Boosting the Transferability of Adversarial Attacks in Deep Neural Networks

Xu Xiaotang[1*], $WeiKangxin^{2}$, $XuJin^{3}$, $ZhuFangyi^{4}$, $ZhangJiayuan^{5}$

1 School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
2 Qian Weichang College, Shanghai University, Shanghai, 200444, China
3 College of Electronic Science and Engineering, Jilin University, Jilin, 130015, China
4 School of Big Data and Software Engineering, Chongqing University, Chongqing 400044, China
5 School of Information Science and Engineering, East China University of Science and Technology, Shanghai 200237, China
*Correspondence: 1022041113@njupt.edu.cn; Tel.:+86 19805186027

1 Abstract

This article presents a means of boosting the transferability of adversarial attacks in deep neural networks. The research includes background, methodologies, and outcomes, encompassing single-attack approaches and ensemble attack strategies such as I-FGSM and MI-FGSM. We delve into the notion of retraining using adversarial examples. Our contributions reveal the limitations of single-attack methods regarding transferability and demonstrate the superiority of ensemble attack methods. We highlight how algorithm selection impacts attack effectiveness and how model variations enhance transferability. Through these investigations, we offer valuable insights for bolstering deep neural networks' adversarial robustness while acknowledging existing constraints.

2 Introduction

With the development of deep neural networks (DNNs), remarkable success has been achieved in various fields, including face recognition, autonomous driving, and speaker verification. Meanwhile, more and more attention has been paid to DNN-related security issues. Goodfellow et al. [1] and Szegedy et al. [2] found that DNNs are vulnerable to adversarial examples, carefully crafted

perturbations to input data designed to fool DNNs into making incorrect predictions or classifications. Adversarial attacks pose security and reliability threats to DNN-based systems, especially in safety-critical applications such as autonomous driving, face recognition, and medical diagnosis.

An adversarial example refers to a sample created by introducing imperceptible perturbations, either visually imperceptible or slightly noticeable after processing, to an original data point in a dataset. Such examples can cause well-trained models to produce classification outputs with high confidence that differ from the classification of the original sample.

Some people have done some adversarial example research already. Szegedy [2] highlighted how minor disturbances can lead neural networks to misclassify images. Goodfellow [1] introduced the gradient-based method for generating adversarial samples, while Madry [9] further refined the approach. Liu et al. [3] proposed ensemble-based methods for Clarifai.com. Lin et al. [4] introduced NI-FGSM and SIM methods to enhance adversarial example transferability. Naseer et al. [5] presented a novel generative approach for transferable adversarial perturbations. Zhao et al. [6] devised practices for evaluating transfer adversarial attacks, categorizing attacks, and evaluating their transferability and stealthiness.

3 Team work

Name	Roles	Responsibilities
XU XIAOTANG	Leader	1. Project overall organization 2. Learning routes design 3.Experimentation 4. Results analysis 5.Algorithm optimization
WEI KANGXIN	Member	1. Papers reading 2. Codes comprehension 3.Algorithms summarization
XU JIN	Member	1. Papers reading 2. Article proofreading 3. Codes comprehension 4. Flowcharting
ZHANG JIAYUAN	Member	1. Papers reading 2. Experimentation 3.Results analysis 4. Algorithm optimization 5. Heatmaps Generation
ZHU FANGYI	Member	1. Papers reading 2. Proposal Presentation 3. Codes comprehension 4. Algorithm optimization

4 Methods and results

single-attack method Initially, we employed a single-attack approach; the Schematic diagram

of the single-attack principle is shown in Figure 1. We selected ImageNet as the source dataset and used a single model's [7–9] logits to compute the loss function. Different attack algorithms [4, 10–12] were applied to calculate perturbations, which were then added to the original input samples to generate adversarial examples that effectively attack the target model. We assessed the attack's transferability by comparing the model's predicted results on the original samples and adversarial examples. We adjusted the perturbations, attack parameters, and algorithms to enhance the attack's transferability.

