# Power of Diversity: Enhancing Data-Free Black-Box Attack with Domain-Augmented Learning

**Yang Wei** [1], **Jingyu Tan** [1], **Guowen Xu** [2], **Zhuoran Ma** [3*], **Zhuo Ma** [3], **Bin Xiao** [1, 4*]

[1] School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, China
[2] School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China
[3] School of Cyber Engineering, Xidian University, Xi'an, China
[4] Jinan Inspur Data Technology Co., Ltd., Jinan, China
weiyang@cqupt.edu.cn, s230231105@stu.cqupt.edu.cn, guowen.xu@foxmail.com,
emmazhr@163.com, mazhuo@mail.xidian.edu.cn, xiaobin@cqupt.edu.cn

## Abstract

Substitute training-based data-free black-box attacks pose a significant threat to enterprise-deployed models. These attacks use a generator to synthesize data and query APIs, then train a substitute model to approximate the target model's decision boundary based on the returned results. However, existing attack methods often struggle to produce sufficiently diverse data, particularly for complex target models and extensive target data domains, severely limiting their practical application. To address this gap, we design domain-augmented learning to improve the quality of the synthetic data domain (SDD) generated by the generator from two perspectives. Specifically, (1) To broaden the SDD's coverage, we introduce textual semantic embeddings into the generator for the first time. (2) To enhance the SDD's discretization, we propose a competitive optimization strategy that forces the generator to self-compete, along with heterogeneity excitation to overcome the constraints of information entropy on diversity. Comprehensive experiments demonstrate that our method is more effective. In non-targeted attacks on the CIFAR-10 and Tiny-ImageNet datasets, our method outperforms the state-of-the-art by 14% and 7% in attack success rate, respectively.

## 1 Introduction

In recent years, machine learning technology has been widely applied across various fields, including self-driving cars, intelligent medical diagnostics, etc., its influence is pervasive. With the rapid advancement of these technologies, Machine Learning as a Service (MLaaS) has emerged. It provides access to complex machine learning models for users without a deep learning background. Therefore, more enterprises are deploying pre-trained machine learning models on cloud platforms and making APIs avaiable to make machine learning services easy to use.

However, this convenience also brings new security challenges. Recent studies indicate that models with public APIs may be vulnerable to black-box attacks based on substitute training. In this type of attacks, attackers query the API and use the API's predictions to train a substitute model that closely approximates the functions of the target model, and then use it to create adversarial samples that attack the target model, resulting in a transferable attack(Szegedy et al. 2013; Goodfellow, Shlens, and Szegedy 2014a; Tramèr et al. 2017). It can be divided into three types: (1) The first type involves directly using real public data to query APIs for substitute training, such as Knockoff (Orekondy, Schiele, and Fritz 2019; Jindal et al. 2024; Papernot et al. 2017). (2) Due to protection of data privacy and the difficulty in obtaining suitable real data, the second type uses the generator to create synthetic data for querying. This type is based on the "min-max" game framework, exemplified by DaST (Zhou et al. 2020; Yu and Sun 2022; Wang et al. 2021; Zhang et al. 2022; Wei et al. 2023), which enhances the attack's effectiveness by simulating the adversarial mechanism of GANs (Goodfellow et al. 2014). (3) The third type is also using synthetic data for querying, but it's based on gradient estimation, such as DFME (Truong et al. 2021; Kariyappa, Prakash, and Qureshi 2021; Beetham et al. 2023; Sanyal, Addepalli, and Babu 2022), which gives the generator extra information by estimating the gradients of the target model. Nevertheless, as the performance of models provided by enterprises gradually improves due to factors such as the increasing complexity of model structures and the expanded distribution of training datasets. This means that the deci-

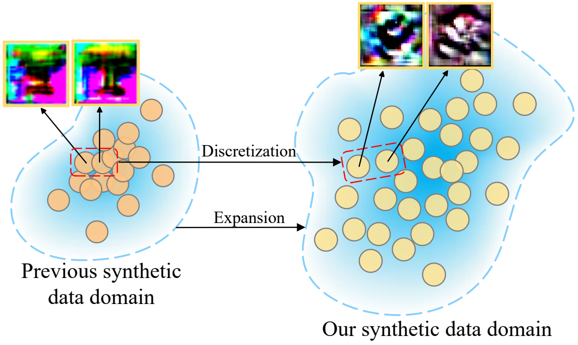

Figure 1: Distribution comparison of synthetic data domains. Previous work usually show high degree of concentration, resulting in limited data domain coverage. Our domain-augmented learning can discretize these data and expand the coverage of the synthetic data domain.

[*] Corresponding authors.

| Method | SSIM ↓ | ASRs (%) ↑ |
|---|---|---|
| DaST (Zhou et al. 2020) | 0.571 | 32.41 |
| DDG (Wang et al. 2021) | 0.563 | 32.94 |
| FE-DaST (Yu and Sun 2022) | 0.542 | 35.74 |
| TEDF (Zhang et al. 2022) | 0.511 | 43.61 |
| EDD (Wei et al. 2023) | 0.415 | 48.80 |

Table 1: Structural similarity (SSIM) and attack success attacks (ASRs) comparison for five methods on CIFAR-100 dataset. We calculate SSIM score of synthetic data, and lower SSIM score indicate higher data diversity. We select the same substitute model for these methods.

| Abbreviation | Description |
|---|---|
| $\mathcal{G}$ | generator |
| $\mathcal{S}$ | substitue model |
| $\mathcal{T}$ | target model |
| SDD | synthetic data domain |
| AGO | adversarial generation optimization |
| DO | diversity optimization |
| SSE | shallow semantic embedding |
| MSE | middle semantic embedding |
| DSE | deep semantic embedding |

Table 2: List of abbreviations used in this paper.

sion boundaries of the models are becoming highly non-linear. Additionally, since the target model is a black box for attackers, it limits the possibility of optimizing the substitute model based on its structure or gradient information. Therefore, to approximate the highly non-convex decision boundaries of the target model, the substitute model requires synthetic data with wide area coverage for training. However, existing generators struggle to produce effective data. We perform a simple experiment to show the relationship between data diversity and attack success rates as shown in Table 1. Obviously, the diversity of synthetic data seriously affects substitute model's ASRs. Therefore, we improve the data diversity from the perspective of the synthetic data domain (SDD) produced by generators.

