{sat}	Natural	93.35%	0%	46.68%	74.86%	0%	37.43%
	SAT	83.87%	46.84%	65.36%	59.19%	25.64%	42.42%
	TRADES	77.95%	49.06%	63.51%	55.41%	23.33%	39.37%
	ARD	83.43%	49.5%	66.47%	60.45%	28.69%	44.57%
	RSLAD	83.2%	53.25%	68.23%	59.01%	30.19%	44.6%
	MTARD	89.26%	44.16%	66.71%	67.01%	25.14%	46.08%
PGD _{trades}	Natural	93.35%	0%	46.68%	74.86%	0%	37.43%
	SAT	83.87%	49.14%	66.51%	59.19%	26.96%	43.08%
	TRADES	77.95%	50.27%	64.11%	55.41%	28.42%	41.92%
	ARD	83.43%	51.7%	67.57%	60.45%	29.63%	45.04%
	RSLAD	83.2%	54.76%	68.98%	59.01%	31.19%	45.1%
	MTARD	89.26%	47.99%	68.63%	67.01%	27.1%	47.06%
CW _∞	Natural	93.35%	0%	46.68%	74.86%	0%	37.43%
	SAT	83.87%	46.62%	65.25%	59.19%	25.01%	42.1%
	TRADES	77.95%	46.06%	62.01%	55.41%	27.72%	41.57%
	ARD	83.43%	48.96%	66.20%	60.45%	26.55%	43.50%
	RSLAD	83.2%	51.78%	67.49%	59.01%	27.98%	43.50%
	MTARD	89.26%	43.42%	66.34%	67.01%	24.14%	45.58%

baseline method under the attack of FGSM on CIFAR-100; For MobileNet-V2, MTARD improves the W-Robust Accuracy by 1.96% against PGD_{trades} on CIFAR-100. Moreover, our method also shows relevant superiority against PGD_{sat}, CW_∞ compared with other methods.

Meanwhile, ARD, RSLAD, and MTARD outperform SAT and TRADES, which shows the adversarial robustness distillation can bring greater improvement to the small models than traditional methods. In addition, our MTARD can achieve better performance than ARD and RSLAD without artificial hyperparameters adjustment, while ARD relies on the adjustment of temperature constant τ and RSLAD relies on the adjustment of loss weight.

Black-box Robustness. In addition, we also test MTARD and other methods against black-box attacks for ResNet-18 and MobileNet-V2 on CIFAR-10 and CIFAR-100 separately. We choose the transfer-based attack and query-based attack in our evaluation. As for the transfer-based attack, we choose our adversarial teachers (WideResNet-34-10 and WideResNet-70-16) as the surrogate model to produce adversarial example against the PGD-20 (PGD_{trades}) and CW_∞ attack; As for the query-based attack, we choose the Square attack (SA) to attack these models. We select the best checkpoint of baseline model and MTARD based on Weighted Robust Accuracy. The result of ResNet-18 is showed in Table 4, while the result of MobileNet-V2 is showed in Table 5.

From the result, Our MTARD achieves better W-Robust Accuracy than any other model against all three black-box attacks in any conditions. Under the

Table 4. Black-box robustness of ResNet-18 on CIFAR-10 and CIFAR-100 dataset.

Attack	Defense	CIFAR-10			CIFAR-100		
		Clean	Robust	W-Robust	Clean	Robust	W-Robust
PGD-20	SAT	84.2%	64.74%	74.47%	56.16%	38.1%	47.13%
	TRADES	83.00%	63.56%	73.28%	57.75%	38.2%	47.98%
	ARD	84.11%	63.59%	73.85%	60.11%	39.53%	49.82%
	RSLAD	83.99%	63.9%	73.95%	58.25%	39.93%	49.09%
	MTARD	87.36%	65.17%	76.27%	64.3%	41.39%	52.85%
CW_∞	SAT	84.2%	64.88%	74.54%	56.16%	39.42%	47.79%
	TRADES	83.00%	62.85%	72.93%	57.75%	38.63%	48.19%
	ARD	84.11%	62.78%	73.45%	60.11%	38.85%	49.48%
	RSLAD	83.99%	63.02%	73.51%	58.25%	39.67%	48.96%
	MTARD	87.36%	64.65%	76.01%	64.3%	41.03%	52.67%
SA	SAT	84.2%	71.3%	77.75%	56.16%	41.27%	48.72%
	TRADES	83.00%	70.33%	76.67%	57.75%	41.96%	49.86%
	ARD	84.11%	73.3%	78.71%	60.11%	48.79%	54.45%
	RSLAD	83.99%	72.1%	78.05%	58.25%	45.34%	51.80%
	MTARD	87.36%	79.99%	83.68%	64.3%	41.03%	52.67%

Table 5. Black-box results of MobileNet-V2 on CIFAR-10 and CIFAR-100 dataset.

