DANAA: Towards Transferable Attacks with Double Adversarial Neuron Attribution

Zhibo Jin[1], Zhiyu Zhu[1], Xinyi Wang[2], Jiayu Zhang[3], Jun Shen[4], and Huaming Chen[1(✉)]

[1] The University of Sydney, Camperdown, Australia
{zjin0915,zzhu2018}@uni.sydney.edu.au, huaming.chen@sydney.edu.au
[2] Jiangsu University, Zhenjiang, China
[3] Suzhou Yierqi, Changshu, China
[4] University of Wollongong, Wollongong, Australia

Abstract. While deep neural networks have excellent results in many fields, they are susceptible to interference from attacking samples resulting in erroneous judgments. Feature-level attacks are one of the effective attack types, which target the learned features in the hidden layers to improve their transferability across different models. Yet it is observed that the transferability has been largely impacted by the neuron importance estimation results. In this paper, a double adversarial neuron attribution attack method, termed 'DANAA', is proposed to obtain more accurate feature importance estimation. In our method, the model outputs are attributed to the middle layer based on an adversarial non-linear path. The goal is to measure the weight of individual neurons and retain the features that are more important toward transferability. We have conducted extensive experiments on the benchmark datasets to demonstrate the state-of-the-art performance of our method. Our code is available at: https://github.com/Davidjinzb/DANAA.

Keywords: Transferability · Adversarial attack · Attribution-based attack

1 Introduction

Deep neural networks (DNNs) have been used in a wide range of applications in different fields, such as face recognition [6], voice recognition [1] and sentiment analysis [30]. DNNs can also achieve state-of-the-art performance in tasks such as security verification in unconstrained environments where very low false positive rate metrics are required [6]. However, deep learning models are shown to be vulnerable to interference from adversarial samples. Attackers can manipulate the model outcome by deliberately adding the perturbations to the original samples to attack the models [28].

In general, the current approaches to attack models can be categorised into two types: white-box attack [12] and black-box attack [22]. For white-box attacks, the attacker knows the relevant parameters of the target model and can formulate

the most suitable attack method. For black-box attacks, on the other hand, the attacker does not have access to the model parameters. In terms of the characteristics of the white-box and black-box attack methods, the black-box attack provides the adversarial performance of the attacking samples, which is useful for improving the robustness of deep learning models in real-world scenarios. Specifically, the black-box attack methods have three types, including query-based method [14], transfer-based method [8] and hybrid method [9].

The objective of the query-based method is to interrogate the model to extract pertinent input or output information, and subsequently utilize this limited information to iteratively generate optimal adversarial samples. However, such method is subject to restrictions imposed by access permissions and often require multiple queries to obtain excellent adversarial samples. The transfer-based method aims to train and generate adversarial samples on a known-information local surrogate model, which are then transferred and tested on the target black-box model for the attack success rate. Compared to query-based methods, transfer-based methods do not require additional access to the model and can bypass certain adversarial defense mechanisms aimed at queries. The hybrid method combines the principles of query and transfer approaches. Although it can achieve sufficiently high attack success rate, it also implies that it is susceptible to adversarial defense mechanisms targeting both queries and transfers. Therefore, in this paper, we focus on transfer-based method.

As a common approach of transfer-based attack, feature-level attack attempts to maximise the internal feature loss by attacking intermediate layers' features to improve the transferability of the attack [32]. The aim is to increase the weight of negative features in the middle layer of the model while decreasing the weight of positive features. More negative features will be retained to assist the diversion of the model's predictions. However, it is still challenging to harmoniously differentiate the middle-level features via feature-level attack method, which is also prone to its local optimum [32]. Moreover, it is well-known that the effectiveness of transfer-based black-box attacks is influenced by the overfitting on surrogate models and specific adversarial defenses. To address these challenges, we propose to utilise the information of neuron importance estimation for the middle layer to identify the adversarial features more accurately. In addition, we also evaluate the transferability of our proposed method on adversarially trained models, which will be specifically discussed in Sect. 4. The results demonstrate that our method achieves favorable attack success rates even on target models protected by adversarial defenses.

To obtain adversarial samples with higher transferability, this paper presents a double adversarial neuron attribution attack (DANAA). DANAA method attributes the model outputs to the middle layer neurons, thus measuring individual neuron weights and retaining features that are more important towards transferability. We use adversarial non-linear path selection to enrich the attacking points, which improves the attribution results. Extensive experiments on the benchmarking datasets following the literature methods have been conducted. The results show that, DANAA can achieve the best performance for the adver-

sarial attacks. We anticipate this work will contribute to the attribution-based neuron importance estimation and provides a novel approach for transfer-based black-box attack. Our contributions are summarised as follows:

- We propose DANAA, an innovative method of non-linear gradient update paths to achieve a more accurate neuron importance estimation, for a more in-depth study of the route to attribution method.
- We present both theoretical and empirical investigation details for the attribution algorithm in DANAA, which is a core part of the method, in Sect. 3.
- A comprehensive statistical analysis is performed based on our benchmarking experiments on different datasets and adversarial attacks. The results in Sect. 4 demonstrates the state-of-the-art performance of DANAA method.

2   Related Work

In this section, we review the literature on white-box attacks, query-based black-box attacks, transfer-based black-box attacks, and hybrid black-box attacks.