**Challenges.** There are two main challenges.

- *How to broaden SDD's coverage*. Without real data supervision, the generator synthesize data absolutely based on the output of the target model, making it hard to synthesize data with wide area coverage from only input Gaussian noise.
- *How to improve SDD's discretization*. On the one hand, the optimization strategies of existing methods are limited to optimizing a single iteration of data, ignoring inter-iteration correlations. On the other hand, they typically use information entropy to evaluate synthetic data diversity. However, information entropy mainly reflects model uncertainty and cannot adequately represent diversity. These may cause the generator to produce data with higher uncertainty but insufficient discretization.

We propose Domain-Augmented Leraning inspired by the aforesaid challenges.

**Contributions.** The following concludes our contributions.

- Text modality information's semantic content can effectively compensate for the prior approaches' noise input constraint. However, the generator's training might become unstable if low-dimensional textual data is directly entered into it. We therefore propose Adaptive Semantic Embedding. With this approach, the text data is encoded using CLIP Text Encoder (Radford et al. 2021) to obtain high-dimensional semantic embedding. The embedding is subdivided into three levels so that the generator can gradually adapt to this beneficial information.
- Adaptive semantic embedding can extend SSD's coverage, but the synthetic data within the domain may still be centralized. We therefore propose two strategies. The first strategy, termed as Competition Optimization, constructs a "loser and winner" game between previous and current iterations, enabling the generator to compete with itself. Compared to existing methods, the strategy further leverages the ignored information between iterations. Besides, we design a regularization strategy, termed as heterogeneity excitation, enforce the generator to maximize the ratio of distance between the synthetic data and input noise, and the ratio of the distance between the synthetic data and semantic embedding. This strategy assists with information entropy to better optimize the generator.
- The comprehensive experiments and synthetic data distribution analysis demonstrate the effectiveness of the proposed domain-augmented learning against the state-of-the-art attacks. Specifically, in target attacks on CIFAR-10 and Tiny-ImageNet datasets, our method outperforms the SOTA by 8.66% and 7.24% in attack success rate, respectively.

As shown in Figure 1, domain-augmented learning expand the coverage of the synthetic data domain, and increase its discretization. This aims to support effective training for substitute model, thereby enhancing attack effectiveness.

# 2 Limitation of Prior Works

For substitute training-based data-free black-box attacks, the key to constructing a substitute model $\mathcal{S}$ that can approximate the decision boundary of the target model $\mathcal{T}$ lies in whether the generator $\mathcal{G}$ can generate sufficiently diverse samples without real data supervision. However, for the above issue, existing works (Zhou et al. 2020; Yu and Sun 2022; Wang et al. 2021; Wei et al. 2023; Zhang et al. 2022) still has shortcomings. In this section, we mainly analyze three limitations of previous work and their negative effects. The abbreviations used in this paper are listed in Table 2.

## 2.1 Ignoring the Impact of Initial Distribution

Firstly, in the process of the data-free substitute training, we cannot supervise the training of $\mathcal{G}$ with effective data, relying solely on the output of $\mathcal{T}$ as guidance. However, the output of $\mathcal{T}$ is often highly concise, such as labels or probability information predicted for images. This information is very limited and cannot achieve the supervision effect similar to real data on $\mathcal{G}$, resulting in poor generation performance. The problem can be formulated as follows:

$$x = G(z), z \sim N(0, I), P(x) \approx N(0, I), \tag{1}$$

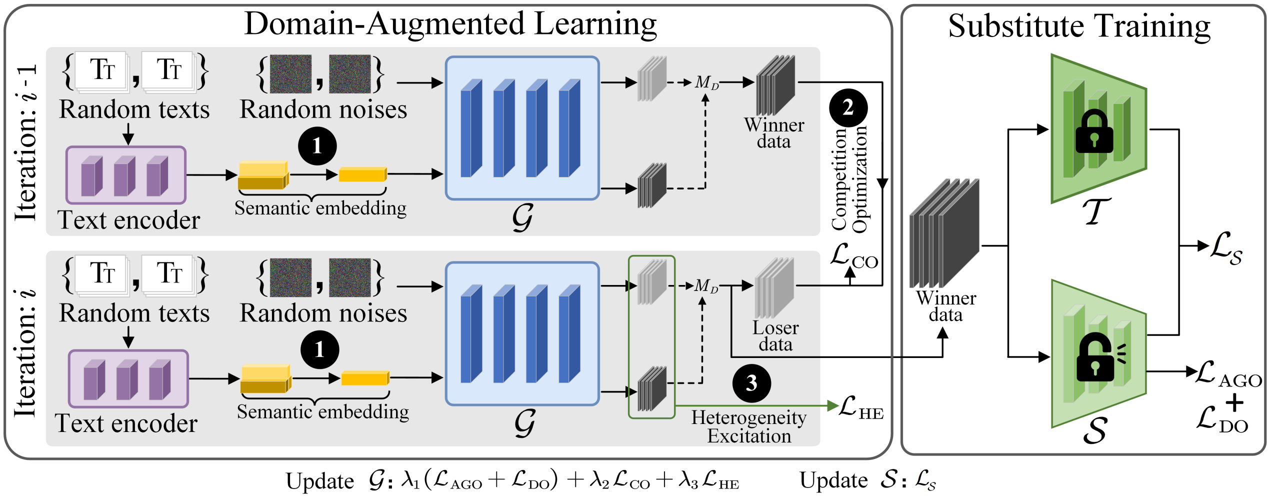
Domain-Augmented Learning
Substitute Training
Iteration: $i$-1
Iteration: $i$
Random texts
Random noises
Text encoder
Semantic embedding
1
$\mathcal{G}$
$M_D$
Winner data
2
Competition Optimization
$\mathcal{L}_{\text{CO}}$
Loser data
3
Heterogeneity Excitation
$\mathcal{L}_{\text{HE}}$
Winner data
$\mathcal{T}$
$\mathcal{S}$
$\mathcal{L}_S$
$\mathcal{L}_{\text{AGO}}$ + $\mathcal{L}_{\text{DO}}$
Update $\mathcal{G}$: $\lambda_1(\mathcal{L}_{\text{AGO}} + \mathcal{L}_{\text{DO}}) + \lambda_2 \mathcal{L}_{\text{CO}} + \lambda_3 \mathcal{L}_{\text{HE}}$
Update $\mathcal{S}$: $\mathcal{L}_S$

