PeerAiD: Improving Adversarial Distillation from a Specialized Peer Tutor

Supplementary Material

1. Appendix

1.1. Detailed Description of Experimental Settings

PeerAiD. For CIFAR-10, we set $\gamma_1 = 1$, $\gamma_2 = 0.1$, $\lambda_1 = 1$, $\lambda_2 = 0$, $\lambda_3 = 1$. The temperature for the peer loss and the student loss are 1 and 5, respectively. We used the same hyperparameters for ResNet-18 and WideResNet34-10. For CIFAR-100, we used the same hyperparameters as CIFAR-10 except for γ_2. We only changed γ_2 to 1 because we found that the knowledge of a student model is useful for the dataset with a large number of classes. We also used the same hyperparamters for ResNet-18 and WideResNet34-10 with CIFAR-100. For TinyImageNet, we set $\gamma_1 = 1$, $\gamma_2 = 100$, $\lambda_1 = 35$, $\lambda_2 = 0.035$, $\lambda_3 = 20$. The temperatures of the peer loss and student loss are 1 and 1, respectively. We used the same coefficients and the temperature in the loss terms for ResNet-18 and WideResNet34-10 with TinyImageNet. In all our experiments, the peer model and student model used the same parameters of training epoch, learning rate, batch size, and weight decay.

Baselines. We followed their original settings for baselines as mentioned in Sec. 4.1. In detail, there are two versions of IAD [11] in the original paper. IAD presented IAD-I and IAD-II depending on whether a naturally trained teacher is used or not. We chose IAD-I in all experiments because the paper mentioned IAD-I generally shows better robustness with the teacher model pretrained by TRADES. We also tested IAD-II with WideResNet34-10 on CIFAR-100 and found a consistent result which shows a higher robust accuracy of IAD-1 than IAD-II with the robust teacher pretrained by TRADES. CAT [7] mentions that any two adversarial training methods can be used to train two student models collaboratively. We chose TRADES [10] and Adversarial Logit Paring (ALP) [6] to train the two student models because the paper mentioned TRADES and ALP show superior performance in terms of AutoAttack (AA) robust accuracy. We measured the robust accuracy of both student models with AutoAttack and reported the higher robust accuracy between the two student models for CAT.

1.2. Robustness against Other Attacks

We checked the robustness of PeerAiD and the baselines with two additional attacks. We chose CW_2 attack [1] and MI-FGSM attack [3] to check the robustness of baselines and PeerAiD. Overall, PeerAiD shows higher robust accuracy against them, as illustrated in Tab. 8. In particular, PeerAiD improves the robust accuracy against CW_2 attack by up to 3.2%p with WideResNet34-10 on TinyImageNet. CW attack uses margin-based loss and minimizes the norm of the perturbation. We conducted CW_2 attack following [4] and set the balance constant c to 0.1. We chose l_2 norm for the norm of the perturbation in CW attack because the original paper [1] mentioned the defenders should show the robustness against l_2 attack. MI-FGSM attack is a momentum-based iterative method to find adversarial examples. We set the iteration number to 10 and the decaying factor to 1 with MI-FGSM attack. All other experimental settings are the same as in Sec. 4.1.

Method	ResNet-18		WideResNet34-10	
	CW_2	MI-FGSM	CW_2	MI-FGSM
PGD-AT	44.76	23.80	49.26	26.81
TRADES	46.55	23.89	50.46	26.91
AKD^2	48.46	26.47	52.43	30.52
RSLAD	41.81	24.04	44.48	26.46
IAD	45.82	25.15	49.00	29.32
CAT	39.10	22.58	39.53	24.01
AdaAD	47.90	25.08	49.96	27.61
PeerAiD	**53.13**	**28.03**	**55.63**	**31.32**

Table 8. Test robust accuracy of the models trained by the baselines and PeerAiD against CW_2 and MI-FGSM attack with TinyImageNet.

1.3. Difference from the Prior Art

Many previous works which aim at adversarial distillation require a pretrained robust model, whereas PeerAiD does not require the pretrained robust model. CAT [7] proposed online adversarial distillation which also collaboratively trains two student models. However, PeerAiD significantly differs from CAT regarding the inner maximization process. CAT independently attacks two student models with different attack methods because their approach is based on the idea that each student model trained by distinct attack methods learns different features. On the other hand, PeerAiD attacks only a single student model, and thus, the computational cost of adversarial distillation is twice times smaller than CAT, as emphasized in Tab. 13. In addition, while previous works did not focus on the non-transferability of adversarial examples, PeerAiD considers this aspect. Specifically, as illustrated in Tab. 2, the adversarial examples generated from the student model are not strong enough to fool the peer model because the peer model becomes the specialist who defends the adversarial examples aimed at the student model. Based on the above finding, PeerAiD lets the peer model guide the student model.

