

Delving into Data: Effectively Substitute Training for Black-box Attack

Wenxuan Wang^{1*†} Bangjie Yin^{2*¶} Taiping Yao^{2*¶} Li Zhang^{1§}
Yanwei Fu^{1†§} Shouhong Ding^{2†¶} Jilin Li^{2¶} Feiyue Huang^{2¶} Xiangyang Xue^{1‡}

¹Fudan University

²Youtu Lab, Tencent

Abstract

Deep models have shown their vulnerability when processing adversarial samples. As for the black-box attack, without access to the architecture and weights of the attacked model, training a substitute model for adversarial attacks has attracted wide attention. Previous substitute training approaches focus on stealing the knowledge of the target model based on real training data or synthetic data, without exploring what kind of data can further improve the transferability between the substitute and target models. In this paper, we propose a novel perspective substitute training that focuses on designing the distribution of data used in the knowledge stealing process. More specifically, a diverse data generation module is proposed to synthesize large-scale data with wide distribution. And adversarial substitute training strategy is introduced to focus on the data distributed near the decision boundary. The combination of these two modules can further boost the consistency of the substitute model and target model, which greatly improves the effectiveness of adversarial attack. Extensive experiments demonstrate the efficacy of our method against state-of-the-art competitors under non-target and target attack settings. Detailed visualization and analysis are also provided to help understand the advantage of our method.

1. Introduction

Despite achieved impressive performance in most computer vision tasks, deep neural networks (DNNs) have been

*indicates equal contributions.

†indicates corresponding author.

‡Wenxuan Wang and Xiangyang Xue are with School of Computer Science, and Shanghai Key Lab of Intelligent Information Processing, Fudan University. Email: {wxwang19, xyxue}@fudan.edu.cn.

§Li Zhang and Yanwei Fu are with School of Data Science, MOE Frontiers Center for Brain Science, and Shanghai Key Lab of Intelligent Information Processing, Fudan University. Email: {yanweifu, lizhangfd}@fudan.edu.cn.

¶Bangjie Yin, Taiping Yao, Shouhong Ding, Jilin Li and Feiyue Huang are with Youtu Lab, Tencent. Email: {bangjieyin, taipingyao, ericshding, jerolinli, garyhuang}@tencent.com.

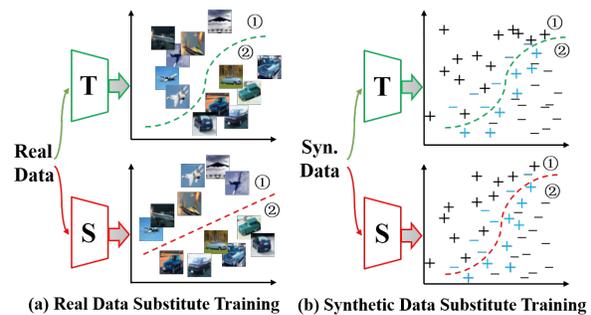


Figure 1. Differences between applying real data and synthetic data for substitute training. The ‘T’/‘S’ means the target/substitute model, the blue (+)/(-) in (b) indicates the adversarial examples, and the dashed green/red lines represent the decision boundary. Comparing (a) and (b), synthetic data generated in our way can train a substitute model with a more similar decision boundary to the target model. Best viewed in color and zoomed in.

shown to be vulnerable to even imperceptible adversarial noise/perturbations [28, 18]. The existence of adversarial examples reveals important security risks in deploying DNNs to real-world applications. The community studies the adversarial attacks in the settings of white-box or black-box attack, by whether or not fully access to the target attack model. Practically, as the information of the full target model for white-box attack is unavailable to real-world deployment, this paper particularly focuses on the black-box attack, which normally produces the adversarial examples only relying on hard-labels or output scores of the target model. Typically, the black-box attack includes the score-based [3, 12, 11, 7] or decision-based methods [5, 1]. Nevertheless, it is required to make an avalanche of queries to the target model in such attacks, still potentially limiting their usability to attack DNNs in real situations.

Recently, the idea of substitute training has been extensively explored in the black-box attack [8, 26, 16, 23, 29]. Normally, rather than directly learning to synthesize adversarial examples, a substitute model is trained to make similar predictions as the target model, queried by the same

input data. Within a certain amount of queries, this type of method is usually capable of learning a substitute model from the target model. Attack can thus be conducted on substitute model, and then transferable to the target model.

Fundamentally, substitute model tries to gain knowledge from the target model, by giving the input data and corresponding queried labels. Critically, shall the input data come from the training data for the target model? By assuming the ‘yes’ answer, it indeed simplifies the substitute training. However, it is even non-trivial to collect real input data in many real-world vision tasks. For example, the data of person images and videos are under very strict control, and the privacy of personal data has been well protected by the laws in many countries. Moreover, are the real images the most effective data for substitute training? The training data of the target model indeed help to get a well-performing substitute model on original task, but it cannot guarantee the transferability of the attack from the substitute model to target model, which has been proved in Tab. 1 and Tab. 2. For improving the attack performance in substitute training, it is necessary to minimize the decision boundary distance between the substitute and target models, which needs not only large-scale and diverse training data, but especially the data distributed near the decision boundary.

