COMBATING FALSE SENSE OF SECURITY: BREAKING THE DEFENSE OF ADVERSARIAL TRAINING VIA NON-GRADIENT ADVERSARIAL ATTACK

Mingyuan Fan[1], *Yang Liu*[2*], *Cen Chen*[3], *Shengxing Yu*[4*], *Wenzhong Guo*[1] *and Ximeng Liu*[1]

[1]College of Computer and Data Science, Fuzhou University, China
[2]School of Cyber Engineering, Xidian University, China
[3]School of Data Science and Engineering, East China Normal University, China
[4]School of Electronics Engineering and Computer Science, Peking University, China

ABSTRACT

Adversarial training is believed to be the most robust and effective defense method against adversarial attacks. Gradient-based adversarial attack methods are generally adopted to evaluate the effectiveness of adversarial training. However, in this paper, by diving into the existing adversarial attack literature, we find that adversarial examples generated by these attack methods tend to be less imperceptible, which may lead to an inaccurate estimation for the effectiveness of the adversarial training. The existing adversarial attacks mostly adopt gradient-based optimization methods and such optimization methods have difficulties in searching the most effective adversarial examples (i.e., the global extreme points). On the contrast, in this work, we propose a novel Non-Gradient Attack (NGA) to overcome the above-mentioned problem. Extensive experiments show that NGA significantly outperforms the state-of-the-art adversarial attacks on Attack Success Rate (ASR) by 2% $\sim$ 7%.

Index Terms— adversarial attack, adversarial training, non-gradient attack

1. INTRODUCTION

Deep learning has obtained great achievements in various real-world problems, such as computer vision [1], natural language processing [2], and etc. [3, 4, 5]. Despite these impressive advances, it was found that Deep Neural Networks (DNNs) are extremely vulnerable to adversarial examples (AEs) [6, 7, 8, 9, 10]. AEs are generated by adding human-imperceptible and carefully crafted noises to natural examples to fool a well-trained model, especially for convolutional neural networks. Such vulnerability of DNNs extremely lowers the reliability of the application of DNNs in real-world applications [11, 12]. For instance, in self-driving, the vulnerability can potentially threaten a driver's life.

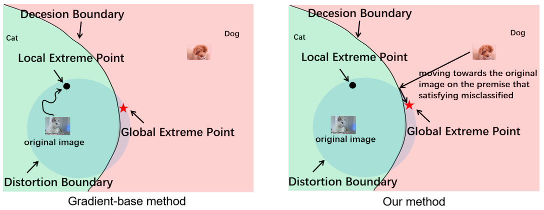


Fig. 1. The running process of NGA compared to gradient-based methods.

In past years, with extensive and in-depth studies of AEs, numerous defense methods are developed to enhance the robustness of DNNs. However, [13, 7, 14, 15, 16] pointed out that most existing defense methods only bring a false sense of security except adversarial training (AdvTrain) [17]. AdvTrain is a straightforward method implemented by adding AEs generated by gradient-based white-box attack methods, e.g., [17, 18, 19, 7], into the training process to increase the robustness of DNNs against AEs. Till now, AdvTrain still remains empirically robust.

To precisely assess the effectiveness of AdvTrain, a powerful attack method is necessary. The existing works mainly adopt *gradient-based white-box attack methods* to evaluate the effectiveness of AdvTrain [20, 21, 22]. However, previous works only suggest that AdvTrain is resilient to gradient-based attack methods (because there is no clear evidence to support that these methods are the most threatening). In addition, the gradient-based optimization methods have difficulties in searching global extreme points in high-dimensional space, also known as the curse of dimensionality [23, 24, 25]. Thus, we strongly believe that current evaluation methods overestimate the effectiveness of AdvTrain.

