Long-term Cross Adversarial Training: A Robust Meta-learning Method for Few-shot Classification Tasks

Fan Liu [1] **Shuyu Zhao** [1] **Xuelong Dai** [1] **Bin Xiao** [1]

Abstract

Meta-learning model can quickly adapt to new tasks using few-shot labeled data. However, despite achieving good generalization on few-shot classification tasks, it is still challenging to improve the adversarial robustness of the meta-learning model in few-shot learning. Although adversarial training (AT) methods such as Adversarial Query (AQ) can improve the adversarially robust performance of meta-learning models, AT is still computationally expensive training. On the other hand, meta-learning models trained with AT will drop significant accuracy on the original clean images. This paper proposed a meta-learning method on the adversarially robust neural network called Long-term Cross Adversarial Training (LCAT). LCAT will update meta-learning model parameters cross along the natural and adversarial sample distribution direction with long-term to improve both adversarial and clean few-shot classification accuracy. Due to cross-adversarial training, LCAT only needs half of the adversarial training epoch than AQ, resulting in a low adversarial training computation. Experiment results[2] show that LCAT achieves superior performance both on the clean and adversarial few-shot classification accuracy than SOTA adversarial training methods for meta-learning models.

1. Introduction

Meta-learning only needs few-shot data that can quickly adapt to even unseen tasks and has been widely applied to many fields (Haoxiang Wang, 2021). However, recent research shows that meta-learning based-models are also vulnerable to adversarial examples (Goldblum et al., 2020; Croce & Hein, 2019). Adversarial training trained on the large-scale dataset can improve the adversarial robustness of deep neural networks (Rice et al., 2020). However, how to train on few-shot tasks to robustly defend adversarial examples, fast adapt to other tasks is still challenging.

Recent robust meta-learning work called Adversarial Query (AQ) (Goldblum et al., 2020) demonstrates that adversarial training only during the query step can compromise the superior adversarially robust performance compared to traditional adversarial training (Madry et al., 2017b). However, AQ is still computationally expensive training throughout the whole query step. In addition, AQ will drop significant accuracy on the original clean image distribution, which means meta-learning models cannot adapt well to other unseen tasks.

This paper proposes a new robust meta-learning method called Long-term Cross Adversarial Training (LCAT). In detail, LCAT will make the meta-learning model's parameter θ cross along the natural and adversarial sample distribution direction. LCAT is a model-agonist meta-learning method, can adversarially adapt to few-shot classification tasks, i.e., perform well on the original clean sample distribution and robust to adversarial examples. Our solved problem can be formulated as Eq. (1), where $T(x, y)$ is training data distribution. θ is meta-learning model f_θ's parameters. A is the fine-tuning algorithm during the inner loop for meta-learning. ϵ is the maximum bound for adversarial perturbation. $\mathbb{L}$ is the loss function.

$$\min_{\theta} \mathbb{E}_{T(x,y)} \left[\max_{\|\epsilon\|_p \leq \delta} \mathbb{L}(f_{A(\theta,T)}(x+\epsilon), y) \right] \tag{1}$$

LCAT only needs half of the adversarial training epoch via cross adversarial training and significantly improves the robust performance compared to the the-state-of-art adversarial training AQ for meta-learning models. The proposed method LCAT is illustrated in Fig. 1. Our contribution can be summarized as follows:

- We propose an adversarially robust meta-learning

[1]Department of Computing, The Hong Kong Polytechnic University, Hong Kong. Correspondence to: Fan Liu <cs-fan.liu@connect.polyu.hk>.

Accepted by the ICML 2021 workshop on A Blessing in Disguise: The Prospects and Perils of Adversarial Machine Learning. Copyright 2021 by the author(s).

[2]The code to reproduce all the experimental results is available in https://github.com/Gnomeek/Long-term-Cross-Adversarial-Training.

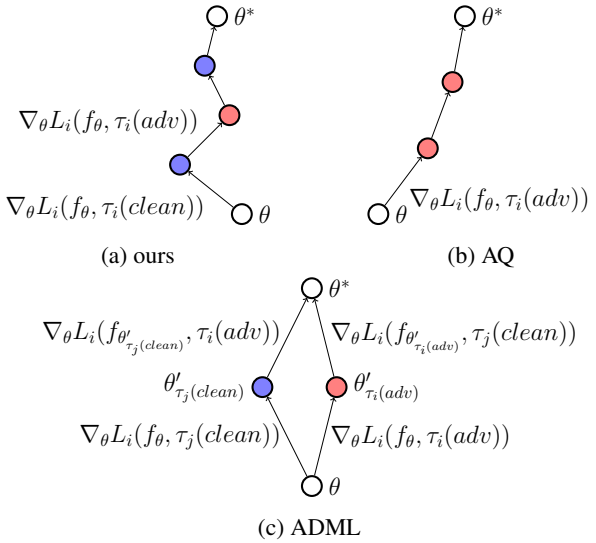

Figure 1. An illustration of our proposed method. LCAT will make the meta-learning model's parameter θ cross along the natural and adversarial sample distribution direction. A precious state-of-the-art adversarial training method AQ only makes models' parameters updated on the adversarial sample space. Another adversarial training method named ADML (Yin et al., 2018) is only designed for MAML (Finn et al., 2017) and cannot well against stronger attackers such as PGD (Madry et al., 2017a). Compared to AQ, our LCAT only needs half of the adversarial training epoch has higher adversarial and clean few-shot classification accuracy.

method called LCAT. LCAT will update meta-learning model parameters cross along the natural and adversarial sample distribution direction with long-term to improve both adversarial and clean few-shot classification accuracy.

- LCAT is a model-agonist meta-learning method, which can improve the adversarial robustness of meta-learning models. In addition, LCAT only needs half of the adversarial training epoch compared to AQ via cross adversarial training, resulting in a low adversarial training computation.
- Experimental results show that LCAT improves both clean and adversarial accuracy compared to SOTA method AQ. i.e. improve 9.7% clean and 2.88% adversarial few-shot classification accuracy compared to previous best results in MetaOptNet (Lee et al., 2019) corresponding to 5-way 5-shot on MiniImageNet dataset.

2. Related Work

There're several methods to train a robust Deep Learning model in normal scenarios rather than a meta-learning model, including defensive distillation (Papernot et al., 2016), adversarial training (Madry et al., 2017a; Su et al., 2016), feature denoising (Xie et al., 2019), TRADES theoretically principled the trade-off between accuracy and robustness (Zhang et al., 2019), and adversarial example detection (Hendrycks & Gimpel, 2016; Xu et al., 2017; Gong et al., 2017; Grosse et al., 2017). There are many adversarial robust deep neural network applications, including (Pang et al., 2020; Dong et al., 2020; Feng et al., 2021; Hendrycks et al., 2020; Dai et al., 2018; Pang et al., 2021; Dong et al., 2021).