Figure 2: The left subplots show the proposed domain-augmented learning. ❶, ❷ and ❸ represents the proposed adaptive semantic embedding, competition optimization and heterogeneity excitation, respectively. The right subplot demonstrate the substitute training-based data-free black-box attacks. Finally, we update $\mathcal{G}$ and $\mathcal{S}$ by Equations 9 and 10.

where $z$ is the input noise from the Gaussian distribution $N(0, 1)$. $P(x)$ represent the distribution of $x$.

In Equation 1, $x$ should ideally cover a complex data distribution. However, due to the limited capacity of $\mathcal{G}$, $P(x)$ remains highly similar to the distribution of $z$, which fails to achieve significant distribution expansion and sample diversification. In summary, this is hard to support the effective training of $\mathcal{S}$, further impacting attack effectiveness.

**Solution.** We need to improve the input strategy of $\mathcal{G}$ to enhance the coverage range of the synthetic data domain (SDD). We describe the strategy in detail in Section 3.2.

### 2.2 Limitations of Existing Optimization Strategies

Currently, the optimization strategies of $\mathcal{G}$ can be roughly divided into two parts: adversarial generation optimization (AGO) and diversity optimization (DO). Specifically, AGO enhances $\mathcal{G}$ by maximizing the difference between the outputs of $\mathcal{S}$ and $\mathcal{T}$. On the other hand, DO improves $\mathcal{G}$ by maximizing the entropy of $\mathcal{S}$ on the synthetic data. However, both optimization methods have certain limitations, which are analyzed as follows:

**Ignoring information interaction between iterations.** Under the data-free condition, we should fully utilize all supervisable information for $\mathcal{G}$ optimization to enhance the diversity of synthetic data. However, existing AGO strategies fall short in this aspect. Specifically, these strategies are usually limited to optimizing single-iteration data, focusing only on the quality of data generated by $\mathcal{G}$ within the current iteration while neglecting the correlation and informational value between iterations. This isolated optimization way restricts the potential of $\mathcal{G}$ to utilize the results from previous and subsequent iterations to improve data discretization, making it difficult to generate wider and more diverse data.

**Solution.** We need to design a novel mechanism to enable $\mathcal{G}$ to fully utilize information between iterations. The specific design of the mechanism is referred to in Section 3.3.

**Using only information entropy as optimization direction.** Existing DO utilizes information entropy to assess the diversity of synthesized data to update $\mathcal{G}$. The goal of $\mathcal{S}$ is to approach the decision boundary of $\mathcal{T}$, with its output representing the feedback of $\mathcal{T}$ to some extent. During the optimization of $\mathcal{G}$, $\mathcal{S}$ takes synthetic data as input and gets information entropy based on its output, which can assist $\mathcal{G}$ to generate data relevant to the target model. However, as shown in Equation 2, information entropy only reflects the uncertainty of $\mathcal{S}$ regarding the synthesized data and does not fully reflect the data diversity. Hence, merely relying on information entropy as the direction of DO may result in $\mathcal{G}$ producing data with higher uncertainty but lower diversity, negatively impacting the training effectiveness of $\mathcal{S}$ and further attack success rate.

$$H(X) = -\sum_{i=1}^{n} p\left(x_i\right) \log p\left(x_i\right), \tag{2}$$

where $x_i$ represents synthesized data, $n$ is the class number of $\mathcal{S}$, and $p(x_i)$ is the predicted probability of $\mathcal{S}$ for $x_i$.

**Solution.** We need to introduce more effective optimization scheme to assist the current optimization strategy, further increasing the discretization of the synthetic data domain. The scheme is extensively explained in Section 3.4.

In conclusion, for the existing optimization challenges, we propose two methods to increase SSD's discretization.

## 3 Methodology

In the previous sections, we analyzed three main limitations of $\mathcal{G}$ in the existing substitute training-based data-free black-box attacks: simplistic input distribution, noninteractive inter-iteration information, and single information entropy optimization. To address these issues, we propose domain-augmented learning to improve $\mathcal{G}$ more suitable for data-free substitute training.

## 3.1 Overview

The proposed domain-augmented learning includes adaptive semantic embedding, competitive optimization strategy, and heterogeneity excitation. These methods complement each other by increasing the breadth of coverage and degree of discretization in the synthetic data domain, effectively promoting training of $\mathcal{S}$, and enhancing the attack success rate on $\mathcal{T}$. To begin, the adaptive semantic embedding enhances the breadth of the synthetic data domain by embedding semantic information into the process of data synthesis and utilizing the text's semantic richness to make up for the limitation of input noise. Then, the competitive optimization forces $\mathcal{G}$ to optimize generation effects in each iteration through a self-competition mechanism. Finally, the heterogeneity excitation assists the information entropy optimization, increasing the degree of discretization in the synthetic data domain. Figure 2 displays the overall framework. The following describes the specific details of each method's implementation.

## 3.2 Adaptive Semantic Embedding

In order to improve the initial distribution's richness to enhance the breadth of the synthetic data domain, we introduced copious semantic embedding information for $\mathcal{G}$.