Method	DenseNet-BC-40		DenseNet-40	
	Clean	AA	Clean	AA
Natural	**92.75**	0.00	**94.39**	0.00
PGD-AT	77.01	41.84	80.79	44.82
TRADES	73.73	39.78	76.81	44.81
AKD^2	75.19	41.70	78.41	46.16
RSLAD	71.63	41.00	74.94	46.05
IAD	71.96	42.48	76.34	46.64
CAT	73.08	41.07	75.91	44.77
AdaAD	72.63	39.45	75.80	43.62
PeerAiD	75.93	**43.26**	78.21	**47.15**

Table 9. Test robust accuracy and clean accuracy of DenseNet-BC-40 and DenseNet-40 models trained by PeerAiD and the baselines on CIFAR-10.

Perturbation budget	$\epsilon = 10/255$			
Method	Clean	FGSM	PGD-20	AA
PGD-AT	55.15	29.73	26.69	23.66
TRADES	51.82	30.25	27.83	23.46
AKD^2	**56.65**	33.49	30.88	26.71
RSLAD	54.53	33.30	31.09	25.99
IAD	53.67	33.01	30.89	26.17
CAT	55.19	**34.14**	**32.17**	26.50
AdaAD	55.72	31.91	29.97	25.29
PeerAiD	54.39	33.45	30.20	**27.19**

Table 10. Test robust accuracy of ResNet-18 trained by various methods with the large perturbation budget $\epsilon = 10/255$ on CIFAR-100.

1.4. Evaluation on Different Models

We also checked the effectiveness of PeerAiD with additional models. We compared PeerAiD and baselines with DenseNet-BC-40 and DenseNet-40 [5]. The number of layers L was set to 40 and the growth rate $k = 12$ was chosen. As illustrated in Tab. 9, PeerAiD shows higher AutoAttack (AA) robust accuracy than the baselines with DensNet and DenseNet-BC. DenseNet-BC has fewer feature maps than DenseNet, and it is a compressed version of DenseNet. Therefore, the robustness of DenseNet-BC is lower than that of DenseNet because DenseNet-BC has fewer parameters and a smaller model capacity [8]. PeerAiD shows higher robust accuracy and a better trade-off between robustness and clean accuracy with both DenseNet-BC and DenseNet. PeerAiD improves the AutoAttack robust accuracy of DenseNet-BC-40 by up to 0.78%p and DenseNet-40 by up to 0.51%p.

Method	AA Robust Accuracy			Clean Accuracy		
	Best	Final	Diff	Best	Final	Diff
PGD-AT	21.84	18.61	3.23	57.30	55.92	1.38
TRADES	23.69	23.65	0.04	54.90	55.39	−0.49
AKD^2	25.83	25.43	0.40	58.84	59.82	−0.98
RSLAD	25.96	26.03	-0.07	55.45	55.28	0.17
IAD	25.44	25.22	0.22	54.98	55.42	−0.44
CAT	25.93	25.11	0.82	57.81	58.48	−0.67
AdaAD	25.03	24.79	0.24	56.08	56.29	−0.21
PeerAiD	27.33	27.28	0.05	59.35	59.38	−0.03

Table 11. Robust overfitting comparison of various methods on CIFAR-100 with ResNet-18. The best checkpoint was selected based on the test robust accuracy using PGD-10.

1.5. Larger Search Radius of ϵ

We tested the baselines and PeerAiD with the student model trained on adversarial examples generated using the larger perturbation budget of $\epsilon = 10/255$. In Sec. 4.1, we compared the baselines and PeerAiD with the student model trained on the adversarial examples generated using the perturbation budget of $\epsilon = 8/255$. All other training settings are the same as in Sec. 4.1. We kept the perturbation budget at $\epsilon = 8/255$ during testing. As illustrated in Tab. 10, PeerAiD shows superior AutoAttack (AA) robust accuracy with the student model trained on the adversarial examples generated using a large perturbation budget. We conducted adversarial training and adversarial distillation with ResNet-18 on CIFAR-100. All baselines and PeerAiD harm clean accuracy with a large perturbation budget due to the trade-off between robustness and clean accuracy. The decrease in the clean accuracy with a large perturbation budget is the smallest in AdaAD [4] as mentioned in the original paper. However, PeerAiD still surpasses the AA robust accuracy of AdaAD by a large margin of 1.9%p.

1.6. Mitigating Robust Overfitting

It is known that robust overfitting is prevalent in adversarial training [2, 9], and many previous works of adversarial training can be defeated by early-stopping. Robust overfitting is the phenomenon where the test robust accuracy peaks shortly after the first learning rate decay and then degrades until the last training epoch. Tab. 11 shows that PeerAiD exhibits less robust overfitting than most of the baselines. Robust overfitting is often measured with the difference between the test robust accuracy of the best and last checkpoints. PeerAiD shows higher robust accuracy at both the best and last checkpoint of the student model than the baselines. AutoAttack robust accuracy of PeerAiD at the last training epoch is even higher than the robust accuracy of all other baselines at their best epochs. This superior performance of PeerAiD comes from the fact that PeerAiD ef-

