Can Contrastive Learning in Embedding Space Improve Adversarial Robustness?

Qifan Zhang
Department of Statistics and Data Science
Yale University
New Haven, CT 06511
qifan.zhang@yale.edu

Abstract

Adversarial robustness is crucial for the safe and trustworthy deployment of deep learning models in practice. The main approaches against evasion attack include adversarial training [Madry et al., 2018] and TRADES [Zhang et al., 2019]. This project investigates whether a contrastive loss component in embedding space can further improve adversarial robustness of models over the two methods, inspired by the diminishing rank phenomenon in [Feng et al., 2022]. The experiment results indicate that contrastive learning in embedding space does not outperform the contrastive learning in output space i.e. the combination of adversarial training and TRADES. In addition, the results show that the improved robustness against the attack used in training may come from overfitting and thus may decrease models' performance against other attacks in inference stage.

1 Problem Definitions

The problem that this project investigates is whether a contrastive loss component in embedding space can improve adversarial robustness of deep learning models. Given the time constraint on this project, We only evaluate the loss component on image classification tasks. Robustness against evasion attack usually indicates a smoother neural network which is insensitive to imperceptible changes on inputs. One approach to achieve this goal, named TRADES [Zhang et al., 2019], imposes a contrastive loss component in output space i.e.

$$\mathbb{E}_{(\mathbf{x}_0,y_0)\sim\mathcal{D}}\left\{\mathcal{L}(f(\mathbf{x}_0),y)+\beta\max_{d(\mathbf{x},\mathbf{x}_0)\leq\epsilon}\mathcal{L}(f(\mathbf{x}),f(\mathbf{x}_0))\right\}.$$

Nevertheless, neural networks such as ResNet and Transformer tends to have diminishing rank on image datasets [Feng et al., 2022], which indicates that the deeper linear mappings are degenerated. Thus to achieve the robustness goal, clean inputs and its perturbed inputs are only constrained to be close in quotient space instead of in embedding space. Thus it is natural to ask whether imposing contrastive loss in embedding space can further improve adversarial robustness. The goal of this project is to test whether the following new loss function can improve adversarial robustness

$$\mathbb{E}_{(\mathbf{x}_0,y_0)\sim\mathcal{D}}\left\{\max_{d(\mathbf{x},\mathbf{x}_0)\leq\epsilon}\mathcal{L}(f(\mathbf{x}),y)+\beta\max_{d(\mathbf{x},\mathbf{x}_0)\leq\epsilon}d_{\text{emb}}(g(\mathbf{x}),g(\mathbf{x}_0))\right\},$$

where g is the mapping from inputs to its embedding. Further discussion on the proposed method can be found in Section 4. The base model in this project is the ResNet-18 [He et al., 2016]. The attack methods are white-box untargeted attack such as FGSM [Goodfellow et al., 2014], PGD [Madry et al., 2018] and Deepfool [Moosavi-Dezfooli et al., 2016]. The baseline methods are regular training, adversarial training [Madry et al., 2018] and TRADES [Zhang et al., 2019].

Preprint. Under review.

2 Relevance to Trustworthy AI Research

This project is related to adversarial robustness in Trustworthy AI and it aims to improve the adversarial robustness of deep learning models with a contrastive learning loss component.

3 Related Work

Adversarial attack is a potential threat to safe and trustworthy deployment of neural networks in real-life scenarios such as autonomous driving [Deng et al., 2020], cybersecurity [Vrejoiu, 2019] and healthcare [Alipanahi et al., 2015]. One major attack occuring in deployment stage is *Evasion Attack*. Existing evasion attack methods mainly utilize the sensitivity to input perturbation of neural networks and can be written as a constrained optimization problem

$$\max_{d(\mathbf{x},\mathbf{x}_0)\leq\epsilon} \mathcal{L}(f(\mathbf{x}), f(\mathbf{x}_0)), \tag{1}$$