To address the limitation of real data and explore a better distribution of substitute training data, we propose a novel task-driven unified framework, which only uses specially-designed generated data for substitute training and achieves high attack performance. As shown in Fig. 1, compared with using the training data of the target model to conduct substitute training, diverse synthetic data combined with adversarial examples will promote the substitute model to further approach the target. More specifically, in our framework, we first propose a novel Diverse Data Generation module (DDG), which samples noise combined with label-embedded information to generate diverse training data. Such distributed generated data can basically guarantee the substitute model to learn knowledge from the target. Moreover, to further encourage the substitute model with similar decision boundary as the target, Adversarial Substitute Training strategy (AST) is proposed to introduce adversarial examples as boundary data into the training process. Overall, the jointly learning of DDG and AST ensures the consistency between the substitute and target model, which greatly improves attack success rate in substitute training for black-box attack without any real data beforehand.

The main contributions of this work are summarized as, (1) We propose a novel effective generation-based substitute training paradigm to boost data-free black-box attacking performance, for the first time, by delving into the essence of input generated substitute training data. (2) To achieve this goal, we firstly propose a diverse data generation module with multiple diverse constraints to broaden

the distribution of synthetic data. And then further improve the consistency of decision boundaries between substitute model and target model by adversarial substitute training strategy. (3) The comprehensive experiments and visualizations over the four datasets and one online machine learning platform demonstrate the effectiveness of our method against the state-of-the-art attacks.

2. Related work

Adversarial attack. Many previous works focus on the white-box attack [28, 24, 2, 16, 20] by generating adversarial examples through accessing gradient-information of target model. Furthermore, there are also some white-box attack methods studying the transferable attacking performance on the unknown black-box models [6, 34, 4]. Unfortunately, such a white-box setting greatly and unrealistically simplifies the attack task in the real-world scenario, as it demands a strong pre-condition of accessing the target models. In contrast, recent efforts are made on black-box attack methods, which has a more practical setting. Normally, the attacker can only obtain the output scores or hard labels of a target victim model. In general, the black-box attack [9, 1] is conducted by finding adversarial examples from trials, which will cross the decision boundary of classes. For example, when processing the class probability output, Chen *et al.* [3] propose utilizing a derivative of zeroth order to estimate the real gradients, and the work has been expanded by [30]. Ilyas *et al.* [11, 12] also propose performing score-based black-box attack by prior knowledge. Nevertheless, previous black-box attacks are limited to prohibitive cost for extensively querying the target model, and significant number of real data for the corresponding target model. Rather than directly discovering the adversarial examples, our model learns to effectively synthesize the data distribution of target model for training a substitute model. Such a substitute model potentially saves plenty amount of queries to the target model during the attack generation.

Substitute training. Substitute training is becoming a flourishing research direction. Papernot *et al.* [23] train the substitute model by utilizing a group of real images, and model theft attacks [29, 35] steal the target model also based on real data. However, considering the privacy or unattainable problems of training data, some works [31, 32, 33] generate synthetic data to train a substitute model. Methods in [31, 32] generate synthetic images from noise or recover training images from teacher model for substitute training based on knowledge distilling (KD). Zhou *et al.* [33] firstly propose an attack method to learn a substitute model under data-free condition. However, they only learn to output same results with target model, instead of further recovering the data distribution and decision boundary of the target, which are more crucial for the transferability of adversarial examples. Different from their

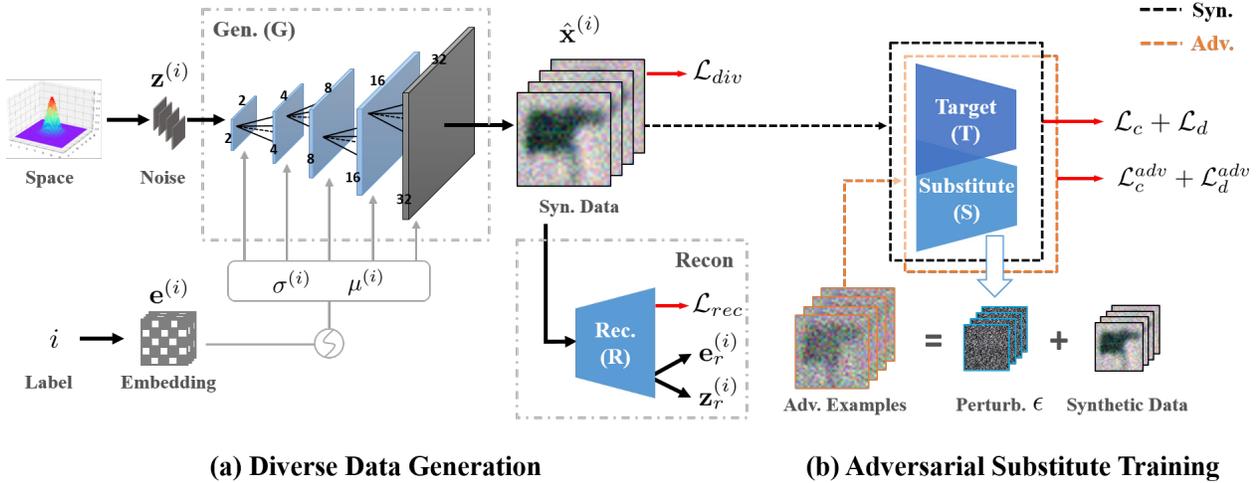


Figure 2. Illustration of the unified proposed architecture, which consists of Diverse Data Generation module (DDG) and Adversarial Substitute Training module (AST). (a) DDG aims to generate diverse data with given label, which used to train a substitute model. (b) AST utilizes the adversarial examples generated from current substitute model to push the substitute model mimic the boundary of the target.

strategy, our proposed method starts from focusing on the distribution of generated data training for substitute model, comprehensively improving the attacking performance on black-box model by two perspectives of the diverse data generation and adversarial substitute training.

