APPENDIX: EXPLORE THE VULNERABILITY OF BLACK BOX MODELS VIA DIFFUSION MODELS

Anonymous for submission

A. APPENDIX

A.0.1. A. Query Budget and Training Settings for Data-Free Model Extraction

During the data generation phase, we employ the Stable Diffusion model to create high-resolution images of 512x512 pixels, operating on the prompts we provided for 50 inference cycles. Once generated, we downscaled these images to align with the dimensions of the target datasets. For CIFAR-10, this meant scaling down from 512x512 to 32x32 pixels, and for ImageNet subsets, from 512x512 to 256x256 pixels. We produced a substantial collection of $200k$ synthetic samples per dataset, which served as the initial training material for our substitute models. To ensure consistency, the same training settings were used for both the original teacher models and the synthetic substitute models. This entailed using the SGD optimizer with a momentum of 0.9, a weight decay set at 5×10^{-4} , and a cosine learning rate scheduler commencing at a learning rate of 0.1. Our experiments conduct on an Nvidia-A40 GPU, and the entire distillation training process take only 10 minutes under a query budget of $5k$.

Query Budget. Our approach emphasizes query efficiency, so we consistently apply the same query constraint across all baseline methods. Specifically, the query budget is set to $Q = 5k$ for CIFAR-10, $Q = 1k$ for ImageNetette, $Q = 130$ for ImageFruit, $Q = 50$ for ImageYellow, and $Q = 30$ for ImageSquaw. In the distillation phase, each teacher model undergoes a single forward pass with the allotted Q queries using synthetic data, which is the sole instance we query the teacher model. The top-1 hard-labels gleaned from this query are preserved for subsequent use. We then leverage the substitute model's logits alongside the hard-labels from the teacher to compute the cross-entropy loss, thus advancing the substitute model's training. We measure performance by conducting three independent trials with 3 random seeds and present the mean top-1 accuracy.

Data and Model. We evaluate our method on a range of datasets to assess its performance under diverse conditions: CIFAR-10 [1], CIFAR-100 [1]. In addition, we use specialized ImageNet subsets: ImageFruit [2], ImageYellow [2], and ImageSquaw [2], and Tiny ImageNet [3]. To validate the generalization and practicality of our method, we experiment with different model architectures as the target model, including AlexNet [4], VGG-16 [5], VGG-19 [5], Wide-ResNet-16 [6], and ResNet34 [7]. Simultaneously, we also explore different model architectures as the substitute model, specifically VGG-16, VGG-19, Wide-ResNet-16, ResNet-18 [7], and ResNet-34 (see details in Tab. 5). All the training parameters for the substitute model follow the settings in [8]. The target model, trained on private, real-world datasets, functions as a black-box and is only accessible to attackers via queries. Conversely, the substitute model is exclusively trained on synthetic data. This setup aims to evaluate the practicality of our method in environments where direct access to the target model's training data is restricted, effectively simulating a real-world adversarial scenario where attackers depend on synthetic approximations to challenge and compromise black-box models.

Baselines. In our experiments, we select two distinct categories of prevalent approaches. MAZE [9], DFME [8], and ZSDB3 [10] originally design for soft-label settings (i.e., probabilities or logits). Furthermore, we evaluate models designed for hard-label settings (i.e., return top-1 prediction), including DFMS [11] and DisGuide [12].

A.0.2. B. Performance Comparison by Soft-Label in Model Extraction

Table 1. Accuracy (%) of substitute models on datasets with CIFAR-10, and ImageNet subsets in the soft-label setting. All results are averaged over three random seeds.

Dataset	Teacher	MAZE	DFME	ZSDB3	DFMS-HL	DisGuide	Ours	Query Budget
CIFAR-10	93.9	10.7	10.8	11.1	11.4	13.4	**88.8**	$5k$
ImageNette	92.2	11.2	10.3	10.5	11.2	11.8	**83.9**	$1k$
ImageSquawk	92.4	10.4	10.6	10.2	11.2	11.7	**83.6**	30
ImageFruit	78.2	10.2	10.4	11.1	10.9	11.3	**70.8**	130
ImageYellow	90.8	10.2	10.4	10.6	11.7	11.8	**82.4**	50

Table 2. Accuracy (%) of student models on datasets of hundreds of classes in the soft-label setting. All results are averaged over three random seeds.