2.1   Common White-Box Attacks

Previous work has demonstrated that neural networks are highly susceptible to misclassification by pre-addition of perturbed test samples. Such processed samples are called adversarial samples. The emergence of adversarial samples has led to the development of a range of adversarial defences to ensure the model performance [16, 28, 29].

Currently, adversarial attacks can be divided into white-box attacks and black-box attacks depending on the level of available information for the model being attacked. There are various approaches for white-box attacks, such as gradient-based and GAN-based. Gradient-based white-box attacks include FGSM [12], I-FGSM [16], PGD [20] and *C&W* [3]. Some recent GAN-based white-box attack methods are AdvGAN [33], GMI [37], KED-MI [4] and *Plug&Play* [24]. While white-box attacks are effective in measuring the robustness of a model under attack, in real-world scenario, the parameters of the model are often not accessible, leading to the development of black-box attacks.

2.2   Query-Based Black-Box Attacks

Query-based attacks are a branch of black-box attacks aiming to train an effective adversarial sample by performing a small-scale attack on the target model to query the model parameters, such as the model labels and confidence levels. These parameters can be used as part of the dataset to assist in training the migration algorithm to verify the migration of the black-box model. Ilyas et al. [14] were the first to propose a query-based black-box attack approach. Following, they proposed combining prior and gradient estimation of historical queries and data structures based on Bandit Optimization, which greatly reduces

the number of queries [15]. Li et al. [17] proposed a query-efficient boundary-based black box attack method (QEBA). It proved that the gradient estimation of the boundary-based attack over the entire gradient space is invalid in terms of the number of queries. Andriushchenko et al. [2] proposed the square search attack method, which selects local square blocks at random locations in the image to search and update the direction of the attack.

2.3   Transfer-Based Black-Box Attacks

The transferability of adversarial attacks refers to the applicability of the adversarial samples generated by the local model to the target model for attack. The attacker firstly uses the parameters obtained from the attack on the local model to train the adversarial samples, then uses these samples to perform a black-box attack on the target model to verify the success rate.

There are three main categories of transfer-based black-box attacks, namely gradient calculation methods, input transformation methods and feature-level attack methods. Gradient calculation methods such as MIM [7], VMI-FGSM [31] and SVRE [35] improve transferability by designing new gradient updates. Input transformation methods such as DIM [34], PIM [11] and SSA [18] boost the transferability by using input transformations to simulate the ensemble process of the model, while feature-level attacks focus on the middle-layer features.

Some state-of-the-art feature-level attack methods include NRDM, FDA, FIA and NAA, etc. NRDM [21] attempts to maximise the degree of distortion between neurons, but it does not take into account the role of positive and negative features in the attack. FDA [10] averages the neuronal activation values to obtain an estimate of the importance of a neuron. However, this method does not distinguish the degree of each neuron's importance and the discrimination between positive and negative features is still too low. FIA [32] multiplies the activation values of neurons and back-propagation gradients for estimation, but its effect on the original input is affected by over-fitting and the results are not accurate. NAA [36] effectively improves the transferability of the model and reduces computational complexity by attributing the model's output to an intermediate layer to obtain a more accurate importance estimation. However, its attribution method focuses more on the gradient iteration process considering linear path, and there is still room for improvement in the non-linear path condition.

2.4   Hybrid Black-Box Attacks

Hybrid method is a combination of query-based method and transfer-based method. It not only considers the priori nature of the transfer but also utilizes the gradient information obtained from the query, which resolves the challenges of high access cost for the query attack and low accuracy for transfer attack.

Dong et al. [5] proposed a hybrid method named P-RGF, which used the gradient of surrogate model as prior knowledge to guide the query direction of RGF and obtained the same success rate as RGF with fewer queries. Fu et al. [9] train Meta Adversarial Perturbation (MAP) on an surrogate model

and perform black-box attacks by estimating the gradient of the model, which has good transferability and generalizability. Ma et al. [19] introduced Meta Simulator to black-box attacks based on the idea of meta-learning. By combining query and transfer based attacks, the researchers not only significantly reduce the number of queries, but also reduce the complexity of queries by transferring the adversarial samples trained on the surrogate model to the target model.

While there are different types of black-box attack methods, transfer-based attacks is considered as the most convenient method which doesn't require additional information queries for the model. However, it poses the challenge of a good transferability for the adversarial samples. Therefore, in this work, we target the transfer-based attack methods. Especially, we introduce the attribution method for the middle-layer feature estimation, which shows a promising performance with our experiments.

3   Method

3.1   Preliminaries

When an adversarial attack to the target model can be successfully launched given an adversarial samples trained with a local DNN model, we consider there is a strong transferability relationship between these two models. Formally, with a deep learning network $N : R^n \to R^c$ and original image sample $x^0 \in R^n$, whose true label is t, if the imperceptible perturbation $\sum_{k=0}^{t-1} \triangle x^k$ is applied on the original sample x^0, we may mislead the network N with the manipulated input $x^t = x^0 + \sum_{k=0}^{t-1} \triangle x^k$ to the label of m, which can also be denoted as x^{adv}. Assuming the output of the sample x as $N(x)$, the optimization goal will be:

$$\left\| x^t - x^0 \right\|_n < \epsilon \quad subject \ to \quad N(x^t) \neq N(x^0) \tag{1}$$

where $\|\cdot\|_n$ represents the n-norm distance. Considering the activation values in the middle layers of network N, we denote the activation value of y-th layer as y and the activation value of j-th neuron as y_j.