Specifically, we first collect a large amount of random text information from sources, *e.g.*, news and social media, to ensure the richness of the text. Then, by a pre-trained CLIP Text Encoder, we encode these texts to extract high-dimensional Semantic Embedding (SE). These embeddings are fused with Gaussian noise features through a cross-attention mechanism to increase the richness of the inputs. To ensure $\mathcal{G}$'s stability when receiving SE in data-free substitute training attacks, we further refined the output of the Text Encoder into three levels: shallow semantic embedding (SSE), middle semantic embedding (MSE), and deep semantic embedding (DSE). Specifically, the shallow layers of CLIP Text Encoder capture basic text information, *e.g.*, words and simple phrases; the middle layers acquire more complex semantic relationships and phrase structures, which provides rich sentence components and semantic dependencies; and the deep layers obtain advanced semantic information of the text's overall meaning and abstract concepts (Peters et al. 2018; Jawahar, Sagot, and Seddah 2019).

During the optimization process of $\mathcal{G}$, we gradually introduce SE's different levels through the dynamically adjusted weighting factor $\alpha$, and then obtain the attention-weighted features (AF) generated by the cross-attention mechanism (CM) (Rombach et al. 2022). The process can be defined as:

$$\text{AF} = \text{CM}(\alpha \cdot \text{SE},\, z),\;\; \alpha \in \{0, 1\}, \tag{3}$$

where $z$ denotes Gaussian noise features.

In the initial stages of training, we select SSE to help $\mathcal{G}$ gradually adapt to the basic semantic embedding information. As $\mathcal{G}$'s generation capability improves, we select MSE to enable $\mathcal{G}$ to adapt to more complex semantic information. Finally, we use DSE to let $\mathcal{G}$ understand advanced semantic concepts. This adaptive embedding strategy allows $\mathcal{G}$ to effectively utilize semantic embedding information to expand the coverage of the synthetic data domain.

Algorithm 1: The proposed data-free black-box attack.

```
Input: Random text $T$, Text Encoder (TE), cross-attention
    mechanism (CM), adjusted weighting factor $\beta$, noise
    feature $z$, epochs $E$, $t$ iterations per epoch, and learning rates $\gamma_1, \gamma_2$.
Initialization: Model parameters $\theta_{\mathcal{G}}, \theta_{\mathcal{S}}$.
for each $e \in E$ do
    for each $i \in t$ do
        Get $\text{SE}_1, \text{SE}_2 \leftarrow \text{TE}(T_1), \text{TE}(T_2)$
        Get $\text{AF}_1, \text{AF}_2 \leftarrow \text{CM}(\beta \cdot \text{SE}_1, z_1), \text{CM}(\beta \cdot \text{SE}_2, z_2)$
        Generate two batches of data:
            $I_1, I_2 \leftarrow \mathcal{G}(\text{AF}_1), \mathcal{G}(\text{AF}_2)$
        Select the loser and winner data using $\mathcal{M}_D$
        Calculate $\mathcal{L}_m$, $\mathcal{L}_{\mathcal{S}} = \mathcal{L}_m$
        Update $\mathcal{S}$: $\theta_{\mathcal{S}} \leftarrow \theta_{\mathcal{S}} - \gamma_1 \nabla_{\theta_{\mathcal{S}}} \mathcal{L}_{\mathcal{S}}(\theta_{\mathcal{S}})$
        Calculate $\mathcal{L}_{\mathcal{G}} = \lambda_1(\mathcal{L}_{\text{AGO}} + \mathcal{L}_{\text{DO}}) + \lambda_2 \mathcal{L}_{\text{CO}} + \lambda_3 \mathcal{L}_{\text{HE}}$
        Update $\mathcal{G}$: $\theta_{\mathcal{G}} \leftarrow \theta_{\mathcal{G}} - \gamma_2 \nabla_{\theta_{\mathcal{G}}} \mathcal{L}_{\mathcal{G}}(\theta_{\mathcal{G}})$
    end for
end for
return $\theta_{\mathcal{S}}$
```

## 3.3 Competition Optimization

As described in Section 2.2, current AGO strategies only utilizes the current iteration data to optimize $\mathcal{G}$, neglecting the information interaction between iterations. Although adaptive semantic embedding can extend the breadth of the synthetic data domain, the synthetic data within the domain may still be centralized, exhibiting lower discretization and lacking sufficient diversity. Therefore, we propose the concept of competitive optimization to fully utilize the overlooked inter-iteration information to optimize $\mathcal{G}$. The core of competitive optimization is to gradually improve the discretization of the synthetic data domain by comparing the generation effect of $\mathcal{G}$ in different iterations. Specifically, during the $(i-1)$-th iteration, for the synthetic data $I_i^1$, $I_i^2$, which generated from two different batches of Gaussian sampling noise and randomized text, we use the diversity measure $M_D$ to evaluate them and obtain the corresponding $M_D(I_i^1)$, $M_D(I_i^2)$. The synthetic data with the larger $M_D$ value is selected as the winner of this iteration, while the one with the smaller $M_D$ value is the loser. In the $i$-th iteration, we repeat the above steps to obtain a new winner and loser. We require that the diversity of the loser in the $i$-th iteration exceed that of the winner in the $(i-1)$-th iteration. The loss function for competitive optimization can be expressed as:

$$\mathcal{L}_{\text{CO}} = \mathcal{M}_D(I_i) - \mathcal{M}_D(I_{i-1}), \tag{4}$$

where $I_i$ is the loser in the $i$-th iteration, and $I_{i-1}$ is the winner in the $(i-1)$-th iteration. For $M_D$, we use the negative of cosine similarity to measure the diversity degree of the synthesized data. Specifically, we calculate the cosine similarity matrix for a round of synthetic data. Then, we accumulate the upper triangular elements of this matrix and

