ATC: Adversarial aTtack for CIFAR-10

Junjie Ye
School of Computer Science
Fudan University
19307130140@fudan.edu.cn

Fubing Yang
School of Computer Science
Fudan University
19307130304@fudan.edu.cn

Abstract

Adversarial attack, which makes the model make wrong decisions in the reasoning process through adding small human imperceptible disturbances to original samples, is gradually valued by people. We conduct a black box attack on CIFAR-10, compare the impact of three common attack algorithms on the experimental performance, and further use the ensemble attack to improve the experimental effect. Finally, in the common pretrained target network, we reduce the classification accuracy of the model to 3%. However, we also find that in some pretrained target networks with low classification accuracy, our adversarial attacks will not play an obvious role, and may even improve the classification accuracy.

1. Introduction

In recent years, with the continuous development of deep learning technology, adversarial attack has become an important research direction. At present, scholars have proposed a variety of effective adversarial attack methods. At the same time, according to whether the attacker can obtain the internal structure information of the target model, the counter attack can be divided into white box attack and black box attack. In the white box attack, the attacker can use back propagation to calculate the gradient of the target model to generate countermeasure samples; while in the black box attack, the attacker cannot obtain the internal structure information of the target model, and can only generate adversarial samples by constructing a proxy network.

In order to compare and analyze the performance of different attack algorithms, we selected 200 CIFAR-10 pictures for black box attack, which is available at `https://drive.google.com/file/d/1fHi1ko7wr80wXkXpqpqpOxuYH1mClXoX/view`, and we randomly selected the well-known model pre trained on CIFAR-10 as the target network. Through experiments, we find that using FGSM [1] to attack the model can indeed have a certain effect and using I-FGSM [2] can further improve the attack performance. Finally, we verify the generalization ability of MI-FGSM [3], which reduces the classification accuracy from 90% to 40%. On this basis, we also use the Ensemble Attack method [4]. By selecting 15 pretrained networks as ensemble, we successfully reduce the classification accuracy of the model to 3%, which effectively improve the experimental performance.

During the experiment, we also found an unexpected experimental result, that is, for some target networks with low classification accuracy, the adversarial attack may not play a significant effect, and may even increase the classification accuracy. We suspect that this is because the network is very different from our proxy network in feature extraction, which is a direction that can be further studied.

In brief, the contribution of our experiment is:

- We compare and analyze the performance of different attack algorithms of attack tasks, and verify their effectiveness and generalization ability.
- We use the method of ensemble attack, which greatly improves the performance of attack.
- We find that adversarial attack has no significant effect on the models with low accuracy, which can be used as a new research direction.

2. Related Work

2.1. Attack methods

So far, scholars have proposed many methods to adversarial attacks. In order to make readers understand them enough, we show the attack methods used in the experiment below.

Fast Gradient Sign Method (FGSM) [1], which was introduced by Goodfellow et al., is a simple method for producing adversarial samples efficiently. The algorithm generates the adversarial perturbation by scaling the sign of the model's gradient by ϵ first, and adds it to the original image to form the adversarial sample. The adversarial image, $\hat{x}$, is defined by Equation 1, where $\nabla l(x)$ is the gradient of the model loss with respect to input x, and ϵ is a hyperparameter to control the intensity of the perturbation.

$$\hat{x} = x + \epsilon * \text{sign}\big(\nabla l(x)\big) \tag{1}$$

Iterative FGSM (I-FGSM), which is also referred to as the Basic Iterative Method, is a simple extension to the FGSM attack introduced by [2]. The algorithm applies FGSM repeatedly to produce a better targeted adversarial sample. The algorithm is defined in Equation 2, where $\widehat{x_n}$ indicates the adversarial sample produced by n steps of the method and α indicates the learning rate. Besides, the function Clip restricts the adversarial sample to remain within the ϵ-ball surrounding x.

$$\widehat{x_n} = \text{Clip}_{x,\epsilon}\left(\widehat{x_{n-1}} + \alpha * \text{sign}\left(\nabla l(x_{n-1})\right)\right) \quad (2)$$

Momentum I-FGSM (MI-FGSM) [3] was introduced by Dong et al. to enhance iterative attack transferability. The algorithm adds momentum to the I-FGSM, increasing the stability of the attack by reducing its susceptibility to being trapped in a local loss maximum. The algorithm is defined in Equation 3 and 4, where g_n indicates the momentum produced by n steps of the method and μ indicates the decay factor.

$$g_n = \mu * g_{n-1} + \nabla l(\widehat{x_{n-1}}) \quad (3)$$

$$\widehat{x_n} = \widehat{x_{n-1}} + \alpha * \text{sign}(g_n) \quad (4)$$

2.2. Ensemble Attack

Ensemble Attack [4], which was introduce by Liu et al., is not a specific attack algorithm, but a method of selecting proxy network. The method forms a proxy network by combining multiple networks and greatly improves the performance of attacks.