3.2   Non-linear Path-Based Attribution

Inspired by [25] and [36], we define the attribution results of input image x^t(with $n \times n$ pixels) as

$$A := \sum_{i=1}^{n^2} \int \triangle x_i^t \frac{\partial N(x^t)}{\partial x^t} \mathrm{d}t \tag{2}$$

As shown in Fig. 1, different from the NAA algorithm [36], our paper proposes a new attribution idea that uses a non-linear gradient update path instead of the original linear path, which allows the model to find the optimal path against the attack itself. In Eq. 2, the gradient of N iterates along the non-linear path $x^t = x^0 + \sum_{k=0}^{t-1} \triangle x^k$, in which $\frac{\partial N}{\partial x^t}(\cdot)$ is the partial derivative of N to the i-th
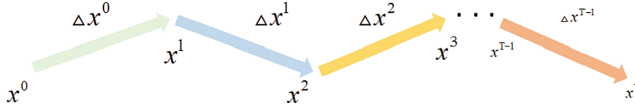
Fig. 1. Non-linear gradient update path diagram

pixel. For each iteration, $\triangle x^t = sign(\frac{\partial N(x^t)}{\partial x^t}) + N(0, \sigma)$. We further apply the learning rate and Gaussian noise to update the perturbation.

Afterwards, we can approximate A as $N(x)$ depending on basic advanced mathematics and extend the attribution results to each layer. The formula of attribution can then be expressed as:

$$A_{y_j} := \sum_{i=1}^{n^2} \int \triangle x_i^t \frac{\partial N(x^t)}{\partial y_j(x^t)} \frac{\partial y_j(x^t)}{\partial x^t} dt \tag{3}$$

where A_{y_j} represents the attribution of j-th neuron in the layer y, $\sum A_{y_j} = A$. We provide the relevant proof of our non-linear path-based attribution in following section.

3.3 Proof of Non-linear Path-Based Attribution

Since we now have A_{y_j} as Eq. 3, assuming that the neurons on the middle layer of the deep neural network are independent from each other, A_{y_j} can be expressed as

$$A_{y_j} := \int \frac{\partial N(x^t)}{\partial y_j(x^t)} \sum_{i=1}^{n^2} \triangle x_i^t \frac{\partial y_j(x^t)}{\partial x^t} dt \tag{4}$$

where $\frac{\partial N(x^t)}{\partial y(x^t)}$ is the gradient of$N(x^t)$ to the j-th neuron, $\sum_{i=1}^{n^2} \triangle x_i^t \frac{\partial y_j(x^t)}{\partial x^t}$ is the sum of the gradient of y_j to each pixel on $x^t (x^t \in R^n)$. Since the two gradient sequences are zero covariance, we then convert Eq. 4 into:

$$A_{y_j} := \int \frac{\partial N(x^t)}{\partial y_j(x^t)} dt \cdot \int \sum_{i=1}^{n^2} \triangle x_i^t \frac{\partial y_j(x^t)}{\partial x^t} dt \tag{5}$$

Combining the principles of calculus, we can prove that

$$\int \sum_{i=1}^{n^2} \triangle x_i^t \frac{\partial y_j(x^t)}{\partial x^t} dt = y_j^t - y_j^0 \tag{6}$$

then we denote $y_j^t - y_j^0$ as $\triangle y_j^t$, Eq. 5 can be converted into

$$A_{y_j} := \triangle y_j^t \int \frac{\partial N(x^t)}{\partial y_j(x^t)} dt \tag{7}$$

Denoting $\int \frac{\partial N(x^t)}{\partial y_j(x^t)} dt$ as $\gamma(y_j)$, which means the gradient of network N along our non-linear path with attention to the j-th neuron. Afterwards, we can get $A_{y_j} = \triangle y_j^t \cdot \gamma(y_j)$. Since the neuron y_j is on the middle layer y, finally the attribution result of the layer y can be expressed as

$$A_y = \sum_{y_j \in y} A_{y_j} = \sum_{y_j \in y} \triangle y_j^t \cdot \gamma(y_j) = \triangle y^t \cdot \gamma(y) \tag{8}$$

Algorithm 1. Double Adversarial Neuron Attribution Attack

Require: Deep network N, target layer y
Require: Manipulated input x^t with label m
Require: Perturbation budget ϵ and iteration number T
Require: Original input x^0 and integrated step τ
1: $\alpha = \frac{\epsilon}{T}$, $\gamma(y_j) = 0$, $g_0 = 0$, $\mu = 1$, $x_0^{adv} = x^t$
2: **for** $t = 0 \leftarrow \tau$ **do**
3:     $x^{t+1} = clip_x^\epsilon \{x^t + lr \cdot sign(\frac{\partial N(x^t)}{\partial x^t}) + N(0, \sigma)\}$
4:     $\gamma(y_j) = \gamma(y_j) + \triangledown_{y(x^t)} N(x^t)$
5: **end for**
6: **for** $s = 0 \leftarrow T - 1$ **do**
7:     $A_y = \triangle y^t \cdot \gamma(y)$
8:     $g_{s+1} = \mu \cdot g_s + \frac{\triangledown_{x^t} A_y}{\|\triangledown_{x^t} A_y\|_1}$
9:     $x_{s+1}^{adv} = Clip_{x^t}^\epsilon \{x_{s+1}^{adv} + \alpha \cdot sign(g_{s+1})\}$
10: **end for**

Algorithm 1 shows the specific pseudocode structure of our DANAA algorithm with Non-linear Path-based attribution.