Attack	Defense	CIFAR-10			CIFAR-100		
		Clean	Robust	W-Robust	Clean	Robust	W-Robust
PGD-20	SAT	83.87%	64.6%	74.24%	59.19%	40.7%	49.95%
	TRADES	77.95%	61.07%	69.51%	55.41%	37.76%	46.59%
	ARD	83.43%	63.34%	73.39%	60.45%	39.15%	49.8%
	RSLAD	83.2%	64.3%	73.75%	59.01%	40.32%	49.67%
	MTARD	89.26%	66.37%	77.82%	67.01%	43.22%	55.12%
CW_∞	SAT	83.87%	64.16%	74.02%	59.19%	40.97%	50.08%
	TRADES	77.95%	60.68%	69.32%	55.41%	38.02%	46.72%
	ARD	83.43%	62.73%	73.08%	60.45%	38.53%	49.49%
	RSLAD	83.2%	63.61%	73.41%	59.01%	39.92%	49.47%
	MTARD	89.26%	65.67%	77.47%	67.01%	42.97%	54.99%
SA	SAT	83.87%	69.94%	76.91%	59.19%	43.35%	51.27%
	TRADES	77.95%	66.3%	72.13%	55.41%	41.39%	48.4%
	ARD	83.43%	71.82%	77.63%	60.45%	47.08%	53.77%
	RSLAD	83.2%	71.11%	77.16%	59.01%	42.95%	50.98%
	MTARD	89.26%	79.73%	84.50%	67.01%	50.77%	58.89%

Square Attack, ResNet-18 and MobileNet-V2 trained by MTARD outperform W-Robust Accuracy by 4.97% and 6.87% respectively on CIFAR-10 compared to the second-best method; Moreover, MTARD brings 2.13% and 5.12% improvements to ResNet-18 and MobileNet-V2. In addition, MTARD has different margins in defending against attacks from PGD-20 and CW_∞ transfer attack, which shows the superior performance of MTARD in defending the black-box attacks.

4.3 Ablation Study

To better understand the impact of each component in our MTARD, we conduct a set of ablation studies. Baseline denotes using one well-trained adversarial teacher (WideResNet-34-10) to guide ResNet-18 student network on CIFAR-10, and Baseline+MT denotes using an adversarial teacher (WideResNet-34-10) and a clean teacher (ResNet-56) to guide student from adversarial and clean aspects, respectively, where the weight w_{adv} and w_{nat} are constant at 0.5. Base-

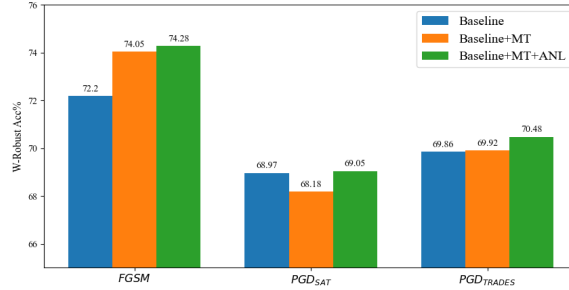


Fig. 2. Ablation study with ResNet-18 student network distilled using variants of our MTARD and Baseline method on CIFAR-10. MT and ANL are abbreviations of multi-teacher and Adaptive Normalization Loss. Baseline+MT means using multiple teachers in Baseline. Baseline+MT+ANL means our MTARD method.