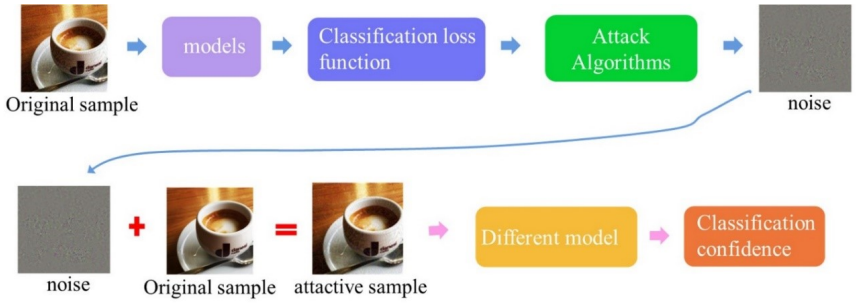

Figure 1: Schematic diagram of the single attack principle

These results represent the highest success rate obtained from iterating the attack a hundred times, as shown in Figure 2. Notably, using a single model for the attack yields relatively low transferability (success rate), with a maximum success rate of at most 80%. Consequently, we began considering better approaches to explore integrated adversarial attacks.

4.1 Ensemble attack methods

In our study, we delve into ensemble attacks, aiming to disrupt the classification performance of DNNs by combining multiple models. Our primary focus is understanding the generation of adversarial samples and assessing the transferability of adversarial attacks across different models.

When generating perturbations, we first employ the algorithm of I-FGSM. The Iterative Fast Gradient Sign Method (I-FGSM) [13] is a prominent technique used in adversarial attacks to perturb DNNs by introducing imperceptible changes into input data. It operates by iteratively modifying the input features based on the gradient information of the loss function concerning the input. I-FGSM belongs to the family of white-box attacks, where the attacker can access the target model's architecture and parameters.

As shown in Figure 3, we select an image from the dataset and feed it into various commonly used DNN models for classification. Each model produces a loss value, which we aggregate by summing the partial derivatives of the loss function concerning the clean samples, assigning equal