Some works published in recent years discussed the robustness of the meta-learning model from different aspects. Yin et al. (2018) introduce a robust variant of MAML (Finn et al., 2017) called ADML, which is the first attempt to achieve robustness on the meta-learning model. However, ADML needs lots of computational power while the attacker used in their experiments is relatively weak. Goldblum et al. (2020) present a method to integrate MAML with adversarial training, called adversarial querying(AQ), which is the state-of-the-art approach in this domain. We choose AQ as one of the baselines in our experiments. In (Wang et al., 2021), they build a principled robustness-regularized meta-learning framework, which can be treated as a generalized AQ model.

3. Proposed Method: LCAT

Our goal is to make meta-learning models perform well both on clean and adversarial few-shot classification tasks. We now formulated the problem we want to solve, which can be seen in Eq. (1).

Cross adversarial training The proposed method called Long-term cross adversarial training (LCAT) is illustrated in Fig. 1. The LCAT across along the direction of the clean and adversarial direction. In the clean meta-update step, the LCAT will optimize the meta-learning model f_θ in the clean batch of tasks. This step aims to give the meta-model a good initial start which quickly adapts to clean few-shot classification tasks. In the adversarial meta-update step, the LCAT will optimize f_θ in the adversarial batch of tasks, which can help the meta-learning model against adversarial attacks.

It is important to notice that LCAT is model-agnostic. Our adversarial training method can help the meta-learning model improve significance accuracy on the clean image distribution and help the meta-learning model improve adversarially robust accuracy with a low computation compared to a previous SOTA adversarial training AQ.

The method of LCAT training method is as follows.

STEP 1 Generate a batch of tasks according to the distribution of tasks $\rho(\tau)$ (Goldblum et al., 2020).

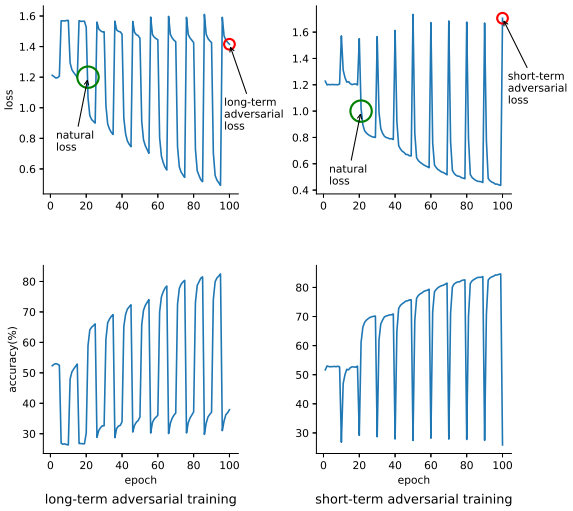

Figure 2. The comparison between long-term and short-term adversarial training.

STEP 2 Fine-tune each task $\{\tau_i\}_{0\leq i<n}$ based on the fine-tuning algorithm $A(\theta, T)$.

STEP 3 Make model parameter theta update along with the clean sample space with long-term T based on Eq. (2).

$$\theta \leftarrow \theta - \frac{\lambda}{n} \sum_{0\leq j<T} \sum_{\tau_i \sim \rho(\tau)} \nabla_\theta \mathbb{L}(f_\theta^{(j)}, \tau_i) \tag{2}$$

Where θ is the meta-leaning model parameter, and $f_\theta^{(j)}$ is the meta-learning model at jth term. λ is the learning rate.

STEP 4 Compute adversarial examples of tasks $\{\tau_i'\}$ based on attacker (Madry et al., 2017a) within a ϵ ball and then make the model update in the space of adversarial examples for long-term T based on Eq. (3).

$$\theta \leftarrow \theta - \frac{\lambda}{n} \sum_{T\leq j'<2T} \sum_{\tau_i' \sim \rho(\tau')} \nabla_\theta \mathbb{L}(f_\theta^{(j')}, \tau_i') \tag{3}$$

The detail of the LCAT training method is shown in Alg. 1.

Why need long-term cross training? Short-term cross adversarial training cannot give suitable initial parameters for meta-learning models, as shown in Fig. 2. Long-term cross adversarial training can make the model's parameters go deeper in the adversarial sample space, resulting in good adversarial training. The model can not be well trained in the adversarial sample space corresponding to short-term adversarial training. In Tab. 2, we test the short-term adversarial training. In detail, we set ten epochs as a loop, the first nine epochs are natural training, and the last epoch is adversarial training. Fig. 2 and Tab. 2 show the training results. The test results show that the model trained with short-term has low adversarial accuracy compared to long-term, cannot sufficiently defend adversarial samples. In addition, to avoid long-term cross-training appearing in the clean and adversarial sample space to go deeper and lead to poor performance of the model in few-shot classification, we also added the denoise module (Xie et al., 2019) for the meta-learning model. The Sec. 5 shows the effectiveness of the denoise module.

Algorithm 1 LCAT training method

Input:
Meta-learning model, f_θ, learning rate, λ, fine-tune algorithm A, tasks distribution $\rho(\tau)$, long-term T;
Initialize all parameters;
while not done **do**
  Sample batch of tasks, $\{\tau_i\}_{0\leq i<n}$;
  for $j = 0; j < T; j++$ **do**
    for $i = 0; i < n; i++$ **do**
      $\theta = A(\theta, \tau_i)$, fine tine on task τ_i;
      Compute updated gradient $\nabla_\theta \mathbb{L}(f_\theta^{(j)}, \tau_i)$;
    end for
  end for
  $\theta \leftarrow \theta - \frac{\lambda}{n} \sum_{0\leq j<T} \sum_{\tau_i \sim \rho(\tau)} \nabla_\theta \mathbb{L}(f_\theta^{(j)}, \tau_i)$;
  for $j' = T; j' < 2T; j'++$ **do**
    for $i = 0; i < n; i++$ **do**
      $\theta = A(\theta, \tau_i)$, fine tine on task τ_i;
      Generate adversarial batch of tasks, $\{\tau_i'\}_{0\leq i<n}$ $\nabla_\theta \mathbb{L}(f_\theta, \tau_i')$;
      Compute updated gradient $\nabla_\theta \mathbb{L}(f_\theta^{(j')}, \tau_i')$;
    end for
  end for
  $\theta \leftarrow \theta - \frac{\lambda}{n} \sum_{T\leq j'<2T} \sum_{\tau_i' \sim \rho(\tau')} \nabla_\theta \mathbb{L}(f_\theta^{(j')}, \tau_i')$;
end while

4. Experiments

4.1. Setup

Baselines To test the effectiveness of LCAT, we compared our adversarial training with 1) Adversarial Training (AT) (Madry et al., 2017b), a directly robust adversarial training method applied in the meta-learning method. 2) The Adversarial Query (AQ) (Goldblum et al., 2020) is a SOTA adversarial training method that only needs adversarial training in the query step. 3) Adversarial Query plus TRADES (Zhang et al., 2019) loss function can be a trade between natural accuracy and robust accuracy. We call this method AQ+TRADES which is suitable as a SOTA method compared to LCAT.