3. Methodology

3.1. Framework Overview

The objective of our work is to train a substitute model effectively for black-box adversarial attack, the whole proposed framework is illustrated in Fig. 2. It consists of two modules: Diverse Data Generation module (DDG) generating diverse data and Adversarial Substitute Training strategy (AST) further mimicking the ‘behaviour’ of the target model. In Fig. 2(a), the DDG generates data $\hat{x}^{(i)} = G(\mathbf{z}^{(i)}, \mathbf{e}^{(i)})$ based on the random noise $\mathbf{z}^{(i)}$ and label-embedded vector $\mathbf{e}^{(i)}$ for the label index i . To guarantee the diversity of synthetic data, the generator G will be trained by three constraints, *i.e.*, the adaptive label normalized generator, noise/label reconstruction, and inter-class diversity, which will be elaborated later. Furthermore, to ensure the substitute model S approximate the decision boundary of the target model T , we feed the synthesized data along with the adversarial examples employed by AST into S for substitute training in Fig. 2(b). Essentially, we take the target model T as a black-box of classifying M classes, where only the label/probability outputs are available. The teacher-student strategy is re-purposed here to learn S from T . Finally, attacks can thus be conducted on substitute model, and then transferable to the target model.

3.2. Diverse Data Generation

To synthesize better data for substitute training, we first propose a novel Diverse Data Generation module (DDG) with three constraints to manipulate the diversities of generated synthetic images. These constraints, in principle, encourage the generator G to learn relatively independent data-distribution for each different class, and keep the inter-class variances, which promote the alternative model to learn the knowledge of the target model.

Adaptive label normalized generator. To better learn from the target model, we need equally distributed data of all categories for substitute training, thus it is necessary to generate label-controlled data. To realize that, we take full advantage of the given label and random noise. Firstly, with the input of the random noise vector $\mathbf{z}^{(i)} \in \mathbb{R}^N$ sampled from standard Gaussian distribution and label i , we calculate the label-embedded vector $\mathbf{e}^{(i)} \in \mathbb{R}^N$ based on embedding layers [21]. Such label embedding process can encode a single discrete label to a continuous learnable vector, which has a wider distribution in the feature space and contains more representation information. Unlike GANs, we have no real images for supervision, such a label embedding process is crucial for data generation. Next, we extract the mean $\mu^{(i)}$ and variance $\sigma^{(i)}$ from the N -dimensional label-embedding vector $\mathbf{e}^{(i)}$ by two full-connected layers. Then, the $\mu^{(i)}$ and $\sigma^{(i)}$ are involved in all deconvolution blocks to iteratively synthesize the image data with the condition of the specific category, which can be expressed as,

$$\hat{\mathbf{x}}_t^{(i)} = DeConv(\hat{\mathbf{x}}_{t-1}^{(i)}) * \sigma^{(i)} + \mu^{(i)} \quad (1)$$

where there are total five de-convolution blocks, and t represents the number of de-convolution block. After obtaining

the final $\hat{\mathbf{x}}^{(i)}$, the output generated data has been decorated with label-normalized information. Such an adaptive label normalized generator can better leverage the relations between input noise and label-embedding vectors to synthesize label-controlled data.

Noise/Label reconstruction. To further ensure the diversity of generated data $\hat{\mathbf{x}}^{(i)}$, we introduce a reconstruction net R to reconstruct the input noise and label embedding $\mathbf{z}_r^{(i)}, \mathbf{e}_r^{(i)} = R(\hat{\mathbf{x}}^{(i)})$. And the corresponding reconstruction loss can be calculated as,

$$\mathcal{L}_{rec} = \sum_{i=0}^{M-1} \left(\|\mathbf{z}_r^{(i)} - \mathbf{z}^{(i)}\|_1 + \text{CE}(f(\mathbf{e}_r^{(i)}), \mathbf{e}), i \right) \quad (2)$$

where we use L_1 to denote the difference between the input $\mathbf{z}^{(i)}$ and reconstructed $\mathbf{z}_r^{(i)}$. As for label reconstruction, we apply function $f(*)$ to calculate the cosine distance between $\mathbf{e}_r^{(i)}$ and \mathbf{e} , which are further processed by Softmax to compute the cross entropy loss with the ground-truth label i . Under this constraint, our G can generate more diverse images for different input noise vectors of each class.

Inter-class diversity. To further enhance the data diversity of different classes, we use a cosine-similarity matrix to maximize the inter-class distance, for all the synthetic images. Particularly, the generator produces one input synthetic data batch of $M_B \ll M$ different classes, and the model S gives the output similarity matrix $O_B \in \mathbb{R}^{M_B \times M_B}$ of this batch. Note that we have the ground-truth similarity matrix $O_B^{gt} \in \{0, 1\}^{M_B \times M_B}$ with all the elements to be 0 except the diagonal elements are set to be 1. Thus the diversity loss function \mathcal{L}_{div} can be formulated as:

$$\mathcal{L}_{div} = \|TRI(O_B - O_B^{gt})\|_2 \quad (3)$$

where $TRI(*)$ is defined as an operation to extract the upper triangle elements of similarity matrix except the diagonal elements. In this way, \mathcal{L}_{div} will ensure the synthetic data owns the independent distribution for each class.

3.3. Adversarial Substitute Training

After DDG generates diverse training data, for better attack performance, we still have to further encourage the substitute model with a more similar decision boundary as the target. As is known to all, adversarial examples are wrongly classified with the visually-indistinguishable perturbations applied on. Due to the perturbations is relatively small, the adversarial examples can be seen as the samples around the decision boundary. Therefore, we propose a novel adversarial substitute training strategy (AST), which utilizes the adversarial examples to further push the decision boundary of S more fitting to the T 's. More specifically, for each iteration during training, our generator firstly synthesizes images through DDG. Then we choose the white-box

Algorithm 1 The proposed black-box attack.