3. Dataset

We conduct our experiment on CIFAR-10 datasets. We selected 20 pictures from each of the given 10 classes as samples. Each picture is an RGB picture with a resolution of 32 by 32. For convenience, we download these pictures directly from the following website instead of the original website:
`https://drive.google.com/file/d/1fHi1ko7wr80wXkXpqpqpOxuYH1mClXoX/view`

4. Experimental Setup

Our experiment is generally divided into two stages. The architecture of our model can be seen in Figure 1. It is generally a common network structure of adversarial attacks. The variable part lies in the selection of attack methods and the construction of proxy networks.

In the first stage, we try to use different attack methods for single proxy attack, that is, there is only one proxy network while training. In this stage, we select three pretrained models on cifar-10 from Pytorchcv for cross validation. The models are resnext29_16x64d_cifar10, sepreresenet56_cifar10, resnet1202_cifar10 as described in [5, 6, 7]. For I-FGSM and MI-FGSM, we set the number of iterations to 30, which is determined empirically via grid search. We set the epsilon to about 0.15 to ensure that the human eye cannot recognize the disturbance of the picture. Meanwhile, for MI-FGSM, we set alpha to one tenth of epsilon and mu to 0.9. This is in line with the usual practice. Besides, we use L-infinity norm to constrain the level of attack.

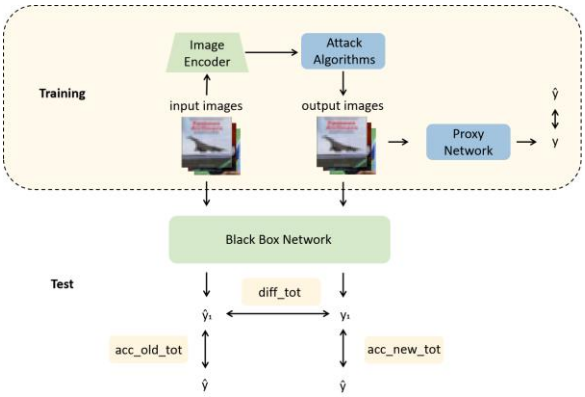

Figure 1: Illustration of the model.

In the second stage, we go further and use the idea of ensemble attack. We select 15 pretrained models on cifar-10 from Pytorchcv. The models are resnext29_16x64 d_cifar10, resnext29_32X4d_cifar10, preresnet56_cifar10, preresnet164bn_cifar10, seresnet110_cifar10, sepreresnet 56_cifar10, sepreresent110_cifar10, diaresnet56_cifar10, renet1001_cifar10, diapreresnet56_cifar10, resnet1202_ cifar10, resnet56_cifar10, diapreresnet110_cifar10 and resnet110_cifar10 as described in [5, 6, 7]. As you can see, the models we choose are generally based on Resnet or its variant. This is because their good performance in classification is recognized. We randomly select seven different networks as the target network, of which three are in the ensemble and four are outside it. The models are resnext29_16x64d_cifar10, sepreresnet56_cifar10, resnet 1202_cifar10, densenet100_k12_cifar10 [8], resnet272bn _cifar10, pyramidnet164_a270_bn_cifar10 [9], and ror_ 110_cifar10 [10]. Similarly, we used FGSM, I-FGSM, MI-FGSM as the attack methods. And the hyperparameters are also the same as those in the first stage. Meanwhile, we test the effect of different iterations on the results.

