One Size Does NOT Fit All: Data-Adaptive Adversarial Training

Shuo Yang[1] and Chang Xu[1]

University of Sydney, Sydney, AU
syang9630@uni.sydney.edu.au, c.xu@sydney.edu.au

Abstract. Adversarial robustness is critical for deep learning models to defend against adversarial attacks. Although adversarial training is considered to be one of the most effective ways to improve the model's adversarial robustness, it usually yields models with lower natural accuracy. In this paper, we argue that, for the attackable examples, traditional adversarial training which utilizes a fixed size perturbation ball can create adversarial examples that deviate far away from the original class towards the target class. Thus, the model's performance on the natural target class will drop drastically, which leads to the decline of natural accuracy. To this end, we propose the **D**ata-**A**daptive **A**dversarial **T**raining (DAAT) which adaptively adjusts the perturbation ball to a proper size for each of the natural examples with the help of a natural trained calibration network. Besides, a dynamic training strategy empowers the DAAT models with impressive robustness while retaining remarkable natural accuracy. Based on a toy example, we theoretically prove the recession of the natural accuracy caused by adversarial training and show how the data-adaptive perturbation size helps the model resist it. Finally, empirical experiments on benchmark datasets demonstrate the significant improvement of DAAT models on natural accuracy compared with strong baselines.

Keywords: Adversarial training, Adversarial attack, Adversarial robustness

1 Introduction

Deep learning has led to significant advances across a broad range of tasks, such as computer vision [11], natural language processing [6]. However, the pervasive brittleness of deep neural networks (DNNs) against adversarial examples [27] has raised particular worrisome of the applications of deep learning. Adversarial examples can induce significant change to the output of DNNs even though they are generated by perturbing the clean data only with "imperceptible" noise. The real-world especially security-related tasks (e.g., autonomous driving [3]) require reliability and robustness of DNN models against adversarial attacks.

A large body of approaches has been proposed to defend the adversarial attacks. Adversarial training [9,21] is regarded as one of the most effective

adversarial defense methods. The fundamental philosophy behind adversarial training is to encourage the similarity of predictions between the clean input and its neighborhoods. For example, the standard adversarial training [21] first generates adversarial examples within an l_p-norm perturbation ball of radius ϵ, and then imposes the model to have the correct prediction of the adversary. Although subsequent methods such as [14] utilize more sophisticated loss to generate adversaries, they basically share the same training strategy.

Despite the empirical success of adversarial training, recent works show that the improvement of robustness comes at the cost of natural accuracy [29,34]. This problem has inspired many works to study the intrinsic trade-off between robustness and accuracy. For example, TRADES [34] explicitly sets a trade-off coefficient on the natural and adversarial training loss to balance the performance of accuracy and robustness. FAT [35] searches for the least adversarial data to moderate the influence of the adversarial training on natural accuracy. However, the origin of the trade-off is still arguable and the solution to improving the degraded accuracy while keeping the robustness still leaves open.

In this paper, we try to reconsider the trade-off problem from a novel perspective. When generating the adversarial examples, traditional adversarial training methods adhere to the principle that the generated adversaries should be projected to a ball with a fixed size around the natural examples. Given sufficient update, the final adversaries tend to appear on the surface of the perturbation ball, since the generation is oriented by the gradient ascending direction which is generally away from the clean example. On the one hand, for the clean data which are more resistive to the adversarial attacks, the generated adversarial example within the perturbation ball is still similar to the original class; however, on the other hand, for the natural example which is more attackable, the generated adversary can extremely diverge from its natural counterpart. At worst, there may be some clean examples from other classes existing in the perturbation ball if the ball is large enough as illustrated in Fig. 1. If trained with such adversarial examples which excessively overstep the decision boundary, the natural accuracy of the model will be inevitably degraded.

To this end, we propose a novel adversarial training scheme named **D**ata-**A**daptive **A**dversarial **T**raining (DAAT) which adaptively adjusts the perturbation ball to a proper size for each of the natural examples. The data-adaptive perturbation size is upper-bounded by the initial preset size and it aims at avoiding generating excessively overstepping examples. Concretely, if the attacker generates an adversary which crosses the line into another category too much, DAAT will shrink the perturbation ball to pull it back, while if the generated adversaries are so benign that can be easily classified, DAAT will enlarge the perturbation ball to give more elbow room to the attacker. The specific perturbation size is determined by a calibration network that is trained merely with the natural data so as not to overfit the adversaries. Therefore, the generalization ability on the natural examples can be better preserved. Besides, to exploit more informative adversaries, we dynamically enlarge the margin of the data-adaptive perturbation ball during different training stages. Empirical experiments demon-

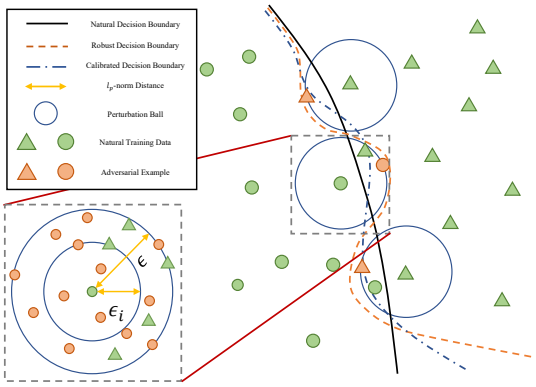


Fig. 1: Illustration of how the adversarial training with a fixed size perturbation ball can cause the degradation of the natural accuracy.

strate that the dynamic training strategy plays a significant role in improving the robustness of DAAT models.