Require:

Input: Random noise $\mathbf{z}^{(i)} \in \mathbb{R}^N$; Label $i \in \{0, 1, \dots, M-1\}$; Generator G ; Target victim model T ; Substitute model S ; Iterations R .

Initialization: Model parameters θ_G, θ_S ; hyper-parameters $\beta_1, \beta_2, \beta_3, \gamma_1, \gamma_2$.

Ensure: Model parameters θ_G^*, θ_S^* .

- 1: **for** each $r \in R$ **do**
 - 2: **Synthetic data generation:**
 - 3: Given the label i and random noise $\mathbf{z}^{(i)}$, extract the mean $\mu^{(i)}$ and variance $\sigma^{(i)}$ from the label-embedded vector $\mathbf{e}^{(i)}$
 - 4: Generate data through adaptive label normalized generator $\hat{\mathbf{x}}_t^{(i)} = DeConv(\hat{\mathbf{x}}_{t-1}^{(i)}) * \sigma^{(i)} + \mu^{(i)}$
 - 5: Generate adversarial examples based on the $\hat{\mathbf{x}}^{(i)}$
 - 6: **Update S:**
 - 7: Compute $\mathcal{L}_S := (\mathcal{L}_d, \mathcal{L}_d^{adv})$ then update $\theta'_S \leftarrow \theta_S - \gamma_1 \nabla_{\theta_S} \mathcal{L}_S(\theta_S)$
 - 8: **Update G:**
 - 9: Calculate $\mathcal{L}_G := (\mathcal{L}_c, \mathcal{L}_c^{adv}, \mathcal{L}_{rec}, \mathcal{L}_{div})$ and then update $\theta'_G \leftarrow \theta_G - \gamma_2 \nabla_{\theta_G} \mathcal{L}_G(\theta_G)$
 - 10: **endfor**
 - 11: $\theta_G^* = \theta'_G, \theta_S^* = \theta'_S$
 - 12: **return** θ_G^*, θ_S^* ;
-

attacking algorithm to obtain the adversarial perturbations ϵ for the synthetic images based on the current S . The objective function to generate adversarial images is defined as,

$$\min_{\epsilon \in [0, 1]^d} \|\epsilon\| + \lambda \cdot \mathcal{L}(\hat{\mathbf{x}}^{(i)} + \epsilon, i^{adv}) \quad (4)$$

where $\mathcal{L}(\cdot)$ denotes an attack objective reflecting the probability or cross-entropy of predicting $\hat{\mathbf{x}}^{(i)} + \epsilon$ to be i^{adv} , if considering the un-targeted attack, $i^{adv} \neq i$, otherwise, $i^{adv} = t$, t is a target label. λ is a regularization coefficient, and the constraint $\epsilon \in [0, 1]^d$ confines the perturbation ϵ to the valid image space. Then the generated images and corresponding adversarial data are used to updating S together.

3.4. Loss Functions

Finally, we apply the basic loss functions as in [33] to train the substitute model,

$$\mathcal{L}_d = \sum_{i=0}^{M-1} \|T(\hat{\mathbf{x}}^{(i)}), S(\hat{\mathbf{x}}^{(i)})\|_F \quad (5)$$

$$\mathcal{L}_c = e^{-\mathcal{L}_d} + \sum_{i=0}^{M-1} \text{CE}(S(G(\mathbf{z}^{(i)}, \mathbf{e}^{(i)})), i) \quad (6)$$

where \mathcal{L}_d measures the distance between the output of T and S , and \mathcal{L}_c denotes the generation loss. $e^{-\mathcal{L}_d}$ implies a

Table 1. Comparing ASRs results using probability as the target model output among our method and competitors over several datasets.

	Dataset	MNIST			CIFAR-10			CIFAR-100		Tiny ImageNet
	Target Model	AlexNet	VGG-16	ResNet-18	AlexNet	VGG-16	ResNet-18	VGG-19	ResNet-50	ResNet-50
Non-Target	Training Data	41.36	29.25	34.81	30.95	23.15	32.66	14.47	18.33	12.86
	ImageNet	44.78	34.86	31.39	36.84	22.94	34.01	17.26	20.93	21.75
	PBBA [23]	52.53	50.31	59.77	45.82	30.19	33.91	22.34	28.11	26.54
	Knockoff [22]	59.21	58.38	65.82	50.93	31.58	39.40	27.73	29.55	29.99
	DaST [33]	58.86	54.82	59.62	50.28	32.45	42.77	27.39	26.18	28.81
	Ours	66.31	62.84	70.27	55.76	42.31	46.82	35.48	39.29	34.28
Target	Training Data	38.45	40.27	43.94	11.45	10.35	11.22	5.02	8.66	6.17
	ImageNet	40.42	43.88	41.72	14.66	10.28	13.43	5.82	10.39	11.25
	PBBA [23]	42.67	55.66	49.24	25.83	15.38	20.44	6.73	17.22	13.88
	Knockoff [22]	48.28	52.89	54.27	30.87	16.92	19.56	12.83	22.37	15.26
	DaST [33]	50.17	52.84	51.29	29.93	16.28	21.44	10.84	15.81	13.92
	Ours	59.29	57.28	64.46	33.81	29.89	25.77	17.23	21.44	19.37