5. Results

5.1. Attack with one proxy network

In order to compare the performance of different attack algorithms, we do comparative experiments and get the results shown in Figure 2. From the result, we can clearly

Method	FGSM								
Target \ Source	resnext29_16x64d_cifar10			sepreresnet56_cifar10			resnet1202_cifar10		
	diff_tot	acc_old_tot	acc_new_tot	diff_tot	acc_old_tot	acc_new_tot	diff_tot	acc_old_tot	acc_new_tot
resnext29_16x64d_cifar10	——	——	——	67/200	162/200	122/200	66/200	162/200	134/200
sepreresnet56_cifar10	65/200	151/200	110/200	——	——	——	67/200	151/200	126/200
resnet1202_cifar10	46/200	178/200	136/200	61/200	178/200	127/200	——	——	——

Method	I-FGSM-30								
Target \ Source	resnext29_16x64d_cifar10			sepreresnet56_cifar10			resnet1202_cifar10		
	diff_tot	acc_old_tot	acc_new_tot	diff_tot	acc_old_tot	acc_new_tot	diff_tot	acc_old_tot	acc_new_tot
resnext29_16x64d_cifar10	——	——	——	87/200	162/200	105/200	96/200	162/200	94/200
sepreresnet56_cifar10	79/200	151/200	101/200	——	——	——	80/200	151/200	101/200
resnet1202_cifar10	72/200	178/200	117/200	79/200	178/200	110/200	——	——	——

Method	MI-FGSM-30								
Target \ Source	resnext29_16x64d_cifar10			sepreresnet56_cifar10			resnet1202_cifar10		
	diff_tot	acc_old_tot	acc_new_tot	diff_tot	acc_old_tot	acc_new_tot	diff_tot	acc_old_tot	acc_new_tot
resnext29_16x64d_cifar10	——	——	——	93/200	162/200	99/200	109/200	162/200	79/200
sepreresnet56_cifar10	93/200	151/200	88/200	——	——	——	105/200	151/200	81/200
resnet1202_cifar10	79/200	178/200	107/200	90/200	178/200	98/200	——	——	——

Figure 2: The result of attacking with one proxy network. “Source” indicates the proxy network. “diff_tot” indicates the numbers of pictures with different output results before and after the attack. “acc_old_atot” indicates the number of pictures correctly classified before the attack. “acc_new_tot” indicates the number of pictures correctly classified after the attack. The same below.

Methods	FGSM			I-FGSM-15			MI-FGSM-15			Type
Target	diff_tot	acc_old_tot	acc_new_tot	diff_tot	acc_old_tot	acc_new_tot	diff_tot	acc_old_tot	acc_new_tot	
resnext29_16x64d_cifar10	83/200	162/200	107/200	172/200	162/200	10/200	166/200	162/200	17/200	
sepreresnet56_cifar10	80/200	151/200	104/200	160/200	151/200	17/200	156/200	151/200	20/200	inside
resnet1202_cifar10	60/200	178/200	126/200	163/200	178/200	20/200	164/200	178/200	21/200	
densenet100_k12_cifar10	79/200	150/200	103/200	155/200	150/200	18/200	158/200	150/200	19/200	
resnet272bn_cifar10	81/200	154/200	98/200	164/200	154/200	16/200	161/200	154/200	19/200	outside
pyramidnet164_a270_bn_cifar10	68/200	161/200	116/200	157/200	161/200	24/200	160/200	161/200	25/200	
ror3_110_cifar10	82/200	157/200	104/200	157/200	157/200	19/200	159/200	157/200	21/200	

Figure 3: The result of attacking with ensemble methods. “15” indicates the number of iterations. “Type” indicates if the target network is in the ensemble list. The same below.

see that FGSM algorithm has certain performance under experimental conditions, but the results are poor. I-FGSM algorithm improves the performance of the algorithm to a certain extent, and MI-FGSM algorithm further enhances the model generalization ability. However, even after 30 iterations, the classification accuracy is only reduced to about 40%, which is not very good.

5.2. Attack with ensemble methods

On the basis of the first stage experiment, in order to improve the performance of the model, we adopt the ensemble attack method and conduct a comparative experiment again. The results are shown in Figure 3 to Figure 5.

As shown in Figure 3, after only 15 iterations, the accuracy of whatever type of target network is reduced to about 10%. Figure 4 and Figure 5 show the result with 20 iterations and 30 iterations. After 30 iterations, we can reduce the classification accuracy of the model to 3%, which is a high performance for black box attack. At this point, increasing the number of iterations will waste computing resources.

5.3. Interesting phenomenon

As shown in Figure 6, we find an interesting phenomenon during the experiment. In some models, the classification accuracy has been improved after attack. We think that this is because these models are not good enough for image feature extraction. Therefore, the attack will not have an important impact on the extracted features, or even conforms to the feature points extracted by the model.