In this paper, we focus on examining whether AdvTrain is truly robust to adversarial attacks. To this end, we propose a powerful Non-Gradient Attack (NGA) method to better assess the robustness of DNNs with adversarial training. The NGA attack method aims to generate adversarial samples around intersection of the decision and distortion boundaries to max-

Thanks to the National Natural Science Foundation of China (No.62072109, No.U1804263) and Natural Science Foundation of Fujian Province (No. 2021J06013) for funding. *Corresponding author.

imize attack effectiveness of NGA.

The intuition for NGA is illustrated in Figure 1. As shown in this figure, given arbitrary x, the adversarial example of x must be at the intersection of decision boundary and distortion boundary (the red star denotes the adversarial example). Therefore, a powerful attack method should push the adversarial example to the intersection. However, existing gradient-based methods always fail, as the generated adversarial example may fall into a local extreme point (i.e., black dot) during the search process. To address this problem, NGA initializes the adversarial example outside the decision boundary and then moves the adversarial example towards the original image on the premise that the adversarial example will be misclassified. As moving towards the original example, adversarial example gradually approaches the distortion boundary. In this way, it is able to gradually approach the decision boundary rather than cross the decision boundary. Hence, after enough iterations, the adversarial example always can reach the intersection of the decision boundary and the distortion boundary, therefore, circumvent the drawback of gradient-based attack methods (i.e., stuck in local extreme point).

2. APPROACH

Overview. A threatening adversarial example has two characteristics: 1) which is misclassified by the target model, 2) and only has small difference to the corresponding natural example. In most of existing works, for crafting the adversarial example of x, the x itself was used as the initialization point to satisfy the second condition. Then, to meet the first requirement, move the example along a certain direction using a prepared optimization algorithm while ensuring the example satisfies the constraints. Nevertheless, due to the limited capacity of such optimization algorithms, the desired result is always not obtained. Hence, to overcome the drawback, we reverse the usual route of finding the adversarial example of x. For NGA, letting the adversarial example satisfies the first condition by proper initialization strategy and then moving the adversarial example to fulfill the second condition by crafted search (not based on gradient). Formally, NGA consists of three basic questions: 1) where to start the search, 2) where to forward each time, 3) and how much forward each time. In addition, the differences between NGA and gradient-based attack approach is demonstrated in Figure 2.

Initialization Strategy. In the most existing literature [19, 7], to craft the adversarial example of x, it is the most popular initialization strategy that x itself serves as the initialization point. But, NGA does not adopt the same initialization strategy as gradient-based attack methods. Specifically, the candidate set is introduced in NGA. The candidate set consists of some natural examples extracted randomly from the test set or other similar sets. When we want to generate the adversarial example for x, NGA randomly selects a natural example with the different label for x from the candidate set as the initialization point, which makes the initial adversarial examples lie outside the decision area with the label of the natural example as shown in Figure 2(a). Now, the initial adversarial example of x is misclassified by the model into the other label, satisfying the first condition. But it has extremely low similarity for the natural example. Therefore, we turn to construct a great direction to meet the second condition.

Step Direction. Apparently, blind search has trouble in gaining the ideal result and increases the overhead of NGA. Hence, we craft a reliable search direction, or step direction, to NGA, which significantly improves the search efficiency of NGA to the most threatening adversarial examples. In detail, given the attacked natural example $x \in R^{w\times h\times c}$ with ground-truth label y and the initial adversarial example $x_{adv} \in R^{w\times h\times c}$, a straightforward way is moving the x_{adv} along $(x - x_{adv})$ in each iteration:

$$x_{adv} = x_{adv} + \alpha(x - x_{adv}),\ \alpha \in (0, 1), \tag{1}$$

where α is step size given by the evaluator. There is a proof that x_{adv} iterated enough times can incredibly close-in x.