4   Experiments

Extensive experiments have been conducted to demonstrate the efficiency of our method. Following sections cover the topic of leveraged datasets, benchmarking models and incorporated metrics. We also provide the experimental settings. We performed five rounds of benchmarking experiments to compare our algorithm with other methods, demonstrating the superiority of our approach to the baselines in terms of transferability for adversarial attacks. Moreover, we conducted the ablation study to investigate our approach, focusing on the impact of various learning rates and noise deviation on attack transferability.

4.1   Dataset

Following other literature methods, the widely-used datasets from NAA work [36] are considered in this paper. The datasets consist 1000 images of different categories randomly selected from the ILSVRC 2012 validation set [23], which we called a multiple random sampling(MRS) dataset.

4.2   Model

We include four widely-used models for image classification tasks, namely Inception-v3 (Inc-v3) [27], Inception-v4 (Inc-v4) [26], Inception-ResNet-v2 (IncRes-v2) [26], and ResNet-v2-152 (Res152-v2) [13], as source models for assessing the attacking performance of our algorithm. We start with four pretrained models without adversarial learning, which include Inc-v3, Inc-v4, IncRes-v2, and Res152-v2. Later on, we construct more robust models for a in-depth comparison, such as including adversarial training for the pretrained models. This results in two adversarial trained models, including Inception-v3(Inc-v3-adv) and Inception-Resnet-v2 (IncRes-v2-adv) [16]. The remaining three models are based on the ensemble models: the ensemble of three adversarial trained Inception-v3(Inc-v3-adv-3), the ensemble of four adversarial trained Inception-v3 (Inc-v3-adv-4), and the ensemble of three adversarial trained Inception-Resnet-v2 (IncRes-v2-adv-3), following the work from [29]. In [29], the models are combined by training the sub-models of the corresponding model independently and finally weighting the results of each sub-model to increase the accuracy and robustness of the model.

4.3   Evaluation Metrics

The attack success rate is selected as the metric to evaluate the performance. It measures the proportion of the dataset where our method produces incorrect label predictions after attacking. Hence, a higher success rate indicates improved performance of the attack method.

4.4   Baseline Methods

For comparison in our experiment, we selected five state-of-the-art attack methods as the baseline, including MIM [7], NRDM [21], FDA [10], FIA [32], and NAA [36]. Furthermore, to test the effect of each model after combining input transformation methods and to verify the superiority of our algorithm, we apply both DIM and PIM to the attack methods. The implementation details can be found in the open source repository. Consequently, we extend the model comparison set with MIM-PD, NRDM-PD, FDA-PD, FIA-PD, NAA-PD and DANAA-PD, respectively.

4.5   Parameter Setting

In the experiment, we set the parameters as following: the learning rate (lr) is 0.0025; the noise deviation is 0.25; and the maximum perturbation rate is 16, which is derived from the number of iterations (15) and the step size (1.07). The batch size is 10, and the momentum of the optimization process is 1. Since we introduced the DIM and PIM algorithms to verify the superiority of our model when combining input transformation methods, we set the transformation probability of DIM to 0.7, and the amplification factor and kernel size of PIM are 2.5

and 3, respectively. For the target layer of the attack, we choose the same layer as in NAA. Specifically, we attack InceptionV3/InceptionV3/Mixed_5b/concat layer for Inc-v3; InceptionV4/InceptionV4/Mixed_5e/concat layer for Inc-v4; InceptionResnetV2/Ince-ptionResnetV2/Conv2d_4a_3x3/Relu layer for IncRes-v2; the ResNet-v2-152/blo-ck2/unit_8/bottleneck_v2/add layer of Res152-v2 [36].

4.6   Result

All the experiments are carried out with the hardware of RTX 2080Ti card. A detailed replication package can be found in the open source repository at https://github.com/Davidjinzb/DANAA. We subsequently compile the results of all the attack methods without and with the input transformation methods (ending with PD) in Table 1.

In Table 1, we can see that, DANAA has retained a strong and robust performance across all the models, in comparison with other attack methods. Especially, DANAA demonstrated notable improvements on five models that are adversarial trained. We can observe a largest improvement of the attacking performance is between our method and NAA method [36], which is the generally second best attacking method in the comparison experiments. The ratio of improvement is 9.0%. Across all local models, our approach demonstrated an overall average improvement of 7.1% as compared to NAA on the adversarial trained models. By introducing the PD concept, our method achieves a maximum improvement of 9.8% over NAA-PD and an overall average improvement of 7.3% on the adversarial trained models.

4.7   Ablation Study

In this section, we investigate the impact of the learning rate and Gaussian noise deviations on the performance of the proposed method.

