上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

Adversarial Attack on Attackers: Post-Process to Mitigate Black-Box Score-Based Query Attacks

Sizhe Chen, Zhehao Huang, Qinghua Tao, Yingwen Wu, Cihang Xie, Xiaolin Huang

Department of Automation, Shanghai Jiao Tong University

ESAT-STADIUS, KU Leuven

Computer Science and Engineering, University of California, Santa Cruz

NeurIPS 2022

AAA paper

饮 水 思 源 • 爱 国 荣 校

Introduction

The adversarial threat has been made feasible by score-based query attacks (SQAs),

which greedily update x_k by a query sample x_{q} (crafted by certain strategies from x_k) if it reduces DNN's loss.

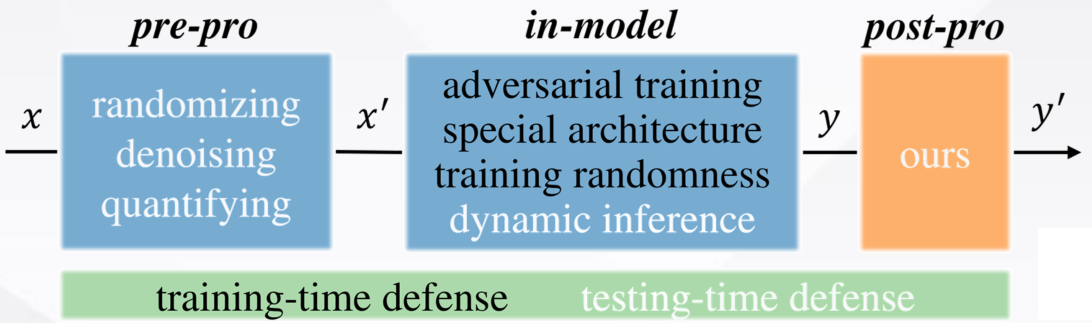
$$\boldsymbol{x}_{k+1} = \begin{cases} \boldsymbol{x}_{\mathrm{q}}, & \mathcal{L}(f(\boldsymbol{x}_{\mathrm{q}}), y) < \mathcal{L}(f(\boldsymbol{x}_k), y), \\ \boldsymbol{x}_k, & \mathcal{L}(f(\boldsymbol{x}_{\mathrm{q}}), y) \geq \mathcal{L}(f(\boldsymbol{x}_k), y). \end{cases} \qquad \mathcal{L}(f(\boldsymbol{x}), y) = f_y(\boldsymbol{x}) - \max_{k \neq y} f_k(\boldsymbol{x})$$

SQAs only use DNN output scores, but could efficiently attack within dozens of queries, posing great danger.

However, existing defenses against worst-case perturbations are not suitable for mitigating real-world SQAs.

Table 1: Expectation and effects of defenses (unpreferable effects are marked in red)

	expectation	adv-train	pre-process	dynamic inference	AAA (ours)
accuracy	=	↓↓	↓	=	=
calibration	↑	/	↓	↓	↑
testing cost	=	=	=	↑↑↑	=
training cost	=	↑↑↑	=	=	=
acc under SQA	↑	↑↑	↑	↑	↑↑↑



We note that in black-box settings, a post-processing module in test time is sufficient to mitigate SQAs.

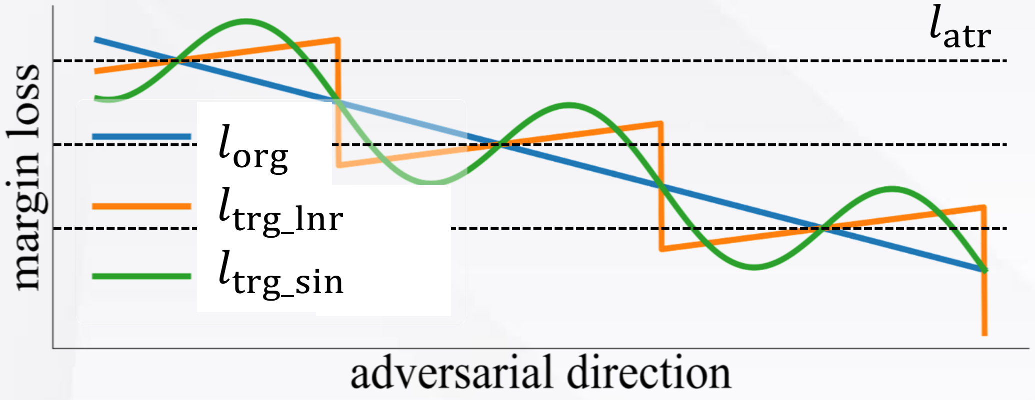
Advantages of post-processing: (1) mitigate SQAs; (2) preserve model accuracy; (3) improve model calibration.