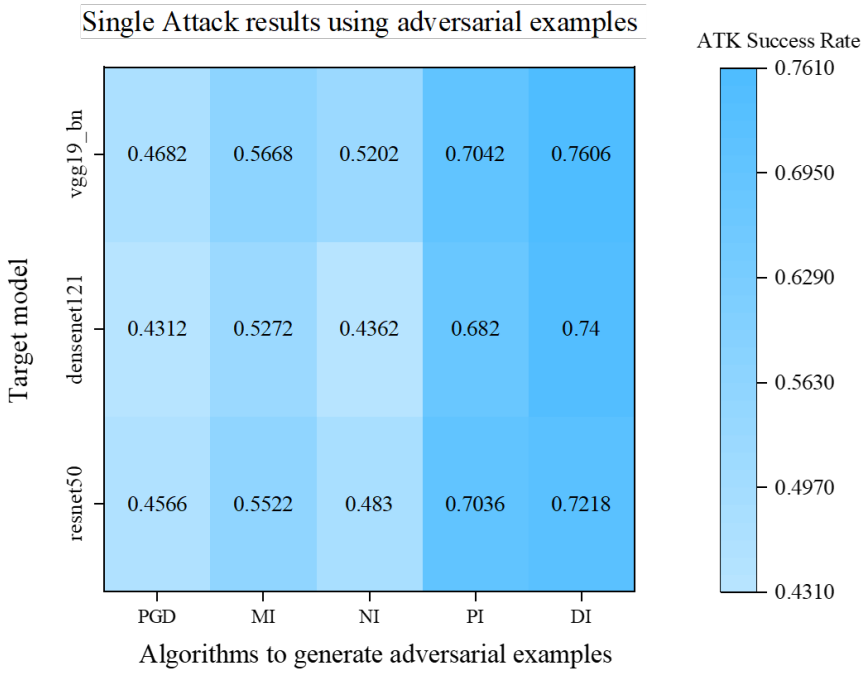

Figure 2: Single attack results

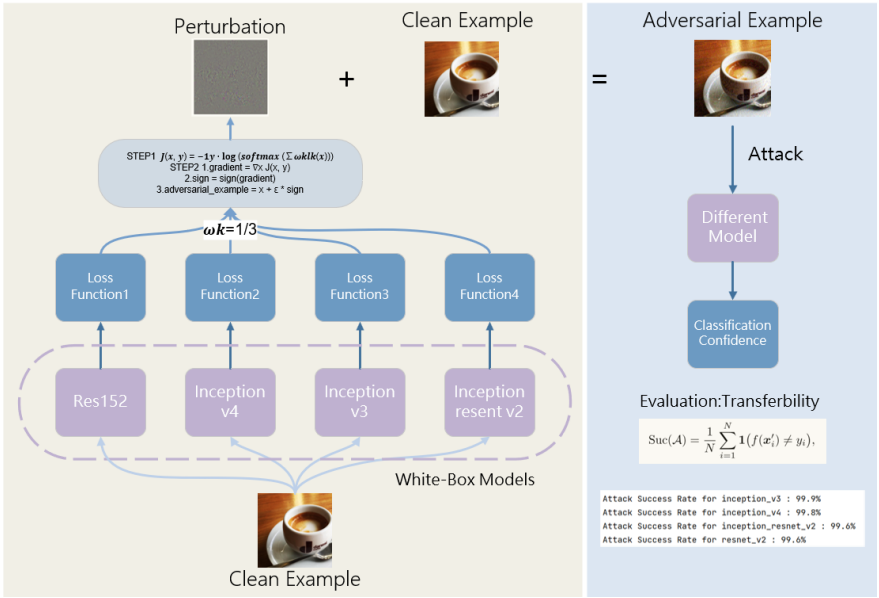

Figure 3: Application of the I-FGSM Algorithm in Ensemble Attack Mode Diagram

weights. This process creates adversarial samples designed to perturb the models' classification performance.

Our evaluation strategy involves inputting the generated adversarial samples into different models. A successful attack occurs if a model produces labels different from the original ones, indicating a misclassification of the image. We define the ratio of successfully deceived models to the total number of dataset images as the success rate. A higher success rate suggests better transferability of the attack method. Next, we analyze the attack success rates across various methods on different datasets.

The obtained results are presented in a heatmap, where colors represent the success rates of the attacks. To analyze the effectiveness of the ensemble attack, we initially exclude the res152 model, generate adversarial samples using the remaining three models, and then proceed to attack the res152 model. Notably, when targeting the res152 model, the success rate is markedly lower than the other models. Similar outcomes are observed when excluding the effects of the other models. Based on these findings, the transferability of the I-FGSM algorithm could be better.

4.2 MI-FGSM: An improvement of the I-FGSM algorithm

Since the transferability of the I-FGSM [10] algorithm is limited, we have employed the MI-FGSM algorithm to enhance it.

So What is the difference? The M introduces the concept of "Momentum" during each perturbation. Momentum can be understood as the accumulation of gradient information, from the loss function in previous iterations, much like inertia.

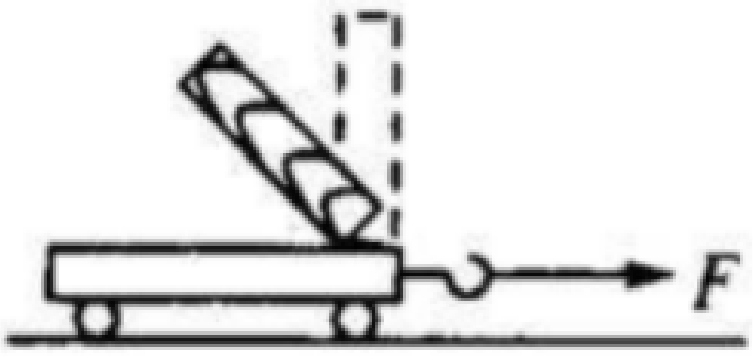

Figure 4: Inertia in the physical sense

In every iteration, it considers the current gradient and accumulated gradients from the past. This results in smoother adversarial perturbations, as it considers a richer history of information with each update. We use the variable "noise momentum" to perturb the input image x, which combines the momentum coefficient and the ensemble noise. After each perturbation, the momentum coefficient is updated. The specific calculation process is as follows:

1. Compute the gradients of the loss functions of multiple models concerning the input image x:$\frac{\mathrm{d}Li}{\mathrm{d}x}$ for each model i

2. Calculate the average of the gradients of multiple models to form the disturbance term noise ensemble

$$noise \quad ensemble = \frac{1}{N}\sum_{i=1}^{N}\frac{\mathrm{d}Li}{\mathrm{d}x}$$

3. Use the momentum coefficient β and the disturbance term noise ensemble to calculate noise momentum

$$noise \quad momentum = \beta \bullet momentum + \frac{noise \quad ensemble}{mean(|noise \quad ensemble|)}$$

Among them, β is the momentum coefficient, and momentum is the momentum term of the last iteration.

