Multi-task Self-supervised Transfer Learning For Adversarial Robustness

Tanmay Jaiswal tj2467@columbia.edu Yingfei Fan yf2549@columbia.edu

December 2021

Abstract

Deep neural networks are fragile to adversarial attacks when dealing with image recognition problems. Motivated by the research results that when models are trained on several tasks simultaneously, they become more robust to adversarial attacks on individual tasks [10] and the fact that the representations pre-trained through self-supervision could enable fast fine-tuning to downstream tasks, leading to better generalization [8], we aim to conduct an empirical analysis to see whether pre-training with a combination of self-supervised tasks in advance will enhance overall adversarial robustness and perform better than the traditional supervised adversarial training. Specifically in experiments, we pre-train the model under the SimCLR framework (A Simple Framework for Contrastive Learning of Visual Representations [2]) with ResNet18 as the backbone model on CIFAR100 unlabeled dataset. We then fine-tune the model with different fine-tuning strategies using CIFAR10 data. Finally, we generate adversarial examples from CIFAR10 test data using FGSM [6] algorithm to evaluate the model. The experimental results show that pre-training with SimCLR and then fine-tuning with 30% adversarial data yields a around 38% adversarial accuracy. We consider it as an efficient workflow for adversarial training compared to the traditional ones.Our code is at GitHub repo.

1 Introduction

Deep learning neural networks can obtain high performance in computer vision tasks. However, such models also remain brittle to adversarial attacks [6]. An adversary aims to attack a target model by adding human imperceptible perturbations to the input and then making the model predict a wrong class with high confidence. The weakness of the model being "fooled" by adversarial examples easily can lead to serious problems in real-world application in the self-driving or medicine industry [17] [13].

Extensive studies on adversarial training and defense have been conducted. Some of them focus on designing robust optimization functions such as PGD adversarial training [9] or FGSM adversarial training [6]. Some studies try to add a regularization term in loss functions to conduct a more robust training strategy [19]. However, these kinds of adversarial training have two major downsides. On the one hand, directly training the model on an adversarial dataset takes lots of time since either generating adversarial examples or training all the layers with full data size for a deep network is resource-extensive [6] [19]. On the other hand, the traditional supervised training of deep neural networks requires massive, labeled datasets, which could be unavailable and expensive. Therefore, a more efficient adversarial training approach is worth exploring.

Fortunately, a decent amount of experiments using transfer learning demonstrate that the self-supervised pre-trained framework can achieve large performance margin, compared with the conventional end-to-end adversarial training baseline [1]. Additionally, different self-supervised pre-trained models have diverse adversarial vulnerabilities, inspiring researchers to ensemble different pre-training tasks. Furthermore, the SimCLR [2] approaches contrastive pre-training in a similar way utilizing a combination of pre-training tasks. Although the original SimCLR paper does not mention adversarial robustness, the experiment results from [1] give us a strong hint that combining SimCLR as the pre-trained job and adversarial training as the downstream task will be a potential approach to attain efficient adversarial training and have a good result. Therefore, in order to tackle the issues of lacking computing resources or enough ground-truth labels, we can take advantage of transfer learning and self-supervision. In our experiments, we explore what kinds of pre-trained tasks, how to combine them, and how many labeled data are needed during the fine-tuning process can boost the overall robustness of the model.

Contributions

- Introduce contrastive learning to adversarial training by using SimCLR pre-training framework.
- Conduct empirical analysis on how SimCLR could enhance model robustness.
- Obtain the conclusion that SimCLR indeed is an efficient and effective pre-training framework and work well with adversarial fine-tuning in the later steps.

2 Related Work

Transfer learning for adversarial robustness. Transfer learning is often used when data or labeled data is scarce, or full-scale training is too costly. It achieves efficient and effective training by studying a network on one task (pre-training) and re-purposed on another (fine-tuning) [14]. Adversarial robustness is introduced to transfer learning and vice versa. On the one hand, using PGD examples during training on the source task generates provably better representations, leading to more general robust features that are easier to transfer [3]. On the other hand, performing adversarial training on top of semi-supervised learning can further improve transferability, suggesting that the two approaches have complementary benefits on representations [4].

