Reinforcement Based Learning on Classification Task Could Yield Better Generalization and Adversarial Accuracy

Anonymous Author(s)
Affiliation
Address
email

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). Humans are very robust against such perturbations, one possible reason could be that humans do not learn to classify on the basis error between "target label" and "predicted label" but rather they learn on the basis of reinforcements which is based on what predictions they make. 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 $> 2\%$.

1 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 [8]. 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 [6]. 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 [9, 2, 12]. On the other hand humans are very robust against such perturbations. Previously, Engstrom et al. [5] and Ilyas et al. [7] shown that how adversarial perturbations corresponds to non-robust but predictive features and that adversarial robustness can indeed help deep neural networks in learning perceptually similar features. So we tried to find a different approach to train deep neural networks on the basis how possibly humans learn to classify images. Humans do not learn to classify on the basis error between "target label" and "predicted label" but instead they predicts a label and then some other people give them feedback which results into a positive or negative reward for the learner and on the basis of this reinforcements they learn to classify different stimuli.

Submitted to 2nd Workshop on Shared Visual Representations in Human and Machine Intelligence (SVRHM).

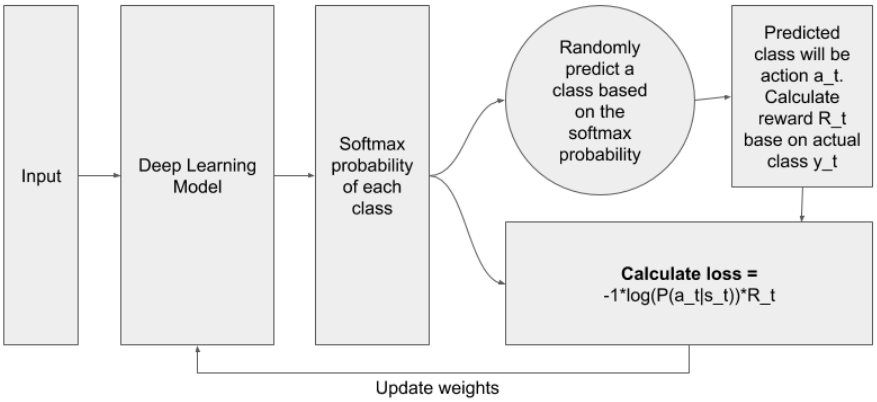


Figure 1: The complete flowchart for our RL based classification model

In this 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 [14]. We 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 [6] using our method and with cross-entropy loss. We observed that even though the model trained using cross-entropy achieved higher accuracy on the test 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.

2 Method

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 [13]. 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.

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

Note that during training we predict the class label i.e. take action by using random selection based on the softmax probability. This is extensively used in RL to bring randomness during training. The complete procedure is shown in Figure 1. 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. Another advantage of using this method is that we can develop different strategies for assigning rewards based on the prediction that the networks makes. For example giving higher negative rewards

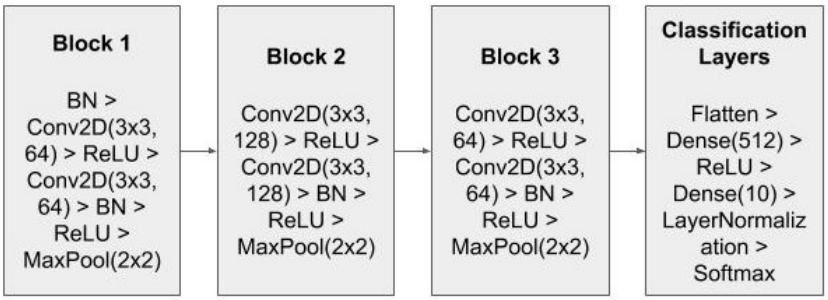


Figure 2: CNN model architecture

to those examples on which networks makes repeated wrong prediction could probably help in fasten the learning.