| | Dataset | MNIST | | CIFAR-10 | | CIFAR-100 | | Tiny-ImageNet |
|---|---|---|---|---|---|---|---|---|
| | Target Model | VGG-16 | ResNet-18 | VGG-16 | ResNet-18 | VGG-19 | ResNet-50 | ResNet-50 |
| Non-targeted | Training Data | 30.53 | 34.03 | 22.36 | 31.74 | 16.53 | 19.49 | 13.79 |
| | ImageNet | 36.07 | 32.46 | 23.73 | 33.91 | 18.37 | 21.68 | 22.54 |
| | Knockoff | 58.38 | 65.82 | 31.58 | 39.40 | 27.73 | 29.55 | 29.99 |
| | DDG | 52.63 | 72.53 | 35.92 | 49.31 | 32.65 | 39.11 | 32.81 |
| | FE-DaST | 69.01 | 70.73 | 49.61 | 53.39 | 34.46 | 43.60 | 34.67 |
| | TEDF | 72.46 | 81.53 | 60.43 | 72.87 | 39.79 | 49.23 | 40.93 |
| | EDD | 73.48 | 69.57 | 72.19 | 71.22 | 40.14 | 53.31 | 41.74 |
| | Ours | **84.37** | **82.16** | **78.36** | **77.92** | **47.82** | **57.36** | **48.37** |
| Targeted | Training Data | 41.98 | 42.37 | 10.76 | 12.35 | 5.49 | 8.73 | 5.98 |
| | ImageNet | 42.81 | 43.35 | 11.86 | 13.67 | 6.31 | 10.97 | 12.15 |
| | Knockoff | 52.89 | 54.27 | 16.92 | 19.56 | 12.83 | 22.37 | 15.26 |
| | DDG | 38.15 | 55.57 | 31.30 | 0.93 | 15.24 | 15.48 | 13.23 |
| | FE-DaST | 53.34 | 56.62 | 35.74 | 31.53 | 18.08 | 19.12 | 19.65 |
| | TEDF | 62.57 | 63.58 | 50.83 | 49.53 | 35.67 | 31.59 | 35.26 |
| | EDD | 61.52 | 56.31 | 36.62 | 42.15 | 26.85 | 29.46 | 33.64 |
| | Ours | **70.77** | **74.48** | **59.24** | **58.19** | **37.61** | **38.81** | **42.68** |

Table 3: Comparing ASRs results (%) using the probability as the target model output among different attack methods.

take the negative of the cumulative sum. A larger $M_D$ value represents a higher degree of diversification.

Competitive optimization forces $\mathcal{G}$ to produce better generative effects in each iteration, thereby increasing the discretization of the synthetic data domain.

### 3.4 Heterogeneity Excitation

As described in Section 2.2, existing DO strategies often rely on information entropy to optimize $\mathcal{G}$, but information entropy is insufficient to comprehensively reflect the diversity of synthetic data and may even limit the generative capability of $\mathcal{G}$. To address this issue, we propose a regularization strategy for heterogeneity excitation. Specifically, we evenly divide the original noise $z$ into two noise groups, $z_1$ and $z_2$, and obtain two sets of random texts. These texts are processed through CLIP's Text Encoder to obtain semantic embeddings $t_1$ and $t_2$. The two groups of noise and semantic embeddings are then mapped by $\mathcal{G}$ to obtain synthetic data $I_1$ and $I_2$. We then enforce $\mathcal{G}$ to maximize the ratio of the distance between $I_1$ and $I_2$ to the distance between $z_1$ and $z_2$, and the ratio of the distance between $I_1$ and $I_2$ to the distance between $t_1$ and $t_2$. Heterogeneity Excitation can be expressed as:

$$\mathcal{L}_{\text{HE}} = \beta \left( \frac{d\left(I_1, I_2\right)}{d\left(z_1, z_2\right)} \right) + (1-\beta) \left( \frac{d\left(I_1, I_2\right)}{d\left(t_1, t_2\right)} \right), \tag{5}$$

where $\beta$ balances the loss components, and $d(*)$ denotes the Euclidean distance.

Heterogeneity excitation assists single information entropy to optimize $\mathcal{G}$, encouraging the generation of significantly diverse synthetic data to further enhance the discretization of the synthetic data domain.

### 3.5 Loss Functions

In this subsection, we describe the total loss functions of $\mathcal{G}$ and $\mathcal{S}$ respectively. For $\mathcal{G}$,

$$\mathcal{L}_m = \left\| \mathcal{T}\left(I\right), \mathcal{S}\left(I\right) \right\|_F, \tag{6}$$

$$\mathcal{L}_{\text{AGO}} = e^{-\mathcal{L}_m}, \tag{7}$$

$$\mathcal{L}_{\text{DO}} = -H(\mathcal{S}(I)), \tag{8}$$

where $\mathcal{L}_m$ measures the distance between the outputs of $\mathcal{T}$ and $\mathcal{S}$, and $I$ represents the data that wins in the competitive optimization. $\mathcal{L}_{\text{AGO}}$ denotes the adversarial optimization loss, and $e^{-\mathcal{L}_m}$ suggests a "min-max" game related to $\mathcal{L}_m$. $\mathcal{L}_{\text{DO}}$ indicates the diversity optimization loss, and $H(*)$ denotes information entropy in Equation 2. Overall, we minimize the following loss function to update $\mathcal{G}$:

$$\mathcal{L}_{\mathcal{G}} = \lambda_1(\mathcal{L}_{\text{AGO}} + \mathcal{L}_{\text{DO}}) + \lambda_2 \mathcal{L}_{\text{CO}} + \lambda_3 \mathcal{L}_{\text{HE}}, \tag{9}$$

where $\lambda_1$, $\lambda_2$, and $\lambda_3$ are the balancing hyper-parameters for each loss term. $\mathcal{L}_{\text{CO}}$ and $\mathcal{L}_{\text{HE}}$ comes from equations 4 and 5, respectively, which can enable $\mathcal{G}$ to produce synthetic data with a wider coverage and higher distribution dispersion.