The Impact of Learning Rates. Experiments are conducted using different scales of learning rates, which are 0.25, 0.025, 0.0025 and 0.00025. In Fig. 2, the DANAA method exhibits the highest attack success rate for nearly all models when the selected learning rate was 0.0025. In Fig. 3, the highest attack success rates are achieved on most models for DANAA-PD method.

Notably, when using Inception-ResNet-v2 as the source model, although at a learning rate of 0.0025 DANAA-PD ranked second best in attack success rate on the models without adversarial training, its effectiveness on the model with adversarial training is still much higher than those at other learning rates.

The Impact of Gaussian Noise Deviation (Scale). To verify the effect of adding Gaussian noise to the gradient update on model transferability in this paper, we selected different noise deviations for testing in this subsection. As shown in Fig. 4 and Fig. 5, five different scales of the Gaussian noise deviation

ranging from 0.2 to 0.4 are used in this experiment. In general, higher value of scale tends to have more superior results for the normal training model while sacrificing performance for the more robust one. Conversely, a lower value of scale results in less improvements for the normal trained model but better performance for the adversarial trained model. Accordingly, the scale value of 0.25 is selected for the optimal performance in this paper.

Table 1. Attack success rate of multiple methods on different models

Model	Attack method	Inc-v3	Inc-v4	IncRes-v2	Res152-v2	Inc-v3-adv	IncRes-v2-adv	Inc-v3-adv-3	Inc-v3-adv-4	IncRes-v2-adv-3