Temperature	1	2	5	10
AKD^2	26.11	26.37	26.44	26.22
RSLAD	26.32	24.70	24.69	24.17
IAD	25.60	25.94	25.26	24.80
AdaAD	24.89	24.37	24.06	24.22
PeerAiD	**27.06**	**27.41**	**27.33**	**27.41**

Table 12. AutoAttack robust accuracy of ResNet-18 trained by PeerAiD and the baselines on CIFAR-100 under various temperature settings.

fectively mitigates robust overfitting with only 0.05%p difference in the robust accuracy between the best checkpoint and the last checkpoint. As illustrated in Fig. 3(a), the test robust accuracy of the student model trained by PeerAiD does not suffer from robust overfitting in the experiment. The test robust accuracy of the student model distilled by the peer model does not reach its peak shortly after the first learning rate decay at epoch 215. The best checkpoint is attained at epoch 265 with PeerAiD in Fig. 3(a). The best robust and clean accuracy of RSLAD is lower than those of the last checkpoint because the best checkpoint was chosen based on PGD-10 test robust accuracy, whereas the robust accuracy in Tab. 11 is measured with AutoAttack robust accuracy and the clean accuracy is measured with clean data.

1.7. The Effect of Temperature for Adversarial Distillation

In Tab. 12, we tested the sensitivity of PeerAiD and the baselines with respect to the temperature parameter of the loss in the student model. Both PeerAiD and the baselines have a distillation term, and we varied the temperature in the distillation term for the sensitivity study. The adversarial distillation is conducted with ResNet-18 on CIFAR-100. All other experimental settings are the same as in Sec. 4.1. We tested four temperature values $\{1, 2, 5, 10\}$ and checked that PeerAiD maintains higher AutoAttack robust accuracy in all tested temperatures. This result shows that the higher robustness of PeerAiD is insensitive to the the temperature parameter of the distillation term in the loss of the student model.

1.8. Analysis of Training Time

We checked the training time of PeerAiD and the baselines to compare the computational cost among various methods. As illustrated in Tab. 13, PeerAiD shows comparable total training time compared to other adversarial distillation methods. Most of the baselines require a pretrained robust model except for CAT [7]. We included the pretraining time in the total training time of the baselines, which required pretraining because the pretraining should be conducted beforehand to run them. CAT requires nearly 2× time to perform adversarial distillation compared to PeerAiD, though CAT is also an online adversarial distillation method. The substantial computational cost of CAT arises from the necessity to attack two student models using different attack methods, whereas PeerAiD only needs to attack one student model and PeerAiD does not attack the peer model. The time cost is measured with a single A100 GPU.

Method	Pretraining Time	Distillation Time	Total Training Time
Natural	-	-	2.01 hours (A)
TRADES	-	-	11.64 hours (B)
AKD^2	13.65 hours (A+B)	15.87 hours	29.52 hours
RSLAD	11.64 hours (B)	26.21 hours	37.85 hours
IAD	11.64 hours (B)	20.54 hours	32.18 hours
CAT	-	59.44 hours	59.44 hours
AdaAD	11.64 hours (B)	28.89 hours	40.53 hours
PeerAiD	-	30.50 hours	30.50 hours

Table 13. Time cost of adversarial distillation methods with WideResNet34-10 on CIFAR-100.

References

[1] Nicholas Carlini and David Wagner. Towards Evaluating the Robustness of Neural Networks. In *S&P*, 2017. 1

[2] Tianlong Chen, Zhenyu Zhang, Sijia Liu, Shiyu Chang, and Zhangyang Wang. Robust Overfitting May Be Mitigated by Properly Learned Smoothening. In *ICLR*, 2021. 2

[3] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *CVPR*, 2018. 1

[4] Bo Huang, Mingyang Chen, Yi Wang, Junda Lu, Minhao Cheng, and Wei Wang. Boosting Accuracy and Robustness of Student Models via Adaptive Adversarial Distillation. In *CVPR*, 2023. 1, 2

[5] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely Connected Convolutional Networks. In *CVPR*, 2017. 2

[6] Harini Kannan, Alexey Kurakin, and Ian Goodfellow. Adversarial Logit Pairing. *arXiv preprint arXiv:1803.06373*, 2018. 1

[7] Qizhang Li, Yiwen Guo, Wangmeng Zuo, and Hao Chen. Collaborative Adversarial Training. *arXiv preprint arXiv:2303.14922*, 2023. 1, 3

[8] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards Deep Learning Models Resistant to Adversarial Attacks. In *ICLR*, 2018. 2

[9] Leslie Rice, Eric Wong, and Zico Kolter. Overfitting in Adversarially Robust Deep Learning. In *ICML*, 2020. 2

[10] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically Principled Trade-off between Robustness and Accuracy. In *ICML*, 2019. 1

[11] Jianing Zhu, Jiangchao Yao, Bo Han, Jingfeng Zhang, Tongliang Liu, Gang Niu, Jingren Zhou, Jianliang Xu, and Hongxia Yang. Reliable Adversarial Distillation with Unreliable Teachers. In *ICLR*, 2022. 1