where $d(\cdot,\cdot)$ is a metric in input space, $\mathcal{L}(\cdot,\cdot)$ is a loss function and f is a neural network. Fast Gradient Sign Method (FGSM) [Goodfellow et al., 2014] is a single-step gradient method designed for ℓ_∞ attack but is shown to be a weak attack method [Tramèr et al., 2017]. Projected Gradient Descent (PGD) [Madry et al., 2018] improves over FGSM through iteration and projection and is suitable for most norm metric such as ℓ_1, ℓ_2 and ℓ_∞. Other approaches such as One Pixel Attack [Su et al., 2019], ℓ_1-APGD [Croce and Hein, 2021] and JSAS [Papernot et al., 2016] also generate attacks through the optimization problem but with a different metric or optimization algorithms. The adversarial attack can also be generated through the dual problem of (1) such as Deepfool [Moosavi-Dezfooli et al., 2016] and C&W attack [Carlini and Wagner, 2016]. These attacks are shown to be stronger and even effective against robust neural networks.

Adversarial robustness aims to defend neural networks against the above attack methods. There are mainly three different directions to improve model's adversarial robustness: *gradient masking or obfuscation*, *robust optimization* and *adversarial example detection* [Silva and Najafirad, 2020]. Gradient masking or obfuscation approach aims to introduce randomness or non-differentiable function in neural networks to avoid gradient-based attacks. One approach is Stochastic Activation Pruning (SAP) [Dhillon et al., 2018], which randomly drops some neurons in inference phase with a multinomial distribution and probability proportional to the magnitude of the activation. That is, the probability of sampling the j'th activation with value h_j is

$$p_j = \frac{|h_j|}{\sum_{i=1}^{N} |h_i|},$$

where N is the number of neurons in the considered layer. However, the performance of models on clean images also drops when using SAP and the gradient masking or obfuscation methods are shown to be vulnerable to attack with surrogate models [Athalye et al., 2018].

The proposed approach in this project can be categorized into robust optimization. The robust optimization approach aims to replace regular loss function with a robust objective. One option is

$$f \in \min_f \mathbb{E}_{(\mathbf{x}_0,y_0)\sim\mathcal{D}} \left\{ \max_{d(\mathbf{x},\mathbf{x}_0)\leq\epsilon} \mathcal{L}(f(\mathbf{x}), y_0) \right\}.$$

Adversarial Training (AT) [Madry et al., 2018] solves this optimization by approximating the inner-max optimization with an adversarial examples. But this method depends on the quality of the adversarial examples and cannot be guaranteed to provide optimal robustness. In addition, the hard label target in AT may lead to a worse performance on regular data. Tradeoff-inspired adversarial defense via surrogate-loss minimization (TRADES) [Zhang et al., 2019] solves the hard label problem by introducing a regularized surrogate loss i.e.

$$f \in \min_f \mathbb{E}_{(\mathbf{x}_0,y_0)\sim\mathcal{D}} \left\{ \mathcal{L}(f(\mathbf{x}_0), y) + \beta \max_{d(\mathbf{x},\mathbf{x}_0)\leq\epsilon} \mathcal{L}(f(\mathbf{x}), f(\mathbf{x}_0)) \right\}.$$

The regularized loss function provides a balance between the regular performance and the adversarial robustness. Also, the regularization term can be regarded as a contrastive learning loss in the output space and thus can be viewed as a special case in this project. Ensemble Adversarial Training

(EAT) [Tramèr et al., 2020] points out that the vanilla AT tends to converge a degenerate global minimum, wherein small curvature artifacts near the data points obfuscate a linear approximation of the loss. Thus it fails to defend against some strong attacks. Instead, the EAT proposed a technique that augments training data with perturbations transferred from other models. However, there are still some black-box models such as generative attack [Baluja and Fischer, 2017] that the EAT may not be able to defend against.