For $\mathcal{S}$, we use the basic loss function as in FE-DaST (Yu and Sun 2022).

$$\mathcal{L}_{\mathcal{S}} = \mathcal{L}_m. \tag{10}$$

The entire training process is described in Algorithm 1.

## 4 Experiment

### 4.1 Experiment Setting

**Datasets and Model Architectures.** We test our method on public datasets and classic models. 1) MNIST (LeCun et al. 1998): $\mathcal{T}$ is pre-trained on VGG-16 (Simonyan 2014), and ResNet-18 (He et al. 2016). $\mathcal{S}$ is a network with 3 convolutional layers. 2) CIFAR-10 (Krizhevsky, Hinton et al. 2009): $\mathcal{T}$ is pre-trained on VGG-16 and ResNet-18. $\mathcal{S}$ is VGG-13. 3) CIFAR-100 (Krizhevsky, Hinton et al. 2009): $\mathcal{T}$ is pre-trained on VGG19 and ResNet50. $\mathcal{S}$ is ResNet-18. 4) Tiny-Imagenet (Russakovsky et al. 2015a): $\mathcal{T}$ is pre-trained on ResNet50. $\mathcal{S}$ is ResNet-34.

**Comparison methods.** To assess the effectiveness of our method, we compare the attack results with four other data-free black-box attack methods: FeDaST (Yu and Sun 2022),

|  | Dataset | MNIST |  | CIFAR-10 |  | CIFAR-100 |  | Tiny-ImageNet |
|---|---|---|---|---|---|---|---|---|
|  | Target Model | VGG-16 | ResNet-18 | VGG-16 | ResNet-18 | VGG-19 | ResNet-50 | ResNet-50 |
| Non-targeted | Training Data | 21.37 | 24.97 | 11.47 | 13.83 | 4.93 | 9.04 | 7.84 |
|  | ImageNet | 22.98 | 23.64 | 13.53 | 13.97 | 8.73 | 12.37 | 12.93 |
|  | Knockoff | 33.18 | 37.72 | 20.74 | 19.87 | 16.48 | 18.31 | 22.33 |
|  | DDG | 37.87 | 49.32 | 42.32 | 16.17 | 25.79 | 32.94 | 20.59 |
|  | FE-DaST | 42.22 | 56.49 | 32.54 | 22.12 | 25.51 | 35.74 | 23.1 |
|  | TEDF | 47.53 | 61.72 | 45.85 | 41.72 | 28.92 | 43.61 | 29.37 |
|  | EDD | 50.46 | 68.49 | 67.68 | 49.97 | 26.31 | 48.80 | 37.93 |
|  | Ours | **78.28** | **80.17** | **73.51** | **64.48** | **30.18** | **52.73** | **42.56** |
| Targeted | Training Data | 11.63 | 12.37 | 11.62 | 9.85 | 5.53 | 7.35 | 4.35 |
|  | ImageNet | 15.74 | 15.37 | 13.62 | 10.57 | 5.97 | 7.55 | 8.94 |
|  | Knockoff | 23.74 | 17.85 | 12.80 | 13.91 | 9.48 | 9.52 | 10.65 |
|  | DDG | 30.65 | 27.69 | 40.40 | 20.91 | 15.68 | 14.69 | 6.34 |
|  | FE-DaST | 32.37 | 22.66 | 26.98 | 17.39 | 15.37 | 20.23 | 7.96 |
|  | TEDF | 42.57 | 37.24 | 45.32 | 40.53 | 22.41 | 26.54 | 22.43 |
|  | EDD | 49.29 | 22.54 | 47.26 | 23.02 | 20.03 | 29.86 | 22.19 |
|  | Ours | **62.11** | **54.26** | **52.37** | **43.94** | **24.59** | **35.69** | **28.94** |

Table 4: Comparing ASRs results using the label as the target model output among different attack methods.

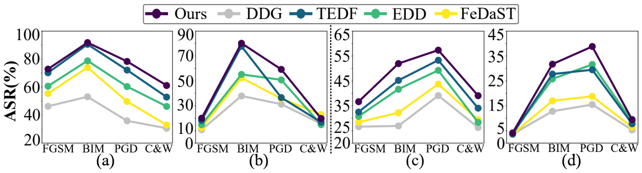

Figure 3: Using probability as the target model output, different attack methods are compared in terms of ASR results, alongside various white-box adversarial example generation methods. Subgraphs (a) and (b) represent non-targeted and targeted attacks in CIFAR-10, and subgraphs (c) and (d) show the corresponding attacks in CIFAR-100.

DDG (Wang et al. 2021), TEDF (Zhang et al. 2022), and EDD (Wei et al. 2023). And a black-box attack method that requires real data, Knockoff (Orekondy, Schiele, and Fritz 2019). Additionally, we also use the original training dataset of the target model and ImageNet (Russakovsky et al. 2015b) to train $\mathcal{S}$ for black-box attacks.

**Implementation details.** Our $\mathcal{S}$ and $\mathcal{G}$ are trained using the Adam with a learning rate of 0.0001. We set the mini-batch size as 500, and the $\mathcal{S}$ for 120 epochs on MNIST, 300 epochs on CIFAR-10/100 datasets, and 400 epochs on Tiny-ImageNet dataset. In Adaptive Semantic Embedding (ASE), we evenly divide the epochs into three stages based on the dataset. During each stage, the weight factor $\delta$ gradually grows from 0 to 1. The hyper-parameter $\beta$ in Heterogeneity Excitation (HE) is set to 0.5, while the hyper-parameters $\lambda_1$, $\lambda_2$, and $\lambda_1$ are equally as 1. In the evaluation phase, we use PGD (Madry et al. 2017) as the main white-box attack to assess ASR. Furthermore, we also evaluate our method using other classic attacks, *e.g.*, FGSM (Goodfellow, Shlens, and Szegedy 2014b), BIM (Kurakin, Goodfellow, and Bengio 2018), and C&W (Carlini and Wagner 2017). We use the same generator in ProGAN (Karras et al. 2017).