Dataset	Teacher	MAZE	DFME	ZSDB3	DFMS-HL	DisGuide	Ours	Query Budget
CIFAR-100	79.89	1.05	1.19	1.25	1.42	1.75	**72.3**	$150k$
Tiny-ImageNet	64.55	0.53	0.61	0.67	0.72	0.86	**58.5**	$200k$

Our method achieves top-one accuracy rates of 88.9% and 83.9% on the CIFAR-10 and Imagenette datasets, respectively, marking a substantial improvement over all previous methods under a constrained query budget. This enhancement stems from the soft label setting during the distillation learning phase, wherein the substitute model accesses the target model's soft labels (i.e., probabilities or logits). This access enables the substitute to learn a richer set of information compared to the hard-label setting. As a result, both our method and the baseline demonstrate improvements over results obtained under hard-label conditions. However, the baseline's performance remains suboptimal, as demonstrated in Fig. 1. Specifically, the GANs training framework, operating under a severely limited query budget, produces only noisy data, and the training loss fails to converge.

A.0.3. Empirical Studies of Previous Methods

To better demonstrate the training deficiencies of state-of-the-art methods DFME [8], DisGuid [12] under a restricted $5k$ query budget, this section provides an empirical analysis of the loss changes in their generator and substitute models. Our experiments are conducted on the CIFAR-10 dataset with a stringent limit of 20 epochs due to the $5k$ query budget, highlighting the challenges traditional GAN-based training methods face under such constraints. As depicted in Fig. 1, the left subplot for DisGuid shows a slight reduction in generator loss (green), decreasing from an initial 5.5 to 4.8 before it begins to oscillate. However, the loss does not converge and remains high, indicating that the generator fails to produce high-quality images effectively. The loss for the substitute model (orange) remains near zero, suggesting that the generator struggles to generate consistently challenging images, thereby hindering the substitute model's ability to learn effectively from the target model. The right subplot in Fig. 1 presents the loss values for DFME, where both the generator and substitute model exhibit significant fluctuations throughout the training process. This unstable convergence pattern indicates that under a minimal query budget, the substitute model struggles to match the target model's output, further highlighting the inherent challenges of training traditional GANs to converge under these conditions.

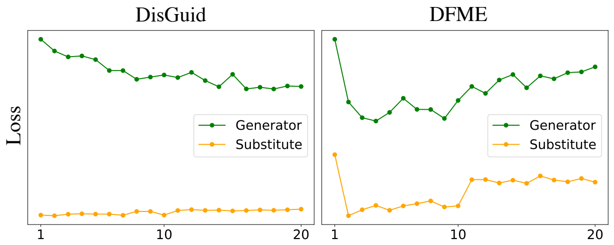

Fig. 1. Training flaws of previous SOTA methods under $5k$ query budget.

A.0.4. Ablation Study on Pre-trained Models

In this section, we assess the impact of the pre-training stage on the performance of substitute models. As indicated in Tab. 3, eliminating the pre-training stage leads to significant performance degradation. Specifically, we evaluated our method on CIFAR-10, an ImageNet subset, CIFAR-100, and Tiny ImageNet, using the same query budgets outlined in (Appendix A). The observed degradations in test accuracy on these datasets were 33.9%, 16.4%, 17.1%, and 9.1%, respectively. Our training framework diverges from traditional adversarial training of generators and substitutes by leverage the online stable diffusion APIs to synthesize high-quality images. This approach is critical for reducing query budgets, as pre-training the substitute model proves essential. An effectively pre-trained substitute model not only reduces the number of queries required but also accelerates and enhances the efficiency of the training convergence process during the knowledge distillation stage.

Table 3. Ablation studies evaluating the impact of the pre-training stage with Top-1 accuracy. All results are averaged over three random seeds.