Unlike the robust optimization approach, adversarial example detection is an approach aiming to detect whether an input is attacked or not in inference phase. It is believed that those attacked images have some distinguishable features making them different from clean inputs [Ilyas et al., 2019]. Thus one can use train a deep learning to detect whether an input is attacked or not. Erase and Restore (E&R) [Zuo and Zeng, 2020] train a binary classifier with clean inputs and ℓ_2-attack input by erasing some pixels and restoring them in an in-painting process. Then the confidence score will be used to decide whether the input is attacked or not.

4 Proposed Approach

In this section, we first provide short answers to the required questions and then explain the proposed method.

4.1 Short Answers

What is your approach to the problem? The approach is to introduce a contrastive loss on embedding space in the regular loss function to improve neural networks' robustness against evasion attacks.

What algorithm have you implemented? We implemented the base model ResNet-18, the attack methods FGSM, PGD and Deepfool, the defense methods Adversarial Training (AT) and TRADES.

What is different? This approach uses a new loss function i.e.

$$\mathbb{E}_{(\mathbf{x}_0,y_0)\sim\mathcal{D}}\left\{\max_{d(\mathbf{x},\mathbf{x}_0)\leq\epsilon}\mathcal{L}(f(\mathbf{x}),y)+\beta\max_{d(\mathbf{x},\mathbf{x}_0)\leq\epsilon}d_{\text{emb}}(g(\mathbf{x}),g(\mathbf{x}_0))\right\},$$

where g is the mapping from inputs to its embedding. It is different from AT since it includes a contrastive loss term and is different from TRADES since it consider an AT loss and a contrastive loss in embedding space.

Training pipeline? The training pipeline is the same as AT and TRADES. We directly use gradient-based method to do empirical risk minimization. The empirical loss can be seen from the following subsection.

4.2 Contrastive Loss in Embedding Space

As discussed in Section 1, the rationale to use contrastive loss in embedding space is that the TRADES loss does not guarantee the proxmity of images within a small neighborhood in the whole forward pass and thus may still be vulnerable to perturbation in inputs. Intuitively, it is similar to a series circuit (the whole forward pass) and the breaking down of any component of it (any hidden layer) may lead to the breaking down of the whole circuit. Thus we conjecture that the proximity of clean input and its perturbed inputs in embedding space can further improve robustness.

In addition, we also include the AT loss in the objective function since the AT loss has also been shown effective in improving generalization ability and robustness.

The empirical form of the proposed loss function is

$$\frac{1}{|\mathcal{D}_{\text{train}}|}\sum_{(\mathbf{x},y)\sim\mathcal{D}_{\text{train}}}\left\{\mathcal{L}(f(\mathbf{x}^{\text{adv}}),y)+\beta d_{\text{emb}}(g(\mathbf{x}^{\text{adv}}),g(\mathbf{x}))\right\},$$

where $\mathbf{x}^{\text{adv}}$ is an adversarial attack on $\mathbf{x}$ with $d(\mathbf{x}^{\text{adv}},\mathbf{x})\leq\epsilon$. In our experiment, we use FGSM and PGD attack in training. The metric d_{emb} is selected as Euclidean metric. We also explore which hidden layers i.e. which g can bring best robustness.

5 Experiments

5.1 Datasets

In this project, we use standard image classification datasets CIFAR-10 [Krizhevsky, 2009] and MNIST [Lecun et al., 1998]. Since these datasets are standard, we do not provide repeated data analysis on them. One can find analysis on them on their homepages.

5.2 Baselines

The baselines are regular training, AT and TRADES. The introduction of these methods can be found in Section 3.

5.3 Metrics

Similar to experiments of baseline approches, the metrics used in this project are Top1, Top3 and Top5 classification accuracy.