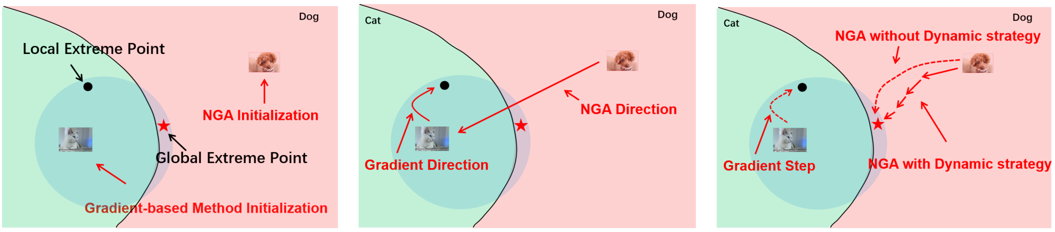
$$\begin{aligned} x^n_{adv} &= x^{n-1}_{adv} + \alpha(x - x^{n-1}_{adv}) \\ &= (1-\alpha)x^{n-1}_{adv} + \alpha \cdot x \\ &= (1-\alpha)^2 x^{n-2}_{adv} + ((1-\alpha)\alpha + \alpha) \cdot x \\ &= \cdots = (1-\alpha)^n x^0_{adv} + (1-(1-\alpha^n))x, \\ x^n_{adv} &= \lim_{n\to+\infty}((1-\alpha)^n x^0_{adv} + (1-(1-\alpha^n))x) = x, \end{aligned} \tag{2}$$

where x^i_{adv} denotes the value of x_{adv} in i-th iteration. In this way, x_{adv} iterated enough times can incredibly close-in x, i.e., $x_{adv} \approx x$ (the second condition of crafting adversarial examples). Notice that it is required to check x^n_{adv} being misclassified by the model in advance. If x^n_{adv} is misclassified by the model, Equation 1 is executed repeatedly until x_{adv} close to x; otherwise, we cannot execute this update due to violating the condition of being misclassified. In other words, NGA is stuck if $F_\theta(x^n_{adv}) \neq y$. We have to improve above direction for circumventing the deadlock. It is highlighted that decreasing the α cannot completely overcome the deadlock, because the update direction is constant in Equation 1. Specifically, there is:

$$\begin{aligned} x - x^n_{adv} &= x - x^{n-1}_{adv} - \alpha x + \alpha x^{n-1}_{adv} \\ &= (1-\alpha)x + (\alpha-1)\alpha x^{n-1}_{adv} \\ &= (1-\alpha)(x - x^{n-1}_{adv}), \end{aligned} \tag{3}$$

which suggests the $x - x^n_{adv}$ and $x - x^{n-1}_{adv}$ (update directions in adjacent iterations) have same direction.

Now, the crafted direction is desired to be diverse, while along the direction, x_{adv} can close to x. Specifically, a feasible solution is initializing a random vector as the initial direction and then reserving the elements of the random vector



(a) Initialization (b) Direction (c) Step Size

Fig. 2. Comparison between gradient-based attack method and NGA in initialization, search direction and step size.

with the same signs to $x - x_{adv}$ (others is set to 0) so that the direction not only guide the x_{adv} forwards x but also have substantial diversity. Formally, it is done by generating a random direction vector $z_0 \in R^{w \times h \times c}$ with certain distribution, such as Gaussian distribution firstly (w, h, c denotes the three dimensions of the x, namely width of the image, height of the image, and channel of the image). Afterward, computing difference vector between the attacked natural example and the adversarial example is conducted:

$$d = x - x_{adv}.$$

Finally, the values with the same symbol as the difference vector are reserved, and the others set to 0:

$$z_1 = z_0 \cdot I(d > 0),$$

$$I(d > 0)_{i,j,k} = \begin{cases} 1, & if \;\; d_{i,j,k} > 0 \\ 0, & if \;\; d_{i,j,k} \leq 0 \end{cases}.$$

The above process is implemented in each iteration:

$$x_{adv} = x_{adv} + \alpha \cdot z_0 \cdot I(d > 0). \tag{4}$$

Figure 2(b) compares the difference between the gradient-based method and NGA with respect to search direction.