‘min-max’ game with \mathcal{L}_d , $\text{CE}(\cdot)$ indicates the cross entropy loss between the prediction of S and input ground-truth label i . Thus by virtue of such an alternately minimization of these two loss functions, the substitute model S can learn to mimic the outputs from the target model T . Further promoted by DDG and AST, with the generated data and adversarial examples, the unified substitute training loss \mathcal{L}_S and generator loss \mathcal{L}_G to train S and G are defined as,

$$\mathcal{L}_S = \mathcal{L}_d + \mathcal{L}_d^{adv} \quad (7)$$

$$\mathcal{L}_G = \beta_1(\mathcal{L}_c + \mathcal{L}_c^{adv}) + \beta_2\mathcal{L}_{rec} + \beta_3\mathcal{L}_{div} \quad (8)$$

where \mathcal{L}_d^{adv} is defined as the same as \mathcal{L}_d in Eq. 5 measuring the distance between the output from T and S using the adversarial examples as inputs. \mathcal{L}_c^{adv} is defined as \mathcal{L}_c in Eq. 6 to constrain the generation with adversarial examples as input, \mathcal{L}_{rec} and \mathcal{L}_{div} are proposed to enhance the diversity of generated data, which will be elaborated later. β_1, β_2 , and β_3 are the balanced hyper-parameters for DDG. Overall, the whole training process is illustrated in Alg.1.

4. Experiments

4.1. Experiment Setup

Datasets and target model. 1) MNIST [17]: The attacked model is pre-trained on AlexNet [14], VGG-16 [27], and ResNet-18 [10]. The default substitute model is a network with 3 convolutional layers. 2) CIFAR-10 [13]: The attacked is pre-trained on AlexNet, VGG-16, and ResNet-18. The default substitute model is VGG-13. 3) CIFAR-100 [13]: The attacked is pre-trained on VGG-19 and ResNet-50. The default substitute model is ResNet-18. 4) Tiny ImageNet [25]: The attacked is pre-trained on ResNet-50. The substitute model is ResNet-34.

Competitors. To verify the efficacy of the proposed method, we compare our attacking results with data-free black-box attack, *i.e.*, DaST [33], and several black-box

attacks which require real data, such as PBBA [23] and Knockoff [22]. We also conduct substitute training using the original training data of the attacked model, and utilize ImageNet [25] to learn the substitute model.

Implementation details. We use Pytorch for Implementation. We utilize Adam to train our substitute model, generator, and reconstruction net from scratch, and all weights are randomly initialized using a truncated normal distribution with std of 0.02. The initial learning rates of all networks are set as 0.0001, they are gradually decreased to zero from the 80th epoch, and stopped after the 150th epoch. We set the mini-batch size as 500, the hyper-parameters β_1, β_2 , and β_3 are equally as 1. Our model is trained by one NVIDIA GeForce GTX 1080Ti GPU. We apply PGD [20] as the default method to generate adversarial images during the AST and evaluation. We also utilize FGSM [8], BIM [15] and C&W [2] to conduct attack for extensive experiments.

Evaluation metrics. Considering there exist two different scenarios as proposed in DaST [33], *i.e.*, only get the output label from the target model and access the output probability well, and we name these two scenarios as Probability-based and Label-based. In the experiments, we report the attack success rates (ASRs) of the adversarial examples generated by the substitute model to attack the target black-box model. Following the setting in DaST [33], in the non-target attack setting, we only generate adversarial examples on the images classified correctly by the attacked model. For target attacks, we only generate adversarial examples on the images which are not classified to the specific wrong labels. For a fair comparison, during all the adversarial example generation, we restrict the perturbation $\epsilon = 8$. We conduct five times over each testing, and report the average results.

4.2. Black-box Attack Results

We evaluate our method with competitors over four datasets and one online machine learning platform for both target and non-target attack settings. As shown in Tab. 1, Tab. 2, and Tab. 3, we conduct extensive compari-

Table 2. Comparing ASRs results using the label as target model output among our proposed method and competitors over several datasets.

	Dataset	MNIST			CIFAR-10			CIFAR-100		Tiny ImageNet
	Target Model	AlexNet	VGG-16	ResNet-18	AlexNet	VGG-16	ResNet-18	VGG-19	ResNet-50	ResNet-50
Non-Target	Training Data	17.45	20.11	24.50	13.76	10.43	13.05	5.01	8.58	7.32
	ImageNet	18.26	23.77	22.56	15.83	12.73	14.11	8.38	11.28	13.29
	PBBA [23]	22.45	28.18	29.00	21.84	13.63	17.66	11.48	16.33	15.37
	Knockoff [22]	25.39	33.18	37.72	20.16	20.74	19.87	16.48	18.31	22.33
	DaST [33]	26.51	29.22	35.81	25.18	19.34	23.01	17.34	17.27	16.28
Ours	31.74	32.70	40.96	29.44	26.92	23.38	23.48	27.88	28.31	
Target	Training Data	15.53	12.55	10.88	9.92	10.24	9.09	3.97	6.44	4.92
	ImageNet	14.29	14.81	15.70	11.01	12.22	9.32	4.82	8.56	7.02
	PBBA [23]	15.26	19.86	18.53	12.84	11.33	10.48	6.91	7.33	8.61
	Knockoff [22]	19.48	23.74	17.85	16.38	12.80	13.91	9.48	9.52	10.65
	DaST [33]	20.03	21.48	19.33	15.72	15.92	14.83	7.48	10.39	10.31
Ours	25.56	27.64	21.83	21.66	18.67	17.90	12.47	16.26	13.39	

Table 3. Comparing ASRs results among our proposed method and competitors for attacking the Microsoft Azure example model.

