Query-Efficient Black-Box Attack by Active Learning

Pengcheng Li
National Key Laboratory
for Novel Software Technology
Nanjing University, Nanjing 210023, China
Email: lipc@lamda.nju.edu.cn

Jinfeng Yi
JD AI Research
Email: yijinfeng@jd.com

Lijun Zhang
National Key Laboratory
for Novel Software Technology
Nanjing University, Nanjing 210023, China
Email: zhanglj@lamda.nju.edu.cn

***Abstract*—Deep neural network (DNN) as a popular machine learning model is found to be vulnerable to adversarial attack. This attack constructs adversarial examples by adding small perturbations to the raw input, while appearing unmodified to human eyes but will be misclassified by a well-trained classifier. In this paper, we focus on the black-box attack setting where attackers have almost no access to the underlying models. To conduct black-box attack, a popular approach aims to train a substitute model based on the information queried from the target DNN. The substitute model can then be attacked using existing white-box attack approaches, and the generated adversarial examples will be used to attack the target DNN. Despite its encouraging results, this approach suffers from poor query efficiency, i.e., attackers usually needs to query a huge amount of times to collect enough information for training an accurate substitute model. To this end, we first utilize state-of-the-art white-box attack methods to generate samples for querying, and then introduce an active learning strategy to significantly reduce the number of queries needed. Besides, we also propose a diversity criterion to avoid the sampling bias. Our extensive experimental results on MNIST and CIFAR-10 show that the proposed method can reduce more than 90% of queries while preserve attacking success rates and obtain an accurate substitute model which is more than 85% similar with the target oracle.**

***Index Terms*—Deep Neural Network, Active Learning**

I. Introduction

Deep neural networks (DNNs) have achieved great successes in a variety of domains [1]. However, recent studies have shown that DNNs may be easily fooled by adversarial examples [2]. For example, in the context of image classification, an adversarial example is an image that is visually indistinguishable to the original image but can mislead the DNN model to output incorrect labels. In addition to image classification, attacks to other DNN-related tasks have also been actively investigated, such as semantic segmentation [3], machine translation [4], visual QA [5], image captioning [6], speech recognition [7], medical prediction [8], and autonomous driving [9].

Depending on how much information the attackers have access to, adversarial attack can be broadly classified into two categories: white-box attack and black-box attack. The adversary in the white-box setting has full access to the target DNN model [2], [10], [11]. In the black-box setting, adversaries can only access the input and output of the underlying DNN but not its internal configurations and parameters [12], [13]. Recent studies have shown that both of these two categories of attacks can reach a extremely high success rate of attack. Although a lot of defense methods [14]–[16] were designed to increase the robustness of the model, the white-box attack [17] can still conquer the model with nearly 100% success rate by estimating the gradient through approximation or expectation [18].

Compared to the white-box setting, the black-box setting is much more practical since a majority of real-world learning systems do not allow white-box access due to security reasons. Most of existing black-box attack methods are based on the transferability phenomenon [13], where an adversary first trains a substitute model and then crafts adversarial examples against it, hoping that the generated adversarial examples can also successfully attack the underlying black-box models. Black-box attack can also bypass most defense methods that change the model structure to increase robustness for the reason that it is isolated from the target model. The black-box variants of JSMA [19] and of the Carlini & Wagner attack [10] both obtain over 95% success rate on adversarial examples. However, a key limitation of these approaches is that training a substitute network requires a large number of queries to collect sufficient information. For example, the number of queries in [13] increases almost exponentially with respect to the number of iterations.

In this paper, we address this issue by employing the active learning strategy. Specifically, we first utilize the state-of-the-art white-box attack methods to generate adversarial examples. We then improve the query-efficiency of transfer-based framework by actively selecting the most informative samples. Furthermore, we propose a diversity criterion to avoid the bias caused by active learning. We summarize our main contributions as follows:

- We propose to use more advanced methods for data augmentation in transfer-based framework, and verify that C&W attack method [10] and Deepfool [11] are more effective than the raw Jacobian-based method [13].
- We propose to use active learning strategy to select the most informative samples for querying. To avoid the bias caused by active learning, we further introduce a diversity criterion to ensure that the sampled queries are both informative and diverse.

978-1-5386-9159-5/18/$31.00 ©2018 IEEE
DOI 10.1109/ICDM.2018.00159

- We conduct extensive experiments to evaluate our method. Our empirical results show that the proposed approach can significantly reduce the number of queries while preserve the success rate of attack.

II. Related Work

In this section, we briefly review the existing methods of adversarial attack and active learning.