Method	CIFAR-10	ImageNet subset	CIFAR-100	Tiny-ImageNet
w/ pre-training stage	**81.8**	**69.7**	**60.5**	**45.9**
w/o pre-training stage	47.9	53.3	43.4	36.8

A.0.5. E. Query Budget and Evaluations For Data-Free Adversarial Transfer Attack

Adversarial transfer presents a more complex challenge than model extraction because it involves creating adversarial samples that must maintain high transferability between the substitute and target models. Consequently, we allocate a query budget of $150k$ for both the CIFAR-10 and CIFAR-100 datasets to accommodate this complexity. Our experiments conduct on an Nvidia-A40 GPU, and the entire adversarial attacking process take about 45 minutes under a query budget of $150k$. To ensure fair comparisons, the same query budget is applied to the baselines. We utilize well-known adversarial attack methods such as BIM [13], FGSM [**?**], PGD [14] for conducting experiments. The specific parameters set for these experiments on CIFAR-10 and CIFAR-100 include a perturbation limit ϵ = 8/255 and the step size at α = 2/255 follow by the setting in [15]. In the untargeted attack mode, adversarial examples are generated only from images that the model initially classifies correctly. In contrast, targeted attack strategies generate adversarial examples solely from images that are not already misclassified into specific incorrect categories. The attack success rate is calculated using the ratio n/m, where n is the number of adversarial examples that successfully fool the attacked model, and m is the total number of adversarial examples created.

Baselines. We select the most prominent baselines for black-box adversarial attacks, including JPBA [16] and

Knockoff [17], which require access to training data. Additionally, we evaluate black-box knowledge distillation methods exploiting probabilities returned by the target model, with a focus on DFME [8]. Furthermore, we critically examine data-free black-box attacks using hard-labels, closely align with our experimental setup, as detailed in (Appendix E), including DaST [18] and TEBA [15], to underscore the comparative effectiveness of our approach.

A.0.6. F. Impact Analysis of Query Efficiency in Adversarial Transfer Attack

An adversarial example x' for the model f_{sub} of the substitute is generated by perturbing x such that $x' = x + \delta$, where δ is chosen to maximize $L_{\text{sub}}(x', y; \theta)$. The goal of the adversarial attack is to find x' such that:

$$\arg\max f_{\text{sub}}(x') \neq y,$$

and ideally:

$$\arg\max f_{\text{target}}(x') \neq y,$$

The transferability of adversarial examples from f_{sub} to f_{target} can be quantified as:

$$T(f_{\text{sub}} \to f_{\text{target}}) = \mathbb{P}(\arg\max f_{\text{target}}(x + \delta) \neq y \mid \arg\max f_{\text{sub}}(x + \delta) \neq y),$$

Training on Synthetic Data.

Objective for non-pretrained substitute model $f_{\text{sub}}^{\text{real}}$:

$$f_{\text{sub}}^{\text{real}} = \min_{f_{\text{sub}}} \mathbb{E}_{x,y \sim \mathcal{D}_{\text{real}}}[L(f_{\text{sub}}(x), y)],$$

Objective for pretrained substitute model $f_{\text{sub}}^{\text{syn}}$:

$$f_{\text{sub}}^{\text{syn}} = \min_{f_{\text{sub}}} \mathbb{E}_{x,y \sim \mathcal{D}_{\text{syn}}}[L(f_{\text{sub}}(x), y)],$$

Effect on Decision Boundary.

The decision boundary in $f_{\text{sub}}^{\text{syn}}$ tends to be broader because the diffusion model generates diverse training data. We define the decision boundary of a model f as $\partial f = \{x \mid \arg\max f(x) \text{ changes}\}$. If the decision boundary ∂f_{syn} of the substitute model is close to that of the target model $\partial f_{\text{target}}$, then the adversarial samples generated on $f_{\text{sub}}^{\text{syn}}$ are more likely to transfer effectively to f_{target}.

Query Efficiency.

Let $Q(f_{\text{sub}} \to f_{\text{target}})$ represent the number of queries required to generate a successful adversarial example for the target using f_{sub}. For $f_{\text{sub}}^{\text{real}}$, the expected number of queries can be denoted as:

$$\mathbb{E}[Q(f_{\text{sub}}^{\text{real}} \to f_{\text{target}})] = \frac{1}{T(f_{\text{sub}}^{\text{real}} \to f_{\text{target}})},$$

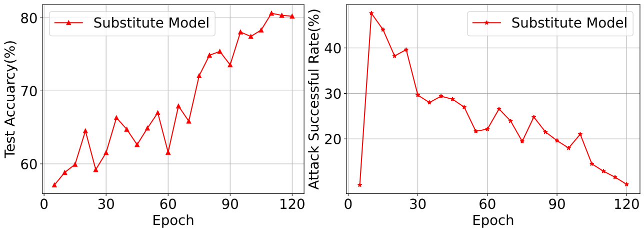

Fig. 2. **Left**: Changes in ACC with different checkpoint in stage one. **Right**: Changes in ASR with different checkpoint in stage one.