**Evaluations.** We evaluate our method in two scenarios defined by DaST: one where the attacker has access to only the probability outputs of the $\mathcal{T}$ (Probability), and another where only label outputs are accessible (Label). Attack Success Rates (ASRs) are used to measure the effectiveness of black-box attacks. For non-targeted attacks, we generate adversarial samples for correctly classified images; for targeted attacks, we create them for misclassified images.A uniform perturbation level $\epsilon$=8 is applied to all adversarial examples. To mitigate the effects of randomness, we conduct each experiment five times, recording the mean results. We also evaluate the substitute model's classification accuracy.

## 4.2 Attack Results

**Comparisons among the state-of-the-art works.** Tables 3 and 4 display a comparison of our method's performance with other attack methods, including a real-data-dependent attack and four data-free attacks. Our method shows superior performance compared to others in both non-targeted and targeted settings. It achieves the highest ASRs across all datasets, for both probability and label scenarios. Especially in the extreme scenario of target attack on the Tiny-ImageNet dataset with labels as outputs, our method can exceed SOTA by 6.51%. In addition, as shown in Figure 3, compared to other methods, our method also outperforms across four white-box attacks in the CIFAR-10 and CIFAR-100 datasets. The results demonstrate that our method significantly improves the quality of synthetic data domains, enabling effective substitute training and generating more powerful transferable adversarial examples.

**Accuracy comparisons between substitute and target models.** The key to implementing an effective transfer attack lies in the $\mathcal{S}$'s ability to accurately mimic the decision boundary of $\mathcal{T}$. Therefore, a high classification accuracy rate of $\mathcal{S}$ is an important indicator of the similarity between its decision boundary and that of the $\mathcal{T}$. As shown in Tables 5, our method outperform others with the highest ACC across

| | Dataset<br>Target Model | MNIST<br>VGG-16 | <br>ResNet-18 | CIFAR-10<br>VGG-16 | <br>ResNet-18 | CIFAR-100<br>VGG-19 | <br>ResNet-50 | Tiny-ImageNet<br>ResNet-50 |
|---|---|---|---|---|---|---|---|---|
| Probability | Training Data | 92.34 | 94.16 | 83.39 | 80.43 | 62.42 | 63.59 | 48.61 |
| | ImageNet | 94.82 | 95.11 | 86.49 | 82.34 | 66.73 | 69.89 | 50.13 |
| | DDG | 53.66 | 49.51 | 34.82 | 32.26 | 12.25 | 23.92 | 7.04 |
| | FE-DaST | 32.37 | 22.66 | 26.98 | 17.39 | 15.37 | 28.12 | 7.96 |
| | TEDF | 65.28 | 63.47 | 55.28 | 52.90 | 24.15 | 35.97 | 16.73 |
| | EDD | 70.35 | 60.76 | 65.15 | 43.98 | 24.89 | 42.17 | 15.96 |
| | Ours | **72.64** | **71.67** | **71.34** | **57.33** | **35.22** | **57.70** | **28.10** |
| Label | Training Data | 91.51 | 93.83 | 81.75 | 79.10 | 59.13 | 62.47 | 47.53 |
| | ImageNet | 92.73 | 94.15 | 86.12 | 80.62 | 65.37 | 67.64 | 49.68 |
| | DDG | 53.72 | 38.53 | 31.00 | 25.71 | 19.12 | 21.33 | 6.83 |
| | FE-DaST | 51.49 | 42.42 | 27.1 | 32.88 | 17.41 | 24.12 | 8.51 |
| | TEDF | 62.35 | 63.49 | 47.08 | 48.83 | 22.03 | 32.73 | 14.35 |
| | EDD | 67.76 | 58.64 | 55.27 | 43.73 | 24.17 | 40.92 | 14.79 |
| | Ours | **69.15** | **68.26** | **65.95** | **54.53** | **27.96** | **52.56** | **19.76** |

Table 5: Accuracy comparison of various attack methods under Probability and Label scenarios.

| | Scenario<br>Dataset | Probability<br>C-10 | <br>C-100 | Label<br>C-10 | <br>C-100 |
|---|---|---|---|---|---|
| Non-target | Ours | **78.36** | **57.36** | **73.51** | **52.73** |
| | w/o ASE | 71.21 | 48.93 | 67.47 | 46.92 |
| | w/o CO | 69.02 | 50.12 | 66.81 | 45.65 |
| | w/o HE | 69.25 | 49.52 | 64.05 | 46.35 |
| Target | Ours | **59.24** | **38.81** | **52.37** | **35.69** |
| | w/o ASE | 49.37 | 32.29 | 48.72 | 30.49 |
| | w/o CO | 47.90 | 32.23 | 47.89 | 31.42 |
| | w/o HE | 51.48 | 31.91 | 48.49 | 30.20 |

Table 6: Ablation Study of ASRs by cutting of different modules. "C-10" and "C-100" refers to CIFAR-10 and CIFAR-100 dataset. In C-10, $\mathcal{T}$ is VGG-16, $\mathcal{S}$ is VGG-13. In C-100, $\mathcal{T}$ is ResNet-50, $\mathcal{S}$ is ResNet-18.

all datasets for both probability and label scenarios. Especially on the CIFAR-100 dataset, it surpass the SOTA by 15.53%. The results suggest that our synthetic data domain exhibits greater diversity, enabling $\mathcal{S}$ to acquire more information about $\mathcal{T}$. By utilizing this knowledge, $\mathcal{S}$ can more accurately mimic the decision boundary of $\mathcal{T}$, achieving closeness between the decision boundaries of the two and facilitating an efficient transfer attack.

**Substitute training with the real data.** As shown in Tables 3, 4 and 5, we train $\mathcal{S}$ with either $\mathcal{T}$'s training set or the ImageNet. The experiments reveal that while real data training may improve the accuracy of $\mathcal{S}$, it will reduce the ASRs of adversarial attacks. We attribute this to the limitations in the quantity and diversity of real images and the insufficient distribution of real data around $\mathcal{T}$'s decision boundary.