Step Size. Generally, the step with fixed magnitude is the most intuitive solution. However, the magnitude of step is tough to adjust properly (similar to the learning rate). Concretely, small step size increases the overhead of NGA, and large step size may fail to obtain the desired result. Accordingly, to bypass the problem, NGA adopts the dynamic step size strategy. In detail, at the beginning of the iteration, the current example is far from the desired adversarial example, and then a relatively larger step size should be taken. Then, the step size gradually decays as the number of iterations increases to avoid missing the desired adversarial example. Therefore, we define the following formula:

$$\alpha = \frac{1}{\beta \cdot log(i) + 1}, \;\; i = 1, 2, \cdots, N,$$

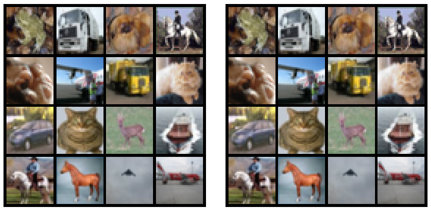
where α is step size with i^{th} iteration , β is hyperparameter utilized to control the decay speed, and N is the total number of iterations. A larger β yields a faster decay speed. If $\beta = 0$, α degrade into a constant (step with fixed magnitude). In summary, dynamic step size strategy improve convergence speed as shown in Figure 2(c).

Others. In practice, there are some inspired tricks to improve the efficiency of NGA further. A common and advanced trick is Restart generally adopted most attack methods [19, 7]. Restart refers to reinitialization when the desired solution cannot be found for a long time. We also import it into NGA to enhance performance further.

3. EXPERIMENT

3.1. Experiment Setup

In the experiments, we select 3 widely-used model architectures, namely PreActResNet18, GoogLeNet, and MobileNetV2 and the models trained with advanced adversarial training methods, PGD adversarial training (PGD AdvTrain) [20], Free adversarial training (Free AdvTrain) [21] and Fast adversarial training (Fast AdvTrain) [22]. For a fair evaluation, we follow all configurations in the latest adversarial training [22]. To validate the performance of NGA, we examine the performance of NGA over two benchmark datasets (CIFAR-10 and CIFAR-100), compared with 4 state-of-the-art adversarial attacks, i.e., FGSM [17], BIM [18], PGD [19], and C&W [7]. For the evaluation metric, we adopt the attack success rate (ASR), i.e., the misclassification rate of AEs, which is commonly used in evaluating the significance of AE attack. Generally, ASR can directly reflect the attack effectiveness of an AE method. In other words, the higher the ASR is, the stronger the attack method is.

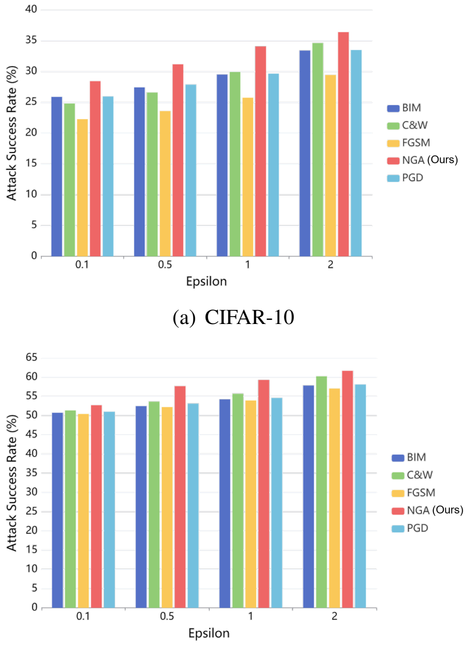


(a) Natural Examples (b) AEs

Fig. 3. The exhibit of the AEs generated by NGA compared to the corresponding natural examples.

Table 1. The ASRs (%) of the NGA with 10 Restarts and 10000 Iterations against the three networks with adversarial training (FGSM AdvTrain, FREE AdvTrain, and PGD AdvTrain) over different ϵ constraints.