5.4 Experiment Settings

In our experiment, we use ResNet-18 pretrained on ImageNet1K [Russakovsky et al., 2015] as our base model. The optimizer is AdamW [Loshchilov and Hutter, 2019] with learning rate 10^{-5} and decay weight 0.1. The maximal perturbation of FGSM and PGD is set as 0.31 and the learning rate of PGD is set as 0.008. The maximal iteration for PGD and Deepfool is 20 and the overshoot of Deepfool is set as 0.02. The trade-off parameter in TRADES is set as 1.0. Each experiment is run 3 times with different random seeds to estimate the mean and standard deviation of the classification accuracy.

5.5 Results

In this part, we only list the Top1 accuracy results on CIFAR-10 and MNIST. The Top3 and Top5 results can be found in Appendix A. In this experiment, the embedding space we used is the fifth block in ResNet-18 (see Table 3) and the trade-off parameter is set to be 6.0. The results are listed in Table 1 and 2. The results on CIFAR-10 indicates that the contrastive learning with PGD adversarial examples can achieve around 2% improvement over adversarial training. In addition, the contrastive learning with FGSM samples can achieve 8% improvement on robustness against PGD samples. However, the model's performance against clean and Deepfool images drops, which may be due to an overfit to FGSM-like examples. The results on MNIST shows consistent improvement of our methods over AT and TRADES. Compared to AT, our method can achieve better performance on Deepfool attack and compared to TRADES, our method can achieve better performance on FGSM and PGD attack.

5.6 Ablation Study

In this section, we provide an ablation study on the embedding space (hidden layer) used in contrastive learning and the trade-off parameter β.

5.6.1 Embedding Space

The ResNet-18 contains multiple hidden layers and can all be used as embeddings in our contrastive learning framework. In Table 3, we divide the ResNet-18 architecture (the name of components correspond to the pytorch implementation of ResNet-18) into 6 parts and refer them to 5 embedding spaces and 1 output spaces. In this experiment, we fix β to be 6. The results are listed in Table 4. The results indicates that the contrastive learning in the output space provides the best improvement on robustness against FGSM and PGD. This shows that a contrastive loss in embedding space does not provide a better robustness over contrastive loss in output space, which can be viewed as a AT loss combined with a TRADES loss. In addition, the results on the first 4 layers are similar to the results of the AT method. This may be due to the parameter β is not optimal for these layers. Also, as the layer get deeper, the model's performance on clean and Deepfool images drops.

Table 1: Top1 classification accuracy (%) under different attack on CIFAR-10 [mean (std)]

	Attack Methods			
	Id	FGSM	PGD	Deepfool
ResNet	69.50	29.15	7.64	10.20
	(0.04)	(0.10)	(0.02)	(0.67)
ResNet+AT-FGSM	75.05	50.79	38.99	11.64
	(0.13)	(0.09)	(0.07)	(0.34)
ResNet+AT-PGD	74.57	54.51	47.79	10.98
	(0.03)	(0.04)	(0.08)	(0.24)
ResNet+TRADES-FGSM	75.07	46.08	29.96	11.93
	(0.10)	(0.06)	(0.00)	(0.75)
ResNet+TRADES-PGD	**75.73**	51.79	41.17	**12.12**
	(0.03)	(0.02)	(0.01)	(0.59)
ResNet+CL-FGSM (Our method)	73.59	53.14	46.52	9.77
	(0.18)	(0.22)	(0.17)	(0.36)
ResNet+CL-PGD (Our method)	73.79	**54.66**	**49.44**	9.90
	(0.19)	(0.12)	(0.13)	(0.39)

Table 2: Top1 classification accuracy (%) under different attack on MNIST [mean (std)]