How to serve users while mitigating SQA attackers when they access the same output information?

Andriushchenko et al. Square attack: A query-efficient black-box adversarial attack via random search, ECCV 2020
Guo et al. On calibration of modern neural networks, ICML 2017

Adversarial attack on attackers (AAA)

- fool attackers into incorrect attack directions by slight perturbations on DNN outputs in the test time
- manipulate the loss trend, which is the only metric SQAs base on
- attackers trying to greedily update samples following the original trend are led to incorrect paths



Algorithm 1 Adversarial Attack on Attackers

Input: the logits $\boldsymbol{z}_{\text{org}}, T, \tau, \alpha, \beta, \kappa$.
Output: post-processed logits $\boldsymbol{z}$

1: get original loss $l_{\text{org}} = \mathcal{L}_{\text{u}}(\boldsymbol{z}_{\text{org}})$ by (2)
2: set target loss l_{trg} by (5)
3: set target confidence $p_{\text{trg}} = \sigma(\boldsymbol{z}_{\text{org}}/T)$
4: initialize $\boldsymbol{z} = \boldsymbol{z}_{\text{org}}$ and optimize it for (6)
5: **return** $\boldsymbol{z}$

$$\mathcal{L}_{\text{u}}(\boldsymbol{z}) \triangleq \mathcal{L}(f(\boldsymbol{x}), \hat{y})$$

$$l_{\text{atr}} = (\text{floor}(l_{\text{org}}/\tau) + 1/2) \times \tau$$

$$l_{\text{trg\_lnr}} = l_{\text{atr}} - \alpha \times (l_{\text{org}} - l_{\text{atr}})$$

$$l_{\text{trg\_sin}} = l_{\text{org}} - \alpha \times \tau \sin(\pi(1 - 2(l_{\text{org}} - l_{\text{atr}})/\tau))$$

$$\min_{\boldsymbol{z}} \ \| \mathcal{L}_{\text{u}}(\boldsymbol{z}) - l_{\text{trg}} \|_1 + \beta \cdot \| \sigma(\boldsymbol{z}) - p_{\text{trg}} \|_1$$

softmax

Line 1: get the original margin loss l_{atr} from unmodified logits z_{org} by assuming the current prediction is correct

Line 2: divide losses into intervals by periodic loss attractors l_{atr}, and set the target loss value l_{trg} accordingly