A. Adversarial Attack Methods

Adversarial attack methods can be broadly classified into two main categories: (i) white-box attacks, which utilize the specific information of the target model to construct perturbations for attack, including FGSM, JSMA, Deepfool and C&W [2], [10], [11], [20]; (ii) black-box attacks, which are isolated from the parameters and settings of the target model and can be roughly divided into three types, i.e., score-based attacks, decision-based attacks and transfer-based attacks [12], [13], [19], [21], [22]. Score-based attacks rely on the predicted

$$\mathcal{S}_{add} = \left\{\mathbf{x} + \frac{|\ \ _l| \times |\nabla_{\mathbf{x}}\ \ _l(\mathbf{x})|}{\|\nabla_{\mathbf{x}}\ \ _l(\mathbf{x})\|_2^2} \times \text{ign}(\nabla_{\mathbf{x}}\ \ _l(\mathbf{x})) | \mathbf{x} \in \mathcal{S}_\rho\right\}$$

where $\ _l$ is the ℓ-th dimension of $\ (\cdot)$.

- C&W [10]: For each instance $\mathbf{x}$ in $\mathcal{S}_\rho$, we solve an optimization problem:

$$\begin{aligned} \min \quad & \|\boldsymbol{\delta}\|_2 + \ell \cdot \ (\mathbf{x} + \boldsymbol{\delta}) \\ \ \ \quad & \mathbf{x} + \boldsymbol{\delta} \in [0\ \ 1]^n \end{aligned}$$

where $\ (\mathbf{x}) = \text{ma}\ (\text{ma}\ _{i=l}(\ \ (\mathbf{x})_i) - \ \ (\mathbf{x})_t - \ ), \quad (\cdot)$ presents the softmax function and $\ $ is a constant to control the confidence. In our framework, we choose ℓ_2-norm to constrain the size of the perturbation. The solutions to these optimization problems constitute $\mathcal{S}_{add}$.

All these methods propose to generate adversarial samples across the decision boundary of the substitute model so that each round of black-box attack can accurately correct the model's parameters to make it more similar to the target oracle.

B. Active Learning Strategy

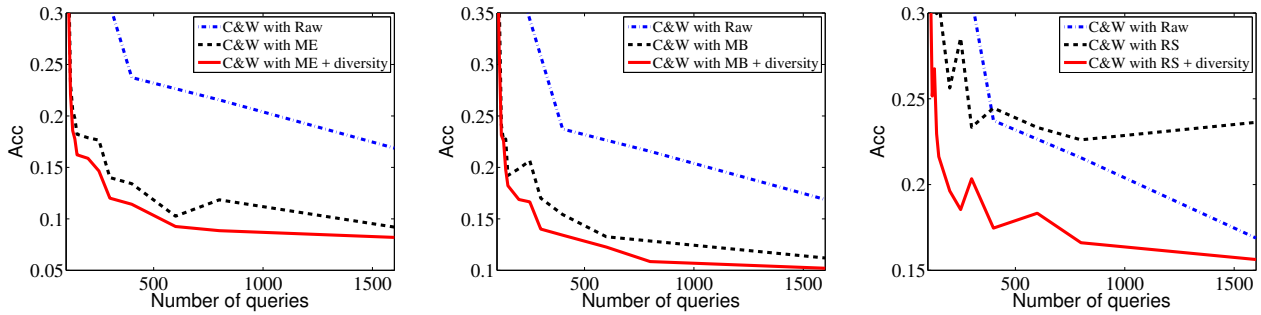
The above framework first trains a substitute DNN, then generates adversarial examples from the substitute DNN. However, it needs to query the oracle too many times, which is not allowed in real applications. So, we propose to use active learning to reduce the number of queries. The new procedure is summarized in Algorithm 2.

Algorithm 2 Substitute DNN training with active learning

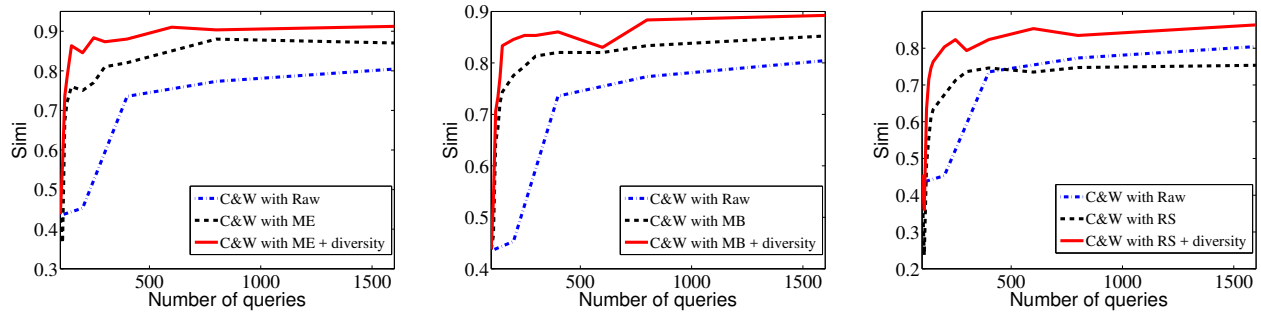
INPUT: target oracle $\tilde{\ }$, a maximum number $\ _{max}$ of training