Metric Learning for adversarial robustness. Distance metric learning (or simply, metric learning) aims at automatically constructing task-specific distance metrics from (weakly) supervised data,which enhances the performance of similarity-based algorithms [16]. Adding an additional constraint (the triplet loss function) to the model will produce more robust classifiers since the triplet loss function will pull all the images of one class, both natural and adversarial, closer while pushing the images of other classes far apart [11].

Contrastive Learning. Contrastive learning is a part of metric learning. It aims to learn the general features of a dataset without labels by teaching the model which data points are similar or different. The SimCLR pretrain framework [2] first learns generic representations of images on an unlabeled dataset using contrastive learning techniques, and then fine-tune model with a small amount of labeled images to achieve improving performance for a given classification task. The generic representations are learned by simultaneously maximizing agreement between differently transformed views of the same image and minimizing agreement between transformed views of different images. In this way, the parameters of a neural network are updated using this contrastive objective causes representations of corresponding views to "attract" each other, while representations of non-corresponding views "repel" each other [2].

Self-supervised pretraining for adversarial robustness. Numerous self-supervised tasks have been proposed in recent years, including image recolorization [20], predicting image rotations [5], and jigsaw puzzle solving [12]. The effective representations that are pre-trained through self-supervision could enable fast fine-tuning to downstream tasks, and lead to better generalization [8]. What's more, learning with more unlabeled data can result in better adversarially robust generalization [18]. Robust pretrained models can benefit the subsequent fine-tuning in two ways: i) boosting final model robustness; ii) saving the computation cost, if proceeding towards adversarial fine-tuning

[1].

Multitask learning for adversarial robustness. Multitask learning [15] aims to resolve several tasks at the same time and researches show that when models are trained on multiple tasks at once, they become more robust to adversarial attacks on individual tasks [10]. This finding again motivates to us pretrain the model using multiple tasks at once.

3 Our Approach

From a high-level perspective, our approach is to combine the pre-trained framework from SimCLR [2] and the downstream adversarial fine-tuning workflow from [1]. View 1 for details.