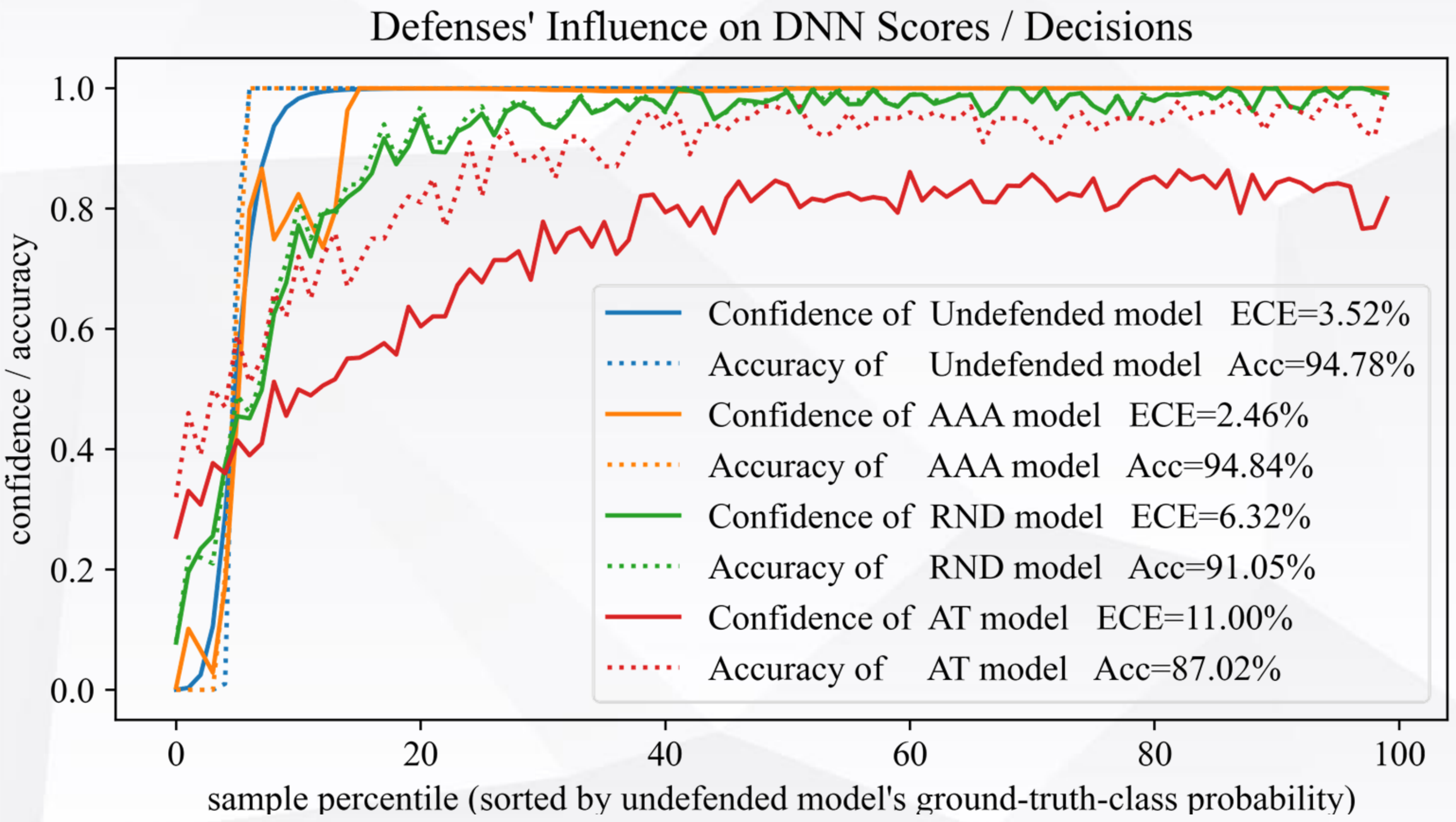
Line 3: set the target prediction confidence p_{atr} by a pre-calibrated temperature T

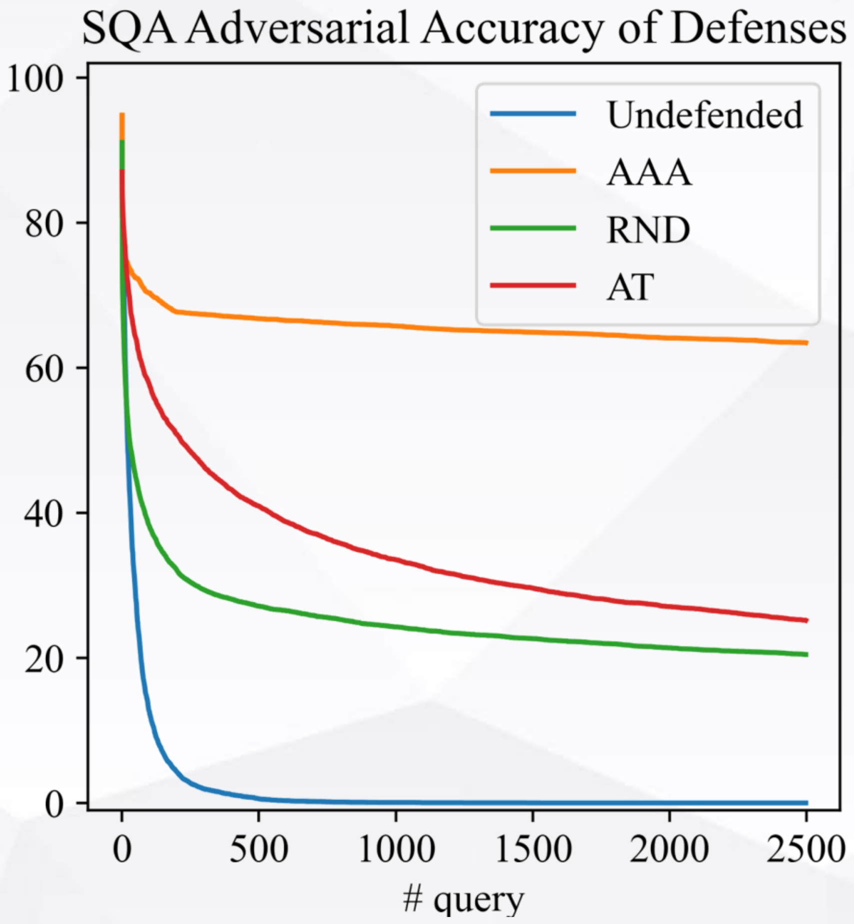
Line 4: optimize the logits z to form the misleading loss curve l_{trg} while outputting accurate confidence p_{trg}

Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods, Advances in large margin classifiers, 1999

Experiments

AAA alters scores most slightly without influencing accuracy, but is outstanding in mitigating SQAs v.s. baselines.