	Attack Methods			
	Id	FGSM	PGD	Deepfool
ResNet	98.64	85.59	10.39	7.38
	(0.04)	(0.10)	(0.02)	(0.67)
ResNet+AT-FGSM	99.19	97.81	94.16	16.18
	(0.12)	(0.09)	(0.06)	(0.40)
ResNet+AT-PGD	99.30	98.49	97.99	10.65
	(0.02)	(0.11)	(0.05)	(0.34)
ResNet+TRADES-FGSM	99.21	97.65	94.70	18.53
	(0.03)	(0.06)	(0.11)	(0.55)
ResNet+TRADES-PGD	99.20	98.06	97.32	13.58
	(0.07)	(0.02)	(0.07)	(0.31)
ResNet+CL-FGSM (Our method)	99.17	98.56	98.15	18.37
	(0.12)	(0.07)	(0.12)	(0.20)
ResNet+CL-PGD (Our method)	**99.31**	**98.84**	**98.60**	**18.77**
	(0.10)	(0.02)	(0.14)	(0.36)

5.6.2 Trade-off Parameter

We set the trade-off parameter β to be 0.1, 0.5, 1, 3, 6 and 7. The results are listed in Table 5. The results indicates that the "stronger" contrastive learning components provides better robustness against FGSM-like attack at a cost of worse performance on clean inputs and Deepfool inputs. When the parameter is small, the results of method get closer to the AT method. The parameter also seems to have an optimal value at around 6 to achieve the best robustness against FGSM.

6 Conclusions

This project investigates whether an contrastive loss in embedding space can improve adversarial robustness. The results on CIFAR-10 and MNIST indicates that the combination of adversarial training and contrastive learning can improve the model's robustness against the adversarial attack used in training but may decrease the model's robustness against other attack. This means that the improved robustness may come from overfitting on the attack used in training. In addition, the ablation study on embeddings indicates that the contrastive loss in embedding space cannot outperform the contrastive loss in output space i.e. TRADES loss.

Table 3: Definitions of embedding spaces and output space

Name	Components
Embedding-1	conv1 + bn1 + relu + maxpool
Embedding-2	conv1 + bn1 + relu + maxpool + layer1
Embedding-3	conv1 + bn1 + relu + maxpool + layer1 + layer2
Embedding-4	conv1 + bn1 + relu + maxpool + layer1 + layer2 + layer3
Embedding-5	conv1 + bn1 + relu + maxpool + layer1 + layer2 + layer3 + layer4 + avgpool
Output	conv1 + bn1 + relu + maxpool + layer1 + layer2 + layer3 + layer4 + avgpool + fc

Table 4: Ablation study on embeddings. Top1 classification accuracy (%) on CIFAR-10 [mean (std)]

	Attack Methods			
	Id	FGSM	PGD	Deepfool
ResNet+CL-Embed-1-FGSM	**75.48**	50.59	38.34	11.32
	(0.22)	(0.02)	(0.16)	(0.16)
ResNet+CL-Embed-2-FGSM	75.28	50.81	39.08	**11.37**
	(0.02)	(0.09)	(0.08)	(0.12)
ResNet+CL-Embed-3-FGSM	75.43	50.85	39.09	11.79
	(0.10)	(0.12)	(0.06)	(0.23)
ResNet+CL-Embed-4-FGSM	75.30	50.82	39.43	11.63
	(0.17)	(0.07)	(0.06)	(0.14)
ResNet+CL-Embed-5-FGSM	73.59	53.14	46.52	9.77
	(0.18)	(0.22)	(0.17)	(0.36)
ResNet+CL-Output-FGSM	72.01	**53.40**	**48.06**	9.49
	(0.12)	(0.14)	(0.26)	(0.16)

However, this project only evaluates the method on two image classification datasets. To draw a more trustworthy conclusion, we also need to consider tasks other than classification and visual modality. What's more, including more evasion attack methods such as black-box attack methods can help test whether the improvement brought by the contrastive learning is from overfitting to adversarial examples.

7 GitHub Repository

Please see https://github.com/zh-qifan/Contrastive_Learning_Adversarial_Training.