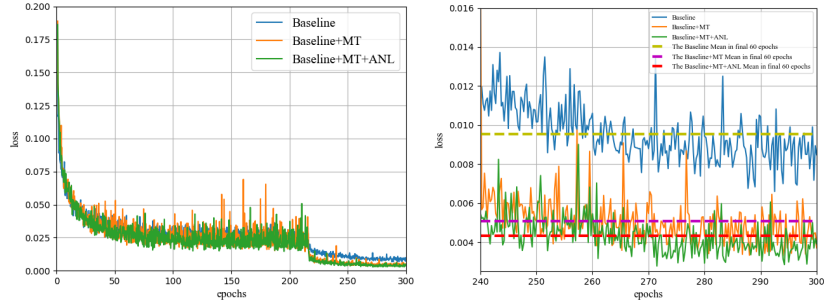


Fig. 3. The training loss with ResNet-18 student network distilled using variants of Baseline, Baseline+MT, and Baseline+MT+ANL (our MTARD) on CIFAR-10. MT and ANL are abbreviations of multi-teacher and Adaptive Normalization Loss. The y axis is the L_{total} in the training epoch x . The left is the change curve of L_{total} in the whole training process, the right is the change curve of L_{total} in final 60 epochs.

line+MT+ANL (MTARD) term denotes adding the Adaptive Normalization Loss to dynamically adjust the weight w_{adv} and w_{nat} based on Baseline+MT.

The final result is shown in Fig. 2. The change of total loss L_{total} is shown in Fig. 3, and the change of relative loss \tilde{L}_{nat} and \tilde{L}_{adv} is shown in Fig. 4. Compared to the baseline method, MTARD’s improvement is remarkable, which confirms the importance of each component. Multiple teachers positively affect the student model to learn both clean and robust accuracy. However, it is not enough to use multiple teacher models without Adaptive Normalization Loss due to the poor performance of Baseline+MT.

In Fig. 4, the lower \tilde{L} means the student has learnt more ability from the corresponding teacher. The gap between \tilde{L}_{nat} and \tilde{L}_{adv} can represent the trade-off between each teacher. Compared with Baseline+MT, the MTARD’s training loss is less oscillating, and relative loss can achieve a better ideal state. In the final

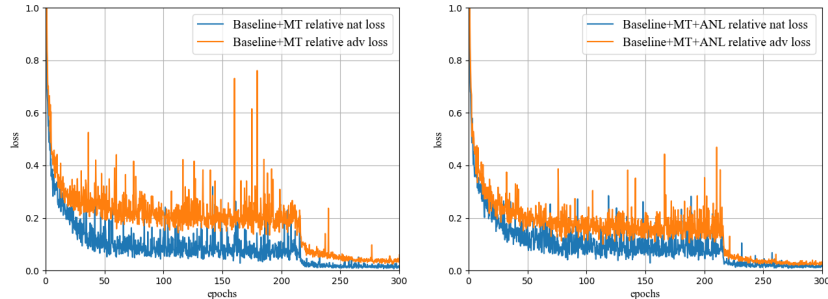


Fig. 4. The relative training loss with ResNet-18 student network distilled on CIFAR-10. The left is Baseline+MT and the right is MTARD (Baseline+MT+ANL). MT and ANL are abbreviations of multi-teacher and Adaptive Normalization Loss. The x axis means the training epochs, the y axis is the adv relative loss \tilde{L}_{adv} and the adv relative loss \tilde{L}_{nat} in the training epoch x .

training period, MTARD’s \tilde{L}_{nat} outperforms the Baseline+MT’s, while \tilde{L}_{nat} can reach the same level as the Baseline+MT’s. Meanwhile, the MTARD’s trade-off between \tilde{L}_{nat} and \tilde{L}_{adv} are more tiny. All the results demonstrate Adaptive Normalization Loss can better balance the influence between the adversarial teacher and the clean teacher to maximize the role of each other and obtain a more capable student with both accuracy and robustness.

5 Conclusion

In this paper, we investigated the problem of enhancing the accuracy and robustness of a small model via adversarial distillation. We revisited several state-of-the-art adversarial training and adversarial robustness distillation methods. To improve both the robust and clean accuracy of small models, we proposed Multi-Teacher Adversarial Robustness Distillation (MTARD) to guide the learning process of the small student models. To balance the influence toward students between adversarial teachers and clean teachers, we designed a method to use Adaptive Normalization Loss in MTARD. The effectiveness of MTARD over existing adversarial training and distillation methods were validated on both benchmark datasets. In the future, our method can be applied by other knowledge distillation tasks with multiple optimization goals, but not just limited to adversarial robustness distillation, which has greater potential to be developed.