For $f_{\text{sub}}^{\text{syn}}$, due to reduced transferability $T(f_{\text{sub}}^{\text{syn}} \to f_{\text{target}})$, the expected number of queries decreased:

$$\mathbb{E}[Q(f_{\text{sub}}^{\text{syn}} \to f_{\text{target}})] = \frac{1}{T(f_{\text{sub}}^{\text{syn}} \to f_{\text{target}})},$$

Since $T(f_{\text{sub}}^{\text{real}} \to f_{\text{target}}) \leq T(f_{\text{sub}}^{\text{syn}} \to f_{\text{target}})$ implies:

$$\mathbb{E}[Q(f_{\text{sub}}^{\text{syn}} \to f_{\text{target}})] \leq \mathbb{E}[Q(f_{\text{sub}}^{\text{real}} \to f_{\text{target}})].$$

Conclusion

Pretraining the substitute model requires fewer queries to achieve an adversarial transfer attack.

A.0.7. G. Interplay Between Substitute Model Training, Model Efficacy, and Attack Success

In the initial stage of our experiments, we leverage off-the-shelf generative models to synthesize high-quality synthetic data based on benign prompts, followed by the pre-training of the substitute model. Our objective was to assess the changes in test accuracy and ASR across different training epochs. Specifically, our experiments were conducted on the CIFAR-10 dataset using a ResNet-18 substitute model. We saved a checkpoint every five epochs and subsequently evaluated each checkpoint for its corresponding test ACC and ASR.The results, as depicted in Fig. 2, demonstrate that: 1) the test accuracy progressively increases with the advancement of the pre-training process, and the checkpoint with the highest accuracy indeed provides an excellent starting point for the distillation training in the subsequent stage. 2) The experiments indicate that the ASR initially rises rapidly within the first 0-5 epochs and then gradually decreases. Consequently, if users require a model with a higher ASR, early stopping is recommended to capture a model that delivers superior performance.

A.0.8. Analyzing the Impact of Query Budget on Substitute Model Efficacy and Attack Success

In this study, we investigate the relationship between two attack methods and the query budget, as well as the interrelation between these attacks. As illustrated in Fig. 3, the top-one accuracy of the substitute model improved from 84.22% to 87.8% as the query budget increased from $10k$ to $200k$. For adversarial transfer attacks employing untargeted and BIM methods, the ASR rose from 68.79% to 99.32%, demonstrating convergence. This enhancement can be attributed to the use of diverse and high-fidelity images generated by off-the-shelf generative models, which pre-train the substitute model and provide a rich knowledge base. This approach significantly enhances the ability to produce more transferable adversarial examples during attacks. Furthermore, we observed a positive correlation between the test accuracy of the substitute model and its ASR, confirming that the performance of data-free model extraction and data-free transfer attack tasks is positively correlated.

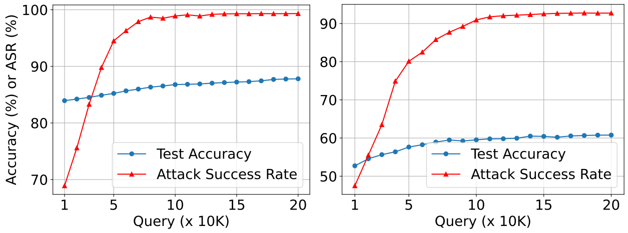

Fig. 3. **Left**: ACC and ASR on CIFAR-10 with different query budget. **Right**: ACC and ASR on CIFAR-100 with different query budget.

A.0.9. Performance Comparison by Soft-Label in Adversarial Attack

In this section, recognizing the suboptimal performance of previous baselines in the hard-label setting, we compare their effectiveness in the more favorable soft-label setting for a fairer and more comprehensive evaluation. Both our method and the baselines show improved performance under soft-label conditions, as this setting allows the substitute model to capture more knowledge. However, the baseline's performance remains suboptimal. Specifically, our method achieves an ASR of 98.68% on CIFAR-10 and 99.68% on CIFAR-100 as shown in Tab. 4, reflecting improvements of 41.85% and 41.74%, respectively, over the previous baselines.