		FGSM		IGS		FGV		Deepfool		C&W		JSMA	
itr	query	Acc	Simi	Acc	Simi	Acc	Simi	Acc	Simi	Acc	Simi	Acc	Simi
0	100	0.4528	0.4521	0.4417	0.4764	0.4577	0.4253	0.4259	0.4009	0.4533	0.4197	0.4532	0.4107
1	200	0.3401	0.5628	0.2895	0.5684	0.3483	0.5853	0.3257	0.4894	0.3792	0.3577	0.2974	0.6173
2	400	0.2085	0.7521	0.2642	0.7648	0.2412	0.7451	0.2161	0.7415	0.2373	0.7504	0.2384	0.7936
3	800	0.2201	0.7706	0.2061	0.7684	0.1989	0.7701	0.1753	0.7679	0.2156	0.7564	0.1936	0.8362
4	1600	0.2253	0.7865	0.2427	0.7962	0.1783	0.8136	0.1242	0.8628	0.1688	0.8328	0.1873	0.8635
5	3200	0.1439	0.8330	0.1426	0.8286	0.1209	0.8460	0.0832	0.8969	0.1196	0.8572	0.1639	0.8935
6	6400	0.1289	0.8623	0.1317	0.8530	0.1374	0.8595	0.0801	0.8739	0.0693	0.9166	0.1373	0.9126
7	12800	0.0906	0.8682	0.0810	0.8778	0.1326	0.8877	0.0639	0.9283	0.0617	0.9310	0.1299	0.9263
8	25600	0.0652	0.8821	0.0656	0.8940	0.0752	0.9053	0.0625	**0.9419**	**0.0563**	0.9217	0.1108	0.9183

TABLE I: Results of using FGSM, Iterative Gradient Sign (IGS), JSMA, Fast Gradient Value (FGV), Deepfool and C&W to craft samples for querying on MNIST. The evaluation metrics are Acc and Simi. The results on this table are averaged over 10 runs.



(a) Acc vs. Number of queries



(b) Simi vs. Number of queries

Fig. 1: Black-box attack by combining C&W and active learning strategy on MNIST (ME: Max Entropy method, MB: Margin based method, RS: Random Select method, diversity: our proposed active learning strategy). The curves on this Fig are averaged over 10 runs.

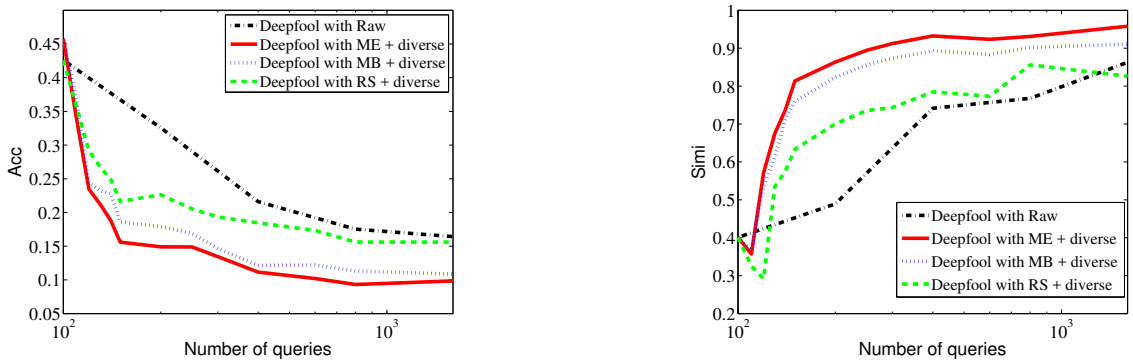
IV. Experiment

In this section, we first compare the performance of different data augmentation methods within the passive learning framework. Then, we validate the performance of our algorithm which considers active learning and diversity of the query set simultaneously.

A. Setup

We evaluate the performance of our algorithm on three datasets, i.e., MNIST [31], Fashion-MNIST [32] and CIFAR-10 [33]. We random select 100 samples as the initial training set (with 10 samples from each class) for each dataset. We assume adversaries can collect such a limited sample set from the oracle task.