Acknowledgement

This work was supported by National Key R&D Program of China (Grant No.2020AAA0104002) and the Project of the National Natural Science Foundation of China (No.62076018).

References

1. Andriushchenko, M., Croce, F., Flammarion, N., Hein, M.: Square attack: a query-efficient black-box adversarial attack via random search. In: European Conference on Computer Vision. pp. 484–501. Springer (2020)
2. Athalye, A., Carlini, N., Wagner, D.: Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In: International conference on machine learning. pp. 274–283. PMLR (2018)
3. Bai, Y., Zeng, Y., Jiang, Y., Xia, S.T., Ma, X., Wang, Y.: Improving adversarial robustness via channel-wise activation suppressing. arXiv preprint arXiv:2103.08307 (2021)
4. Bhagoji, A.N., He, W., Li, B., Song, D.: Practical black-box attacks on deep neural networks using efficient query mechanisms. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 154–169 (2018)
5. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: 2017 IEEE Symposium on Security and Privacy (SP). pp. 39–57. IEEE (2017)
6. Chen, P.Y., Zhang, H., Sharma, Y., Yi, J., Hsieh, C.J.: Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In: Proceedings of the 10th ACM workshop on artificial intelligence and security. pp. 15–26 (2017)
7. Chen, T., Zhang, Z., Liu, S., Chang, S., Wang, Z.: Robust overfitting may be mitigated by properly learned smoothing. In: International Conference on Learning Representations (2020)
8. Croce, F., Hein, M.: Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In: International conference on machine learning. pp. 2206–2216. PMLR (2020)
9. Demontis, A., Melis, M., Pintor, M., Jagielski, M., Biggio, B., Oprea, A., Nita-Rotaru, C., Roli, F.: Why do adversarial attacks transfer? explaining transferability of evasion and poisoning attacks. In: 28th USENIX security symposium (USENIX security 19). pp. 321–338 (2019)
10. Deng, J., Pan, Y., Yao, T., Zhou, W., Li, H., Mei, T.: Relation distillation networks for video object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7023–7032 (2019)
11. Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., Song, D.: Robust physical-world attacks on deep learning visual classification. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1625–1634 (2018)
12. Goldblum, M., Fowl, L., Feizi, S., Goldstein, T.: Adversarially robust distillation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 3996–4003 (2020)
13. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572 (2014)
14. Goyal, S., Qin, C., Uesato, J., Mann, T., Kohli, P.: Uncovering the limits of adversarial training against norm-bounded adversarial examples. arXiv preprint arXiv:2010.03593 (2020)
15. Gupta, S., Agrawal, A., Gopalakrishnan, K., Narayanan, P.: Deep learning with limited numerical precision. In: International conference on machine learning. pp. 1737–1746. PMLR (2015)
16. Gürel, N.M., Qi, X., Rimanic, L., Zhang, C., Li, B.: Knowledge enhanced machine learning pipeline against diverse adversarial attacks. In: International Conference on Machine Learning. pp. 3976–3987. PMLR (2021)