Inc-v3	MIM	**100**	41.9	39.7	32.8	22.1	18.4	14.9	15.7	8.2
	NRDM	90.4	61.4	52.5	49.9	26.1	19.2	9.5	12.9	4.7
	FDA	81.7	42.9	37.1	35.1	19.4	12.6	9.3	12.2	5.0
	FIA	96.5	79.1	77.8	71.8	54.8	53.9	43.1	44.2	23.2
	NAA	97.0	83.0	80.6	74.7	56.2	59.4	49.5	50.4	31.5
	DANAA	98.1	**86.8**	**84.8**	80.3	**64.4**	**68.4**	**55.4**	**56.5**	**33.1**
Inc-v4	MIM	58.2	**99.9**	45	40.4	23.5	20.4	17.7	20.3	9.7
	NRDM	78.0	96.4	62.8	62.3	26.1	25	17.3	16.6	6.8
	FDA	84.6	99.6	71.8	68.8	28.2	26.1	17.4	17.1	7.0
	FIA	74.6	91.0	69.6	65.7	43.5	47.3	39.3	39.9	23.5
	NAA	83.3	95.8	77.9	73.3	49.5	53.2	48.0	46.5	31.4
	DANAA	**86.8**	97.2	**82.4**	**76.9**	**54.9**	**61**	**53.8**	**53.5**	**35**
IncRes-v2	MIM	60	51.9	**99.2**	42.2	25.9	30.5	21.7	23.3	12.3
	NRDM	72.8	67.9	77.9	59.7	35.7	30.8	16.4	17.1	7.3
	FDA	69.0	68.0	78.2	56.2	34.5	29.7	16.2	15.4	7.7
	FIA	71.0	68.2	78.8	63.9	53.8	56.4	47.4	45.8	37.6
	NAA	79.5	76.4	89.3	71.1	60.3	64.8	56.9	55.0	47.3
	DANAA	**82.7**	**80.4**	91.5	**77.7**	**66.3**	**72.2**	**64.7**	**60.8**	**56**
Res152-v2	MIM	52.9	47.3	44.9	**99.4**	26.6	25.1	24.3	24.4	13.3
	NRDM	72.7	68.8	59.5	89.9	39.1	31.0	20.3	18.1	9.3
	FDA	15.7	9.2	8.3	26.2	13.1	6.8	9.3	9.7	4.0
	FIA	80.7	78.2	77.5	98.0	58.5	58.2	53.0	48.4	34.4
	NAA	84.7	83.5	82.3	97.6	61.8	67.0	59.1	58.1	46.1
	DANAA	**86.4**	**86.8**	**85.9**	98.8	**68.1**	**71.7**	**65.1**	**62.0**	**48.4**
Inc-v3	MIM-PD	**99.7**	72.8	66.9	54.1	31.7	29.1	20.2	21.7	9.7
	NRDM-PD	86.3	68.6	64.3	58.0	31.1	22.6	10.6	13.8	5.9
	FDA-PD	74.7	49.3	46.5	40.9	23.7	15.4	10.5	13.1	6.2
	FIA-PD	96.9	83.5	82.7	79.8	61.4	62.1	47.0	48.2	27.5