2 Related Work

2.1 Adversarial Defense

A large body of research has been conducted to improve the model's defensive power against adversarial examples from various perspectives. For example, many works [10,8,24,12,18] try to detect the adversarial examples and reject them. Another branch of works [31,32,19] view the adversarial examples as contaminated natural examples and aim at recovering the clean examples by employing denoising or feature squeezing methods. However, several detectors and denoisers have been shown to have a limited benefit on certain kinds of attacks [4]. Currently, adversarial training is considered to be one of the most effective defense strategies. The key idea of adversarial training is to train the non-robust model with the generated adversaries. Based on the seminal work [21], many works try to improve the performance of the standard adversarial training by utilizing resultful tools from other domains, such as logit pairing [14], metric learning [23], self-supervised learning [15]. However, most of the aforementioned improvements only focus on how to better align the adversaries with their natural counterparts. Thus, a side effect with the increasing robustness is that the natural accuracy will decline rapidly.

2.2 Decline of the Natural Accuracy

[29] first finds that the natural accuracy may be at odd with robustness. [29] claims that the natural trained model and adversarially trained model may learn different features for classification. This idea is further confirmed by [13]. [34] later

proposes TRADES which uses a trade-off parameter β to balance the training between the natural and adversarial examples. However, TRADES only aims at adjusting the attention between accuracy and robustness, but not at harmonizing the conflict between them. FAT [35] assumes that the reason for the decline of natural accuracy is that the adversaries are so invasive that some of them have crossed over the original decision boundary by a large margin. By generating adversarial examples which are weaker, FAT shows a remarkable improvement of natural accuracy. Nonetheless, the objective of FAT encourages the attacker to find the weakest adversarial examples under certain constraints which cannot provide sufficient information to improve the robustness. Thus, the empirical robust accuracy of FAT is usually much lower. Besides, FAT employs the model trained on the adversaries to determine the strength of attack which may be overfitting to the adversaries. Parallel to our work, [1] also shares a similar idea of instance adaptive adversarial training. However, we employ different adjustment strategies, and the natural trained calibration network guarantees better performance of our method. We defer the empirical comparison of DAAT and [1] to the Appendix.

3 Review of Standard Adversarial Training

We denote $S = \{(x_i, y_i)\}_{i=1}^n$ as the training dataset, where $x_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}^K$. In this paper, we consider a multiclass classification task $f(g(\cdot;\theta);\omega) : \mathbb{R}^d \to \mathbb{R}^K$, where $g(\cdot;\theta)$ is a feature extractor parameterized by θ and $f(\cdot;\omega)$ is a classifier parameterized by ω. The perturbation ball around an input example x is defined as:

$$\mathcal{B}_p(x,\epsilon) := \{x' | \left\|x - x'\right\|_p \leq \epsilon\}, \tag{1}$$

where ϵ is a preset perturbation size, and $\left\|\cdot\right\|_p$ refers to the l_p-norm metric. In the experimental section, except where explicitly stated, we typically choose the l_∞ norm for training and evaluation as it commonly leads to a smaller perturbation size.

Adversarial example x' is an example in the perturbation ball around a natural example x, i.e., $x' \in \mathcal{B}_p(x,\epsilon)$. The harmfulness of the adversarial example is reflected in that it can alter the prediction of a model for the original natural example as follow:

$$\arg\max_k f(g(x'))_k \neq \arg\max_k f(g(x))_k, \tag{2}$$

where $f(g(\cdot))_k$ is the predicted probability of the k-th class. The existence of the adversarial example reflects the sensitivity of the model to adversarial perturbations. Adversarial training can be a natural way to smooth the prediction of the model within the perturbation ball [21]. The objective of the standard adversarial training can be formulated as follow:

$$\min_{\omega,\theta} \mathbb{E}\left[\max_{x' \in \mathcal{B}_p(x,\epsilon)} \mathcal{L}\left(f(g(x';\theta);\omega), y\right)\right], \tag{3}$$

where $\mathcal{L}$ is the loss function to measure the difference between the ground-truth label and prediction (e.g., soft-max cross-entropy loss).

From Eq. (3) we can find that the standard adversarial training implies a minimax game between the attacker and defender. The attacker aims at generating the adversarial examples which maximize the loss within the perturbation ball, while the defender tries to correct the misclassification on the generated attack. However, in practice, the inner maximization is generally intractable due to the extremely high dimension of the input space. Thus, the Projected Gradient Descent (PGD) method [21] is proposed to approximate the inner maximization by generating the worst-case example within T-step iterations. A PGD iteration can be written as follow:

$$x^{t+1} = \mathrm{Proj}_{\mathcal{B}_p(x,\epsilon)}\left[x^t + \alpha \mathrm{sign}\left(\nabla_{x^t}\mathcal{L}(f(g(x^t);\theta);\omega), y)\right)\right], \tag{4}$$

where $\mathrm{Proj}_{\mathcal{B}_p(x,\epsilon)}$ is to project the generated example back to $\mathcal{B}_p(x,\epsilon)$ and α is the step size. The last step output is utilized as the final adversarial example, i.e., $x' := x^T$. It can be imagined that with the increasing iteration step, the generated examples will deviate from the origin further and further.

4 Adversarial Perturbation Size Matters

In standard adversarial training, the perturbation size is usually set to be the same for every example. Empirically, given sufficient PGD iteration steps, the generated adversarial example is more likely to be located on the surface of the perturbation ball (see experiments in Section 6.3). According to [36], *more attackable/robust data are closer to/farther away from the decision boundary.* Therefore, as illustrated in Fig. 1, the generated adversaries of the attackable data may cross the decision boundary to another class leading to the accuracy decline. Similar idea is also mentioned in [28].