17. Han, S., Mao, H., Dally, W.J.: Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding. arXiv preprint arXiv:1510.00149 (2015)
18. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
19. Huang, Q., Katsman, I., He, H., Gu, Z., Belongie, S., Lim, S.N.: Enhancing adversarial example transferability with an intermediate level attack. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4733–4742 (2019)
20. Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., Madry, A.: Adversarial examples are not bugs, they are features. *Advances in neural information processing systems* **32** (2019)
21. Javanmard, A., Soltanolkotabi, M., Hassani, H.: Precise tradeoffs in adversarial training for linear regression. In: Conference on Learning Theory. pp. 2034–2078. PMLR (2020)
22. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
23. Kumar, R.S.S., Nyström, M., Lambert, J., Marshall, A., Goertzel, M., Comissioner, A., Swann, M., Xia, S.: Adversarial machine learning—industry perspectives. In: 2020 IEEE Security and Privacy Workshops (SPW). pp. 69–75. IEEE (2020)
24. Liu, Y., Chen, X., Liu, C., Song, D.: Delving into transferable adversarial examples and black-box attacks. arXiv preprint arXiv:1611.02770 (2016)
25. Ma, X., Li, B., Wang, Y., Erfani, S.M., Wijewickrema, S., Schoenebeck, G., Song, D., Houle, M.E., Bailey, J.: Characterizing adversarial subspaces using local intrinsic dimensionality. arXiv preprint arXiv:1801.02613 (2018)
26. Ma, X., Niu, Y., Gu, L., Wang, Y., Zhao, Y., Bailey, J., Lu, F.: Understanding adversarial attacks on deep learning based medical image analysis systems. *Pattern Recognition* **110**, 107332 (2021)
27. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083 (2017)
28. Mao, H., Han, S., Pool, J., Li, W., Liu, X., Wang, Y., Dally, W.J.: Exploring the regularity of sparse structure in convolutional neural networks. arXiv preprint arXiv:1705.08922 (2017)
29. Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z.B., Swami, A.: The limitations of deep learning in adversarial settings. In: 2016 IEEE European symposium on security and privacy (EuroS&P). pp. 372–387. IEEE (2016)
30. Raghunathan, A., Xie, S.M., Yang, F., Duchi, J., Liang, P.: Understanding and mitigating the tradeoff between robustness and accuracy. arXiv preprint arXiv:2002.10716 (2020)
31. Rice, L., Wong, E., Kolter, Z.: Overfitting in adversarially robust deep learning. In: International Conference on Machine Learning. pp. 8093–8104. PMLR (2020)
32. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4510–4520 (2018)
33. Sarikaya, R., Hinton, G.E., Deoras, A.: Application of deep belief networks for natural language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **22**(4), 778–784 (2014)
34. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2818–2826 (2016)

35. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199 (2013)
36. Tang, R., Lu, Y., Liu, L., Mou, L., Vechtomova, O., Lin, J.: Distilling task-specific knowledge from bert into simple neural networks. arXiv preprint arXiv:1903.12136 (2019)
37. Vandenhende, S., Georgoulis, S., Van Gansbeke, W., Proesmans, M., Dai, D., Van Gool, L.: Multi-task learning for dense prediction tasks: A survey. *IEEE transactions on pattern analysis and machine intelligence* (2021)
38. Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., Liu, W.: Cosface: Large margin cosine loss for deep face recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 5265–5274 (2018)
39. Wang, Y., Zou, D., Yi, J., Bailey, J., Ma, X., Gu, Q.: Improving adversarial robustness requires revisiting misclassified examples. In: *International Conference on Learning Representations* (2019)
40. Wei, X., Yan, H., Li, B.: Sparse black-box video attack with reinforcement learning. *International Journal of Computer Vision* **130**(6), 1459–1473 (2022)
41. Wu, B., Chen, J., Cai, D., He, X., Gu, Q.: Do wider neural networks really help adversarial robustness? *Advances in Neural Information Processing Systems* **34** (2021)
42. Xiang, L., Ding, G., Han, J.: Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification. In: *European Conference on Computer Vision*. pp. 247–263. Springer (2020)
43. Yan, H., Wei, X.: Efficient sparse attacks on videos using reinforcement learning. In: *Proceedings of the 29th ACM International Conference on Multimedia*. pp. 2326–2334 (2021)
44. Zagoruyko, S., Komodakis, N.: Wide residual networks. arXiv preprint arXiv:1605.07146 (2016)
45. Zhang, H., Yu, Y., Jiao, J., Xing, E., El Ghaoui, L., Jordan, M.: Theoretically principled trade-off between robustness and accuracy. In: *International conference on machine learning*. pp. 7472–7482. PMLR (2019)
46. Zhang, T., Zhu, Z.: Interpreting adversarially trained convolutional neural networks. In: *International Conference on Machine Learning*. pp. 7502–7511. PMLR (2019)
47. Zhu, J., Yao, J., Han, B., Zhang, J., Liu, T., Niu, G., Zhou, J., Xu, J., Yang, H.: Reliable adversarial distillation with unreliable teachers. arXiv preprint arXiv:2106.04928 (2021)
48. Zi, B., Zhao, S., Ma, X., Jiang, Y.G.: Revisiting adversarial robustness distillation: Robust soft labels make student better. In: *International Conference on Computer Vision* (2021)