Reinforcement Based Learning on Classification Task Yields Better Generalization and Adversarial Accuracy (Student Abstract)

Shashi Kant Gupta
Department of Electrical Engineering, Indian Institute of Technology Kanpur, India
D211, Hall 9, IIT Kanpur, India, Ph: +91-797-908-8653
shashikg@iitk.ac.in

Abstract

Deep Learning has become interestingly popular in the field of computer vision, mostly attaining near or above human-level performance in various vision tasks. But recent work has also demonstrated that these deep neural networks are very vulnerable to adversarial examples (adversarial examples - inputs to a model which are naturally similar to original data but fools the model in classifying it into a wrong class). In this work, we proposed a novel method to train deep learning models on an image classification task. We used a reward-based optimization function, similar to the vanilla policy gradient method in reinforcement learning to train our model instead of conventional cross-entropy loss. An empirical evaluation on cifar10 dataset showed that our method outperforms the same model architecture trained using cross-entropy loss function (on adversarial training). At the same time, our method generalizes better to the training data with the difference in test accuracy and train accuracy $< 2\%$ for most of the time as compared to cross-entropy one, whose difference most of the time remains $> 2\%$.

Introduction

There's a tremendous increase in using deep learning models for various perceptual tasks in computer vision. Thousands of new works are being published every year on attaining better accuracy on different datasets. But these deep neural networks are very vulnerable to adversarial examples (adversarial examples - inputs to a model which are naturally similar to original data but fools the model in classifying it into a wrong class) which raises great concern about whether the model should be used for real-time application purpose or not. While the past few years have seen intense research in training robust models against adversarial attacks most of them have focused on using various adversarial training approaches, unlabelled data, or revisiting misclassified examples. In our work, we focused on introducing a new kind of objective function. We used a reward-based optimization function, similar to the vanilla policy gradient method in reinforcement learning to train our model instead of conventional cross-entropy loss. The formulation of this method is fairly simple and similar to the vanilla policy gradient (Williams 1992). We just design the reward environment, which is as simple as giving positive rewards for correct classification and negative rewards for the wrong classification. And in the end, we train the model to maximize reward using a policy gradient. Here our policy is simply the softmax probability distribution to classify the given input image among the various classes. We trained a very minimal CNN architecture against FGSM attacks (Goodfellow, Shlens, and Szegedy 2014) using our method and with cross-entropy loss. We observed that even though the model trained using cross-entropy achieved higher training accuracy on the training set as compared to our method, the accuracy against adversarial examples was higher for our method. One more very interesting result that came out was that our method generalizes much better than the model trained using cross-entropy loss i.e. our training and validation accuracy remains almost close to each other.

Methods

Some of the previous work has focused on Reinforcement Based learning in classification tasks but none of them evaluated its adversarial robustness, as well as most of them, are complex formulation (Wiering et al. 2011). In this work, we propose a fairly simple implementation. We will also release a simple RL environment for the classification task which will act similar to other RL environments. This environment can be used to run various RL algorithms to train on the classification tasks. In this work, we used the basic formulation of the Vanilla Policy Gradient (VPG) method to test our model. The state s_t is the input image, action a_t is the predicted class, and reward R_t depends upon a_t and actual class label y_t. If $a_t = y_t$, we give a positive reward (+1) on the other hand if $a_t \neq y_t$, we give a negative reward (-1). Finally, we train the model using VPG loss (refer to Eq. 1). Here t is the example image in the given training batch of size B. Please note one direct benefit of this implementation is that we are penalizing/rewarding our network based on what predictions it makes. So if the network predicts a wrong class, it will be penalized based on the gradients calculated using that specific wrong class which in comparison to cross-entropy depends only on the gradients calculated using the correct label.

$$\sum_{t}^{B} -\frac{1}{B} log(P(a_t|s_t)) * R_t \qquad (1)$$

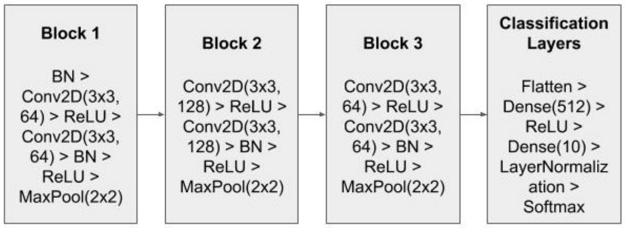
Block 1
BN > Conv2D(3x3, 64) > ReLU > Conv2D(3x3, 64) > BN > ReLU > MaxPool(2x2)
Block 2
Conv2D(3x3, 128) > ReLU > Conv2D(3x3, 128) > BN > ReLU > MaxPool(2x2)
Block 3
Conv2D(3x3, 64) > ReLU > Conv2D(3x3, 64) > BN > ReLU > MaxPool(2x2)
Classification Layers
Flatten > Dense(512) > ReLU > Dense(10) > LayerNormalization > Softmax