3 Experiment

We trained a very minimal CNN architecture on cifar10. The architecture used was shown in Figure 2. 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 [4]).

4 Results

The training performance for both the model is shown in Figure 3. 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 3, 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 also tested the robustness accuracy for both the models i.e. CE and RL one against different level of perturbations added to the original image. Results are shown in Figure 4. RL model always performed better than the CE model which was trained till 220 epochs. We also tested CE model which was trained till 60th epochs we found that for $eps \leq 3/255$, CE model performed better than RL model but its performance got worse on reaching $eps = 8/255$. Note that we also performed the evaluation without any adversarial training against FGSM attack for which RL model indeed showed better FGSM adversarial accuracy as compared to CE model. But it was very easy to attack that model via AutoAttack.

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 of both the models against AutoAttack (eps = $8/255$).

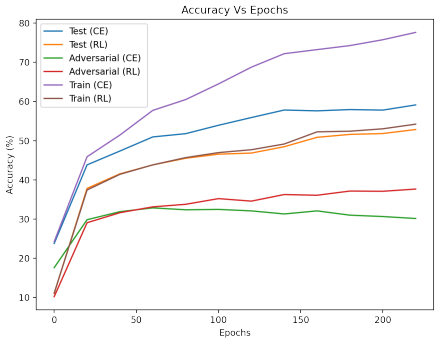


Figure 3: Training accuracy

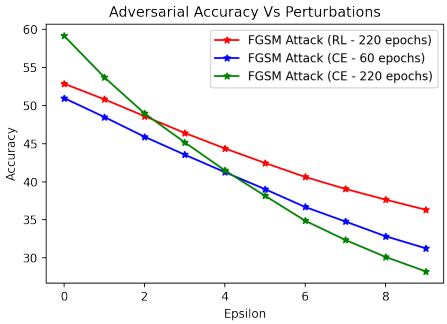


Figure 4: Adversarial accuracy vs perturbation strength

We further tested the robustness of both of the model using AutoAttack (contains ensemble of various adversarial attacks - consisting of apgd-ce, apgd-t, fab-t, and square attack [4, 1, 3]) to confirm the robustness against several other adversarial attacks. We found that our model still performs much better than the CE model (Table 1). Please note that all the reported accuracy are on cifar10's test data.

5 Discussion

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 [11] which shows 29.35% accuracy on cifar10 against AutoAttack [4]. But considering that this is a priliminary result, we still need to evaluate our method on a larger scale with more complex model like which are known to perform best on cifar10 and better parameter tuning.

On the other hand, our RL formulation for classification task could probably benefit other computer vision problems such as datasets having imbalanced class in which we can simply change the magnitude of positive or negative reward for different classes to remove discrepancy due to imbalance data. For future research, it would also be interesting to evaluate how the model performs if we change the method for reward assignment. It would also be interesting to test model trained using this method on predicting human similarity judgements as done by Peterson et al. [10].

And finally, we hope that this result will bring on interesting discussion on intersection of cognitive science and artificial intelligence in terms of understanding how humans might be learning to classify different objects.

References

[1] Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search, 2020.

[2] Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, Percy Liang, and John C. Duchi. Unlabeled data improves adversarial robustness, 2019.

[3] Francesco Croce and Matthias Hein. Minimally distorted adversarial examples with a fast adaptive boundary attack, 2020.

[4] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks, 2020.

[5] Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Brandon Tran, and Aleksander Madry. Adversarial robustness as a prior for learned representations, 2019.

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

[7] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 125–136. Curran Associates, Inc., 2019. URL `http://papers.nips.cc/paper/8307-adversarial-examples-are-not-bugs-they-are-features.pdf`.

[8] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, May 2015. doi: 10.1038/nature14539. URL `https://doi.org/10.1038/nature14539`.

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

[10] Joshua C. Peterson, Joshua T. Abbott, and Thomas L. Griffiths. Adapting deep network features to capture psychological representations: An abridged report. *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, Aug 2017. doi: 10.24963/ijcai.2017/697. URL `http://dx.doi.org/10.24963/ijcai.2017/697`.