4. Multiply the noise momentum with the step size α, then take its sign and add it to the current input image x to produce adversarial samples

$$x = x + \alpha \bullet sign(noise \quad moment)$$

In this process, x represents the current input image, and each model's gradient of the loss function is calculated, averaged, and then combined with the momentum term to generate adversarial samples. This process is repeated over multiple iterations to generate increasingly disruptive adversarial samples that fool the model's predictions.

To better explain the algorithm, here is a metaphor: Imagine you are a mountain biker trying to pedal uphill on a slope called "loss function." I-FGSM is like adjusting your pedaling strength based only on how steep the slope looks in front of you. However, with MI-FGSM, you gather the energy you gained from your previous uphill efforts, which helps you with each pedal push. This way, you can climb more steadily and smoothly, as it considers both the current situation and your past progress.

After conducting several attacks similar to I-FGSM, we were pleasantly surprised to observe a significant increase.

In the attack, success rates against both black-box and defense models. Moreover, the success rate in white-box attacks nearly approached 100%! This indicates that the adversarial samples generated using the MI-FGSM algorithm exhibit strong transferability.

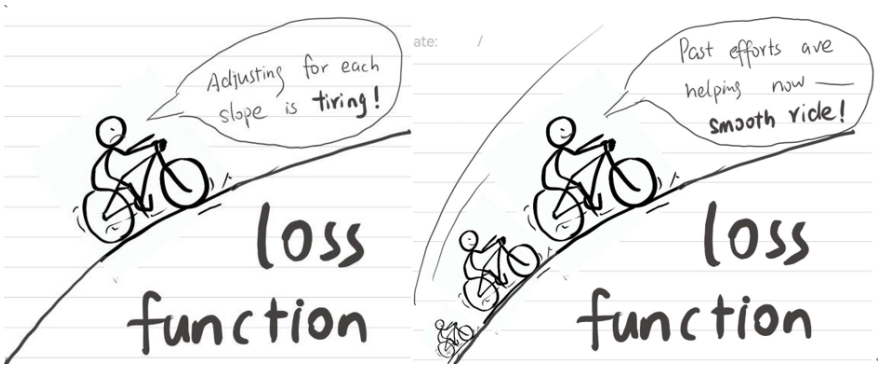

Figure 5: The superiority of MI-FGSM

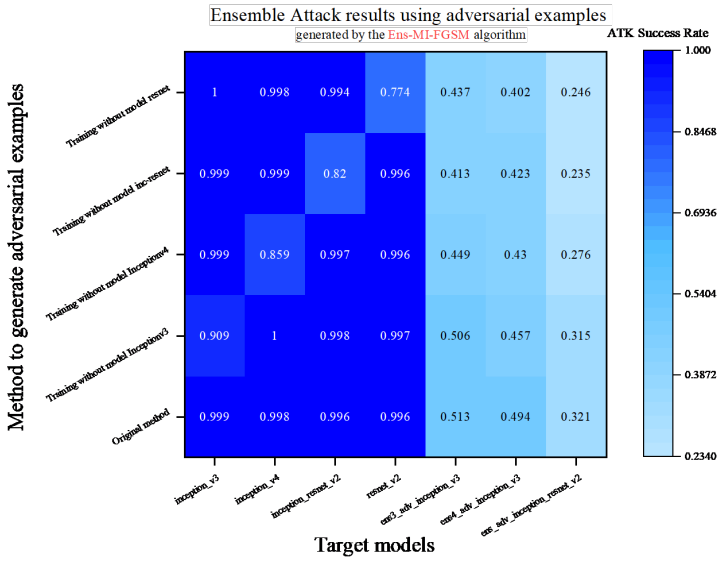

Figure 6: Ensemble Attack results using adversarial examples generated by the Ens-MI-FGSM algorithm

4.3 Further Thinking

After completing all the work mentioned, we took a step further into boosting adversarial example transferability.