The learning rate is set to 0.1. The epoch and the batch size are set to 50 and 8. We use Adam (Kingma & Ba, 2014) to optimize the meta-learning models, which has been proved to obtain better performance than SGD (see Appendix A.3).

Table 1. Experiments results of 50 epoch on MiniImageNet dataset with three models, Acc_{nat} and Acc_{adv} stand for natural and robust accuracy of the model, respectively. Robust accuracy is computed for a 20-step PGD attack on the model. Top accuracy scores and the shortest training time are shown in bold font.

Dataset	Methods	ProtoNet			R2D2			MetaOptNet		
		Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time
MiniImageNet	AT (5way-1shot)	30.19 % (0.43 %)	19.15 % (0.36 %)	17.7h	26.84 % (0.40 %)	18.86 % (0.36 %)	19.7h	33.27 % (0.50 %)	13.25 % (0.36 %)	24.8h
	AQ (5way-1shot)	29.59 % (0.43 %)	19.78 % (0.37 %)	8.4 h	**27.05 % (0.39 %)**	19.06 % (0.35 %)	10.1h	32.43 % (0.49 %)	13.11 % (0.35 %)	**9.4h**
	AQ+TRADES (5way-1shot)	29.61 % (0.43 %)	20.48 % (0.37 %)	10.2h	26.85 % (0.39 %)	**19.34 % (0.35 %)**	12.8h	29.23 % (0.44 %)	16.75 % (0.36 %)	12.1h
	LCAT (ours, 5way-1shot)	**32.55 % (0.49 %)**	19.72 % (0.41 %)	**6.6h**	26.74 % (0.40 %)	18.73 % (0.35 %)	**8.4h**	**34.88 % (0.52 %)**	14.25 % (0.37 %)	9.6h
	LCAT+TRADES (ours, 5way-1shot)	30.71 % (0.46 %)	**20.56 % (0.39 %)**	7.7h	26.25 % (0.39 %)	19.29 % (0.33 %)	10.3h	30.15 % (0.46 %)	**18.35 % (0.39 %)**	10.3h
	AT (5way-5shot)	41.09 % (0.47 %)	25.85 % (0.41 %)	17.7h	34.95 % (0.42 %)	23.48 % (0.39 %)	19.7h	45.82 % (0.50 %)	18.86 % (0.44 %)	24.8h
	AQ (5way-5shot)	39.89 % (0.46 %)	26.01 % (0.41 %)	8.4h	35.06 % (0.42 %)	23.89 % (0.39 %)	10.1h	45.44 % (0.51 %)	19.17 % (0.44 %)	**9.4h**
	AQ+TRADES (5way-5shot)	39.05 % (0.46 %)	26.71 % (0.42 %)	10.2h	33.23 % (0.40 %)	23.77 % (0.38 %)	12.8h	38.13 % (0.48 %)	21.87 % (0.41 %)	12.1h
	LCAT (ours, 5way-5shot)	**44.81 % (0.52 %)**	27.89 % (0.47 %)	**6.6h**	**39.18 % (0.49 %)**	24.84 % (0.45 %)	**8.4h**	**47.83 % (0.53 %)**	21.34 % (0.45 %)	9.6h
	LCAT+TRADES (ours, 5way-5shot)	42.14 % (0.49 %)	**28.27 % (0.45 %)**	7.7h	37.46 % (0.47 %)	**25.31 % (0.42 %)**	10.3h	40.21 % (0.49 %)	**24.75 % (0.44 %)**	10.3h

We follow the experimental setting of AQ in (Goldblum et al., 2020), training the state-of-the-art meta-learning models including PROTONET (Snell et al., 2017a), R2D2 (Bertinetto et al., 2018a) , and MetaOptNet (ResNet-12 as backbone (He et al., 2016)). We conduct all the experiments on the Pytorch with Ubuntu 20.04 and GPU RTX3090.

4.2. Experimental results

The experimental results can be seen in Tab. 1. Due to the space limit, training details and other results can be seen in Appendix A.1. To sum up, our LCAT achieves superior performance both on the clean and adversarial few-shot classification accuracy than SOTA adversarial training methods for meta-learning models. For example, we improve 9.7% clean few-shot classification accuracy compared to previous best results in MetaOptNet and improve 2.88% adversarial few-shot classification accuracy compared to previous best results in MetaOptNet corresponding to 5-way 5-shot on MiniImageNet dataset.

5. Ablation Study

We also conducted the following experiments: 1) short-term cross adversarial training (SCAT): we set ten epochs as a loop, the first nine epochs are norm training, and the last epoch is adversarial training (T=1). 2) Long-term cross adversarial training (LCAT): we set ten epochs as a loop, the first five epochs are norm training, and the last five epochs are adversarial training (T=5). 3) LCAT with/without denoise (w/o denoise). The results are shown in Tab. 2. which can be seen that our LCAT compare to short-term and (w/o denoise) achieve supervisor performance.

In addition, we also test the adversarial robustness of meta-learning models trained with LCAT. With the increase of attack step, PROTONET trained with LCAT still has higher adversarial few-shot classification accuracy compared to other adversarial training methods in Fig. 3 (See Appendix A.4 for R2D2 and MetaOptNet). Further investigations about the influence of TRADES are shown in Tab. 10 in Appendix A.2.

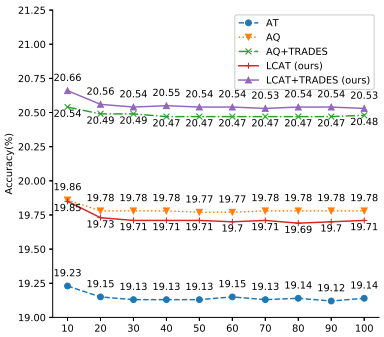

Figure 3. Acc_{adv} of five methods on MiniImageNet dataset versus PGD attack at different attack steps on ProtoNet training with 50 epoch

Table 2. 5-way, 1-shot MiniImageNet results for ProtoNet, Acc_{nat} and Acc_{adv} stand for natural and robust accuracy of the model, respectively. The top accuracy is bolded.