	Method	Probability-based	Label-based
Non-Target	PBBA [23]	82.34	80.29
	Knockoff [22]	88.91	92.88
	DaST [33]	90.63	96.97
	Ours	96.73	98.91
Target	PBBA [23]	39.23	49.39
	Knockoff [22]	46.97	63.99
	DaST [33]	45.66	65.91
	Ours	57.92	69.81

son with multiple target models for each dataset under both probability-based and label-based scenarios.

Comparisons with the real data for substitute training.

Here we study the substitute training for attacking with real images, as listed in Tab. 1 and Tab. 2, we directly use the original training data of the target model or ImageNet to apply substitute training instead of the synthesized. The results show that the real image can let the substitute model learn a little from the target and may higher accuracy on the classification, but weaker attack strength compared to the generated data. We believe that this problem is caused by the number and diversity limitation of the real images, which may lead to failure of the substitute model learning and mimicking from the target one. Thus, we propose a DDG strategy to synthesize large-scale and diversified data.

Comparisons with the state-of-the-art. As shown in Tab. 1 and Tab. 2, we compare our method with black-box attacks. For both non-target and target attacking settings, our method achieves the best ASRs over Probability-based and Label-based scenarios under all datasets. In addition, compared to similar generative DaST, our method significantly outperforms it with a large margin. The results verify the efficacy of proposed method to encourage the substitute model better approximate the target’s decision bound-

Table 4. ASRs results of variants of the proposed attack method. The components are overlaid gradually with the rows. The target model is based on AlexNet for MNIST, and VGG-16 for CIFAR-10, the substitute models are the default ones according to the datasets. ‘C-100’ refers to the CIFAR-100 dataset.

	Components	Probability-based		Label-based	
		MNIST	C-100	MNIST	C-100
Non-Target	Baseline	29.42	8.27	13.84	4.27
	+ ALNG	49.18	21.38	20.85	12.66
	+ N/LR	55.21	26.31	24.91	15.99
	+ ICR	62.82	31.27	28.20	19.94
	+ AST (Ours)	66.31	35.48	31.74	23.48
Target	Baseline	26.29	3.27	11.48	1.29
	+ ALNG	44.48	10.47	16.28	7.83
	+ N/LR	51.87	11.83	19.59	9.38
	+ ICR	54.01	14.89	22.48	11.03
	+ AST (Ours)	59.29	17.23	25.56	12.47

ary and achieve high ASRs for data-free black-box attack.

Comparisons with competitors on Microsoft Azure. To better evaluate the attack method ability under the real-world applications, we conduct experiments for attacking the online model on Microsoft Azure. Target at attacking the example MNIST model of the machine learning tutorial on Azure, we compare the results between our methods and competitors. The results shown in Tab. 3, indicate our method can achieve the best ASRs over the online model, which further prove the efficacy of our method under the real scenario without prior knowledge of the attack one.

4.3. Ablation Study

4.3.1 Quantitative Results

The efficacy of different components in the proposed method. To generate label-controlled and diversity data for substitute training and make the substitute model better fit the decision boundary of the target, our method applies the

Table 5. ASRs results applying various attacks to generate adversarial examples for AST under different attack evaluation in MNIST. The target model is AlexNet and the substitute model is the default. The 3rd and 4th columns represent applying FGSM to conduct AST, the last two columns utilize PGD for AST, and the row indicates the attack to evaluate. The ‘-P’ and ‘-L’ in table mean the Probability-based and Label-based scenarios, respectively.

	Attacks	FGSM [8]		PGD [20]	
		-P	-L	-P	-L
Non-Target	FGSM [8]	70.26	36.29	57.35	33.10
	BIM [15]	66.38	36.97	68.45	29.58
	PGD [20]	62.63	33.72	66.31	31.74
	C&W [2]	49.92	20.91	46.93	22.02
	Target	FGSM [8]	50.82	27.38	29.48
	BIM [15]	67.29	32.33	44.82	18.14
	PGD [20]	52.77	33.39	39.29	25.56
	C&W [2]	49.38	20.39	28.57	19.66

following components: (a) ‘Adaptive label normalized generator’ (ALNG): generates data from input random noise with label-embedded vector; (b) ‘Noise/Label reconstruction’ (N/LR): applies a reconstruction net to reconstruct the input noise and given label; (c) ‘Inter-class diversity’ (ICR): constrains the distance of generated data in inter-classes; (d) ‘Adversarial substitute training’ (AST): uses the adversarial examples to further train the substitute model. In Tab. 4, we list the variants by adding the above components gradually, and the ‘Baseline’ refers to the framework using random noise and given label as the contacted input to directly generate data and train substitute model.

As the results shown in the Tab. 4, we can find that without ALNG, the substitute model ‘Baseline’ can hardly learn knowledge from the attacked one, which may due to the poor generation without powerful controlled label constrains. Besides, models with the N/LR and ICR can achieve much higher ASRs results compared to the former, these verify that more diverse label-controlled generated data can make the substitute models learn more knowledge from the target. To ensure that the substitute model approximates the attacked decision boundary, the AST technique is applied to generate adversarial examples as the boundary data to make the substitute model mimic the attacked one and further boost the attack results. The ASRs results clearly demonstrate the important roles of components for data-free black-box attack.

The affect of different attacks. Consider that we require adversarial examples during the substitute training, here we evaluate the affect of attack method for our algorithm. As shown in Tab. 5, the column means the attacks to generate adversarial examples for substitute training, and the row refers to the method to attack the target model in evaluation. The results indicate that different attacks may not have obvious impacts on our method, which means that using dif-

Table 6. ASRs results with different substitute models attacking VGG-16 trained on CIFAR-10. The ‘-P’ and ‘-L’ in table mean the Probability-based and Label-based scenarios, respectively.