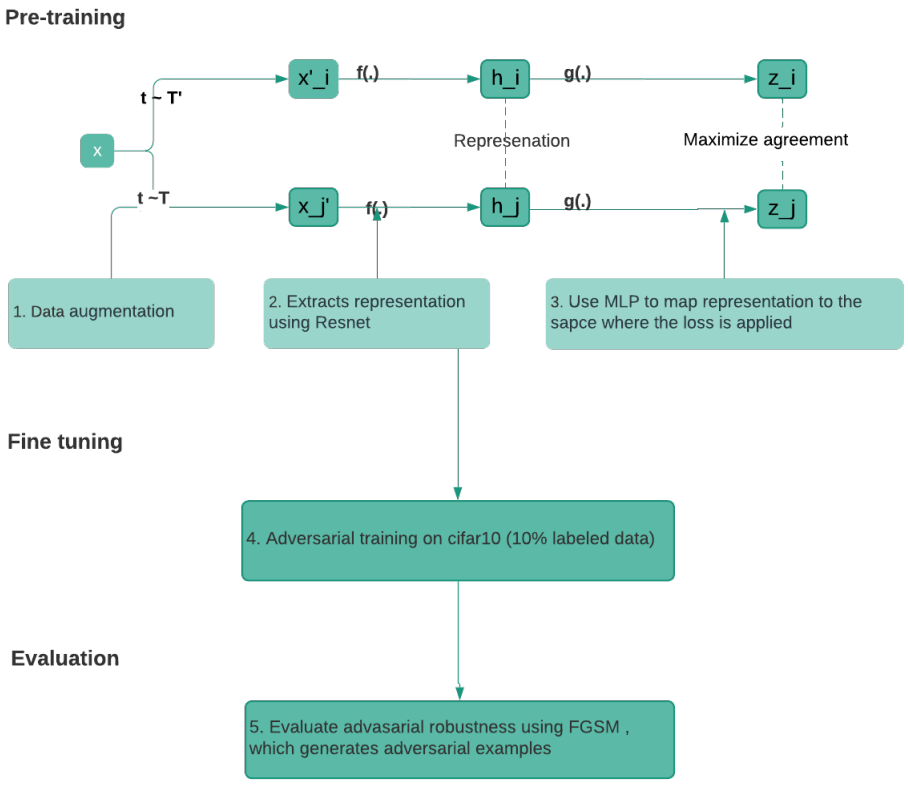

Figure 1: Workflow

3.1 Pre-training

In this step we pre-train a model with resnet as the backbone using SimCLR framework on CIFAR100 dataset.

Firstly, three data augmentation tasks will be applied to the original sample data to transform a data point randomly resulting in two correlated views of the same example x'_i, x'_j, which we consider as a positive pair. At this step, three augmentation tasks: random cropping, random color distortions, and then random Gaussian blur are sequentially applied.

Secondly, a neural network base encoder $f()$ that extracts representation vectors from augmented data examples is needed. At this step, we adopt the commonly used ResNet18 [7] as the original SimCLR suggested. After this step, we obtain $h_i = f(x'_i) = ResNet(x'_i)$ where $h_i \in R^d$ is the output after the average pooling layer.

Thirdly, a small neural network projection head $g()$ that maps representations to the space where contrastive loss can be applied on. We use a MLP with one hidden layer to obtain $z_i = g(h_i) = W^{(2)}(W^{(1)}h_i)$ where is a ReLU function.

Lastly, a contrastive loss function $l_{i,j} = -log\frac{exp(sim(z_i,z_j)/\tau}{\sum_{k=1}^{2N} I_{k\neq i} exp(sim(z_i,z_j)/\tau)}$, where $sim(u, v)$ is the cosine similarity between u and v, is defined for the contrastive prediction task. In this loss function, given a set x'_k including a positive pair of examples x'_i and x'_j, we aims to identify x'_j in $x'_{k(k\neq i)}$ for a given x'_i.

3.2 Fine-tuning

Firstly, we generate adversarial training examples $x_adv = x + \epsilon sign(\nabla x J(\theta, x, y))$ for CIFAR-10 using Fast Gradient Sign Method (FGSM) algorithms [6].

Secondly, we fine tune the pre-train model using 1%, 10%, and 30% of the original and adversarial examples separately (six different fine-tuning). At this step, we train a prediction layer with the labeled data. A more sophisticated transfer learning strategy may be worth exploring such as freezing the first 10 layers.

3.3 Evaluation

Similar to the previous step, we generate adversarial test example from CIFAR-10 test dataset using FGSM algorithms. We feed to adversarial test data into our model then calculate the cross entropy loss as the mis-classification rate.

4 Experiment

In this section, we design and conduct several experiments to examine the network robustness against different configurations for image classification. First, we show how adversarial accuracy will change under different fine-tuning strategies. The experiment results are computed from using 1%, 10%, and 30% labeled data either trained both with and without adversarial examples. 1. We also evaluate the efficacy of fine-tuning using 30% of the dataset with adversarial examples when the model is pre-trained with rotation prediction, Jigsaw puzzle solving and SimCLR. 2.

4.1 Experiment setting

We conduct all of our experiments using Pytorch v1.70 on a single Tesla P4 with 4 CPUs and 15 GB of memory. During pre-training, we use CIFAR100 dataset with 256 batch size, 100 epochs, and exponentially decaying learning rate. In the fine-tuning process, we add a final prediction layer and retrain only this with labeled data with 128 batch size, 50 epochs.

4.2 Experiment Results

Table 1: Adversarial accuracy under different fine-tuning strategies. The experiment results are computed from using 1%, 10%, and 30% labeled data either with or without adversarial examples.

labeled data	original accuracy	robustness accuracy	original w robustness training	robustness accuracy w robustness training
supervised(no pretraining)	68%	23%	66%	36%
1%	58%	26%	49%	29%
10%	61%	32%	60%	36%
30%	63%	33%	62%	38%
100%	63%	35%	62%	40%

Table 2: Comparison With Other Pre-train Task. After the self-supervised pretraining, a partial adversarial fine-tuning is conducted.