Methods	Acc_{nat}	Acc_{adv}	Time
SCAT (T=1,epoch = 50)	25.71 % (0.38 %)	17.97 % (0.32 %)	**3.3h**
LCAT (w/o denoise)	31.46 % (0.48 %)	19.16 % (0.40 %)	5.7 h
LCAT (T=5, epoch =50)	**32.55 % (0.49 %)**	**19.72 % (0.41 %)**	6.6h
SCAT (T=1,epoch = 100)	24.29 % (0.36 %)	18.02 % (0.30 %)	**6.6h**
LCAT (w/o denoise)	34.47 % (0.51 %)	19.37 % (0.41 %)	11.3h
LCAT (T=5, epoch =100)	**35.29 % (0.51 %)**	**19.65 % (0.41 %)**	13.4h

6. Conclusion

In this paper, we propose an adversarially robust meta-learning method called LCAT. LCAT is a model-agonist meta-learning method, which can improve the adversarial robustness of meta-learning models. LCAT will update meta-learning model parameters cross along the natural and adversarial sample distribution direction with long-term to improve both adversarial and clean few-shot classification accuracy. In addition, LCAT only needs half of the adversarial training epoch compared to AQ via cross adversarial training, resulting in a low adversarial training computation. In the future, we will establish a more in-depth understanding of the theoretical analysis of LCAT.

References

Bertinetto, L., Henriques, J. F., Torr, P. H., and Vedaldi, A. Meta-learning with differentiable closed-form solvers. *arXiv preprint arXiv:1805.08136*, 2018a.

Bertinetto, L., Henriques, J. F., Torr, P. H., and Vedaldi, A. Meta-learning with differentiable closed-form solvers. *arXiv preprint arXiv:1805.08136*, 2018b.

Croce, F. and Hein, M. Sparse and imperceivable adversarial attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4724–4732, 2019.

Dai, H., Li, H., Tian, T., Huang, X., Wang, L., Zhu, J., and Song, L. Adversarial attack on graph structured data. In *International conference on machine learning*, pp. 1115–1124. PMLR, 2018.

Dong, Y., Deng, Z., Pang, T., Su, H., and Zhu, J. Adversarial distributional training for robust deep learning. *arXiv preprint arXiv:2002.05999*, 2020.

Dong, Y., Xu, K., Yang, X., Pang, T., Deng, Z., Su, H., and Zhu, J. Exploring memorization in adversarial training. *arXiv preprint arXiv:2106.01606*, 2021.

Feng, S., Yan, X., Sun, H., Feng, Y., and Liu, H. X. Intelligent driving intelligence test for autonomous vehicles with naturalistic and adversarial environment. *Nature Communications*, 12:748, 2021.

Finn, C., Abbeel, P., and Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pp. 1126–1135. PMLR, 2017.

Goldblum, M., Fowl, L., and Goldstein, T. Adversarially robust few-shot learning: A meta-learning approach. *Advances in Neural Information Processing Systems*, 33, 2020.

Gong, Z., Wang, W., and Ku, W.-S. Adversarial and clean data are not twins. *arXiv preprint arXiv:1704.04960*, 2017.

Grosse, K., Manoharan, P., Papernot, N., Backes, M., and McDaniel, P. On the (statistical) detection of adversarial examples. *arXiv preprint arXiv:1702.06280*, 2017.

Haoxiang Wang, Han Zhao, B. L. Bridging multi-task learning and meta-learning: Towards efficient training and effective adaptation. In *International Conference on Machine Learning (ICML)*, 2021.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hendrycks, D. and Gimpel, K. Early methods for detecting adversarial images. *arXiv preprint arXiv:1608.00530*, 2016.

Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M., et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. *arXiv preprint arXiv:2006.16241*, 2020.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Lee, K., Maji, S., Ravichandran, A., and Soatto, S. Meta-learning with differentiable convex optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10657–10665, 2019.

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

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

Oreshkin, B. N., Rodriguez, P., and Lacoste, A. Tadam: Task dependent adaptive metric for improved few-shot learning. *arXiv preprint arXiv:1805.10123*, 2018.

Pang, T., Yang, X., Dong, Y., Su, H., and Zhu, J. Bag of tricks for adversarial training. *arXiv preprint arXiv:2010.00467*, 2020.

Pang, T., Zhang, H., He, D., Dong, Y., Su, H., Chen, W., Zhu, J., and Liu, T.-Y. Adversarial training with rectified rejection. *arXiv preprint arXiv:2105.14785*, 2021.

Papernot, N., McDaniel, P., Wu, X., Jha, S., and Swami, A. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE symposium on security and privacy (SP)*, pp. 582–597. IEEE, 2016.

Ren, M., Triantafillou, E., Ravi, S., Snell, J., Swersky, K., Tenenbaum, J. B., Larochelle, H., and Zemel, R. S. Meta-learning for semi-supervised few-shot classification. *arXiv preprint arXiv:1803.00676*, 2018.

Rice, L., Wong, E., and Kolter, Z. Overfitting in adversarially robust deep learning. In *International Conference on Machine Learning*, pp. 8093–8104. PMLR, 2020.

Snell, J., Swersky, K., and Zemel, R. Prototypical networks for few-shot learning. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information*

Processing Systems, volume 30. Curran Associates, Inc., 2017a.

Snell, J., Swersky, K., and Zemel, R. S. Prototypical networks for few-shot learning. *arXiv preprint arXiv:1703.05175*, 2017b.

Su, H., Zhu, J., Yin, Z., Dong, Y., and Zhang, B. Efficient and robust semi-supervised learning over a sparse-regularized graph. In *European Conference on Computer Vision*, pp. 583–598. Springer, 2016.

Vinyals, O., Blundell, C., Lillicrap, T., Kavukcuoglu, K., and Wierstra, D. Matching networks for one shot learning. *arXiv preprint arXiv:1606.04080*, 2016.

Wang, R., Xu, K., Liu, S., Chen, P.-Y., Weng, T.-W., Gan, C., and Wang, M. On fast adversarial robustness adaptation in model-agnostic meta-learning. *arXiv preprint arXiv:2102.10454*, 2021.

Xie, C., Wu, Y., Maaten, L. v. d., Yuille, A. L., and He, K. Feature denoising for improving adversarial robustness. In *Proceedings of the IEEE CVF Conference on Computer Vision and Pattern Recognition*, pp. 501–509, 2019.

Xu, W., Evans, D., and Qi, Y. Feature squeezing: Detecting adversarial examples in deep neural networks. *arXiv preprint arXiv:1704.01155*, 2017.

Yin, C., Tang, J., Xu, Z., and Wang, Y. Adversarial meta-learning. *arXiv preprint arXiv:1806.03316*, 2018.

Zhang, H., Yu, Y., Jiao, J., Xing, E., El Ghaoui, L., and Jordan, M. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning*, pp. 7472–7482. PMLR, 2019.

A. Appendix

The code to reproduce all the experimental results is available in https://github.com/Gnomeek/Long-term-Cross-Adversarial-Training.