[11] Jianyu Wang and Haichao Zhang. 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)*, October 2019.

[12] Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. In *International Conference on Learning Representations*, 2020. URL `https://openreview.net/forum?id=rklOg6EFwS`.

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

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

Appendices

A. Results Without Any Adversarial Training Against FGSM Attacks

We also performed the evaluation without any adversarial training against FGSM attack for which RL model indeed showed better FGSM adversarial accuracy as compared to CE model. But as stated before this was very easily attacked by AutoAttack.

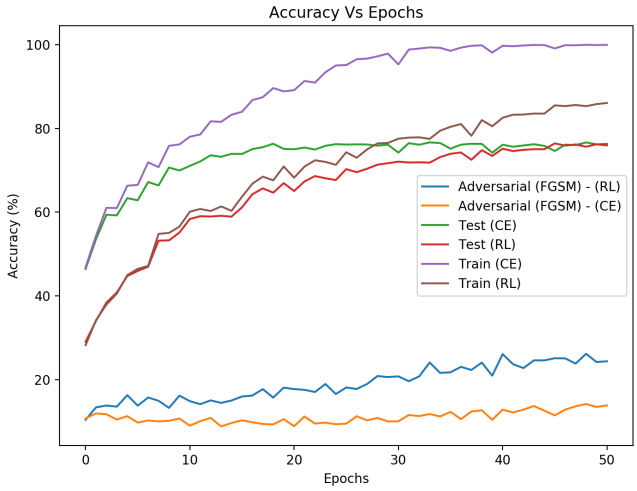


Figure 5: Results without adversarial training. We can see RL method still performs better than the CE method. Adversarial accuracy (FGSM) is calculated for $eps = 8/255$

B. Adversarial Accuracy vs Epochs on Another Training Instance (with adversarial training)

We initially made checkpoints of our model at intervals of 20 epochs. To test the performance for the model at near the 60th epoch after which the CE model performance starts to decrease and for the sake of generalisation, We ran a second training instance in which we recorded the training history till 150 epochs. It again resulted into similar results. Results are shown in Figure 6