A.0.10. Comparisons with Different Substitute Model

In this section, we demonstrate the generalization and practicality of our method as shown in Tab. 5. We prove that an attacker can use various model architectures to perform model extraction attacks on black-box systems. Fixing the target model as ResNet-34, we conducted attacks using both heterogeneous architectures (VGG16, VGG19) and homogeneous ones (ResNet-18, ResNet-34, Wide-ResNet-16-8). The results show that all substitute models in our framework outperform previous baselines. Notably, using VGG16 as the substitute model, our method improves the attack success rate by 68% on CIFAR-10 and 56% on CIFAR-100 compared to prior baselines.

Table 4. ASR(%) comparisons between our proposed method and baselines over CIFAR-10 and CIFAR-100 under soft-label settings with a query budget Q = $150k$. Best result in bold. All results are averaged over three random seeds.

Dataset	Type	Targeted, soft-label			Untargeted, soft-label		
	Method	FGSM	BIM	PGD	FGSM	BIM	PGD
CIFAR-10	JPBA	2.74	3.86	3.93	8.29	10.92	8.62
	Knockoff	2.16	3.53	3.37	7.55	10.18	9.06
	DaST	3.95	4.19	4.33	8.98	12.53	8.32
	DFME	3.55	11.28	8.93	15.13	20.17	17.89
	TEBA	10.38	31.8	27.9	34.48	56.83	50.72
	Ours	**17.27**	**88.71**	**85.85**	**63.95**	**98.68**	**98.63**
CIFAR-100	JPBA	3.25	4.19	4.24	9.39	11.45	9.75
	Knockoff	2.67	4.02	3.87	8.55	11.28	10.13
	DaST	4.45	4.69	4.83	9.98	13.53	9.92
	DFME	4.55	12.28	9.93	16.13	21.17	18.89
	TEBA	11.38	32.8	28.9	35.48	57.94	51.72
	Ours	**18.27**	**89.71**	**86.85**	**64.95**	**99.68**	**99.63**

Table 5. Top-1 accuracy comparison between our method and previous baselines using different substitute model architectures. To evaluate the generalization of our approach, the target model was fixed as ResNet-34, while VGG-16 (V-16), VGG-19 (V-19), Wide-ResNet-16-8 (WRN16), ResNet-18 (R-18), and ResNet-34 (R-34) were employed as substitute models in the hard-label setting. CIFAR-10 and CIFAR-100 employ query budgets of 5k and 150k, respectively. All results are averaged over three random seeds.

Dataset	Method	V-16	V-19	WRN16	R-18	R-34
CIFAR-10	DFME	10.52	10.21	10.36	10.96	10.85
	ZSDB3	10.43	10.31	10.20	10.83	10.47
	DisGuide	12.49	11.20	12.12	12.57	13.37
	Ours	80.73	80.44	83.74	81.51	84.41
CIFAR-100	DFME	1.06	1.08	1.05	1.05	1.07
	ZSDB3	1.03	1.05	1.07	1.05	1.09
	DisGuide	1.13	1.27	1.19	1.27	1.30
	Ours	57.56	58.86	59.14	60.51	58.82

A.0.11. Visualization of Synthetic Data

In this subsection, we present synthesized examples from DFME [8], DisGuide [12], and our method to investigate the underlying reasons for the baseline performance approximating random guessing. Our objective is to corroborate the experimental results shown in **??** and **??** by demonstrating the synthesized images and training stage loss (see details Appendix A.0.3) produced by DFME and DisGuide within a $5k$ query budget constraint. As illustrated in Fig. 4, the images generated by DFME and DisGuide under this limited query budget appear nearly as random noise, lacking discernible patterns. This similarity in noisy patterns indicates that these GAN-based generators struggle with training under

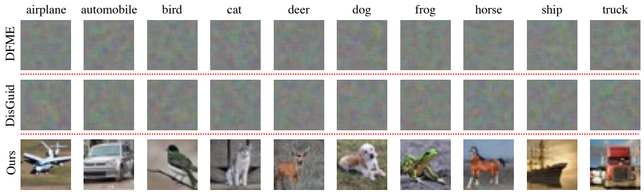

Fig. 4. Visualization of synthetic images generated by baseline DFME and DisGuide with a $5k$ query budget.

data scarcity, leading to model collapse and difficulties in achieving convergence.