A.1. Experiments result with models trained in 50/100 epoch on MiniImageNet, TieredImageNet, CIFAR-FS, and FC100

Training details. We adopt models including PROTONET (Snell et al., 2017b), R2D2 (Bertinetto et al., 2018b), and MetaOptNet (Lee et al., 2019) and dataset including MiniImageNet (Vinyals et al., 2016), TieredImageNet (Ren et al., 2018), CIFAR-FS (Bertinetto et al., 2018b), and FC100 (Oreshkin et al., 2018). We follow the experimental setting (Goldblum et al., 2020) that the maximum adversarial perturbation magnitude ϵ is 8/255. In training, the attacker is 7-step PGD with a 2/255 step size. We use Adam (Kingma & Ba, 2014) to optimize the meta-learning models with learning 0.1 and 50 epochs. In testing, we use a 20-step PGD attacker with a 2/255 step size.

Table 3. Experiments results of 100 epoch on MiniImageNet dataset with three models, Acc_{nat} and Acc_{adv} stand for natural and robust accuracy of the model, respectively. Robust accuracy is computed for a 20-step PGD attack on the model. Top accuracy scores and the shortest training time are shown in bold font.

Dataset	Methods	ProtoNet			R2D2			MetaOptNet		
		Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time
MiniImageNet	AT (5way-1shot)	34.22 % (0.50 %)	20.44 % (0.41 %)	36.3h	29.03 % (0.43 %)	19.05 % (0.36 %)	37.8h	36.92 % (0.55 %)	11.59 % (0.35 %)	46.6h
	AQ (5way-1shot)	33.61 % (0.48 %)	20.44 % (0.39 %)	16.9h	**29.35 % (0.45 %)**	18.73 % (0.38 %)	20.6h	32.09 % (0.46 %)	11.89 % (0.32 %)	**18.3h**
	AQ+TRADES (5way-1shot)	30.97 % (0.46 %)	**21.04 % (0.39 %)**	17.6h	28.17 % (0.41 %)	**19.36 % (0.37 %)**	27.7h	32.73 % (0.48 %)	15.97 % (0.36 %)	26.7h
	LCAT (ours, 5way-1shot)	**35.29 % (0.51 %)**	19.65 % (0.41 %)	**13.4h**	26.47 % (0.38 %)	18.67 % (0.33 %)	**15.9h**	**39.71 % (0.54 %)**	12.51% (0.34 %)	19.2h
	LCAT+TRADES (ours, 5way-1shot)	29.58 % (0.43 %)	18.79 % (0.37 %)	17.2h	25.88 % (0.39%)	18.39 % (0.35 %)	23.1h	31.24 % (0.46 %)	**18.51 % (0.39 %)**	23.6h
	AT (5way-5shot)	48.76 % (0.54 %)	**29.36 % (0.49 %)**	36.3h	41.31 % (0.50 %)	**25.58 % (0.45 %)**	37.8h	51.07 % (0.51 %)	17.67 % (0.43 %)	46.6h
	AQ (5way-5shot)	47.73 % (0.53 %)	29.13 % (0.48 %)	16.9h	**42.13 % (0.51 %)**	25.32 % (0.46 %)	20.6h	45.16 % (0.51 %)	16.78 % (0.40 %)	**18.3h**
	AQ+TRADES (5way-5shot)	42.30 % (0.49 %)	28.61 % (0.44 %)	17.6h	37.21 % (0.45 %)	25.20 % (0.42 %)	27.7h	44.11 % (0.50 %)	22.79 % (0.43 %)	26.7h
	LCAT (ours, 5way-5shot)	**48.93 % (0.53 %)**	29.08 % (0.49 %)	**13.4h**	39.77 % (0.49 %)	25.06 % (0.43 %)	**15.9h**	**54.91 % (0.50 %)**	18.92 % (0.41 %)	19.2h
	LCAT+TRADES (ours, 5way-5shot)	39.53 % (0.48 %)	25.38 % (0.43 %)	17.2h	32.28 % (0.42 %)	22.35 % (0.38 %)	23.1h	42.49 % (0.51 %)	**26.18 % (0.46 %)**	23.6h

Table 4. Experiments results of 50 epoch on TieredImageNet dataset with three models, Acc_{nat} and Acc_{adv} stand for natural and robust accuracy of the model, respectively. Robust accuracy is computed for a 20-step PGD attack on the model. Top accuracy scores and the shortest training time are shown in bold font.

Dataset	Methods	ProtoNet			R2D2			MetaOptNet		
		Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time
TieredImageNet	AT (5way-1shot)	30.88 % (0.52 %)	**23.93 % (0.51 %)**	20.9h	29.31 % (0.50 %)	22.95 % (0.47 %)	15.2h	**47.39 % (0.66 %)**	20.01 % (0.55 %)	24.7h
	AQ (5way-1shot)	36.71 % (0.59 %)	23.19 % (0.50 %)	8.7h	**35.11 % (0.55 %)**	23.85 % (0.51 %)	15.2h	44.16 % (0.63 %)	20.73 % (0.54 %)	10.8h
	AQ+TRADES (5way-1shot)	34.41 % (0.55 %)	23.79 % (0.49 %)	11.0h	33.49 % (0.55 %)	**24.49 % (0.49 %)**	17.5h	40.26 % (0.60 %)	**24.22 % (0.56 %)**	13.7h
	LCAT (ours, 5way-1shot)	**36.87 % (0.58 %)**	22.02 % (0.49 %)	**6.6h**	33.41 % (0.55 %)	23.13 % (0.49 %)	**10.0h**	44.39 % (0.64 %)	20.48 % (0.53 %)	10.3h
	LCAT+TRADES (ours, 5way-1shot)	34.88 % (0.56 %)	23.71 % (0.50 %)	10.0h	30.71 % (0.50 %)	21.99 % (0.45 %)	10.4h	36.91 % (0.60 %)	23.38 % (0.54 %)	**10.1h**
	AT (5way-5shot)	39.83 % (0.53 %)	30.69 % (0.52 %)	20.9h	38.21 % (0.53 %)	29.66 % (0.52 %)	15.2h	**62.51 % (0.55 %)**	29.68 % (0.58 %)	24.7h
	AQ (5way-5shot)	52.62 % (0.59 %)	34.31 % (0.58 %)	8.7h	**48.52 % (0.56 %)**	33.61 % (0.56 %)	15.2h	58.80 % (0.56 %)	30.07 % (0.56 %)	10.8h
	AQ+TRADES (5way-5shot)	48.01 % (0.58 %)	34.19 % (0.57 %)	11.0h	46.63 % (0.55 %)	**34.06 % (0.56 %)**	17.5h	53.54 % (0.54 %)	**34.06 % (0.56 %)**	13.7h
	LCAT (ours, 5way-5shot)	**52.90 % (0.59 %)**	33.39 % (0.58 %)	**6.6h**	47.29 % (0.55 %)	32.52 % (0.56 %)	**10.0h**	59.49 % (0.55 %)	30.28 % (0.57 %)	10.3h
	LCAT+TRADES (ours, 5way-5shot)	49.07 % (0.58 %)	**34.51 % (0.58 %)**	10.0h	42.02 % (0.54 %)	30.12 % (0.54 %)	10.4h	48.47 % (0.54 %)	32.23 % (0.56 %)	**10.1h**

Table 5. Experiments results of 100 epoch on TieredImageNet dataset with three models, Acc_{nat} and Acc_{adv} stand for natural and robust accuracy of the model, respectively. Robust accuracy is computed for a 20-step PGD attack on the model. Top accuracy scores and the shortest training time are shown in bold font.