Table 5: Ablation study on trade-off parameter β. Top1 classification accuracy (%) on CIFAR-10 [mean (std)]

	Attack Methods			
	Id	FGSM	PGD	Deepfool
ResNet+CL-0.1-FGSM	**74.92**	51.17	40.90	**11.63**
	(0.54)	(0.24)	(0.08)	(0.03)
ResNet+CL-0.5-FGSM	74.71	52.14	44.10	10.58
	(0.14)	(0.10)	(0.05)	(0.43)
ResNet+CL-1-FGSM	74.79	52.36	45.07	10.24
	(0.09)	(0.18)	(0.09)	(0.70)
ResNet+CL-3-FGSM	74.15	52.95	46.53	10.04
	(0.16)	(0.53)	(0.45)	(0.33)
ResNet+CL-6-FGSM	73.59	**53.14**	46.52	9.77
	(0.18)	(0.22)	(0.17)	(0.36)
ResNet+CL-7-FGSM	74.41	52.75	**46.77**	9.93
	(0.27)	(0.09)	(0.16)	(0.46)

References

Babak Alipanahi, Andrew Delong, Matthew T Weirauch, and Brendan J Frey. Predicting the sequence specificities of dna-and rna-binding proteins by deep learning. *Nature biotechnology*, 33(8):831–838, 2015.

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples, 2018.

Shumeet Baluja and Ian Fischer. Adversarial transformation networks: Learning to generate adversarial examples, 2017.

Nicholas Carlini and David A. Wagner. Towards evaluating the robustness of neural networks. *CoRR*, abs/1608.04644, 2016. URL `http://arxiv.org/abs/1608.04644`.

Francesco Croce and Matthias Hein. Mind the box: l_1-apgd for sparse adversarial attacks on image classifiers. In *International Conference on Machine Learning*, pages 2201–2211. PMLR, 2021.

Yao Deng, Xi Zheng, Tianyi Zhang, Chen Chen, Guannan Lou, and Miryung Kim. An analysis of adversarial attacks and defenses on autonomous driving models. In *2020 IEEE international conference on pervasive computing and communications (PerCom)*, pages 1–10. IEEE, 2020.

Guneet S. Dhillon, Kamyar Azizzadenesheli, Zachary C. Lipton, Jeremy Bernstein, Jean Kossaifi, Aran Khanna, and Anima Anandkumar. Stochastic activation pruning for robust adversarial defense, 2018.

Ruili Feng, Kecheng Zheng, Yukun Huang, Deli Zhao, Michael Jordan, and Zheng-Jun Zha. Rank diminishing in deep neural networks, 2022.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

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.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features, 2019.

Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. URL `https://api.semanticscholar.org/CorpusID:18268744`.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019.

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*, 2018. URL `https://openreview.net/forum?id=rJzIBfZAb`.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2574–2582, 2016.

Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *2016 IEEE European symposium on security and privacy (EuroS&P)*, pages 372–387. IEEE, 2016.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

Samuel Henrique Silva and Peyman Najafirad. Opportunities and challenges in deep learning adversarial robustness: A survey, 2020.

Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23(5):828–841, 2019.

Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*, 2017.

Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses, 2020.

Mihnea Horia Vrejoiu. Neural networks and deep learning in cyber security. *Romanian Cyber Security Journal*, 1(1):69–86, 2019.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning*, 2019.

Fei Zuo and Qiang Zeng. Exploiting the sensitivity of l_2 adversarial examples to erase-and-restore, 2020.

A Top3 and Top5 Accuracy Result

Table 6: Top3 classification accuracy (%) under different attacks on CIFAR-10 [mean (std)]

	Attack Methods			
	Id	FGSM	PGD	Deepfool
ResNet	92.07 (0.04)	70.28 (0.10)	52.56 (0.02)	64.72 (0.67)
ResNet+AT-FGSM	94.04 (0.13)	85.29 (0.09)	80.44 (0.07)	66.14 (0.34)
ResNet+AT-PGD	93.70 (0.03)	**87.04** (0.04)	84.91 (0.08)	64.80 (0.24)
ResNet+TRADES-FGSM	**94.53** (0.10)	83.17 (0.06)	75.88 (0.00)	**66.84** (0.75)
ResNet+TRADES-PGD	94.41 (0.09)	85.98 (0.18)	82.41 (0.22)	66.83 (0.59)
ResNet+CL-FGSM (Our method)	93.59 (0.22)	86.27 (0.10)	84.19 (0.27)	65.09 (0.33)
ResNet+CL-PGD (Our method)	93.69 (0.19)	**87.04** (0.12)	**85.41** (0.13)	64.87 (0.39)