Figure 1: CNN model architecture

Experiment

We trained a very minimal CNN architecture on cifar10. The architecture used was shown in Figure 1. The model was trained on adversarial images generated using FGSM, these adversarial images were generated at the beginning of each training step on the trained model. We trained both the model i.e. cross-entropy (CE) one and our method (RL) one for 220 epochs with RMSprop optimiser at learning_rate=0.0001 and decay=1e-6 (This choice of hyperparameter was taken from tensorflow example model on cifar10). We initially made checkpoints of our model at intervals of 20 epochs and reporting training, testing, and adversarial accuracy (on test data) at those intervals. We ran a second instance in which we recorded the training history till 150 epochs have included the corresponding plot in our supplementary pdf. We then picked the checkpoint at which the respective method performed best on FGSM attack. Which was RL model at 220th epoch and CE model at 60th epoch. We tested the robustness of both of the model using AutoAttack (an ensemble of diverse parameter-free attacks) (Croce and Hein 2020).

Result and Discussion

The training performance for both the model is shown in Figure 2. The model trained using cross-entropy loss is abbreviated as ‘CE’ while the model trained using our reinforcement-based method is abbreviated as ‘RL’. The CE model reaches it maximum adversarial accuracy at 60th epoch with adversarial accuracy on FGSM attack 32.86% and reduces afterwards whereas the RL model performance improved over successive epochs with adversarial accuracy on FGSM attack 37.66%. From the Figure 2, we can also see that the RL model generalises much better than the CE model with its training and testing curve to be very close to each other. We further tested the robustness of both of the model using AutoAttack to confirm the robustness against several other adversarial attacks. We found that our model still performs much better than the CE model (Table 1). The reported accuracy are on cifar10’s test data.

Model	Natural Acc.	Adversarial Acc.
RL (220th epoch)	52.85%	30.41%
CE (60th epoch)	50.96%	26.36%
CE (220th epoch)	59.12%	17.95%

Table 1: Accuracy against AutoAttack (eps = $8/255$).

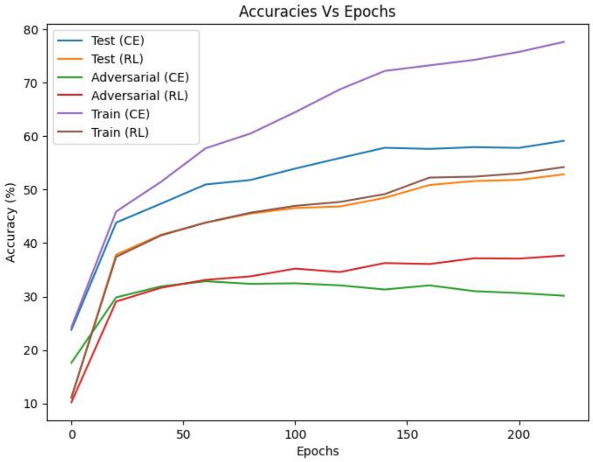

Figure 2: Training accuracy

Conclusion

In this work, we proposed a fairly simple method to introduce reinforcement based learning for classification tasks. Our method shows an initial sign of improvements for better generalisation as well as more robustness to adversarial attacks when compared with same model architecture trained on cross-entropy loss. Even though our present model does not beat the SOTA results. If we consider some recent work involving faster training for robust model against adversarial attack our model performance is quite comparable and outperforms (Wang and Zhang 2019) which shows 29.35% accuracy on cifar10 against AutoAttack (2020). But we still need to evaluate our method on a larger scale with more complex model like WRN which shows best performance on cifar10. Our RL formulation for classification tasks could potentially benefit other domains of research as well.

References

Croce, F.; and Hein, M. 2020. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. *ArXiv* abs/2003.01690.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* .

Wang, J.; and Zhang, H. 2019. Bilateral Adversarial Training: Towards Fast Training of More Robust Models Against Adversarial Attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV).*

Wiering, M. A.; van Hasselt, H.; Pietersma, A.; and Schomaker, L. 2011. Reinforcement learning algorithms for solving classification problems. In *2011 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*, 91–96.

Williams, R. J. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning* 8(3-4): 229–256. doi:10.1007/bf00992696.