Dataset	Methods	ProtoNet			R2D2			MetaOptNet		
		Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time
TieredImageNet	AT (5way-1shot)	**38.05 % (0.61 %)**	**25.81 % (0.57 %)**	38.4h	33.19 % (0.54 %)	23.33 % (0.50 %)	32.0h	**47.34 % (0.65 %)**	19.07 % (0.53 %)	48.4h
	AQ (5way-1shot)	37.26 % (0.59 %)	23.25 % (0.51 %)	17.7h	**34.37 % (0.54 %)**	23.73 % (0.50 %)	30.8h	44.98 % (0.64 %)	18.94 % (0.52 %)	22.9h
	AQ+TRADES (5way-1shot)	34.90 % (0.55 %)	23.71 % (0.50 %)	21.9h	33.50 % (0.55 %)	**24.51 % (0.50 %)**	34.0h	40.65 % (0.60 %)	**24.77 % (0.56 %)**	26.7h
	LCAT (ours, 5way-1shot)	37.20 % (0.58 %)	21.87 % (0.50 %)	**13.2h**	32.11 % (0.53 %)	22.82 % (0.46 %)	**18.1h**	45.74 % (0.65 %)	18.06 % (0.50 %)	**17.5h**
	LCAT+TRADES (ours, 5way-1shot)	35.38 % (0.56 %)	23.73 % (0.50 %)	20.5h	32.82 % (0.55 %)	22.61 % (0.51 %)	18.9h	37.31 % (0.59 %)	23.09 % (0.53 %)	18.3h
	AT (5way-5shot)	51.53 % (0.56 %)	**36.05 % (0.57 %)**	38.4h	46.13 % (0.56 %)	32.36 % (0.55 %)	32.0h	**62.55 % (0.55 %)**	28.58 % (0.57 %)	48.4h
	AQ (5way-5shot)	53.82 % (0.60 %)	34.76 % (0.59 %)	17.7h	**47.75 % (0.56 %)**	33.29 % (0.56 %)	30.8h	59.69 % (0.57 %)	27.97 % (0.54 %)	22.9h
	AQ+TRADES (5way-5shot)	48.91 % (0.58 %)	34.38 % (0.58 %)	21.9h	46.55 % (0.56 %)	**34.04 % (0.55 %)**	34.0h	53.57 % (0.56 %)	**34.71 % (0.57 %)**	26.7h
	LCAT (ours, 5way-5shot)	**54.10 % (0.59 %)**	33.80 % (0.58 %)	**13.2h**	47.49 % (0.56 %)	32.73 % (0.57 %)	**18.1h**	61.34 % (0.55 %)	27.25 % (0.55 %)	**17.5h**
	LCAT+TRADES (ours, 5way-5shot)	50.66 % (0.58 %)	35.22 % (0.58 %)	20.5h	42.34 % (0.55 %)	30.40 % (0.55 %)	18.9h	49.70 % (0.56 %)	32.89 % (0.56 %)	18.3h

Table 6. Experiments results of 50 epoch on CIFAR-FS dataset with three models, Acc_{nat} and Acc_{adv} stand for natural and robust accuracy of the model, respectively. Robust accuracy is computed for a 20-step PGD attack on the model. Top accuracy scores and the shortest training time are shown in bold font.

Dataset	Methods	ProtoNet			R2D2			MetaOptNet		
		Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time
CIFAR-FS	AT (5way-1shot)	38.11 % (0.62 %)	**28.42 % (0.59 %)**	12.2h	37.11 % (0.62 %)	27.63 % (0.57 %)	12.7h	39.49 % (0.64 %)	25.61 % (0.61 %)	17.4h
	AQ (5way-1shot)	**43.22 % (0.66 %)**	25.94 % (0.62 %)	2.5h	**41.03 % (0.62 %)**	28.93 % (0.60 %)	5.0h	47.70 % (0.69 %)	25.44 % (0.65 %)	6.4h
	AQ+TRADES (5way-1shot)	39.58 % (0.63 %)	27.39 % (0.59 %)	3.1h	38.89 % (0.61 %)	**29.66 % (0.60 %)**	6.7h	42.72 % (0.69 %)	30.09 % (0.66 %)	6.4h
	LCAT (ours, 5way-1shot)	42.26 % (0.65 %)	25.67 % (0.61 %)	**2.5h**	40.43 % (0.64 %)	28.26 % (0.60 %)	**3.8h**	**48.41 % (0.68 %)**	25.19 % (0.64 %)	**3.8h**
	LCAT+TRADES (ours, 5way-1shot)	38.52 % (0.63 %)	26.39 % (0.59 %)	3.8h	34.68 % (0.58 %)	26.28 % (0.54 %)	6.3h	40.22 % (0.66 %)	**30.42 % (0.64 %)**	7.5h
	AT (5way-5shot)	52.02 % (0.59 %)	38.11 % (0.62 %)	12.2h	49.56 % (0.59 %)	37.51 % (0.59 %)	12.7h	52.32 % (0.58 %)	34.70 % (0.60 %)	17.4h
	AQ (5way-5shot)	**62.59 % (0.62 %)**	39.68 % (0.67 %)	2.5h	**54.86 % (0.59 %)**	**39.87 % (0.59 %)**	5.0h	63.82 % (0.58 %)	36.73 % (0.67 %)	6.4h
	AQ+TRADES (5way-5shot)	56.30 % (0.62 %)	**40.76 % (0.64 %)**	3.1h	50.95 % (0.58 %)	39.75 % (0.59 %)	6.7h	55.66 % (0.59 %)	40.39 % (0.63 %)	6.4h
	LCAT (ours, 5way-5shot)	60.29 % (0.61 %)	38.50 % (0.65 %)	**2.5h**	53.34 % (0.61 %)	38.88 % (0.61 %)	**3.8h**	**64.76 % (0.59 %)**	36.11 % (0.66 %)	**3.8h**
	LCAT+TRADES (ours, 5way-5shot)	52.21 % (0.62 %)	36.50 % (0.62 %)	3.8h	46.93 %(0.58 %)	36.77 % (0.57 %)	6.3h	52.87 % (0.64 %)	**41.30 % (0.64 %)**	7.5h

Table 7. Experiments results of 100 epoch on CIFAR-FS dataset with three models, Acc_{nat} and Acc_{adv} stand for natural and robust accuracy of the model, respectively. Robust accuracy is computed for a 20-step PGD attack on the model. Top accuracy scores and the shortest training time are shown in bold font.