Table 7: Top5 classification accuracy (%) under different attacks on CIFAR-10 [mean (std)]

	Attack Methods			
	Id	FGSM	PGD	Deepfool
ResNet	97.36 (0.03)	86.72 (0.04)	74.20 (0.02)	**81.80** (0.05)
ResNet+AT-FGSM	98.32 (0.03)	94.59 (0.04)	92.29 (0.06)	78.29 (0.02)
ResNet+AT-PGD	98.12 (0.06)	**95.33** (0.05)	94.45 (0.08)	76.45 (0.11)
ResNet+TRADES-FGSM	98.33 (0.10)	93.48 (0.06)	89.64 (0.02)	79.46 (0.07)
ResNet+TRADES-PGD	**98.43** (0.12)	94.70 (0.08)	92.99 (0.11)	77.79 (0.14)
ResNet+CL-FGSM (Our method)	97.87 (0.22)	94.84 (0.12)	93.98 (0.06)	77.33 (0.25)
ResNet+CL-PGD (Our method)	97.99 (0.19)	95.30 (0.17)	**94.75** (0.18)	76.47 (0.22)

Table 8: Top3 classification accuracy (%) under different attacks on MNIST [mean (std)]

	Attack Methods			
	Id	FGSM	PGD	Deepfool
ResNet	99.77	96.45	59.05	43.70
	(0.04)	(0.10)	(0.02)	(0.13)
ResNet+AT-FGSM	99.95	99.78	99.52	**74.80**
	(0.27)	(0.09)	(0.07)	(0.33)
ResNet+AT-PGD	**99.98**	**99.92**	**99.87**	71.15
	(0.13)	(0.04)	(0.08)	(0.21)
ResNet+TRADES-FGSM	99.95	99.77	99.60	66.64
	(0.05)	(0.16)	(0.01)	(0.43)
ResNet+TRADES-PGD	99.95	99.81	99.78	72.06
	(0.19)	(0.04)	(0.32)	(0.29)
ResNet+CL-FGSM (Our method)	99.93	99.86	99.87	73.20
	(0.42)	(0.21)	(0.27)	(0.23)
ResNet+CL-PGD (Our method)	99.91	99.83	99.80	73.56
	(0.09)	(0.11)	(0.13)	(0.16)

Table 9: Top5 classification accuracy (%) under different attacks on MNIST [mean (std)]

	Attack Methods			
	Id	FGSM	PGD	Deepfool
ResNet	99.93	98.68	77.48	57.04
	(0.01)	(0.10)	(0.12)	(0.23)
ResNet+AT-FGSM	99.99	99.94	99.81	86.84
	(0.02)	(0.09)	(0.17)	(0.14)
ResNet+AT-PGD	99.99	**99.98**	**99.98**	87.29
	(0.03)	(0.04)	(0.13)	(0.32)
ResNet+TRADES-FGSM	99.99	99.95	99.71	79.19
	(0.05)	(0.16)	(0.09)	(0.13)
ResNet+TRADES-PGD	99.99	99.96	99.95	82.82
	(0.09)	(0.04)	(0.10)	(0.22)
ResNet+CL-FGSM (Our method)	99.99	99.97	99.96	**91.31**
	(0.02)	(0.07)	(0.11)	(0.13)
ResNet+CL-PGD (Our method)	**99.99**	99.97	99.97	86.90
	(0.03)	(0.01)	(0.04)	(0.06)