The metric we used to evaluate the attack pattern is divided into two parts: Accuracy of the target oracle (Acc) and Similarity (Simi). Acc is an indicator of the success rate of attacks (the lower, the better). We use FGSM to generate adversarial examples over the substitute model and denote the accuracy of oracle when tested with these adversarial examples as Acc. Simi represents the similarity between our substitute model and the oracle (the higher, the better). We query the



(a) Acc vs. Number of queries

(b) Simi vs. Number of queries

Fig. 2: Black-box attack by combining Deepfool and active learning strategy on MNIST (ME: Max Entropy method, MB: Margin based method, RS: Random Select method, diversity: our proposed active learning strategy). The curves on this Fig are averaged over 10 runs.

oracle with the entire dataset and treat the result as a new dataset. We denote the accuracy of the substitute model when tested with this dataset by Simi.

For dataset MNIST, we use a pre-trained CNN model with accuracy 99.24% as the target oracle and a simple CNN model which contains a convolutional layer of 32 convolution kernels, a max-pooling layer, and a fully-connected layer as the substitute network. The structure of the target oracle is invisible to our algorithms. Due to the limitation of space, we only present results over MNIST and more settings and results can be found in the full version of our paper [1].

B. Passive Learning Framework

We construct the samples with different white-box attack methods and compare the performance of the substitute models trained thereby. In black-box attack, the number of queries is the main cost. The more we query, the more likely to cause the target oracle's attention. The experimental results on the MNIST dataset are shown in Table 1, from which we observe that when using C&W or Deepfool method to craft query samples, the iteration we need for an effective attack is less than FGSM and other methods. We can also verify that with the same number of queries, the substitute model trained by data generated with C&W or Deepfool method has a higher attack success rate and a higher similarity with the oracle.

The reason for the better performance of C&W and Deepfool is that these methods solve an optimization problem to craft samples which can cross the boundary of the current model. In contrast, other methods like FGSM construct samples from the original samples along the direction of the gradient towards the decision boundary but may not cross the boundary. This confirms our thought that there is more information about the boundary within the samples crafted by solving optimization problems than samples crafted by gradient based attack methods.

[1] http://lamda.nju.edu.cn/lipc/papers/ICDM2018.pdf

C. Active Learning Strategy

Since C&W and Deepfool show a better performance than other methods in previous experiments, we combine these two methods with different active learning strategies in this part.

Fig. 1 shows the performance of the active learning method, as well as our improved version that takes the diversity of samples into consideration, where the parameter is set to be 10. The Raw algorithm in Fig. 1 follows the setting in [13] which doubles the number of queries in the first few iterations each time, and then uses reservoir sampling method [34] which is a Random Selection strategy in the later few iterations to make the number of queries grow linearly.

As we can see in Fig. 1(a), the Acc of each active learning method is lower than that of the Raw algorithm except RS method. The reason is that the Raw algorithm queries more samples than the RS method in the first few iterations. However, as the number of iterations increases, the effects of Raw algorithm and RS method become comparable. Furthermore, in all cases, Acc of our improved version is lower than that of the original active learning method. For example, to achieve 10% Acc, the original Max Entropy algorithm queries over 1600 times, our improved version only queries 600 times, while the Raw algorithm queries more than 10000 times. The curves of Simi in Fig. 1(b) also validate our motivation that the diversity of sample set can effectively identify different parts of the decision boundary. We observe that Simi increases much more quickly when we take diversity into consideration. For example, to achieve 85% Simi, the original Max Entropy algorithm queries over 600 times, while the improved version only needs 200 times. This situation suggests that, the distortions in the decision boundary of our substitute models are more similar to that of oracle and considering the diversity of query set can help modify the decision boundary better than only using active learning strategy.

In summary, while the original active learning algorithm (Max Entropy method and Margin based method) can be used

to reduce the number of queries significantly, our improved version can further boost the performance.

In Fig. 2, we change the method from C&W to Deepfool (Deepfool achieve as good performance as C&W in the experiments of the previous passive learning framework) and report the performance of our improved algorithm which considers active learning strategy and diversity simultaneously. Again, the results show that the combination of active learning and the diversity of samples indeed reduces the number of queries in transfer-based black-box attack significantly. For example, when the number of queries reaches up to 400, the Acc of Max Entropy algorithm is 10% lower than that of the Raw algorithm, and the Simi is 20% higher.

V. Conclusion and Future Work

In this paper, we have tested a number of white-box attack methods and found that C&W attack and Deepfool yield the overall best performance. In addition, we introduced active learning to address the query-efficiency issue occurred in transfer-based attack. To alleviate the bias caused by active learning, we propose to maximize the diversity of query set and our empirical study verifies its effectiveness. In the future, we will apply our method to a variety of machine learning models, rather than neural networks and apply more advanced active learning strategy.

ACKNOWLEDGMENT

This work was partially supported by the National Key R&D Program of China (2018YFB1004300), YESS (2017QNRC001), and the Collaborative Innovation Center of Novel Software Technology and Industrialization.

References

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.