Dataset	Methods	ProtoNet			R2D2			MetaOptNet		
		Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time
CIFAR-FS	AT (5way-1shot)	42.67 % (0.65 %)	**29.78 % (0.62 %)**	20.4h	40.24 % (0.64 %)	29.11 % (0.61 %)	22.9h	42.51 % (0.67 %)	23.55 % (0.62 %)	27.8h
	AQ (5way-1shot)	**43.75 % (0.65 %)**	25.77 % (0.61 %)	5.4h	41.44 % (0.63 %)	28.78 % (0.59 %)	10.6h	48.76 % (0.67 %)	23.18 % (0.61 %)	14.1h
	AQ+TRADES (5way-1shot)	39.87 % (0.64 %)	27.27 % (0.59 %)	6.4h	39.41 % (0.62 %)	**30.26 % (0.60 %)**	12.4h	42.24 % (0.65 %)	**29.73 % (0.64 %)**	13.9h
	LCAT (ours, 5way-1shot)	42.51 % (0.65 %)	24.95 % (0.61 %)	**4.9h**	**41.49 % (0.62 %)**	28.27 % (0.59 %)	**7.3h**	**50.29 % (0.70 %)**	24.13 % (0.63 %)	**9.4h**
	LCAT+TRADES (ours, 5way-1shot)	37.94 % (0.63 %)	25.75 % (0.58 %)	7.5h	35.88 % (0.61 %)	27.85 % (0.56 %)	13.9h	41.15 % (0.65 %)	29.51 % (0.62 %)	13.9h
	AT (5way-5shot)	58.59 % (0.59 %)	**41.37 % (0.63 %)**	20.4h	53.06 % (0.60 %)	39.33 % (0.60 %)	22.9h	56.81 % (0.59 %)	33.39 % (0.62 %)	27.8h
	AQ (5way-5shot)	**63.66 % (0.60 %)**	39.78 % (0.67 %)	5.4h	**55.59 % (0.59 %)**	39.83 % (0.60 %)	10.6h	65.84 % (0.57 %)	34.23 % (0.63 %)	14.1h
	AQ+TRADES (5way-5shot)	57.44 % (0.62 %)	41.25 % (0.65 %)	6.4h	51.52 % (0.58 %)	**40.27 % (0.59 %)**	12.4h	55.45 % (0.60 %)	40.12 % (0.62 %)	13.9h
	LCAT (ours, 5way-5shot)	61.69 % (0.60 %)	38.51 % (0.66 %)	**4.9h**	54.97 % (0.60 %)	39.28 % (0.61 %)	**7.3h**	**66.95 % (0.58 %)**	35.22 % (0.66 %)	**9.4h**
	LCAT+TRADES (ours, 5way-5shot)	51.14 % (0.62 %)	36.50 % (0.62 %)	7.5h	47.14 % (0.59 %)	37.37 % (0.59 %)	13.9h	55.72 % (0.61 %)	**41.33 % (0.64 %)**	13.9h

Table 8. Experiments results of 50 epoch on FC-100 dataset with three models, Acc_{nat} and Acc_{adv} stand for natural and robust accuracy of the model, respectively. Robust accuracy is computed for a 20-step PGD attack on the model. Top accuracy scores and the shortest training time are shown in bold font.

Datasets	Methods	ProtoNet			R2D2			MetaOptNet		
		Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time
FC100	AT(5way-1shot)	32.32 % (0.51 %)	20.71 % (0.46 %)	5.2h	**33.26 % (0.55 %)**	21.31 % (0.51 %)	6.3h	32.04 % (0.50 %)	16.08 % (0.46 %)	10.7h
	AQ(5way-1shot)	32.43 % (0.52 %)	20.75 % (0.47 %)	4.8h	33.21 % (0.54 %)	21.54 % (0.50 %)	5.3h	32.47 % (0.51 %)	15.87 % (0.46 %)	5.7h
	AQ+TRADES(5way-1shot)	30.54 % (0.47 %)	21.83 % (0.43 %)	3.0h	31.69 % (0.53 %)	**22.65 % (0.50 %)**	2.4h	**33.28 % (0.57 %)**	**22.29 % (0.53 %)**	5.3h
	LCAT(ours, 5way-1shot)	**34.77 % (0.54 %)**	19.69 % (0.49 %)	3.4h	32.60 % (0.52 %)	20.53 % (0.48 %)	4.0h	33.02 % (0.50 %)	13.40 % (0.41 %)	**4.8h**
	LCAT+TRADES(ours, 5way-1shot)	31.86 % (0.51 %)	**22.00 % (0.48 %)**	**3.0h**	31.18 % (0.51 %)	22.42 % (0.48 %)	**2.4h**	30.94% (0.50%)	21.49% (0.46%)	6.5h
	AT(5way-5shot)	43.41 % (0.54 %)	27.59 % (0.54 %)	5.2h	42.87 % (0.55 %)	28.03 % (0.55 %)	6.3h	41.82 % (0.52 %)	21.56 % (0.51 %)	10.7h
	AQ(5way-5shot)	43.30 % (0.55 %)	27.28 % (0.53 %)	4.8h	**42.93 % (0.54 %)**	28.29 % (0.54 %)	5.3h	42.58 % (0.52 %)	21.09 % (0.50 %)	5.7h
	AQ+TRADES(5way-5shot)	40.96 % (0.53 %)	29.01 % (0.52 %)	3.0h	40.14 % (0.53 %)	**29.06 % (0.53 %)**	2.4h	40.70 % (0.54 %)	**27.90 % (0.53 %)**	5.3h
	LCAT(ours, 5way-5shot)	**45.17 % (0.55 %)**	26.95 % (0.54 %)	3.4h	41.89 % (0.54 %)	27.42 % (0.53 %)	4.0h	**43.54 % (0.51 %)**	19.10 % (0.47 %)	**4.8h**
	LCAT+TRADES(ours, 5way-5shot)	41.28 % (0.55 %)	**29.31 % (0.54 %)**	**3.0h**	39.17 % (0.54 %)	28.25 % (0.53 %)	**2.4h**	39.61 % (0.55%)	27.72 % (0.53%)	6.5h

Table 9. Experiments results of 100 epoch on FC-100 dataset with three models, Acc_{nat} and Acc_{adv} stand for natural and robust accuracy of the model, respectively. Robust accuracy is computed for a 20-step PGD attack on the model. Top accuracy scores and the shortest training time are shown in bold font.