Table 6. Ablation studies evaluating the impact of the pre-training stage with Top-1 accuracy. All results are averaged over three random seeds.

Method	CIFAR-10	ImageNet subset	CIFAR-100	Tiny-ImageNet
w/ pre-training stage	**81.8**	**69.7**	**60.5**	**45.9**
w/o pre-training stage	47.9	53.3	43.4	36.8

A.0.12. Effect Investigation of Pre-training model in Model Extraction

In model extraction, the objective is to approximate the target model's behavior by training a substitute model on the target model's outputs. The parameters of the substitute model, denoted as θ_{sub}, are optimized using the cross-entropy loss function, which is minimized through gradient-based updates. Given an input x and its corresponding one-hot label y, the predicted probability for class y_i by the substitute model is $P_{\theta_{\text{sub}}}(y_i|x)$. The cross-entropy loss function is defined as follows:

$$L_{\text{CE}}(P_{\theta_{\text{sub}}}(y|x), y) = -\sum_{i=1}^{C} y_i \log P_{\theta_{\text{sub}}}(y_i|x), \tag{1}$$

where C denotes the total number of classes, and y_i is the i-th component of the label y, where $y_i = 1$ if $y = i$ and $y_i = 0$ otherwise. This loss function quantifies the divergence between the predicted and true distributions. To optimize the substitute model parameters θ_{sub}, we compute the gradient of the loss function:

$$\nabla_{\theta_{\text{sub}}} L_{\text{CE}}(P_{\theta_{\text{sub}}}(y|x), y) = -\sum_{i=1}^{C} y_i \nabla_{\theta_{\text{sub}}} \log P_{\theta_{\text{sub}}}(y_i|x), \tag{2}$$

where $\nabla_{\theta_{\text{sub}}} \log P_{\theta_{\text{sub}}}(y_i|x)$ denotes the gradient of the log-probability function with respect to θ_{sub}. Using the chain rule, this gradient expands to:

$$\nabla_{\theta_{\text{sub}}} \log P_{\theta_{\text{sub}}}(y_i|x) = \frac{1}{P_{\theta_{\text{sub}}}(y_i|x)} \nabla_{\theta_{\text{sub}}} P_{\theta_{\text{sub}}}(y_i|x), \tag{3}$$

where $\nabla_{\theta_{\text{sub}}} P_{\theta_{\text{sub}}}(y_i|x)$ denotes the gradient of the predicted probability with respect to θ_{sub}. Substituting this into the original gradient formula yields the final expression:

$$\nabla_{\theta_{\text{sub}}} L_{\text{CE}}(P_{\theta_{\text{sub}}}(y|x), y) = -\sum_{i=1}^{C} \frac{y_i}{P_{\theta_{\text{sub}}}(y_i|x)} \nabla_{\theta_{\text{sub}}} P_{\theta_{\text{sub}}}(y_i|x), \tag{4}$$

This formula outlines the gradient update process for the substitute model in model extraction. Iterative updates to θ_{sub} progressively align the substitute model's behavior with that of the target model, enabling efficient model extraction.

Case 1: Pretrain with Synthetic Images.

We first pretrain the substitute model $\theta_{\text{sub}}^{\text{pretrain}}$ on synthetic images, aiming to minimize the expected cross-entropy loss between the substitute model's predictions and the true labels:

$$\theta_{\text{sub}}^{\text{pretrain}} = \arg\min_{\theta_{\text{sub}}} \mathbb{E}_{x \sim \mathcal{D}_{\text{syn}}} \left[L_{\text{CE}} \left(P_{\theta_{\text{sub}}}(y|x), y \right) \right], \tag{5}$$

where $\mathcal{D}_{\text{syn}}$ denotes the synthetic dataset, and L_{CE} is the cross-entropy loss. Given that the synthetic data distribution $\mathcal{D}_{\text{syn}}$ is designed to approximate the real data distribution $\mathcal{D}_{\text{real}}$, the pretrained parameters $\theta_{\text{sub}}^{\text{pretrain}}$ are expected to be closely align to the target model's parameters θ_{target}.

Fine-tune Step.