Expected Calibration Error (ECE ↓) is a measure of calibration (the difference between accuracy and confidence).

RND: Random Noise Defense, AT: Adversarial Training

Qin et al. Random noise defense against query-based black-box attacks, NeurIPS 2021.
Dai et al. Parameterizing activation functions for adversarial robustness, IEEE S&P, 2022.

Experiments

- AAA mitigates SQAs most effectively with improvements on calibration and without hurting accuracy.
- AAA could be easily plugged into existing defenses, e.g., adversarial training.
- It is also easy to mislead adaptive attackers in real-world scenarios by, e.g., AAA-sine.

Table 2: The defense performance under attacks (#query = 100/2500)

Model	Metric / Attack	None	adv-train	random-input	AAA-linear
CIFAR-10 $\ell_\infty = \frac{8}{255}$	ECE (%)	3.52	11.00	6.32	**2.46**
	Acc (%)	94.78	87.02	91.05	**94.84**
Wide-ResNet28	Square	39.38 / 00.09	78.30 / 67.44	60.83 / 49.15	**81.36 / 80.59**
	SignHunter	41.14 / 00.04	78.87 / 66.79	61.02 / 47.82	**79.41 / 76.71**
	SimBA	53.04 / 03.95	84.21 / 75.85	76.39 / 64.34	**88.86 / 83.36**
	NES	83.42 / 12.24	85.92 / 81.01	86.23 / 68.19	**90.62 / 85.95**
	Bandit	69.86 / 41.03	**83.62** / 76.25	70.44 / 41.65	80.86 / **78.36**
ImageNet $\ell_\infty = \frac{4}{255}$	ECE (%)	5.42	5.03	5.79	**4.30**
	Acc (%)	77.11	66.30	75.32	**77.17**
Wide-ResNet50	Square	52.27 / 09.25	59.20 / 51.11	58.67 / 50.54	**63.13 / 62.51**
	SignHunter	53.05 / 13.88	59.47 / 56.22	59.36 / 52.98	**62.35 / 56.80**
	SimBA	71.79 / 20.90	65.64 / 47.60	66.36 / 63.27	**74.16 / 67.14**
	NES	77.11 / 64.93	66.30 / 64.38	71.33 / 66.05	**77.12 / 67.06**
	Bandit	71.33 / 65.77	65.30 / 63.98	65.15 / 61.38	**72.15 / 70.53**

Table 4: Generalization of AAA tested by Square attack (#query = 100/2500, CIFAR-10)

Metric / Attack	None	AAA-linear	adv-train (AT)	AT-AAA-linear
ECE (%)	3.52	**2.46**	11.00	**10.56**
Acc (%)	94.78	**94.84**	**87.02**	**87.02**
untargeted $\ell_\infty = 8/255$	39.38 / 00.09	**81.36 / 80.59**	78.30 / 67.44	**80.80 / 80.13**
targeted $\ell_\infty = 8/255$	75.59 / 02.84	**92.05 / 91.62**	85.75 / 82.72	**86.22 / 86.13**
untargeted $\ell_2 = 0.5$	81.53 / 18.75	**92.66 / 92.63**	84.26 / 78.97	**85.12 / 84.31**
untargeted $\ell_2 = 2.5$	12.77 / 00.01	**70.35 / 63.46**	57.88 / 25.19	**74.03 / 73.72**

Table 6: AAA under adaptive attacks (100 queries)

Defense	None	AAA-linear	AAA-sine
Square	39.38	81.36	78.34
bi-Square	57.09	62.91	76.69
op-Square	94.78	57.31	76.41

Conclusion

- Propose that post-processing could be an **effective, user-friendly, and plug-in defense** against score-based query attacks.
- Design a defense to **attack score-based attackers into incorrect directions** by slightly **perturbing the model outputs in test time**.
- Extensive study show AAA outperforms existing defenses significantly in the **accuracy, calibration, and protection performance**.
- Defending against other types of attacks is beyond our scope, e.g., white-box attacks, transfer-based attacks, and decision-based query attacks, which are either unfeasible or inefficient in the real world.

Thanks for listening
Welcome discussions
sizhe.chen@sjtu.edu.cn

AAA code

AAA poster

About me