Datasets	Methods	ProtoNet			R2D2			MetaOptNet		
		Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time	Acc_{nat}	Acc_{adv}	Time
FC100	AT(5way-1shot)	33.60 % (0.52 %)	19.15 % (0.46 %)	10.5h	33.46 % (0.55 %)	20.73 % (0.50 %)	13.9h	32.86 % (0.49 %)	12.94 % (0.39 %)	21.2h
	AQ(5way-1shot)	33.67 % (0.52 %)	19.83 % (0.47 %)	8.1h	33.30 % (0.53 %)	20.39 % (0.49 %)	10.0h	32.63 % (0.49 %)	14.20 % (0.43 %)	13.0h
	AQ+TRADES(5way-1shot)	31.72 % (0.49 %)	**22.03 % (0.45 %)**	5.5h	32.93 % (0.54 %)	**22.51 % (0.52 %)**	4.8h	32.88 % (0.57 %)	**21.10 % (0.52 %)**	11.6h
	LCAT(ours, 5way-1shot)	**35.28 % (0.55 %)**	18.42 % (0.49 %)	7.6h	**34.11 % (0.54 %)**	20.35 % (0.50 %)	7.5h	**34.09 % (0.50 %)**	10.72 % (0.39 %)	**10.6h**
	LCAT+TRADES(ours, 5way-1shot)	31.44 % (0.50 %)	21.34 % (0.46 %)	**5.4h**	31.22 % (0.51 %)	21.75 % (0.47 %)	**4.8h**	31.94% (0.53%)	20.80% (0.48%)	13.1h
	AT(5way-5shot)	46.05 % (0.55 %)	26.42 % (0.54 %)	10.5h	43.40 % (0.56 %)	27.27 % (0.55 %)	13.9h	44.15 % (0.54 %)	18.67 % (0.49 %)	21.2h
	AQ(5way-5shot)	46.17 % (0.55 %)	27.16 % (0.53 %)	8.1h	**43.75 % (0.55 %)**	27.33 % (0.55 %)	10.0h	43.52 % (0.53 %)	19.54 % (0.51 %)	13.0h
	AQ+TRADES(5way-5shot)	42.18 % (0.54 %)	**28.93 % (0.53 %)**	5.5h	41.65 % (0.53 %)	**29.06 % (0.54 %)**	4.8h	40.65 % (0.54 %)	26.54 % (0.53 %)	11.6h
	LCAT(ours, 5way-5shot)	**46.66 % (0.57 %)**	25.91 % (0.55 %)	7.6h	43.68 % (0.55 %)	27.12 % (0.54 %)	7.5h	**44.92 % (0.54 %)**	15.35 % (0.46 %)	**10.6h**
	LCAT+TRADES(ours, 5way-5shot)	41.17 % (0.53 %)	28.72 % (0.53 %)	**5.4h**	39.69 % (0.53 %)	28.35 % (0.53 %)	**4.8h**	40.98 % (0.55%)	**27.13 % (0.54%)**	13.1h

A.2. Experiments results on applying TRADES at different times

Table 10. Experiments results on MiniImageNet dataset with ProtoNet model, Acc_{nat} and Acc_{adv} stand for natural and robust accuracy of the model, respectively. TRADE_{10}^{10} and TRADE_{10}^{5} represent using TRADES during all the epoch, last 5 epoch in every 10 epoch, respectively, i.e., using TRADES all the time or using it only when attack method is applied. Robust accuracy is computed for a 20-step PGD attack on the model. Top accuracy scores and the shortest training time are shown in bold font.

METHODS	Acc_{nat}	Acc_{adv}	TIME
LCAT+TRADES_{10}^{10} (EPOCH =50)	30.23 % (0.45 %)	19.81 % (0.39 %)	8.9H
LCAT+TRADES_{10}^{5} (EPOCH = 50)	**30.71 % (0.46 %)**	**20.56 % (0.39 %)**	**7.7H**
LCAT+TRADES_{10}^{10} (EPOCH = 100)	29.19 % (0.43 %)	**19.00 % (0.37 %)**	17.5H
LCAT+TRADES_{10}^{5} (EPOCH = 100)	**29.58 % (0.43 %)**	18.79 % (0.37 %)	**17.2H**

A.3. Experiments results on LCAT applied with different optimizer

Table 11. Experiments results on MiniImageNet dataset with ProtoNet model, Acc_{nat} and Acc_{adv} stand for natural and robust accuracy of the model, respectively. Robust accuracy is computed for a 20-step PGD attack on the model. Top accuracy scores and the shortest training time are shown in bold font.

METHODS	Acc_{nat}	Acc_{adv}	TIME
LCAT (ADAM, EPOCH =50)	**32.55 % (0.49 %)**	**19.72 % (0.41 %)**	**6.6H**
LCAT (SGD, EPOCH = 50)	32.48 % (0.48 %)	18.50 % (0.39 %)	8.2H
LCAT (ADAM, EPOCH = 100)	**35.29 % (0.51 %)**	**19.65 % (0.41 %)**	**13.4H**
LCAT (SGD, EPOCH = 100)	31.41 % (0.46 %)	17.88 % (0.37 %)	16.1H

A.4. Extended experiments results on R2D2 and MetaOptNet versus PGD attack at different attack steps

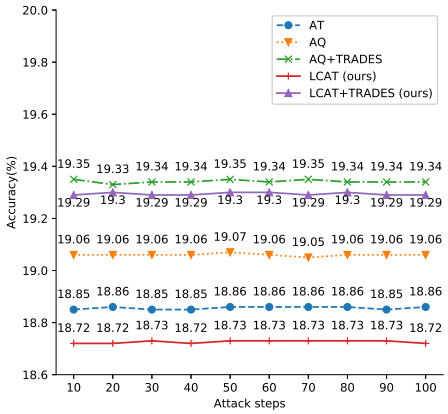

Figure 4. Acc_{adv} of five methods on MiniImageNet dataset versus PGD attack at different attack steps on R2D2 training with 50 epoch

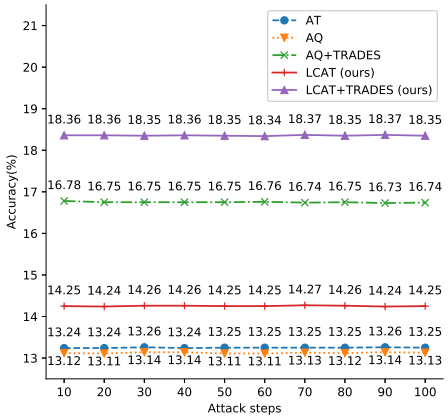

Figure 5. Acc_{adv} of five methods on MiniImageNet dataset versus PGD attack at different attack steps on MetaOptNet training with 50 epoch