In what follows, we theoretically demonstrate how standard accuracy is influenced by the standard adversarial training based on a toy example. Different from the setup of [13] to study the robust and non-robust feature, we adopt the toy experiment to illustrate how adversarial training hurts accuracy. We consider a binary classification problem where the input-label pairs (x, y) are sampled from a distribution D as follows:

$$y \overset{u.a.r.}{\sim} \{-1, 1\}, \quad x \sim \mathcal{N}(y \cdot \boldsymbol{\mu_*}, \boldsymbol{\Sigma_*}). \tag{5}$$

Our goal is to correctly classify new examples which are sampled from D. Based on the *maximum likelihood classification* criteria, our learning objective can be formulated as:

$$\min_{\boldsymbol{\mu},\boldsymbol{\Sigma}} \mathbb{E}_{(x,y)\sim D}\left[\mathcal{L}(x; y \cdot \boldsymbol{\mu}, \boldsymbol{\Sigma})\right], \tag{6}$$

where $\mathcal{L}(x; y \cdot \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the negative log-likelihood function of Gaussian. Due to the symmetry of the data distribution, the resulting optimal linear classifier can be easily obtained as follow:

$$y = \arg\max_y \mathcal{L}(x; y \cdot \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}) = \mathrm{sign}(x^\intercal \hat{\boldsymbol{\Sigma}}^{-1} \hat{\boldsymbol{\mu}}), \tag{7}$$

where $(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}})$ are the estimated parameters of the Gaussian. With the classifier (7), we have the following theorem:

Theorem 1. *When the optima of objective (6) is obtained, classifier (7) can achieve the Bayesian error rate as*

$$err_* = 1 - \frac{1}{\sqrt{2\pi}} \int_c^{\infty} \exp\left(-\frac{1}{2}x^2\right) dx,$$

where $c = -\frac{\boldsymbol{\mu}_*^\intercal \mathbf{w}}{\sqrt{\mathbf{w}^\intercal \boldsymbol{\Sigma}_* \mathbf{w}}}$, $\mathbf{w} = \boldsymbol{\Sigma}_*^{-1}\boldsymbol{\mu}_*$, *and the integral can be evaluated by the error function.*

The proof of Theorem 1 can be found in the Appendix. It indicates that the model which is only trained with the natural data according to the maximum likelihood rule can achieve the highest natural accuracy. Next, we will investigate how standard adversarial training leads to a reduction of natural accuracy. According to the standard adversarial training objective as Eq. (3), we can derive the robust objective of this toy experiment as follow:

$$\min_{\boldsymbol{\mu}, \boldsymbol{\Sigma}} \mathbb{E}_{(x,y)\sim D} \left[\max_{\delta \in \mathcal{B}_p(0,\epsilon)} \mathcal{L}(x+\delta; y \cdot \boldsymbol{\mu}, \boldsymbol{\Sigma}) \right]. \tag{8}$$

By solving Eq. (8) within the l_2-norm perturbation ball, we have the following propositions:

Proposition 1. *Given the robust objective in Eq. (8), the optimal perturbation* δ^* *with respect to input* x *can be derived as:*

$$\delta^* = (\lambda \boldsymbol{\Sigma} - \boldsymbol{I})^{-1}(x - \boldsymbol{\mu}),$$

where λ *is set such that* $\|\delta^*\|_2 = \epsilon$.

A straightforward result of Proposition 1 is that the optimal adversarial perturbation can be obtained at the surface of the perturbation ball. Imagine that for the more attackable examples which are closer to the natural decision boundary, the generated adversaries are more likely to cross the decision boundary with a sufficiently large perturbation size.

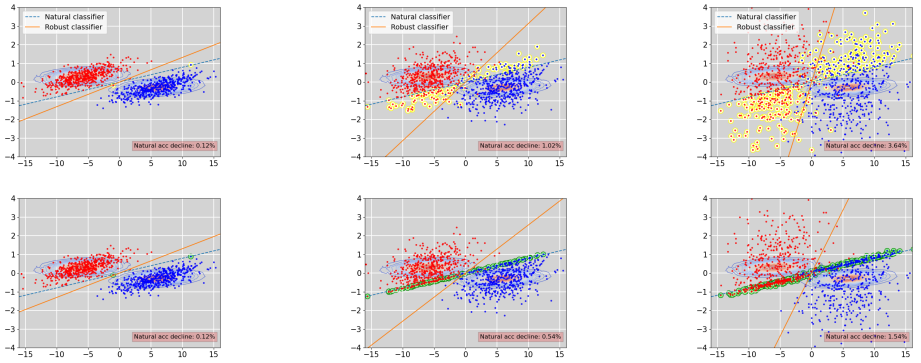
Proposition 2. *For a fixed* $tr(\boldsymbol{\Sigma}^*) = k$, *the objective (8) can be optimized at:*

$$\boldsymbol{\mu_r} = \boldsymbol{\mu^*}, \quad \boldsymbol{\Sigma_r} = \frac{k}{d}\boldsymbol{I},$$

where d *is the dimension of the input space.*

Proposition 2 further demonstrates the consequence of the improper perturbation size for these more attackable examples. Although the mean of the robust model is the same as that of the natural trained model (i.e., $\boldsymbol{\mu_r} = \boldsymbol{\mu^*}$), the covariance matrix becomes proportional to the identity matrix. As a result, the classifier induced by the standard adversarial learning will transit to:

$$y_r = \text{sign}(x^\intercal \boldsymbol{\Sigma_r^{-1}} \boldsymbol{\mu_r}) = \text{sign}(x^\intercal \boldsymbol{\mu_*}), \tag{9}$$



