Supplementary Material for:
Efficient Adversarial Training with Transferable Adversarial Examples

Haizhong Zheng Ziqi Zhang Juncheng Gu Honglak Lee Atul Prakash
University of Michigan, Ann Arbor
{hzzheng, ziqizh, jcgu, honglak, aprakash}@umich.edu

1. Overview

This supplementary material provides details on our experiment and additional evaluation results. In Section 2, we introduce the detailed setup of our experiment. In Section 3, we compare adversarial examples generated by ATTA and PGD and show that, even with one attack iteration, ATTA-1 can generate similar perturbations to PGD-40 (PGD-10) on MNIST (CIFAR10). We also provide the complete evaluation results in Section 3.2.

2. Experiment setup

We provide additional details on the implementation, model architecture, and hyper-parameters used in this work.

MNIST. We use the same model architecture used in [6, 11, 12], which has four convolutional layers followed by three fully-connected layers. The adversarial examples used to train the model are bounded by l_∞ ball with size $\epsilon = 0.3$ and the step size for each attack iteration is 0.01. We do not apply any data augmentation (and inverse data augmentation) on MNIST and set the epoch period to reset perturbation as infinity which means that perturbations are not reset during the training. The model is trained for 60 epochs with an initial 0.1 learning rate and a 0.01 learning rate after 55 epochs, which is the same as [11]. To evaluate the model robustness, we perform the PGD [5], M-PGD [2] and CW [1] attack with a 0.01 step size and set decay factor as 1 for M-PGD (momentum PGD).

CIFAR10. Following other works [6, 8, 11, 12], we use Wide-Resnet-34-10 [10] as the model architecture. The adversarial examples used to train the model are bounded by l_∞ ball with size $\epsilon = 0.031$. For ATTA-$1, 2, 3$, we use $0.015, 0.01, 0.01$ as the step size, respectively. For ATTA-k ($k > 3$), we use 0.007 as the step size. The data augmentation used is a random flip and 4-pixel padding crop, which is same with other works [6,8,11,12]. We set the epoch period to reset perturbation as 10 epochs. Following YOPO [11], the model is trained for 40 epochs with an initial 0.1 learning rate, a 0.01 learning rate after 30 epochs, and a 0.001 learning rate after 35 epochs. To evaluate the model robustness, we perform the PGD, M-PGD and CW attack with a 0.003 step size and set decay factor as 1 for M-PGD (momentum PGD).

ImageNet. We use Resnet-50 [3] as the model architecture. The adversarial examples used to train the model are bounded by l_∞ ball with size $\epsilon = 2/255$, and we use we use $1/255$ as the step size. The model is trained for 20 epochs with a piece-wise learning rate which is 0.4 when epoch is less than 6, 0.04 when epoch is between 6 and 11, 0.004 when epoch is between 12 and 14, 0.0004 when epoch is larger than 14.

For the baseline, we use the author implementation of MAT[1] [6], TRADES[2] [12], YOPO[3] [12], and Free[4] [8] with the hyper-parameters recommended in their works, and we select $1/\lambda$ as 6 for TRADES (both ATTA and PGD).

In Section 3, which analyzes the transferability between training epochs, we use MAT with PGD-10 to train models and PGD-20 to calculate loss value and error rate.

Each experiment is taken on one idle NVIDIA GeForce RTX 2080 Ti GPU. Except PGD attack, we implement other attacks with Adversarial Robustness Toolbox [7].

3. Experiment details

3.1. Qualitative study on training images

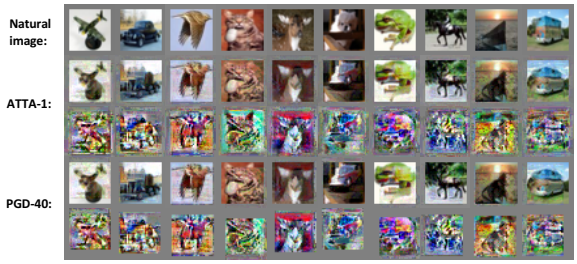
To compare the quality of adversarial examples generated by PGD and ATTA, we visualize some adversarial examples generated by both methods. For MNIST, we choose the model checkpoint trained by MAT-ATTA-1 at epoch 40. For CIFAR10, we choose the model checkpoint trained by MAT-ATTA-1 at epoch 30. Figure 1 shows the adversarial examples and perturbations used to train the model (ATTA-1) and genereated by PGD-40 (PGD-10) attack on MNIST (CIFAR10) model in each class. To better visualize the per-

[1] https://github.com/MadryLab/mnist_challenge https://github.com/MadryLab/cifar10_challenge
[2] https://github.com/yaodongyu/TRADES
[3] https://github.com/a1600012888/YOPO-You-Only-Propagate-Once
[4] https://github.com/ashafahi/free_adv_train

turbation, we re-scale the perturbation by calculating $\frac{2x_p}{\epsilon}$ (where x_p is the perturbation and ϵ is the L_∞ bound of adversarial attack). This shifts the L_∞ ball to the scale of $[0, 1]$.