Methods	FGSM			I-FGSM-20			MI-FGSM-20			Type
Target	diff_tot	acc_old_tot	acc_new_tot	diff_tot	acc_old_tot	acc_new_tot	diff_tot	acc_old_tot	acc_new_tot	
resnext29_16x64d_cifar10	83/200	162/200	107/200	174/200	162/200	7/200	172/200	162/200	11/200	
sepreresnet56_cifar10	80/200	151/200	104/200	167/200	151/200	12/200	163/200	151/200	13/200	inside
resnet1202_cifar10	60/200	178/200	126/200	172/200	178/200	12/200	173/200	178/200	12/200	
densenet100_k12_cifar10	79/200	150/200	103/200	172/200	150/200	7/200	169/200	150/200	9/200	
resnet272bn_cifar10	81/200	154/200	98/200	167/200	154/200	13/200	169/200	154/200	11/200	outside
pyramidnet164_a270_bn_cifar10	68/200	161/200	116/200	164/200	161/200	20/200	158/200	161/200	23/200	
ror3_110_cifar10	82/200	157/200	104/200	162/200	157/200	15/200	164/200	157/200	14/200	

Figure 4: The result of attacking with ensemble methods (20 iterations).

Methods	FGSM			I-FGSM-30			MI-FGSM-30			Type
Target	diff_tot	acc_old_tot	acc_new_tot	diff_tot	acc_old_tot	acc_new_tot	diff_tot	acc_old_tot	acc_new_tot	
resnext29_16x64d_cifar10	83/200	162/200	107/200	181/200	162/200	3/200	178/200	162/200	6/200	
sepreresnet56_cifar10	80/200	151/200	104/200	170/200	151/200	10/200	175/200	151/200	6/200	inside
resnet1202_cifar10	60/200	178/200	126/200	175/200	178/200	9/200	175/200	178/200	9/200	
densenet100_k12_cifar10	79/200	150/200	103/200	167/200	150/200	8/200	170/200	150/200	7/200	
resnet272bn_cifar10	81/200	154/200	98/200	170/200	154/200	10/200	175/200	154/200	6/200	outside
pyramidnet164_a270_bn_cifar10	68/200	161/200	116/200	166/200	161/200	17/200	162/200	161/200	19/200	
ror3_110_cifar10	82/200	157/200	104/200	166/200	157/200	14/200	167/200	157/200	12/200	

Figure 5: The result of attacking with ensemble methods (30 iterations).

Methods	FGSM			I-FGSM-30			MI-FGSM-30		
Target	diff_tot	acc_old_tot	acc_new_tot	diff_tot	acc_old_tot	acc_new_tot	diff_tot	acc_old_tot	acc_new_tot
diapreresnet56_cifar10	92/200	19/200	22/200	107/200	19/200	20/200	109/200	19/200	21/200

Figure 6: An example of model's accuracy increasing after attacked.

6. Conclusion

Our experiments mainly verify the performance of different attack algorithms, and prove the effectiveness of the attack algorithm, the superiority of I-FGSM algorithm and the generalization ability of MI-FGSM algorithm. At the same time, we use the ensemble attack method to greatly improve the attack performance, which proves the effectiveness of this method in the attack.

On the basis of this experiment, we can consider further more general migration attack experiments. We can also consider adding smaller disturbances to achieve stronger attack effect. Finally, corresponding to the adversarial attack, we can continue to study the relevant algorithms of defense.

References

[1] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In International Conference on Learning Representations, 2015. 1, 2

[2] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In International Conference on Learning Representations, 2016. 1, 2

[3] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In IEEE Conference on Computer Vision and Pattern Recognition, pages 9185–9193, 2018. 2, 3, 7

[4] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into Transferable Adversarial Examples and Black-box Attacks. In International Conference on Learning Representations, 2017.

[5] Saining Xie, Ross Girshick, Piotr Dollar, Zhuowen Tu, and Kaiming He. Aggregated Residual Transformations for Deep Neural Networks. In IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[6] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. Squeeze-and-Excitation Networks. 2017.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In IEEE Conference on Computer Vision and Pattern Recognition, 2016. 4

[8] Gao Huang, Zhuang Liu, and Laurens van der Maaten. Densely Connected Convolutional Networks. In IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[9] Dongyoon Han, Jiwhan Kim, and Junmo Kim. Deep Pyramidal Residual Networks. In IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[10] Ke Zhang, Miao Sun, Tony X. Han, Xingfang Yuan, Liru Guo, and Tao Liu. Residual Networks of Residual Networks: Multilevel Residual Networks. In IEEE Transactions on Circuits and Systems for Video Technology, 2017.