(d) DAAT with $\epsilon = 1$ (e) DAAT with $\epsilon = 3$ (f) DAAT with $\epsilon = 5$

Fig. 2: An empirical illustration of the toy example. The contour lines depict the natural data distribution, the red and blue dots in different subfigures are the generated adversaries with increasing perturbation size (i.e., $\epsilon = 1, 3, 5$ from left to right). The yellow circles are the overstepping examples, and the green ones are the corresponding examples adjusted by the adaptive perturbation ball. The blue dashed and orange lines are natural and robustified decision boundaries, respectively.

which is perpendicular to the line between the two mean points. If $\boldsymbol{\Sigma}_* \neq \boldsymbol{I}$, according to Theorem 1, the natural error of classifier (9) will be enlarged .

We give an empirical illustration of the toy examples in Fig. 2. The subfigures in the left column depict the standard adversarial training process. It is obvious that with the increasing perturbation size, more and more adversaries are crossing the optimal natural classifier, causing the induced robust classifier to have a lower and lower natural accuracy. In contrast, the adversaries in the right column are constrained in the adaptive perturbation balls which ensures that the adversaries can be concentrated around the original decision boundary. Consequently, the induced model has a lower natural accuracy decline.

5 Data-Adaptive Adversarial Training

As discussed in the previous sections, the fixed perturbation size employed in the standard adversarial training is not appropriate for every training example. For the examples which are far away from other classes, the adversaries in the ϵ-large perturbation ball can smooth the output of the neighborhoods around the natural examples which is beneficial to the robustness. On the contrary, for the natural examples which are closer to the decision boundary, the adversaries generated within the ϵ-large perturbation ball are likely to cross the decision boundary into another class. If the model is still trained on these overstepping adversaries, it is not difficult to imagine that the model will be biased to misclassified the natural examples from the target classes, which leads to the accuracy decline.

To solve the problem above, we propose the Data-Adaptive Adversarial Training (DAAT) method in this section. The basic idea of DAAT is to apply an

Algorithm 1 Data-Adaptive Adversarial Training (DAAT)

Input: Initialized feature extractor $g(\cdot)$, classifier $f(\cdot)$, calibration network $c(\cdot)$, training data $S = \{(x_i, y_i)\}_{i=1}^n$, initialized perturbation size $\epsilon_i^0 = \epsilon$, number of steps T, margin ρ, temperature τ, number of epochs E, minibatch size m
Output: Adversarial robust network $f(g(\cdot))$
for epoch $= 1, \ldots, E$ **do**
  for mini-batch $= 1, \ldots, M$ **do**
    Sample a mini-batch $\{(x_i, y_i)\}_{i=1}^m$ from S
    Train $c(\cdot)$ with the natural data by $\frac{1}{m}\sum_{i=1}^m \mathcal{L}\left(c(g(x_i)), y_i\right)$
    Generate data-adaptive adversarial example by $x_i' \leftarrow \Pi_{\mathcal{B}_p(x_i, \epsilon_i^e)}^T \left[\alpha \text{sign}\left(\nabla_{x_i^t} \mathcal{L}(f(g(x_i^t)), y)\right) + x_i'\right]$
    Train $f(g(\cdot))$ with the adversarial examples by $\frac{1}{m}\sum_{i=1}^m \mathcal{L}\left(f(g(x_i')), y_i\right)$
    Obtain the similarity s_i by Eq. (13)
    Update the data-adaptive perturbation size ϵ_i^{e+1} by Eq. (12)
  end for
end for

adaptive perturbation size ϵ_i to different training data x_i so that the generated adversaries can be constrained and do not deviate too far away from the clean one. Thus, DAAT generates the adversary x_i' within a calibrated perturbation ball $\mathcal{B}_p(x_i, \epsilon_i)$ as:

$$x_i' = \underset{x_i' \in \mathcal{B}_p(x_i, \epsilon_i)}{\arg\max} \mathcal{L}\left(f(g(x_i')), y_i\right). \tag{10}$$

A critical point of DAAT is how to determine the adjustment of perturbation size. Recall that the job of ϵ_i is to constrain x_i' not to overstep the natural decision boundary, thus we employ a calibration network $c(\cdot; \psi)$ to estimate how far the generated adversary x_i' has crossed over the decision boundary. Naturally, the adjusted ϵ_i should satisfy:

$$\max_y c(g(x_i'))_y - c(g(x_i'))_{y_i} \leq \rho, \quad x_i' \in \mathcal{B}_p(x_i, \epsilon_i), \tag{11}$$

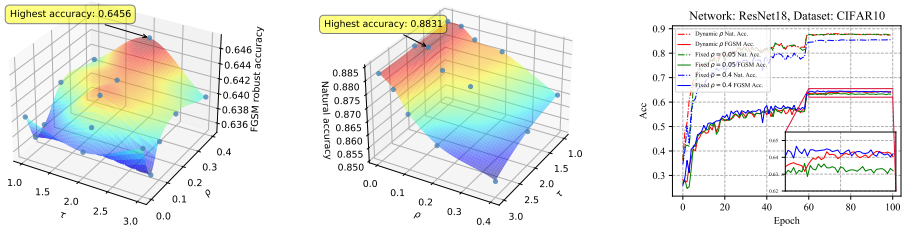
where the constraint (11) makes sure that x_i' which generated in $\mathcal{B}_p(x_i, \epsilon_i)$ will not overstep the natural decision boundary by a margin larger than ρ, otherwise the perturbation size ϵ_i is supposed to be scaled down. Note that, $c(\cdot)$ is supposed to be trained only with the natural examples, thus it can judge x_i' from the perspective of the natural decision boundary. If we replace $c(\cdot)$ with the adversarially trained $f(\cdot)$, the calibrated ϵ_i will adapt to the adversarial decision boundary, which cannot provide accurate information.