	NAA-PD	97.2	87.0	85.6	81.1	64.9	65.8	53.4	51.6	33.6
	DANAA-PD	97.9	**89.4**	**89.4**	**84.8**	**70.6**	**72.3**	**61.7**	**60.9**	**40.1**
Inc-v4	MIM-PD	81.3	**99.4**	71.0	59.7	31.6	28.0	22.9	23.3	12.7
	NRDM-PD	90.3	97.0	79.5	76.8	34.1	34.4	21.1	19.7	8.6
	FDA-PD	93.2	99.2	86.4	82.4	36.7	37.4	20.3	21.1	10.0
	FIA-PD	84.0	92.4	81.2	77.1	55.2	58.6	48.9	47.5	29.3
	NAA-PD	**90.5**	96.9	87.6	83.9	58.4	64.3	54.0	53.4	34.6
	DANAA-PD	90.4	96.5	**87.9**	**84.9**	**63.9**	**71.0**	**61.9**	**60.1**	**42.7**
IncRes-v2	MIM-PD	80.7	76.5	**98.0**	65.8	36.9	42.7	29.4	28.6	17.1
	NRDM-PD	76.4	74.1	78.7	64.1	40.7	32.4	17.5	18.8	6.7
	FDA-PD	78.1	76.2	80.7	66.5	41.3	35.6	18.4	17.0	7.6
	FIA-PD	76.5	73.4	81.7	71.1	60.0	62.5	50.3	47.0	36.4
	NAA-PD	81.4	78.2	89.9	76.4	65.2	67.7	59.9	57.1	46.0
	DANAA-PD	**83.7**	**80.4**	89.8	**80.6**	**70.3**	**73**	**65.8**	**63.1**	**55.8**
Res152-v2	MIM-PD	81.5	77.5	76.2	**99.4**	41.5	44.5	34.8	33.6	18.4
	NRDM-PD	84.1	82.1	73.1	90.1	51.6	43.5	28.3	22.5	11.2
	FDA-PD	22.1	12.7	11.4	23.4	19.6	10.4	9.9	11.7	5.4
	FIA-PD	88.6	86.1	87.0	98.3	70.9	71.0	63.6	58.6	43.4
	NAA-PD	90.2	88.5	89.0	98.0	73.5	76.1	70.3	66.3	52.2
	DANAA-PD	**92.0**	**91.7**	**91.8**	98.7	**79.3**	**82.1**	**76.1**	**73.4**	**60.8**
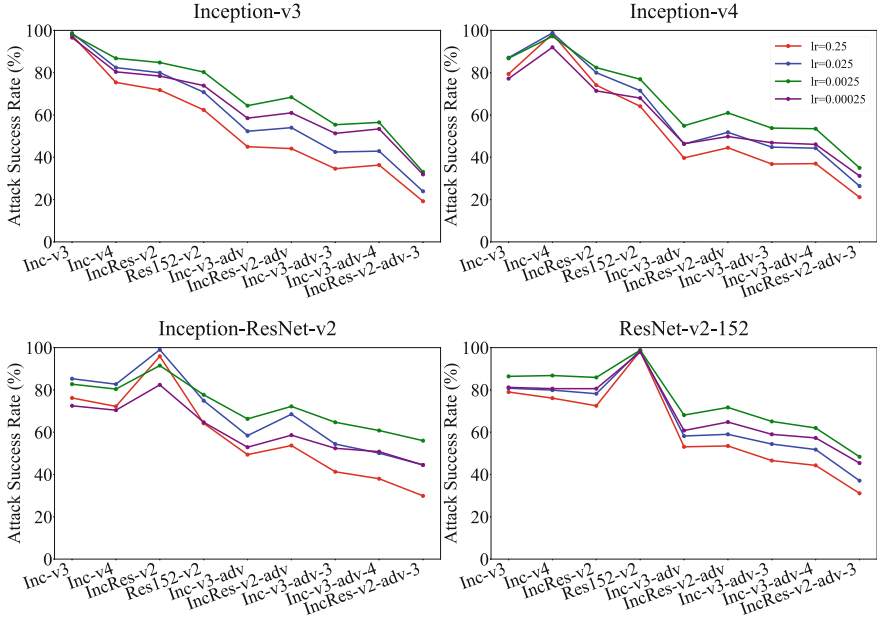
Fig. 2. DANAA attack success rate performance at different learning rates
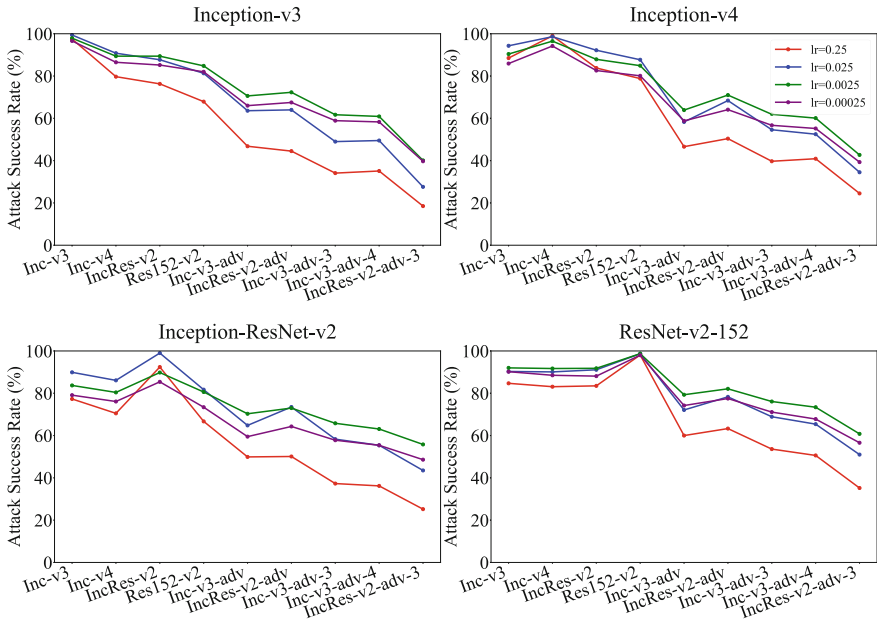
Fig. 3. DANAA-PD attack success rate performance at different learning rates
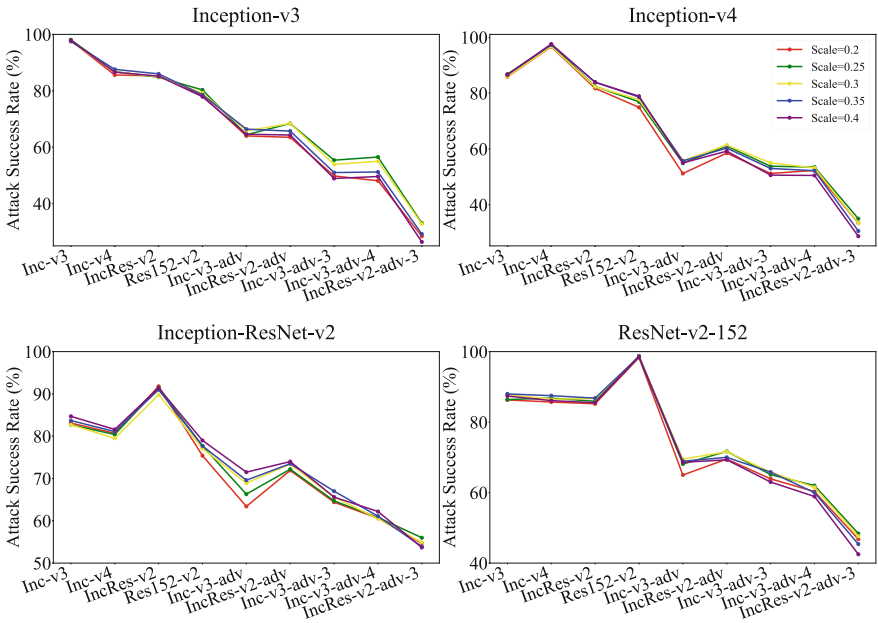
Fig. 4. DANAA attack success rate performance at different noise deviation
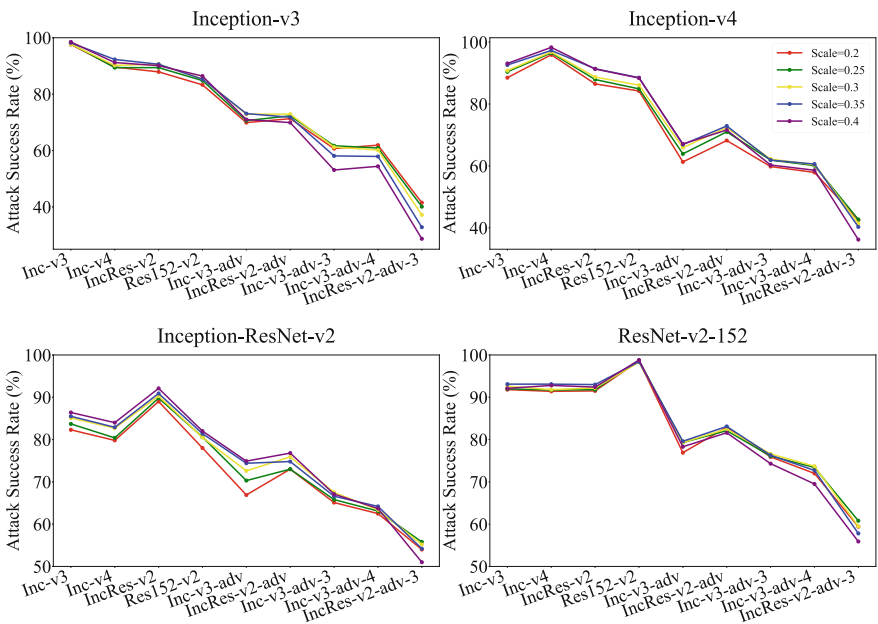