We find that, although ATTA-1 just performs one attack iteration in each epoch, it generates similar perturbations to PGD-40 (PGD-10) in MNIST (CIFAR10). The effect of inverse data augmentation is shown in Figure 1b. There are some perturbations on the padded pixels in the third row (ATTA-1), but perturbations just generated by PGD-10 (shown in fifth row) just appear on cropped pixels.

(a) MNIST



(b) CIFAR10

Figure 1: Visualization of natural images, adversarial examples and corresponding perturbations in each class for MNIST and CIFAR10. The first row in (a) and (b) shows the natural images. The second and third rows show the adversarial examples and perturbations generated by ATTA-1. The fourth and fifth rows show the adversarial examples and perturbations generated by PGD-40 in (a) and PGD-10 in (b).

3.2. Complete evaluation results

We put the complete evaluation result in this section as a supplement to Section 5.2.

We evaluate defense methods under additional attacks and the evaluation results are shown in Table 2 and Table 3. Similar to the conclusion stated in Section 5.2, compared to other methods, ATTA achieves comparable robustness with much less training time, which shows a better trade-off on training efficiency and robustness. With the same number of attack iterations, ATTA needs less time to train the model. As mentioned in Section 3.2, with the accumulation of adversarial perturbations, ATTA can use the same number of attack iterations to achieve a higher attack strength, which helps the model converge faster.

Also, we test our models with stronger attacks which have a different attack size and multipule random starts. We evaluate ATTA-1 model trained with both MAT loss and TRADES loss against PGD-20 attack with an $\epsilon/4$ step-size and 20 independent random starts.

Defense \ Attack	$\epsilon/4$ step-size	20 random starts
MAT-ATTA-1	49.56%	48.48%
TRADES-ATTA-1	53.71%	52.90%

Table 1: Accuracy against PGD-20 attack with different settings.

4. Discussion

Natural accuracy v.s. Adversarial accuracy. In this paper, we find that higher adversarial accuracy can lower natural accuracy. This trade-off has been observed and explained in [9, 12]. A recent work [4] points out that features used by naturally trained models are highly-predictive but not robust and adversarially trained models tend to use robust features rather than highly-predictive features, which may cause this trade-off. Table 3 also shows that, when models are trained with stronger attacks (more attack iterations), the models tend to have higher adversarial accuracy but lower natural accuracy.

Transferability between training epochs. Adversarial attacks augment the training data to improve the model robustness. Our work points out that, unlike images augmented by traditional data augmentation methods that are independent between epochs, adversarial examples generated by adversarial attacks show high relevance transferability between epochs. We hope this finding can inspire other researchers to enhance adversarial training from a new perspective (e.g., improving transferability between epochs).

Inter-epoch reuse in ATTA v.s. inner-batch reuse in Free. Both ATTA and Free reuse perturbations during training. The difference between the two methods is how to reuse perturbations (Fig. 2). Free reuses perturbations through iterations on the top of the same batch of images (**inner-batch**), but ATTA reuses perturbations across epochs (**inter-epoch**), which is complementary to inner-batch reuse. Inter-epoch reuse is a non-trivial problem involving new challenges discussed in Section 4.1. The notion of connection function, which is a crucial feature of ATTA to overcome these challenges, which only applies when exploiting transferability across epochs, does not exist in Free. We found that ATTA can outperform Free, even without any inner-batch reuse and Free's warm start be-

tween different batches (In Free, when a new batch comes, the perturbations of the previous batch are directly used as a warm start, but the mismatch between images and perturbations could be resulting in reduced effectiveness.).

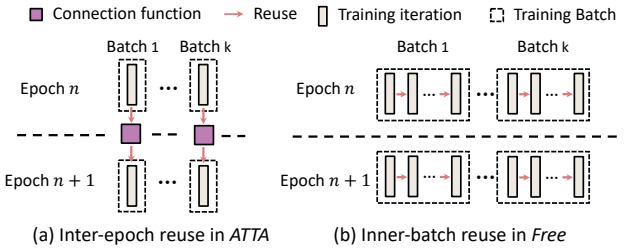


Figure 2: Comparison on reuse between Free and ATTA.