### 4.3 Ablation Study

To demonstrate the effectiveness of each module, we sequentially cut off a single module at a time. As shown in Table 6, cutting any single module leads to a significant drop in ASRs, but the contribution of each module is relatively similar, indirectly proving that the three methods we proposed are complementary to each other, jointly improving the quality of the synthetic data domain.

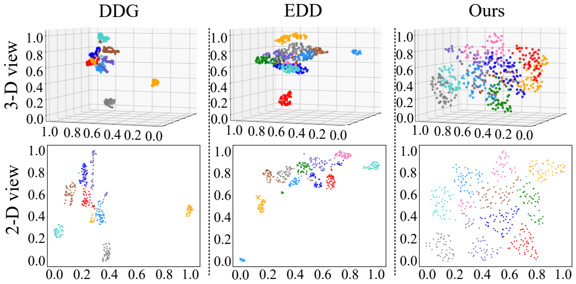

Figure 4: Applying umap to reduce the dimension of data generated by three different methods in 2-D and 3-D views.

### 4.4 Diversity Analysis

We use umap (McInnes, Healy, and Melville 2018) to exhibit the diversity differences of DDG, EDD, and Ours on the CIFAR-100 dataset. As shown in Figure 4, we visualize the feature distributions of generated data extracted by $\mathcal{T}$ for the first ten classes due to spatial limitations. Compared to the other two methods, our synthetic data domains show superior a broader coverage in 3-D view and better discreteness in 2-D view. This more intuitively demonstrates that our method do able to improve the coverage and discreteness of the synthetic data domains. For more visualization and analysis, please refer to the appendix.

## 5 Conclusion

In this paper, we analyze data diversity from the perspective of synthetic data domains and identify two key challenges in existing methods. To address these issues, we propose Domain-Augmented Learning, which includes Adaptive Semantic Embedding, Competition Optimization, and Heterogeneity Excitation, to improve the quality of the synthetic data domain. Through extensive experiments, we prove the effectiveness of the proposed approaches.

## Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (62406047, 62436002, 62302361), the Natural Science Foundation of Chongqing for Distinguished Young Scholars (CSTB2022NSCQ-JQX0001), the Natural Science Foundation of Chongqing for Innovation and Development Joint Foundation (CSTB2024NSCQ-LMX0002), the Science and Technology Research Program of Chongqing Education Commission (KJQN202400608), Special Grants for Postdoctoral Research Program of Chongqing (2023CQB-SHTB3160), China Postdoctoral Science Foundation (No. 2023M732735, 2024T170697). The authors are sincerely grateful to the reviewers for their valuable suggestions, which have significantly enhanced the quality and integrity of this work.

## References

Beetham, J.; Kardan, N.; Mian, A.; and Shah, M. 2023. Dual student networks for data-free model stealing. *arXiv preprint arXiv:2309.10058*.

Carlini, N.; and Wagner, D. 2017. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, 39–57. Ieee.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014a. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014b. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Jawahar, G.; Sagot, B.; and Seddah, D. 2019. What does BERT learn about the structure of language? In *ACL 2019-57th Annual Meeting of the Association for Computational Linguistics*.

Jindal, A.; Goyal, V.; Anand, S.; and Arora, C. 2024. Army of Thieves: Enhancing Black-Box Model Extraction via Ensemble based sample selection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 3823–3832.

Kariyappa, S.; Prakash, A.; and Qureshi, M. K. 2021. Maze: Data-free model stealing attack using zeroth-order gradient estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13814–13823.

Karras, T.; Aila, T.; Laine, S.; and Lehtinen, J. 2017. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Kurakin, A.; Goodfellow, I. J.; and Bengio, S. 2018. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, 99–112. Chapman and Hall/CRC.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11): 2278–2324.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

McInnes, L.; Healy, J.; and Melville, J. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Orekondy, T.; Schiele, B.; and Fritz, M. 2019. Knockoff nets: Stealing functionality of black-box models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4954–4963.

Papernot, N.; McDaniel, P.; Goodfellow, I.; Jha, S.; Celik, Z. B.; and Swami, A. 2017. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, 506–519.

Peters, M. E.; Neumann, M.; Zettlemoyer, L.; and Yih, W.-t. 2018. Dissecting contextual word embeddings: Architecture and representation. *arXiv preprint arXiv:1808.08949*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PMLR.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. 2015a. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115: 211–252.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. 2015b. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115: 211–252.

Sanyal, S.; Addepalli, S.; and Babu, R. V. 2022. Towards data-free model stealing in a hard label setting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 15284–15293.

Simonyan, K. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Tramèr, F.; Papernot, N.; Goodfellow, I.; Boneh, D.; and McDaniel, P. 2017. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*.

Truong, J.-B.; Maini, P.; Walls, R. J.; and Papernot, N. 2021. Data-free model extraction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4771–4780.

Wang, W.; Yin, B.; Yao, T.; Zhang, L.; Fu, Y.; Ding, S.; Li, J.; Huang, F.; and Xue, X. 2021. Delving into data: Effectively substitute training for black-box attack. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4761–4770.

Wei, Y.; Ma, Z.; Ma, Z.; Qin, Z.; Liu, Y.; Xiao, B.; Bi, X.; and Ma, J. 2023. Effectively Improving Data Diversity of Substitute Training for Data-Free Black-Box Attack. *IEEE Transactions on Dependable and Secure Computing*.

Yu, M.; and Sun, S. 2022. FE-DaST: Fast and effective data-free substitute training for black-box adversarial attacks. *Computers & Security*, 113: 102555.

Zhang, J.; Li, B.; Xu, J.; Wu, S.; Ding, S.; Zhang, L.; and Wu, C. 2022. Towards efficient data free black-box adversarial attack. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15115–15125.

Zhou, M.; Wu, J.; Liu, Y.; Liu, S.; and Zhu, C. 2020. Dast: Data-free substitute training for adversarial attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 234–243.