Dataset		CIFAR-10				CIFAR-100			
Epsilon		0.1	0.5	1.0	2.0	0.1	0.5	1.0	2.0
FGSM AdvTrain	GoogLeNet	21.12	22.89	27.99	30.90	50.25	57.13	59.46	62.78
	MobileNetV2	27.00	28.90	31.54	36.25	52.64	58.22	59.20	61.53
	PreActResNet18	21.71	23.10	24.91	29.52	40.26	47.50	50.89	56.03
FREE AdvTrain	GoogLeNet	23.24	25.58	28.47	31.84	46.62	54.98	57.49	62.81
	MobileNetV2	35.26	36.74	38.49	41.24	40.64	49.78	53.62	58.48
	PreActResNet18	22.87	25.05	27.20	31.08	44.35	49.68	52.10	55.28
PGD AdvTrain	GoogLeNet	22.69	24.88	27.24	30.87	46.67	55.20	58.10	62.69
	MobileNetV2	30.81	32.88	34.63	38.39	40.99	50.11	53.78	58.53
	PreActResNet18	24.46	25.88	27.49	31.07	44.43	49.99	52.32	55.89



(a) CIFAR-10

(b) CIFAR-100

Fig. 4. The ASRs of NGA compared to state-of-the-art attack methods. The ASRs were measured over 500 AEs crafted by NGA with 10 Restarts and 10000 Iterations (the corresponding natural examples were extracted from the test datasets).

3.2. Comparison with State-of-the-Art

Figure 4 report the average ASRs of different attack methods against those models. Moreover, to intuitively illustrate attack effectiveness of NGA, we plot the several natural examples and corresponding AEs when ϵ set to 2 in Figure 3.

Overall, as shown in Figure 4, NGA achieves better estimations of the robustness of the models with AdvTrain than gradient-based attack methods and particularly for FGSM. In detail, AEs crafted by NGA are about 2%∼7% higher ASR than ones crafted by gradient-based methods on average, which indicates that the robustness of the models with AdvTrain is overestimated as described before. There is a reason why NGA can achieve more precise estimations. Note that C&W, except NGA, obtained the best estimation in both CIFAR-10 and CIFAR-100, because C&W adopts multi-step gradient iteration (compared to FGSM), Restarts (compared to BIM), and better loss function (compared to PGD), and all of the technologies are applied to avoid convergence to the local extreme points. Furthermore, compared to C&W, NGA is no longer based on gradient-based methods, suggesting that NGA can further evade the local extreme points. Therefore, the robustness of the models with AdvTrain can be estimated more accurately.

Model Architecture. Above all, in Table 1, we see that PreActResNet18 achieves the highest CC on CIFAR-10 and CIFAR-100 with FGSM AdvTrain and FREE AdvTrain, respectively. It implies that PreActResNet18 is the best model choice currently if we consider CC. Likewise, there is another piece of evidence to support the conclusion. Notice that, if we omit the result of PreActResNet18 with PGD AdvTrain in CIFAR-10, all experiment results provide a consistent sign that PreActResNet18 holds the best CC in any circumstance and does especially great in CIFAR-100.

4. CONCLUSION

In this paper, with the aim of examining the effectiveness of adversarial training, we proposed a novel adversarial attack method, dubbed Non-Gradient Attack, which overcomes the drawback of existing adversarial gradient-based attack methods. By comparing four state-of-the-art adversarial attack methods in the experiments, we showed that NGA achieves a better attack success rate, thereby precisely evaluating the performance of the model with adversarial training. Finally, we believe the foremost extension of our work is how to reduce the overhead of NGA. Because from the current point of view, NGA is still a relative brute-force approach and has to consume considerable resources. Therefore, reducing the overhead of NGA is our future work.

5. REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. Hinton, “Imagenet classification with deep convolutional neural networks,” in *NIPS*, 2012.