Defense \ Attack		Natural	PGD-40	PGD-100	M-PGD-40	FGSM	CW-20	Time (sec)
MAT	PGD-1	99.52%	15.82%	4.17%	13.99%	81.84%	98.99%	226
	PGD-10	99.52%	34.92%	16.90%	36.88%	94.32%	99.38%	1649
	PGD-40	99.37%	**96.21%**	94.69%	96.79%	97.37%	99.06%	**3933**
	YOPO-5-10	99.15%	93.69%	87.93%	99.04%	94.51%	99.03%	789
	ATTA-1	99.45%	**96.31%**	95.11%	96.15%	98.31%	99.14%	**297**
	ATTA-10	99.41%	97.36%	96.75%	97.45%	98.37%	99.3%	1687
	ATTA-40	99.23%	97.28%	96.85%	97.51%	98.55%	98.02%	4650
TRADES	PGD-1	99.41%	39.53%	31.28%	36.42%	71.01%	99.17%	583
	PGD-10	99.37%	50.06%	25.71%	50.59%	95.12%	99.24%	1823
	PGD-40	98.89%	**96.54%**	95.54%	96.79%	97.83%	98.87%	**6544**
	ATTA-1	99.03%	**96.10%**	94.24%	95.86%	98.38%	98.93%	**460**
	ATTA-10	98.83%	96.86%	96.34%	96.90%	98.11%	98.68%	1686
	ATTA-40	98.21%	96.03%	96.33%	96.80%	97.85%	98.32%	4660

Table 2: The result of various attacks on MNIST dataset.

Defense \ Attack		Natural	PGD-20	PGD-100	M-PGD-20	FGSM	CW-20	Time (min)
MAT	PGD-1	93.18%	22.3%	21.63%	23.51%	49.88%	31%	435
	PGD-2	91.33%	35.16%	34.46%	38.61%	55.84%	44.12%	600
	PGD-3	89.95%	41.38%	40.59%	42.17%	59.82%	50.94%	785
	PGD-10	87.49%	**47.07%**	46.77%	47.95%	63.5%	56.7%	**2027**
	Free($m = 8$)	85.54%	47.68%	47.22%	48.02%	60.22%	57.58%	640
	YOPO-5-3	86.43%	48.24%	47.74%	51.87%	59.78%	81.98%	335
	ATTA-1	85.71%	**50.96%**	48.18%	52.31%	64.16%	76%	**134**
	ATTA-2	85.95%	51.90%	49.45%	53.08%	64.42%	77.02%	196
	ATTA-3	85.44%	52.56%	50.77%	53.92%	64.82%	75.92%	267
	ATTA-10	83.80%	54.33%	52.6%	55.38%	63.49%	75.37%	690
TRADES	PGD-1	93.58%	35.52%	31.84%	38.42%	65.03%	86.46%	592
	PGD-2	90.61%	49.75%	45.66%	51.19%	60.86%	85.48%	730
	PGD-3	88.52%	54.20%	52.19%	55.91%	62.88%	84.48%	861
	PGD-10	84.13%	**56.6**%	55.36%	57.76%	63.02%	79.4%	**2028**
	ATTA-1	85.04%	54.50%	52.66%	55.70%	64.04%	82.62%	199
	ATTA-2	84.58%	54.94%	53.27%	56.17%	63.72%	81.92%	261
	ATTA-3	84.23%	**56.36%**	54.85%	57.24%	64.03%	81.82%	**320**
	ATTA-10	83.67%	57.34%	56.39%	58.15%	64.1%	82.01%	752

Table 3: The result of various attacks on CIFAR10 dataset.

References

[1] Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 3–14. ACM, 2017.

[2] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9185–9193, 2018.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[4] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. *arXiv preprint arXiv:1905.02175*, 2019.

[5] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016.

[6] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*, 2018.

[7] Maria-Irina Nicolae, Mathieu Sinn, Minh Ngoc Tran, Beat Buesser, Ambrish Rawat, Martin Wistuba, Valentina Zantedeschi, Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, Ian Molloy, and Ben Edwards. Adversarial robustness toolbox v1.2.0. *CoRR*, 1807.01069, 2018.

[8] Ali Shafahi, Mahyar Najibi, Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! *arXiv preprint arXiv:1904.12843*, 2019.

[9] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations (ICLR)*, 2019.

[10] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[11] Dinghuai Zhang, Tianyuan Zhang, Yiping Lu, Zhanxing Zhu, and Bin Dong. You only propagate once: Accelerating adversarial training via maximal principle. In *Neural Information Processing Systems (NeurIPS)*, 2019.

[12] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P Xing, Laurent El Ghaoui, and Michael I Jordan. Theoretically principled trade-off between robustness and accuracy. 2019.