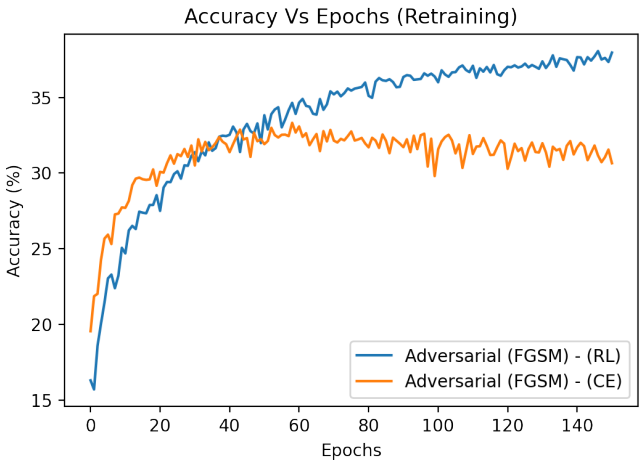
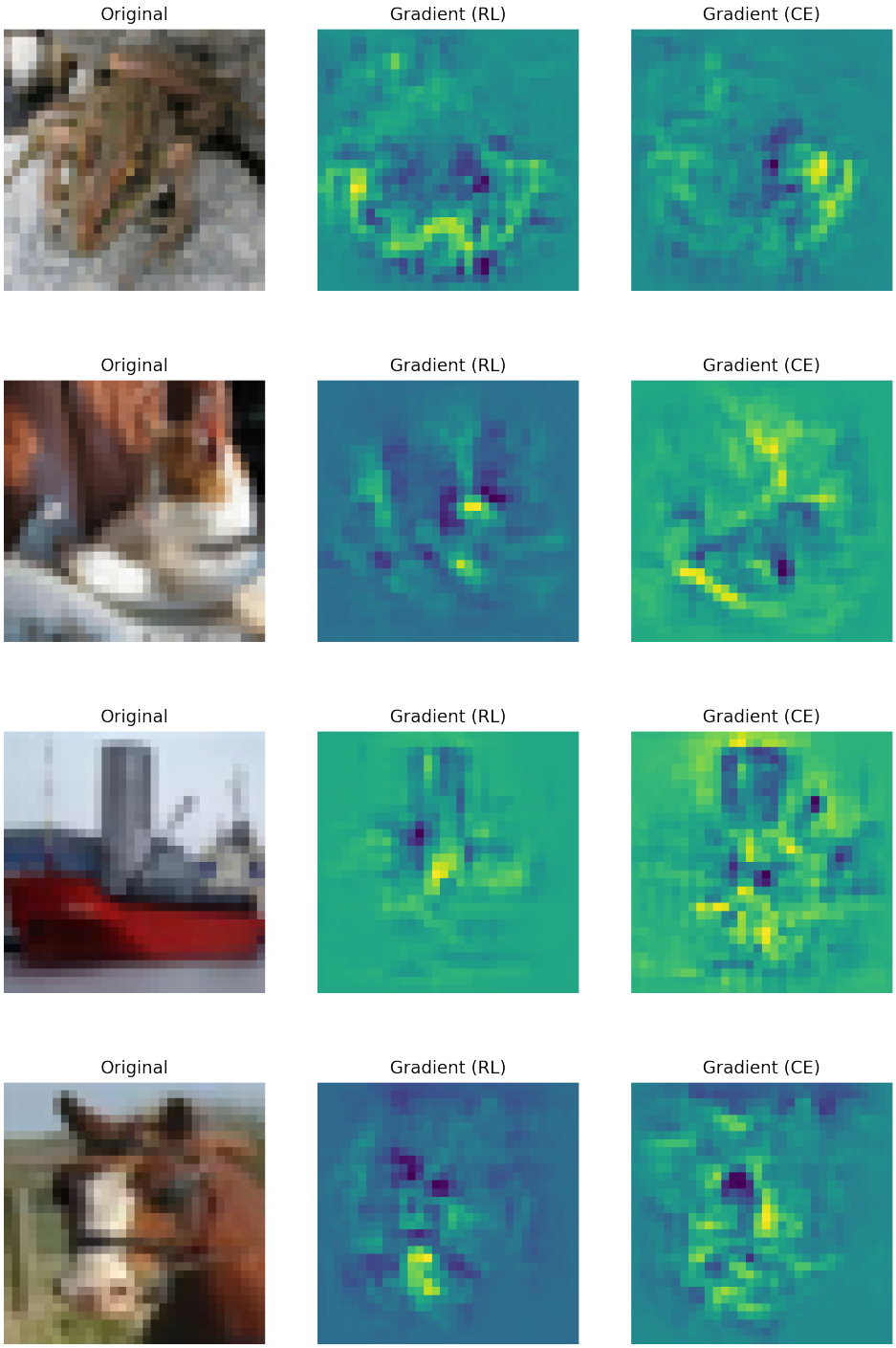
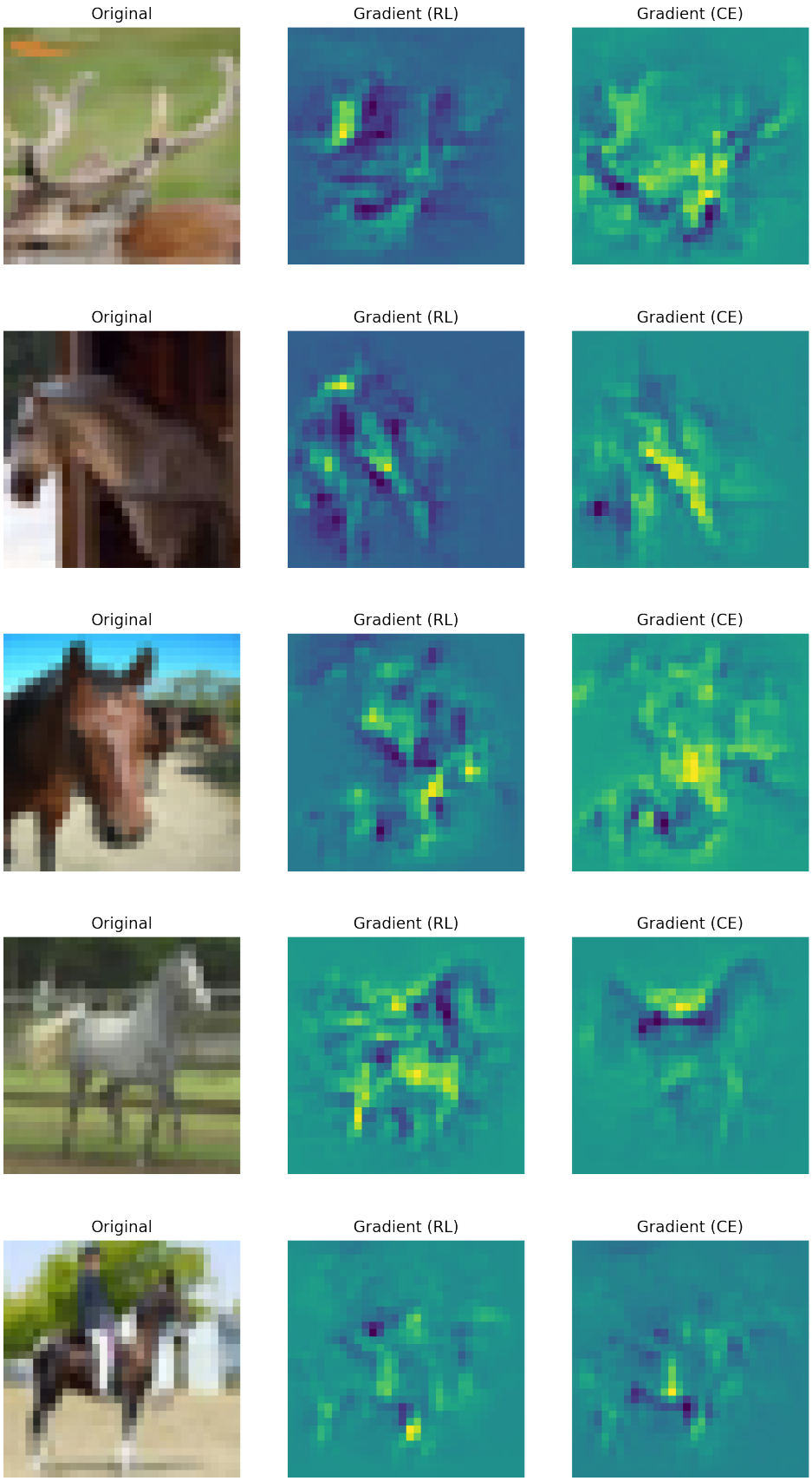


Figure 6: Adversarial accuracy (FGSM) vs epochs for $eps = 8/255$

C. Gradient Examples Calculated on the Both Model

To look further into the kind of features learned by both type of objective function. We checked the gradients calculated during FGSM attack on both the models. While for many of the examples we did not find any useful differences between the two method, some of the examples were worth adding here in which we did find difference among the two model.




Original
Gradient (RL)
Gradient (CE)
Original
Gradient (RL)
Gradient (CE)
Original
Gradient (RL)
Gradient (CE)
Original
Gradient (RL)
Gradient (CE)
Original
Gradient (RL)
Gradient (CE)