	Non-Target Attack		Target Attack	
	-P	-L	-P	-L
AlexNet [14]	39.78	22.57	24.90	18.45
VGG-13 [27]	42.31	26.92	29.89	18.67
VGG-16 [27]	45.24	25.28	30.41	22.46
VGG-19 [27]	45.92	27.69	32.59	21.94
ResNet-18 [10]	49.28	26.83	33.20	20.58
ResNet-34 [10]	48.94	28.72	30.48	20.4

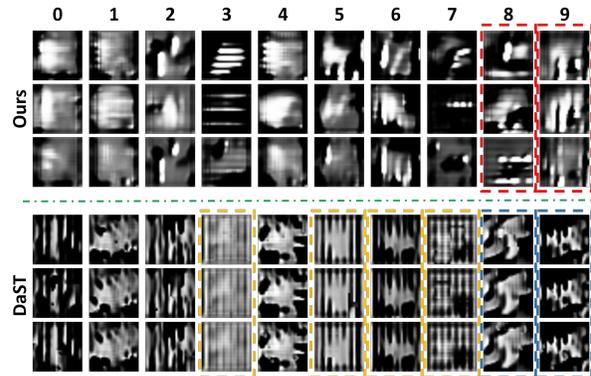


Figure 3. Samples of generated images on MNIST. The upper part represents the images generated from ours, and the below from DaSt. Left to right represents 0–9 hand-written number classes, and the samples belong to the same column have the same label.

ferent attacks for substitute training hardly affects the final attacking results. Thus, our method is effective under various attacks and there is no need to restrict the attack method between the substitute training and evaluation.

The impact of different substitute model architecture. We aim to achieve a successful attack for black-box under data-free condition, thus we have no prior knowledge of the attacked model structure. To further evaluate the impact of different substitute model architecture, we apply several substitute models for the same attacked one, which is VGG-16 pre-trained on CIFAR-10. As shown in Tab. 6, we try various architecture as the substitute model, i.e., AlexNet, VGGNet, and ResNet, and the results show that there does not exist the most suitable structure which can achieve the best ASRs under all settings. Except for the simplest AlexNet, the others reach similar high ASRs results, which demonstrates that the different substitute model architecture may not have a huge impact on the attack strength, but still recommend choosing a deeper network.

4.3.2 Qualitative Results

Our model can improve the diversity of generated data across different categories. (1) As for the generated data

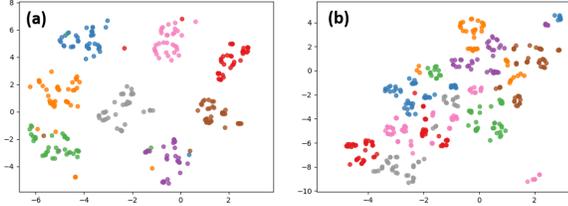


Figure 4. Visualization of generated data in 8 classes (one color indicates one category) using t-SNE [19] on CIFAR-100. (a) Data generated by DDG module. (b) Data generated by the DaST.

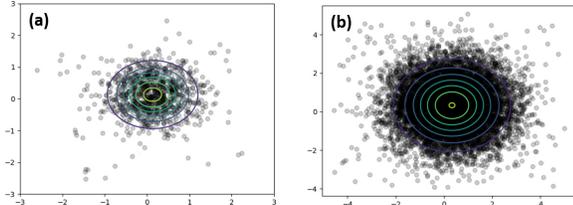


Figure 5. The distribution of data in one class on MNIST. (a) Original real data in MNIST. (b) Data generated by our DDG module.

shown in Fig. 3, we illustrate the data synthesized in different ways. It is clear that, compared with ours, the generated data of DaST without ALNG, N/LR, and ICD strategies, are much more similar in extra-classes, such as the data in yellow dashed boxes have the approximate vertical lines. (2) In terms of feature, we visualize the feature distribution of synthesized data extracted by the target model in Fig. 4. Comparing the DaST with ours, it is obvious that our generated data is widely distributed in the feature space with clearer categorical differences. While, the data generated by DaST have relatively small gap between the data classes and concentrate on the part of the feature space, which is not conducive for the substitute learning and decreases the attack strength. These also further verify the importance of generated data distribution for substitute training.

Our model can generate diverse data of each class. (1) As shown in Fig. 3, compared with ours, the DaST generates more similar data in inter-classes, such as the synthetic data are similar in the blue dotted box but various in the red. (2) We also visualize the distribution of the same class data collected from MNIST and generated by our method in Fig. 5. In terms of the amount of data, ours is much larger than MNIST. At the same time, within the class, the data we generate is more widely distributed. These qualitative results demonstrate that our method can generate more diverse data of each class, to further encourage the substitute model to learn from the target.