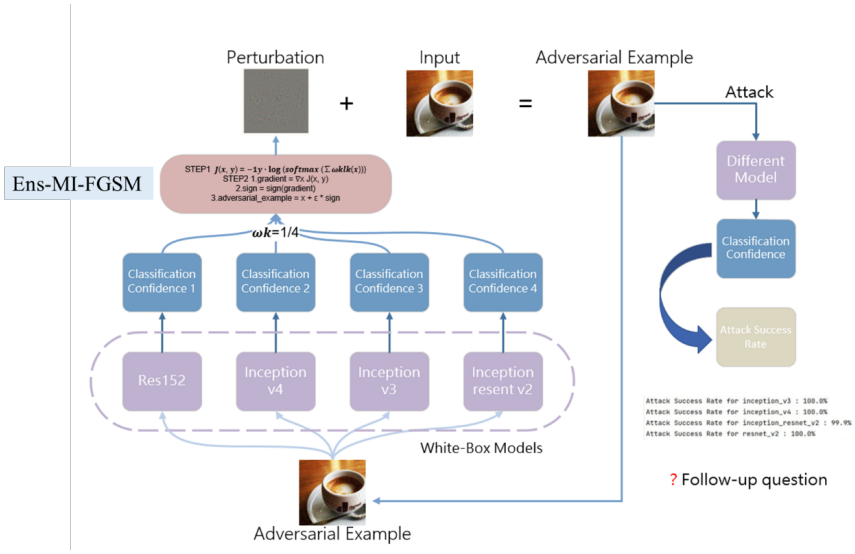

Figure 7: Algorithms flowchart on retraining using adversarial examples

In Figure 7, we describe our algorithm's flow. We use adversarial examples from previous attacks as inputs (denoted as x) and clean examples labels as classification targets (denoted as y). Using our improved method above-mentioned, Ens-MI-FGSM, we circulated this process after generating additional adversarial examples and evaluated their attack success rate against seven different models.

The result of their transferability is seen in Figure 8. When training with adversarial examples generated from a one-time attack, we found this approach led to further improvements upon the already promising results.

Figure 9, shows three heatmaps reflecting different methods we have used. These three columns within the red box represent the results of our adversarial examples attacking defense models: [14]Inc-v3ens3 Inc-v3ens4 IncRes-v2ens.Through analysis, we can find that the results using previous methods we used could be better, with only a few slightly over 50%. However, using an adversarial example retraining approach, only three times did the success rate against these defense models improve obviously, and some reached 57.4%.

The flowchart Figure 10. represents the idea of adversarial training. Models trained together with adversarial samples are added, allowing the model to learn the features of the clean examples and the adversarial ones, thus changing the model's decision boundary, and then improving the