Specific strategy to satisfy the constraint (11) is still an open question. In this paper, the proposed DAAT estimates the perturbation size as follow:

$$\epsilon_i^{e+1} := \min\{\epsilon, \frac{\rho}{s_i} \cdot \epsilon_i^e\}, \tag{12}$$

$$s_i := \max_y \sigma\left[c(g(x_i'))/\tau\right]_y - \sigma\left[c(g(x_i'))/\tau\right]_{y_i}, \tag{13}$$

where $\sigma(\cdot)$ is the softmax operation, $\tau \geq 1$ is the temperature coefficient to smooth the output of $c(\cdot)$, e denotes the training epoch, $s_i \in (0, 1)$ in (13) is to



(a) Grid searching of τ and ρ for the optimal FGSM robust accuracy.

(b) Grid searching of τ and ρ for the optimal natural accuracy.

(c) Learning curves of the DAAT models trained with different ρs.

Fig. 3: Investigation of the fixed (ρ, τ) pairs and dynamic ρ strategy.

measure how far x_i' has overstepped the decision boundary, and ρ is a margin. Thus, if an adversary x_i' has crossed the decision boundary too far (e.g., $\rho/s_i < 1$), the perturbation bound will be shrunken in the next epoch and vice versa. In the following experiments, we also try different adjustment strategies for comparison. To sum up, the final objective of DAAT can be formulated as follows:

$$\min_{\psi} \mathbb{E}_{(x,y)} \left[\mathcal{L} \left(c(g(x); \psi), y \right) \right] + \min_{\omega,\theta} \mathbb{E}_{(x',y)} \left[\mathcal{L} \left(f(g(x'); \theta), y \right) \right], \tag{14}$$

$$x_i' \in \mathcal{B}_c(x_i, \epsilon_i) = \{ x_i' | \, \|x_i - x_i'\| \leq \epsilon_i, \quad \max_y c(g(x_i'))_y - c(g(x_i'))_{y_i} \leq \rho \}, \tag{15}$$

where $\mathcal{B}_c$ represents the calibrated perturbation ball. The detailed training routine of DAAT is summarized in Algorithm 1 for convenience.

6 Experiment

In this section, our proposed DAAT is evaluated on benchmark datasets, including CIFAR10 [16], and SVHN [26]. Strong baselines including AT [22], TRADES [34], MART [30] and FAT [35] are employed to verify the advantages of DAAT. The models are tested under prevalent attack types, including FGSM, PGD-T, C&W [5] optimized by PGD, and the commonly used benchmark attack AutoAttack (AA) [7]. Moreover, comprehensive experiments are conducted to investigate the thorough capability of DAAT. Due to space limitation, the training and attacking details for DAAT and baselines are all deferred to the Appendix. Implementation is available at here[1].

6.1 Investigation of Hyper-parameters

As we mentioned in Section 5, the margin ρ and temperature τ are two important hyper-parameters that affect the performance of DAAT. For the sake of optimal

[1] https://github.com/eccv2022daat/daat.git

performance, we first investigate the model's response to different hyper-parameter pairs. We conducted a grid search to investigate the influence of hyper-parameters. The searching spaces for ρ and τ are $\{0.01, 0.05, 0.1, 0.2, 0.4\}$ and $\{1, 2, 3\}$, respectively. Fig. 3 demonstrates the FSGM robust accuracy and natural accuracy of the model trained with different hyper-parameter pairs. As we can see from Fig. 3a and 3b, for a fixed τ, the robustness of the model generally increases with the growing ρ, while the natural accuracy of the model declines. A convincing reason for this trend is that the job of ρ is to control how much the prediction confidence of the adversarial examples can exceed that of the correct class (i.e., Eq. (13)). Consequently, if ρ is set to be larger, the generated adversary will be further away from its natural counterpart, which leads to the decreasing of the natural accuracy but the promoting of the robustness, and vice versa.

Fig. 3 illustrates that the optimal model for robust and natural accuracy can be both achieved at $\tau = 2$, but with different ρ values. To harmonize the conflict, we devise a novel training scheme by employing a dynamic ρ during different stages of training. Specifically, the model is initially trained with a small value of ρ with which only mild adversaries can be generated. As a consequence, the model can better generalize to the natural examples so that the induced calibration network can provide informative knowledge for the perturbation size adjustment. Then, ρ will be gradually enlarged to make sure that the model can adapt to stronger attacks. Meanwhile, with the help of the adaptive perturbation size, the natural accuracy would be retained as much as possible. In Fig. 3c, we plot the learning curves of the models trained with $\rho = 0.05$, $\rho = 0.4$, and dynamic ρ which is initialized with 0.05 and then enlarged to 0.2 and 0.3 at #75 and #90 epoch, respectively. From the learning curves, we can find that DAAT with dynamic ρ inherits the advantages of the fixed ρ models, e.g., it achieves a high level of natural accuracy while enjoying remarkable robustness. In the following experiments, we will all employ the dynamic ρ to train DAAT models unless otherwise specified.

6.2 White-box and Black-box Attacks

In this subsection, we evaluate the proposed DAAT method with both white-box and black-box attacks. The powerful ResNet18 was employed as the basic feature extractor[2]. For comparison, we combined the baselines with the DAAT training scheme denoted as the DAAT- models to investigate the performance promotion that DAAT brings.