Our AST strategy can improve the consistency of the decision boundary between the substitute and target models. As shown in Fig. 6, the boundaries of the target and substitute models are visualized by input data features using t-SNE. Comparing Fig. 6(a) and Fig. 6(b), although they are

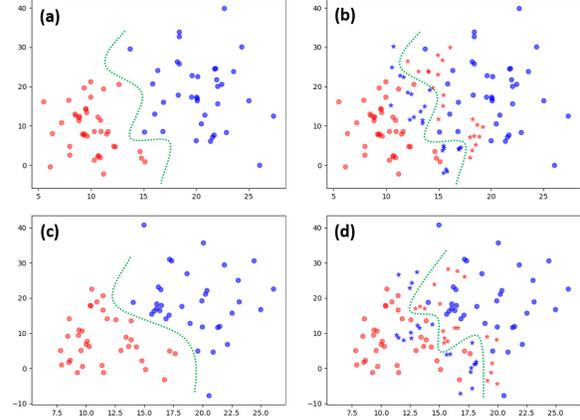


Figure 6. Visualization of decision boundaries between two classes (one color indicates one category) in CIFAR-10. The circles represent the normal data, stars mean the adversarial examples, and green dashed lines are the decision boundaries. (a) The real data on target model. (b) The real data and corresponding adversarial examples on target model. (c) The generated data on substitute model. (d) The generated data and corresponding adversarial examples on substitute model. Best viewed in color and zoomed in.

the same model, the decision boundary is more clear with the adversarial examples around the surface, which demonstrate adversarial examples can help to precisely identify the decision boundary. Meanwhile, it is clear that, compared to Fig. 6(b), the visualized decision boundary of substitute model with adversarial examples as in Fig. 6(c) is more approximated to the target model as shown in Fig. 6(a), and this intuitively verifies the efficacy of AST to further encourage the mimicking of target ‘behaviour’.

5. Conclusion

This paper focuses on the distribution of generated data for substitute training on black-box attack. It proposes a unified substitute model training framework, which contains a diverse data generation module (DDG) and an adversarial substitute training strategy (AST). DDG can generate label-controlled and diverse data to train substitute model. AST utilizes adversarial examples as boundary data to make the substitute model better fit the decision boundary of the target. Extensive experiments are conducted and results show the method can achieve high attack performance.

6. Acknowledgement

This work is supported by NSFC Projects (U62076067), Science and Technology Commission of Shanghai Municipality Projects (19511120700, 19ZR1471800), Shanghai Research and Innovation Functional Program (17DZ2260900), Shanghai Municipal Science and Technology Major Project (2018SHZDZX01) and ZJLab.

References

- [1] Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *arXiv preprint*, 2017.
- [2] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, 2017.
- [3] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, 2017.
- [4] Sizhe Chen, Zhengbao He, Chengjin Sun, and Xiaolin Huang. Universal adversarial attack on attention and the resulting dataset damagenet. *IEEE TPAMI*, 2020.
- [5] Minhao Cheng, Thong Le, Pin-Yu Chen, Jinfeng Yi, Huan Zhang, and Cho-Jui Hsieh. Query-efficient hard-label black-box attack: An optimization-based approach. *arXiv preprint*, 2018.
- [6] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *CVPR*, 2018.
- [7] Yinpeng Dong, Hang Su, Baoyuan Wu, Zhifeng Li, Wei Liu, Tong Zhang, and Jun Zhu. Efficient decision-based black-box adversarial attacks on face recognition. In *CVPR*, 2019.
- [8] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint*, 2014.
- [9] Chuan Guo, Jacob R. Gardner, Yurong You, Andrew Gordon Wilson, and Kilian Q. Weinberger. Simple black-box adversarial attacks. *arXiv preprint*, 2019.
- [10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [11] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. *arXiv preprint*, 2018.
- [12] Andrew Ilyas, Logan Engstrom, and Aleksander Madry. Prior convictions: Black-box adversarial attacks with bandits and priors. *arXiv preprint*, 2018.
- [13] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.
- [14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.
- [15] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.
- [16] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint*, 2017.
- [17] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [18] Bo Luo, Yannan Liu, Lingxiao Wei, and Qiang Xu. Towards imperceptible and robust adversarial example attacks against neural networks. *arXiv preprint arXiv:1801.04693*, 2018.
- [19] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.
- [20] Aleksander Madry, Aleksandar Makelov, and Ludwig Schmidt. Towards deep learning models resistant to adversarial attacks. *arXiv preprint*, 2017.
- [21] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.
- [22] Tribhuvanesh Orekondy, Bernt Schiele, and Mario Fritz. Knockoff nets: Stealing functionality of black-box models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4954–4963, 2019.
- [23] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, 2017.
- [24] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *2016 IEEE European symposium on security and privacy (EuroSP)*, 2016.
- [25] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.
- [26] Yucheng Shi, Siyu Wang, and Yahong Han. Curls whey: Boosting black-box adversarial attacks. In *CVPR*, 2019.
- [27] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [28] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint*, 2013.
- [29] Florian Tramèr, Fan Zhang, and Ari Juels. Stealing machine learning models via prediction apis. In *25th USENIX Security Symposium (USENIX Security 16)*, 2016.
- [30] Chun-Chen Tu, Paishun Ting, Pin-Yu Chen, Sijia Liu, Huan Zhang, Jinfeng Yi, Cho-Jui Hsieh, and Shin-Ming Cheng. Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks. In *AAAI*, 2019.
- [31] Hongxu Yin, Pavlo Molchanov, Jose M. Alvarez, and Zhizhong Li. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *CVPR*, 2020.
- [32] Jaemin Yoo, Minyong Cho, Taebum Kim, and U Kang. Knowledge extraction with no observable data. In *NeurIPS*, 2019.
- [33] Mingyi Zhou, Jing Wu, Yipeng Liu, Shuaicheng Liu, and Ce Zhu. Dast: Data-free substitute training for adversarial attacks. In *CVPR*, 2020.

- [34] Wen Zhou, Xin Hou, Yongjun Chen, Mengyun Tang, Xiangqi Huang, Xiang Gan, and Yong Yang. Transferable adversarial perturbations. In *ECCV*, 2018.
- [35] Yuankun Zhu, Yueqiang Cheng, Husheng Zhou, and Yantao Lu. Hermes attack: Steal dnn models with lossless inference accuracy. *arXiv preprint arXiv:2006.12784*, 2020.