Fig. 5. DANAA-PD attack success rate performance at different noise deviation

5   Conclusion

In this paper, we propose a double adversarial neuron attribution attack method (DANAA) to achieve enhanced transfer-based adversarial attack results. Compared with other literature methods, our method obtains a better transferability for the adversarial samples. To derive more accurate importance estimates for the middle layer neurons, we firstly employ a non-linear path to the perturbation update process. Considering the calculation of gradient on the non-linear path, for all examined models, the performance of DANAA algorithm has substantially improved by up to 9.0% in comparison with the second best method with adversarial trained models, and has an average overall improvement by 7.1%. With the information transformation methods of DIM and PIM, our DANAA-PD algorithm also has a maximum enhancement of 9.8% and an average overall improvement of 7.3% compared to NAA-PD algorithm. Extensive experiments have demonstrated that the attribution model proposed in this paper achieves the state-of-the-art performance, with greater transferability and generalisation capabilities.

References

1. Aizat, K., Mohamed, O., Orken, M., Ainur, A., Zhumazhanov, B.: Identification and authentication of user voice using DNN features and I-vector. Cogent Eng. **7**(1), 1751557 (2020)
2. Andriushchenko, M., Croce, F., Flammarion, N., Hein, M.: Square attack: a query-efficient black-box adversarial attack via random search. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020, Part XXIII. LNCS, vol. 12368, pp. 484–501. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58592-1_29
3. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: 2017 IEEE Symposium on Security and Privacy (SP), pp. 39–57. IEEE (2017)
4. Chen, S., Kahla, M., Jia, R., Qi, G.J.: Knowledge-enriched distributional model inversion attacks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 16178–16187 (2021)
5. Cheng, S., Dong, Y., Pang, T., Su, H., Zhu, J.: Improving black-box adversarial attacks with a transfer-based prior. In: Advances in Neural Information Processing Systems, vol. 32 (2019)
6. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: additive angular margin loss for deep face recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4690–4699 (2019)
7. Dong, Y., et al.: Boosting adversarial attacks with momentum. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9185–9193 (2018)
8. Dong, Y., Pang, T., Su, H., Zhu, J.: Evading defenses to transferable adversarial examples by translation-invariant attacks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4312–4321 (2019)
9. Fu, J., Sun, J., Wang, G.: Boosting black-box adversarial attacks with meta learning. In: 2022 41st Chinese Control Conference (CCC), pp. 7308–7313. IEEE (2022)
10. Ganeshan, A., BS, V., Babu, R.V.: FDA: feature disruptive attack. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8069–8079 (2019)

11. Gao, L., Zhang, Q., Song, J., Liu, X., Shen, H.T.: Patch-wise attack for fooling deep neural network. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020, XXVIII. LNCS, vol. 12373, pp. 307–322. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58604-1_19

12. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572 (2014)

13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)

14. Ilyas, A., Engstrom, L., Athalye, A., Lin, J.: Black-box adversarial attacks with limited queries and information. In: International Conference on Machine Learning, pp. 2137–2146. PMLR (2018)

15. Ilyas, A., Engstrom, L., Madry, A.: Prior convictions: black-box adversarial attacks with bandits and priors. arXiv preprint arXiv:1807.07978 (2018)

16. Kurakin, A., Goodfellow, I., Bengio, S.: Adversarial machine learning at scale. arXiv preprint arXiv:1611.01236 (2016)

17. Li, H., Xu, X., Zhang, X., Yang, S., Li, B.: QEBA: query-efficient boundary-based blackbox attack. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1221–1230 (2020)

18. Long, Y., et al.: Frequency domain model augmentation for adversarial attack. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) ECCV 2022, Part IV. LNCS, vol. 13664, pp. 549–566. Springer, Cham (2022)

19. Ma, C., Chen, L., Yong, J.H.: Simulating unknown target models for query-efficient black-box attacks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11835–11844 (2021)

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

21. Naseer, M., Khan, S.H., Rahman, S., Porikli, F.: Task-generalizable adversarial attack based on perceptual metric. arXiv preprint arXiv:1811.09020 (2018)

22. Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z.B., Swami, A.: Practical black-box attacks against machine learning. In: Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, pp. 506–519 (2017)

23. Russakovsky, O., et al.: Imagenet large scale visual recognition challenge. Int. J. Comput. Vision **115**, 211–252 (2015)

24. Struppek, L., Hintersdorf, D., Correia, A.D.A., Adler, A., Kersting, K.: Plug & play attacks: towards robust and flexible model inversion attacks. arXiv preprint arXiv:2201.12179 (2022)

25. Sundararajan, M., Taly, A., Yan, Q.: Axiomatic attribution for deep networks. In: International Conference on Machine Learning, pp. 3319–3328. PMLR (2017)

26. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31 (2017)

27. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826 (2016)

28. Szegedy, C., et al.: Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199 (2013)

29. Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., McDaniel, P.: Ensemble adversarial training: attacks and defenses. arXiv preprint arXiv:1705.07204 (2017)

30. Wadawadagi, R., Pagi, V.: Sentiment analysis with deep neural networks: comparative study and performance assessment. Artif. Intell. Rev. **53**(8), 6155–6195 (2020)
31. Wang, X., He, K.: Enhancing the transferability of adversarial attacks through variance tuning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1924–1933 (2021)
32. Wang, Z., Guo, H., Zhang, Z., Liu, W., Qin, Z., Ren, K.: Feature importance-aware transferable adversarial attacks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 7639–7648 (2021)
33. Xiao, C., Li, B., Zhu, J.Y., He, W., Liu, M., Song, D.: Generating adversarial examples with adversarial networks. arXiv preprint arXiv:1801.02610 (2018)
34. Xie, C., et al.: Improving transferability of adversarial examples with input diversity. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2730–2739 (2019)
35. Xiong, Y., Lin, J., Zhang, M., Hopcroft, J.E., He, K.: Stochastic variance reduced ensemble adversarial attack for boosting the adversarial transferability. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14983–14992 (2022)
36. Zhang, J., et al.: Improving adversarial transferability via neuron attribution-based attacks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14993–15002 (2022)
37. Zhang, Y., Jia, R., Pei, H., Wang, W., Li, B., Song, D.: The secret revealer: generative model-inversion attacks against deep neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 253–261 (2020)