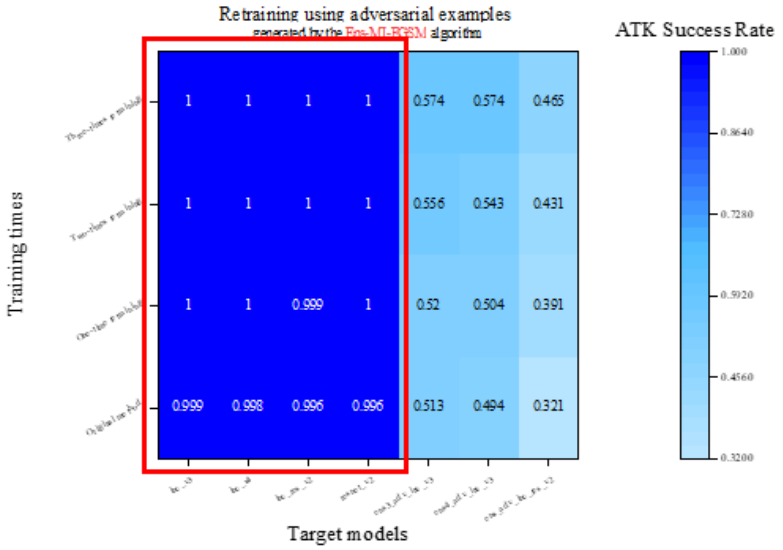

Figure 8: The superiority of retraining-Ens-MI-FGSM

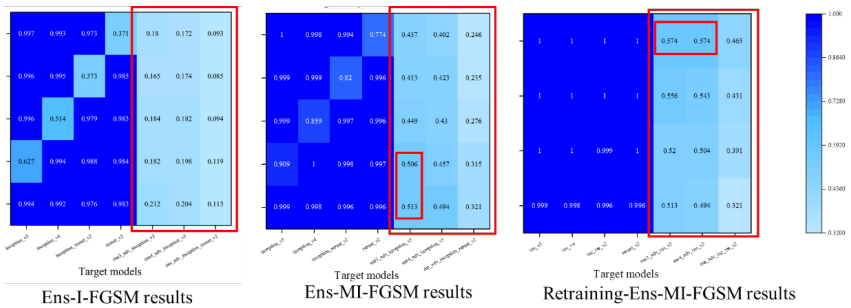

Figure 9: Comparison results of Ens-I-FGSM, Ens-MI-FGSM, and Retraining-Ens-MI-FGSM

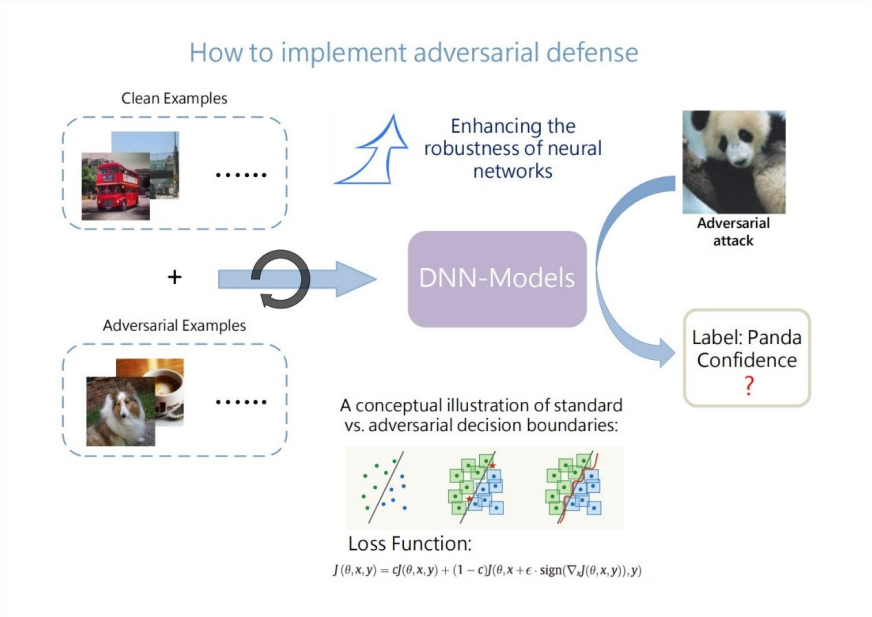

Figure 10: Flowchart of how defense model obtained through adversarial training

robustness of the model against attacks.

To find the underlying reason behind the improvement of the retraining approach, we speculate there may be similarities between this approach and adversarial training's changes to the model's decision boundary.

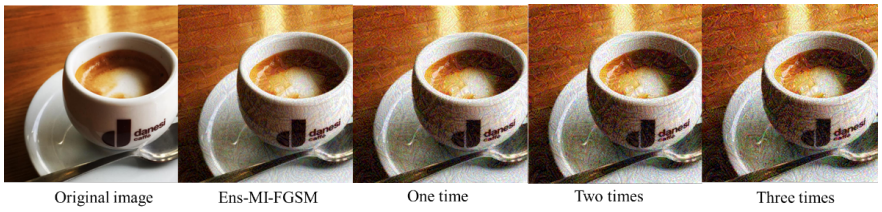

Figure 11: Adversarial examples retraining using adversarial examples

However, the approach we use has its limitations, or rather, the limitation is present in all approaches that enhance transferability. Specifically, methods with strong transferability tend to produce pictures with poorer stealthiness, which means higher visibility, illustrated in Figure 11. From left to right, the perturbations become increasingly evident. This is also one of the directions we can focus on in the future: how to balance transferability and stealthiness.