After pre-training, we fine-tune the substitute model using a limited set of queries.

$$\theta_{\text{sub}}^{\text{fine-tune}} = \arg\min_{\theta_{\text{sub}}} \mathbb{E}_{x \sim \hat{\mathcal{X}}_{\text{query}}} \left[L_{\text{CE}} \left(P_{\theta_{\text{sub}}}(y|x), P_{\theta_{\text{target}}}(y|x) \right) \right], \tag{6}$$

where $\hat{\mathcal{X}}_{\text{query}}$ represents the set of input queries sampled for fine-tuning. Given that $\theta_{\text{sub}}^{\text{pretrain}}$ is already close align to θ_{target}, the gradient of the loss function with respect to the pretrained parameters is expected to be small:

$$\nabla_{\theta_{\text{sub}}^{\text{pretrain}}} L_{\text{CE}} \left(P_{\theta_{\text{sub}}^{\text{pretrain}}}(y|x), P_{\theta_{\text{target}}}(y|x) \right) \approx 0, \tag{7}$$

This small gradient implies that only minor adjustments are necessary during fine-tuning, making the process efficient and requiring fewer queries to the target model.

Case 2: Train from Scratch.

In contrast, when training the substitute model from scratch, the optimization problem is formulated as:

$$\theta_{\text{sub}}^{\text{scratch}} = \arg\min_{\theta_{\text{sub}}} \mathbb{E}_{x \sim \hat{\mathcal{X}}_{\text{query}}} \left[L_{\text{CE}} \left(P_{\theta_{\text{sub}}}(y|x), P_{\theta_{\text{target}}}(y|x) \right) \right], \tag{8}$$

Since the parameters $\theta_{\text{sub}}^{\text{scratch}}$ start from an uninitialized state, far from the target model's parameters θ_{target}, the initial gradients will be large:

$$\nabla_{\theta_{\text{sub}}^{\text{scratch}}} L_{\text{CE}} \left(P_{\theta_{\text{sub}}^{\text{scratch}}}(y|x), P_{\theta_{\text{target}}}(y|x) \right) \gg 0. \tag{9}$$

Conclusion.

The comparison of these two scenarios illustrates that a synthetically pre-trained substitute model significantly reduces the need for extensive parameter adjustments, requiring fewer queries and leading to a more efficient extraction process, while training from scratch involves larger gradients and demands more queries to the target model, resulting in a longer training period.

B. REFERENCES

[1] Alex Krizhevsky, Geoffrey Hinton, et al., "Learning multiple layers of features from tiny images," 2009.

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[3] Ya Le and Xuan Yang, "Tiny imagenet visual recognition challenge," *CS 231N*, vol. 7, no. 7, pp. 3, 2015.

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[5] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[6] Sergey Zagoruyko and Nikos Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146*, 2016.

[7] 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*, 2016, pp. 770–778.

[8] Jean-Baptiste Truong, Pratyush Maini, Robert J Walls, and Nicolas Papernot, "Data-free model extraction," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 4771–4780.

[9] Kariyappa, "Maze: Data-free model stealing attack using zeroth-order gradient estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 13814–13823.

[10] Zi Wang, "Zero-shot knowledge distillation from a decision-based black-box model," in *International Conference on Machine Learning*. PMLR, 2021, pp. 10675–10685.

[11] Sanyal, "Towards data-free model stealing in a hard label setting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 15284–15293.

[12] Jonathan Rosenthal, Eric Enouen, Hung Viet Pham, and Lin Tan, "Disguide: Disagreement-guided data-free model extraction," 2023.

[13] Alexey Kurakin, Ian Goodfellow, and Samy Bengio, "Adversarial machine learning at scale," *arXiv preprint arXiv:1611.01236*, 2016.

[14] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu, “Towards deep learning models resistant to adversarial attacks,” *arXiv preprint arXiv:1706.06083*, 2017.

[15] Jie Zhang, Bo Li, Jianghe Xu, Shuang Wu, Shouhong Ding, Lei Zhang, and Chao Wu, “Towards efficient data free black-box adversarial attack,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 15115–15125.

[16] 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, pp. 506–519.

[17] Tribhuvanesh Orekondy, Bernt Schiele, and Mario Fritz, “Knockoff nets: Stealing functionality of black-box models,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4954–4963.

[18] Mingyi Zhou, Jing Wu, Yipeng Liu, Shuaicheng Liu, and Ce Zhu, “Dast: Data-free substitute training for adversarial attacks,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 234–243.