White-box attack: The white-box attack assumes that all the information about the model is completely exposed to the attacker. As shown in Table 1, thanks to the data-adaptive perturbation size, the proposed DAAT achieves significantly higher natural accuracy compared with the baselines. Especially in the SVHN dataset, DAAT has a natural accuracy of 93.82% which is much closer to that of the natural trained model. Meanwhile, most of the DAAT-enhanced models' robustness is also improved or keep comparable. We attribute this benefit

[2] Experimental results with more network architectures are deferred to the Appendix.

Network: ResNet18, Attack: l_∞***-norm,*** $\epsilon = 0.031$***, Dataset: CIFAR10***								
Attacks / Methods	Clean	White-box FGSM	PGD-20	PGD-40	C&W-20	AA	Black-box (PDG-20) Non-Robust	AT
Natural	94.63%	17.32%	0%	0%	0%	0%	–	70.80%
Standard AT	84.66%	62.83%	47.83%	46.26%	46.24%	43.66%	83.67%	–
FAT	88.26%	63.73%	46.52%	43.73%	45.34%	43.28%	86.40%	64.11%
DAAT	**88.31%**	**64.56%**	**48.89%**	**46.93%**	**49.43%**	**44.32%**	**86.79%**	**64.25%**
TRADES	81.98%	63.24%	53.70%	52.48%	50.93%	49.22%	80.22%	62.08%
DAAT-TRADES	**83.55%**	**64.55%**	**54.57%**	**53.28%**	**51.30%**	**49.83%**	**82.05%**	**63.02%**
MART	80.67%	63.75%	53.87%	52.19%	50.25%	49.73%	79.78%	61.44%
DAAT-MART	**83.87%**	**65.75%**	**54.02%**	**52.63%**	**50.32%**	**49.82%**	**82.16%**	**62.01%**

Network: ResNet18, Attack: l_∞***-norm,*** $\epsilon = 0.031$***, Dataset: SVHN***								
Attacks / Methods	Clean	White-box FGSM	PGD-20	PGD-40	C&W-20	AA	Black-box (PDG-20) Non-Robust	AT
Natural	96.02%	45.38%	0.90%	0.30%	0.83%	0%	–	54.41%
Standard AT	92.74%	72.58%	54.89%	52.33%	52.10%	51.09%	88.35%	–
FAT	93.53%	70.48%	52.98%	49.85%	49.73%	50.63%	89.11%	61.12%
DAAT	**93.82%**	**73.03%**	**56.24%**	**53.72%**	**53.06%**	**52.33%**	**89.50%**	**61.84%**
TRADES	88.99%	69.68%	57.38%	55.69%	**51.74%**	**50.82%**	83.98%	60.84%
DAAT-TRADES	**90.28%**	**71.07%**	**58.59%**	**56.60%**	51.33%	50.59%	**85.19%**	**61.16%**
MART	89.91%	70.65%	58.00%	56.17%	51.47%	50.12%	84.89%	60.72%
DAAT-MART	**92.16%**	**73.81%**	**59.68%**	**57.42%**	**51.81%**	**50.66%**	**87.49%**	**61.28%**

Table 1: Robust accuracy on CIFAR10 and SVHN under different attacks.

to the dynamic ρ, for it can provide the model with a learning environment as Curriculum Learning [2], with which the robustness can be smoothly and better improved. It is undeniable that the FAT method can achieve comparable natural accuracy as DAAT, however, the robustness of the FAT trained model seems to suffer a severe decline. A possible explanation is that the objective of FAT replaces the inner maximization in Eq. 3 with a minimization objective under certain constraints. Thus, FAT cannot find the most effective adversaries to improve the robustness with the early-stopped PGD technique [35]. In comparison, although DAAT also shrinks the perturbation space, it still tries to find the most informative adversaries in a smaller perturbation ball.

Black-box attack: In the black-box attack setting, the attacker has no access to the target model. The attacker has to generate adversaries by attacking a surrogate model.

In this paper, we employ the natural trained model and the standard AT model as the surrogate models. As in the last two columns in Table 1, an interesting trend is that the models with higher robustness towards the white-box attacks (e.g., TRADES, MART) usually have lower robustness towards the black-box attacks. We suspect that this is probably because the robustness towards a specific kind of attack will impede the transferability of robustness. Nonetheless,

the DAAT-enhanced models can generally keep or surpass the robustness against black-box attacks compared with their baselines.

6.3 Distribution of Adversaries

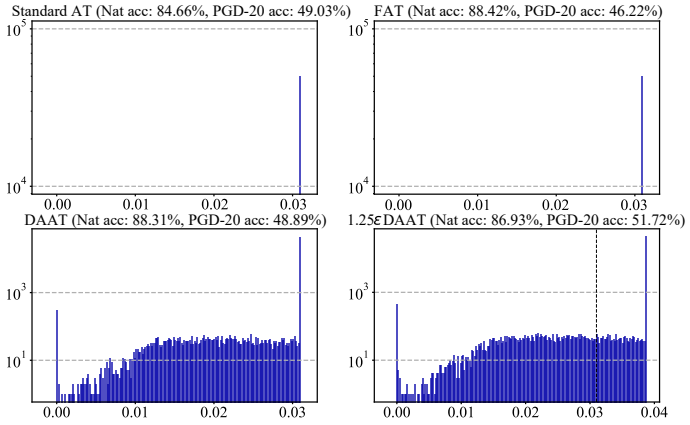
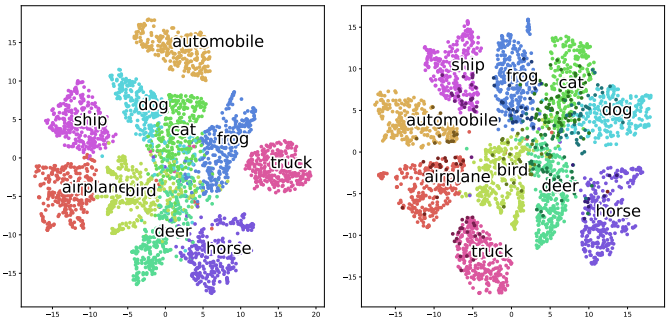