[2] B. Xu, R. Cai, Z. Zhang, X. Yang, Z. Hao, Z. Li, and Z. Liang, “Nadaq: Natural language database querying based on deep learning,” *IEEE Access*, vol. 7, pp. 35012–35017, 2019.

[3] S. Gu, E. Holly, T. Lillicrap, and S. Levine, “Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates,” in *IEEE International Conference on Robotics Automation*, 2017.

[4] V. Kepuska and G. Bohouta, “Next-generation of virtual personal assistants (microsoft cortana, apple siri, amazon alexa and google home),” in *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, 2018.

[5] Geert, Litjens, Thijs, Kooi, Babak, Ehteshami, Bejnordi, Arnaud, Arindra, and Adiyoso, “A survey on deep learning in medical image analysis,” *Medical Image Analysis*, 2017.

[6] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D Erhan, I. Goodfellow, and R. Fergus, “Intriguing properties of neural networks,” *Computer Science*, 2013.

[7] N. Carlini and D. Wagner, “Towards evaluating the robustness of neural networks,” *IEEE*, 2017.

[8] Chen Wan, Biaohua Ye, and Fangjun Huang, “Pid-based approach to adversarial attacks,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021, vol. 35, pp. 10033–10040.

[9] Satya Narayan Shukla, Anit Kumar Sahu, Devin Willmott, and Zico Kolter, “Simple and efficient hard label black-box adversarial attacks in low query budget regimes,” in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, pp. 1461–1469.

[10] Thibault Maho, Teddy Furon, and Erwan Le Merrer, “Surfree: a fast surrogate-free black-box attack,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 10430–10439.

[11] N. Akhtar and A. Mian, “Threat of adversarial attacks on deep learning in computer vision: A survey,” *IEEE Access*, vol. 6, pp. 14410–14430, 2018.

[12] Siddhant Garg, Adarsh Kumar, Vibhor Goel, and Yingyu Liang, “Can adversarial weight perturbations inject neural backdoors,” in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 2020, pp. 2029–2032.

[13] A. Athalye, N. Carlini, and D. Wagner, “Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples,” 2018.

[14] Nicholas Carlini, “Is ami (attacks meet interpretability) robust to adversarial examples?,” *arXiv preprint arXiv:1902.02322*, 2019.

[15] Logan Engstrom, Andrew Ilyas, and Anish Athalye, “Evaluating and understanding the robustness of adversarial logit pairing,” *arXiv preprint arXiv:1807.10272*, 2018.

[16] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok, “Synthesizing robust adversarial examples,” in *International conference on machine learning*. PMLR, 2018, pp. 284–293.

[17] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” *Computer Science*, 2014.

[18] A. Kurakin, I. Goodfellow, and S. Bengio, “Adversarial examples in the physical world,” 2016.

[19] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards deep learning models resistant to adversarial attacks,” 2017.

[20] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards deep learning models resistant to adversarial attacks,” 2017.

[21] Ali Shafahi, Mahyar Najibi, Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein, “Adversarial training for free!,” *arXiv preprint arXiv:1904.12843*, 2019.

[22] E. Wong, L. Rice, and J. Z. Kolter, “Fast is better than free: Revisiting adversarial training,” 2020.

[23] K. Fukumizu and S. Amari, “Local minima and plateaus in hierarchical structures of multilayer perceptrons,” *Neural Networks*, vol. 13, no. 3, pp. 317–327, 2000.

[24] Mario Köppen, “The curse of dimensionality,” in *5th Online World Conference on Soft Computing in Industrial Applications (WSC5)*, 2000, vol. 1, pp. 4–8.

[25] Gao Huang, Yixuan Li, Geoff Pleiss, Zhuang Liu, John E Hopcroft, and Kilian Q Weinberger, “Snapshot ensembles: Train 1, get m for free,” *arXiv preprint arXiv:1704.00109*, 2017.