Rotation	Jigsaw	SimCLR
25%	29%	38%

From Table 1, we can infer that pre-training with SimCLR improves robustness compared to training from scratch even though the accuracy on regular test samples is

marginally lower. This holds even when the model is trained with an adversarial loss while fine-tuning.

Further, Table 2 tells us that SimCLR provides greater adversarial robustness than other pretraining tasks.

4.3 Design Limitation

Our current implementation validates the methodology on the CIFAR dataset with a single transfer learning policy. This does not necessarily imply that the approach can extend to other datasets and transfer learning policies. More experiment on different dataset are worth exploring, since CIFAR100 and CIFAR10 are relatively smaller than other datasets such as Imagenet and Cityspaces both in size of images and number of images. More dynamic transfer learning policies can be explored such as freezing the first 10 layers or only retraining the last layer. Further, the final step would be to pre-train the model with adversarial contrastive loss and fine-tune without adversarial loss to understand if adversarial pretraining is enough to prevent attacks.

5 Conclusion

Motivated by the studies on robustness through multi-task learning, transfer learning, and self-supervision in the adversarial robustness field, we conducted experiments by combining the SimCLR [2] pre-training framework and the downstream adversarial fine-tuning workflow[1]. Our experimental results shows that this combination indeed is an efficient and effective adversarial training workflow within the tested paramaters. However, more diverse experiments are needed in to validate the complete parameters under which this framework remains effective.

References

[1] Tianlong Chen, Sijia Liu, Shiyu Chang, Yu Cheng, Lisa Amini, and Zhangyang Wang. Adversarial robustness: From self-supervised pre-training to fine-tuning, 2020.

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, 2020.

[3] Todor Davchev, Timos Korres, Stathi Fotiadis, Nick Antonopoulos, and Subramanian Ramamoorthy. An empirical evaluation of adversarial robustness under transfer learning, 2019.

[4] Zhun Deng, Linjun Zhang, Kailas Vodrahalli, Kenji Kawaguchi, and James Zou. Adversarial training helps transfer learning via better representations, 2021.

[5] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations, 2018.

[6] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[8] Hong Liu, Mingsheng Long, Jianmin Wang, and Michael I. Jordan. Towards understanding the transferability of deep representations, 2019.

[9] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2019.

[10] Chengzhi Mao, Amogh Gupta, Vikram Nitin, Baishakhi Ray, Shuran Song, Junfeng Yang, and Carl Vondrick. Multitask learning strengthens adversarial robustness, 2020.

[11] Chengzhi Mao, Ziyuan Zhong, Junfeng Yang, Carl Vondrick, and Baishakhi Ray. Metric learning for adversarial robustness, 2019.

[12] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles, 2017.

[13] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana. Deepxplore. *Proceedings of the 26th Symposium on Operating Systems Principles*, Oct 2017.

[14] Ali Shafahi, Parsa Saadatpanah, Chen Zhu, Amin Ghiasi, Christoph Studer, David Jacobs, and Tom Goldstein. Adversarially robust transfer learning, 2020.

[15] Trevor Standley, Amir R. Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning?, 2020.

[16] Juan Luis Suárez-Díaz, Salvador García, and Francisco Herrera. A tutorial on distance metric learning: Mathematical foundations, algorithms, experimental analysis, prospects and challenges (with appendices on mathematical background and detailed algorithms explanation), 2020.

[17] Yuchi Tian, Kexin Pei, Suman Jana, and Baishakhi Ray. Deeptest: Automated testing of deep-neural-network-driven autonomous cars, 2018.

[18] Runtian Zhai, Tianle Cai, Di He, Chen Dan, Kun He, John Hopcroft, and Liwei Wang. Adversarially robust generalization just requires more unlabeled data, 2019.

[19] 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.

[20] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization, 2016.