Fig. 4: Histograms of the distance between the adversarial examples for training and the natural examples. The horizontal and vertical axes represent the distance to the natural example and the number of examples, respectively.



(a) AT model natural data. (b) DAAT model natural data.

Fig. 5: T-SNE visualizations of natural representations obtained from AT and DAAT models. The darker points in the right figure are the examples whose data-adaptive perturbation size is smaller than the preset ϵ.

Traditional adversarial training methods usually search the worst-case adversarial example in a fixed size perturbation ball. As a consequence, the generated adversaries tend to appear on the surface of the perturbation ball. However,

DAAT explicitly restrains the perturbation space of the adversaries, thus the adversaries should be more broadly distributed in the preset perturbation ball. For verification, we plot the histograms representing the distance between the adversarial examples and their corresponding natural examples during the training in Fig. 4.

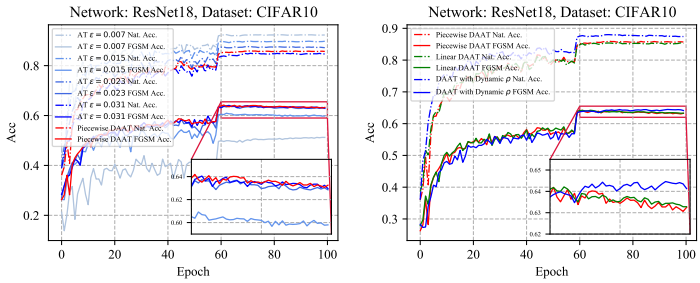
What we can find from the first histograms is that the adversaries generated by the standard AT are all located on the surface of the perturbation ball, which means that for the relatively attackable data, the generated adversaries may change too much. That may explain why the natural accuracy of AT usually significantly declines. Surprisingly, just as the standard AT method, the adversarial examples generated by FAT are also on the perturbation ball surface even with the early stopping technique. However, the robustness of FAT is still comparatively lower since it does not aim at looking for the strongest adversaries. In contrast, the adversaries generated by DAAT are widely distributed throughout the l_∞-norm perturbation ball. Thus, DAAT is able to achieve a significantly higher natural accuracy while keeping robustness. Note that, for a fair comparison with the baselines, the DAAT is upper bounded by the preset perturbation size ϵ as in Eq. (12). Nonetheless, for some robust natural examples, the upper bound ϵ may be too small to find an informative adversarial example. Thus, we further enlarge the upper bound to 1.25ϵ to see the adversary distribution of the enlarged DAAT model. As we can see in the fourth histogram, some of the adversarial examples generated by the enlarged DAAT method can be found outside the perturbation ball, which implies that for the robust natural examples, the most useful adversaries lie outside the preset ball. This may explain why the enlarged DAAT model is more robust than the DAAT model.

Besides, in Fig. 5, we further visualize the representations of the natural data extracted from AT model and DAAT model via T-SNE [20]. As we can see from the figure, the data clusters derived by DAAT are more separable. Moreover, the darker points in Fig. 5b represent the natural examples whose adversaries are generated in a shrunken perturbation ball. Interestingly, these points are usually at the edge of the clusters, which indicates that the attackable examples are near the natural decision boundary.

6.4 DAAT with Different Calibration Strategies.

As we discuss in Section 5, the adjustment of the perturbation size is an open problem. In this subsection, we investigate the influence of different adjustment strategies on the performance of DAAT.

Piecewise DAAT: A direct idea is the piecewise DAAT, which assigns a piecewise perturbation size (i.e., $\epsilon \in \{0, \frac{2}{255}, \frac{4}{255}, \frac{6}{255}, \frac{8}{255}\}$) for an adversary based on the output of the calibration network. For comparison, we first plot the learning curves of the AT models trained with the aforementioned perturbation sizes in Fig. 6a. It is obvious that the AT model with a larger fixed perturbation size can achieve higher robustness but lower natural accuracy. However, the piecewise DAAT can not only keep outstanding robustness but also slightly promote natural accuracy.



(a) AT models with different preset perturbation sizes.

(b) DAAT models with different adjustment strategies.

Fig. 6: Learning curves of AT models with different preset ϵs and DAAT models with different perturbation size adjustment strategies.

Linear DAAT: Linear DAAT employs a linear mapping to derive the adjusted perturbation size as follows:

$$\epsilon_i^{e+1} := \min\{\epsilon, (1 - s_i + \rho) \cdot \epsilon\}, \tag{16}$$

where s_i is the same as Eq. (13), ρ is the margin value. We plot the learning curves of DAAT with different adjustment strategies in Fig. 6b. As we can see from the figure, the default DAAT trained with Eq. (12) receives the best empirical performance. A possible reason is that Eq. (12) utilizes the historical information of ϵ_i^e which can smooth the variation of the perturbation size to stabilize the training process. More discussion about the calibration network can be found in the Appendix.