5 Contributions and Limitations

5.1 Contributions

This study found that ensemble attack methods outperform individual attack methods regarding black-box transferability. Ensemble attacks enhance the portability of adversarial examples by collecting the average gradient directions from multiple models as the update direction for gradient ascent. Neural networks with similar input-output relationships often have similar decision boundaries, thus exhibiting some similarity. By utilizing more models, we can better approximate this common direction. Furthermore, experimental results demonstrate that greater diversity among the models used in ensemble attacks leads to better transferability of generated adversarial examples.

Additionally, the choice of attack algorithm also impacts the attack's effectiveness. Under the premise of using the same models for ensemble attacks, employing superior attack algorithms can improve the transferability of adversarial samples.

Our work thoroughly investigates the factors influencing the effectiveness of ensemble attacks. In future work, we aim to understand better the inherent differences between neural network models with similar input-output relationships and train more advanced ensemble attack models.

5.2 Limitations

In our ensemble attack experiment, we only utilized four models for participation. However, the selection of models needed to be improved, leading to poor compatibility among the models used in the experiment. As a result, we could not fully explore the impact of differences between models on transferability. Furthermore, the addition of more models may enhance transferability in certain ways. However, due to current GPU technology limitations, we could not conduct large-scale training. These limitations are crucial to consider in the context of adversarial sample experiment-related papers.

Author Contributions

Topic proposer:X.X.; project instantiation: X.X.,W.K.,X.J.,Z.F.,Z.J.; investigation: X.X.,W.K.,X.J., Z.F.,Z.J.;algorithm collection and optimization:X.X.,Z.F.,Z.J.;algorithms summarization:W.K.,X.J.; experimentation:X.X,Z.J.;results analysis:X.X.,Z.F.Z.J.;visualization:X.J.,Z.J..All authors have read and agreed to the published version of the manuscript.

Funding

This research received no external funding.

Research Guidelines

This study has followed the research guidelines of Deep Learning and Neural Networks, Cambridge Summer Academic Programme 2023.

Informed Consent Statement
Not Applicable.

Data Availability
Not Applicable.

Acknowledgments
We would like to give our thanks to our faculty professors and teachers at Cambridge University for their guidance.

Conflicts of Interest
The authors declare no conflict of interest.

Intellectual Property
The authors attest that copyright belongs to them, the article has not been published elsewhere, and there is no infringement of any intellectual property rights as far as they are aware.

References

[1] Goodfellow, I., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. In International Conference on Learning Representations (ICLR).

[2] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2014). Intriguing properties of neural networks. In International Conference on Learning Representations (ICLR), Workshop Track.

[3] Liu, Y., Chen, X., Liu, C., & Song, D. (2017). Delving into transferable adversarial examples and black-box attacks. In International Conference on Learning Representations (ICLR).

[4] Lin, J., Song, C., He, K., Wang, L., & Hopcroft, J. E. (2019). Nesterov accelerated gradient and scale invariance for adversarial attacks in International Conference on Learning Representations (ICLR).

[5] Naseer, M., Khan, S., Hayat, M., Khan, F. S., & Porikli, F. (2021). On generating transferable targeted perturbations. In Proceedings of the IEEE/CVF International Conference on Computer Vision.

[6] Zhao, Z., Zhang, H., Li, R., Sicre, R., Amsaleg, L., & Backes, M. (2022). Towards Good Practices in Evaluating Transfer Adversarial Attacks. arXiv preprint arXiv:2211.09565.

[7] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708).

[8] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[9] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083.

[10] Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., & Li, J. (2018). Boosting adversarial attacks with momentum. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 9185-9193).

[11] Wu, B., Pan, H., Shen, L., Gu, J., Zhao, S., Li, Z., ... & Liu, W. (2021). Attacking adversarial attacks as a defense. arXiv preprint arXiv:2106.04938.

[12] Huang, S., Papernot, N., Goodfellow, I., Duan, Y., & Abbeel, P. (2017). Adversarial attacks on neural network policies. arXiv preprint arXiv:1702.02284.

[13] Kurakin, A., Goodfellow, I. J., & Bengio, S. (2018). Adversarial examples in the physical world. In Artificial intelligence safety and security (pp. 99-112). Chapman and Hall/CRC.

[14] Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., & McDaniel, P. (2017). Ensemble adversarial training: Attacks and defenses. arXiv preprint arXiv:1705.07204.