7 Conclusions

In this paper, we first find that, in adversarial training, the decline of the natural accuracy may come from the fixed perturbation size, since one perturbation size does not fit all training data. Thus, we propose a novel adversarial training strategy DAAT which adaptively adjusts the perturbation size for each training data. To achieve better natural accuracy, the adjustment is performed by a natural trained calibration network, and a dynamic training strategy further empowers the DAAT models with impressive robustness. Although the experimental results have demonstrated the empirical superiority of our method, the better perturbation size adjustment strategy is still an open problem to explore.

Acknowledgements

This work was supported in part by the Australian Research Council under Project DP210101859 and the University of Sydney Research Accelerator (SOAR) Prize.

References

1. Balaji, Y., Goldstein, T., Hoffman, J.: Instance adaptive adversarial training: Improved accuracy tradeoffs in neural nets. arXiv preprint arXiv:1910.08051 (2019)
2. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: Proceedings of the 26th annual international conference on machine learning. pp. 41–48 (2009)
3. Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L.D., Monfort, M., Muller, U., Zhang, J., et al.: End to end learning for self-driving cars. arXiv preprint arXiv:1604.07316 (2016)
4. Carlini, N., Wagner, D.: Adversarial examples are not easily detected: Bypassing ten detection methods. In: Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security. pp. 3–14 (2017)
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. Collobert, R., Weston, J.: A unified architecture for natural language processing: Deep neural networks with multitask learning. In: Proceedings of the 25th international conference on Machine learning. pp. 160–167 (2008)
7. Croce, F., Hein, M.: Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In: ICML (2020)
8. Feinman, R., Curtin, R.R., Shintre, S., Gardner, A.B.: Detecting adversarial samples from artifacts. arXiv preprint arXiv:1703.00410 (2017)
9. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572 (2014)
10. Grosse, K., Manoharan, P., Papernot, N., Backes, M., McDaniel, P.: On the (statistical) detection of adversarial examples. arXiv preprint arXiv:1702.06280 (2017)
11. 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)
12. Hendrycks, D., Gimpel, K.: Early methods for detecting adversarial images. In: ICLR (2017)
13. Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., Madry, A.: Adversarial examples are not bugs, they are features. In: NeurIPS. pp. 125–136 (2019)
14. Kannan, H., Kurakin, A., Goodfellow, I.: Adversarial logit pairing. arXiv preprint arXiv:1803.06373 (2018)
15. Kim, M., Tack, J., Hwang, S.J.: Adversarial self-supervised contrastive learning. Advances in Neural Information Processing Systems **33**, 2983–2994 (2020)
16. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
17. Kundu, S., Nazemi, M., Beerel, P.A., Pedram, M.: A tunable robust pruning framework through dynamic network rewiring of dnns. arXiv preprint arXiv:2011.03083 (2020)
18. Li, X., Li, F.: Adversarial examples detection in deep networks with convolutional filter statistics. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 5764–5772 (2017)
19. Liao, F., Liang, M., Dong, Y., Pang, T., Hu, X., Zhu, J.: Defense against adversarial attacks using high-level representation guided denoiser. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1778–1787 (2018)
20. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(11) (2008)

21. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. In: ICLR (2018)
22. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. In: International Conference on Learning Representations (2018), https://openreview.net/forum?id=rJzIBfZAb
23. Mao, C., Zhong, Z., Yang, J., Vondrick, C., Ray, B.: Metric learning for adversarial robustness. Advances in Neural Information Processing Systems **32** (2019)
24. Metzen, J.H., Genewein, T., Fischer, V., Bischoff, B.: On detecting adversarial perturbations. In: ICLR (2017)
25. Moosavi-Dezfooli, S.M., Fawzi, A., Uesato, J., Frossard, P.: Robustness via curvature regularization, and vice versa. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9078–9086 (2019)
26. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y.: Reading digits in natural images with unsupervised feature learning (2011)
27. 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)
28. Tramèr, F., Behrmann, J., Carlini, N., Papernot, N., Jacobsen, J.H.: Fundamental tradeoffs between invariance and sensitivity to adversarial perturbations. In: International Conference on Machine Learning. pp. 9561–9571. PMLR (2020)
29. Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., Madry, A.: Robustness may be at odds with accuracy. In: International Conference on Learning Representations (2018)
30. 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)
31. Xie, C., Wu, Y., Maaten, L.v.d., Yuille, A.L., He, K.: Feature denoising for improving adversarial robustness. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 501–509 (2019)
32. Xu, W., Evans, D., Qi, Y.: Feature squeezing: Detecting adversarial examples in deep neural networks. arXiv preprint arXiv:1704.01155 (2017)
33. Zagoruyko, S., Komodakis, N.: Wide residual networks. In: Richard C. Wilson, E.R.H., Smith, W.A.P. (eds.) Proceedings of the British Machine Vision Conference (BMVC). pp. 87.1–87.12. BMVA Press (September 2016). https://doi.org/10.5244/C.30.87, https://dx.doi.org/10.5244/C.30.87
34. 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)
35. Zhang, J., Xu, X., Han, B., Niu, G., Cui, L., Sugiyama, M., Kankanhalli, M.: Attacks which do not kill training make adversarial learning stronger. In: International Conference on Machine Learning. pp. 11278–11287. PMLR (2020)
36. Zhang, J., Zhu, J., Niu, G., Han, B., Sugiyama, M., Kankanhalli, M.: Geometry-aware instance-reweighted adversarial training. In: International Conference on Learning Representations (2021), https://openreview.net/forum?id=iAX0l6Cz8ub