Supplementary for Towards Data-Free Model Stealing in a Hard Label Setting

Sunandini Sanyal **Sravanti Addepalli** **R. Venkatesh Babu**
Video Analytics Lab, Department of Computational and Data Sciences
Indian Institute of Science, Bangalore

1. Datasets

We perform experiments using different proxy datasets similar to prior works [1, 2] to evaluate the effectiveness of our method DFMS-HL. This section contains a description of the different datasets that we used to evaluate our attack with CIFAR-10 as the true dataset.

- **40-unrelated classes from CIFAR-100 [1]:** This consists of training data from CIFAR-100 belonging to non-overlapping classes with respect to CIFAR-10. The classes from the following categories are included: food containers, household electric devices, household furniture, large man-made outdoor things, large natural outdoor scenes, flowers, fruits and vegetables, trees.
- **10 random classes of CIFAR-100:** From the above 40 unrelated classes, we choose 10 classes randomly to demonstrate this setting. The classes used are : plate, rose, castle, keyboard, house, forest, road, television, bottle and wardrobe.
- **Synthetic Dataset:** We construct synthetic images which are far from the manifold of the training data distribution to simulate this setting. The images contain multiple overlapping shapes on top of a planar background. The creation of synthetic images is described in Sec. 1.1.

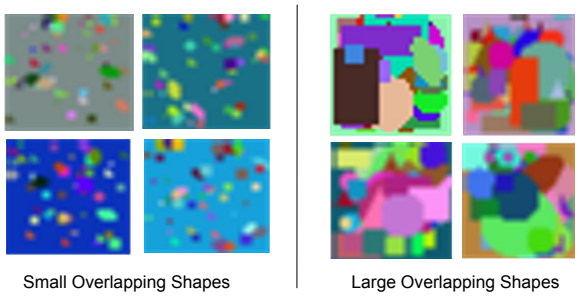

Figure 1. **Types of synthetic images used.** An equal share of large(right) and small(left) overlapping shapes on planar background used to train the clone model.

Algorithm 1 Algorithm for creating synthetic data

Require: Number of images to be generated N_P, num_shapes, max_size, min_size
while $N_P \neq 0$ **do**
  Generates shapes on an image of size (100 x 100), with parameters: num_shapes, min_size, max_size
  Assign a random RGB colour to background pixels
  Perform blurring on the image using a 4 x 4 filter
  Resize image to (32 x 32)
  $N_P \leftarrow N_P - 1$
end while

1.1. Creation of Synthetic Dataset

The algorithm to create a synthetic dataset is presented in Algorithm 1. At first, randomly sampled shapes (triangle, rectangle, circle or ellipse) are generated at random locations in the image with a randomly sampled colour. The shapes are generated using python skimage module[1]. A total of 50K images are generated. We generate two kinds of images. The first variant contains large overlapping shapes with number of shapes in the image (num_shapes) as 50 and the (min_size, max_size) of each shape as (20,50). The initial image generated is of size (100 x 100) which is scaled down to (32 x 32). The other variant contains textured images with (min_size, max_size) as (5,10) and num_shapes=50 to get small overlapping shapes on top of a planar background. A random colour is sampled and assigned to the background pixels. These images are then used to steal an ML model trained on CIFAR-10 and CIFAR-100. The generated images are shown in Fig. 1. We share our dataset here[2].

[1] https://scikit-image.org/docs/stable/auto_examples/edges/plot_random_shapes.html

[2] https://drive.google.com/drive/folders/1CCMCYVRnvqZig9dYUYO_BupI8tImGZ2x

2. Insights on Query Budget

Chandrasekaran *et al*. [3] formulated the model extraction task as a query synthesis algorithm where an adversary $\mathcal{A}$ can ask for labels of the data x which could be completely unrelated to the training data distribution. They show that, given a maximum query budget of $q_A(\epsilon, \delta)$ and a victim model $\mathcal{V}$ trained with a specific hypothesis $f^* \in \mathcal{F}$, there exists an adversary which implements an ϵ-extraction attack with confidence $1-\delta$. Adversary $\mathcal{A}$ trains a clone model $\mathcal{C}$ with hypothesis $\hat{f}$ such that the following holds true.

$$Pr[\mathcal{A} \text{ trains } \hat{f} \text{and Err}(\hat{f}) \leq \epsilon] \geq 1-\delta \tag{1}$$

where $Err(\hat{f}) = ||w^* - w||_2$, w and $\hat{w}$ being the parameters of $\hat{f}$ and f^*, respectively. This shows that an adversary can implement a model stealing algorithm in a Query Synthesis scenario using active learning. Further, the authors [3] show that even when a victim employs a randomized procedure for returning labels such that the upper bound on the probability of returning wrong labels $\rho_D(f^*) < \frac{1}{2}$, an adversary can implement an ϵ-extraction attack with confidence $1-2\delta$ within the following query bound:

$$q = \frac{8}{(1-2\rho_D(f^*))^2} q(\epsilon, \delta) ln \frac{q(\epsilon, \delta)}{\delta} \tag{2}$$

3. Experimental Details

For the evaluation of DFMS-HL, we consider victim models trained on two datasets, CIFAR-10 and CIFAR-100. For each dataset, a victim model is trained upto a comparable accuracy of the teacher model used in prior works [1,2,9]. The initial Clone model is trained with an SGD optimizer of momentum 0.9, learning rate of 0.1 and weight decay of 5×10^{-4}. We train the initial clone model for 200 epochs. The learning rate is reduced to 0.01 once the alternate training of clone and generator starts. After this, the clone is trained alternately till the query budget is exhausted. We use a cosine annealing scheduler to decay the learning rate across epochs. For the generator, a DCGAN architecture is trained with an Adam Optimizer and a learning rate of 2×10^4 with (β_1, β_2) as (0.5, 0.999). We use NVIDIA GeForce GTX 1080 Ti and GeForce RTX 3090 to train our models. Our code takes a total training time of approximately 5 hours for CIFAR-10 and 10 hours for CIFAR-100 datasets on NVIDIA GeForce RTX 3090.

4. Ablation Experiments

4.1. Impact of Synthetic Data

We tried two variants of the synthetic dataset. The first variant, "Large overlapping shapes" contains multiple overlapping shapes on a planar background. The second variant "Small overlapping shapes" contains multiple shapes of smaller size in an image. Each variant is shown in Fig 1. We report results obtained by using each of these datasets individually and both combined in Table-1. In this experiment, we use grey scale images for training. After combining the two datasets, we obtain a competent accuracy of 85.92%.

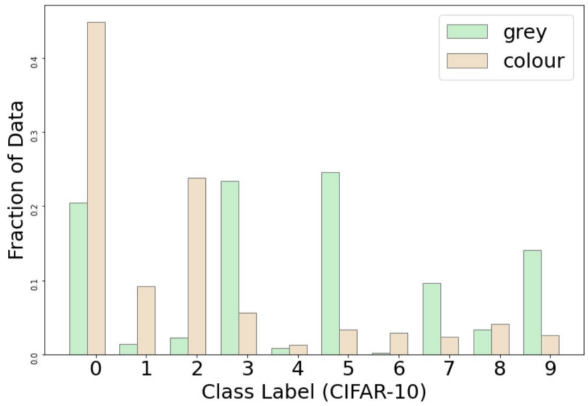

Figure 2. **Distribution of classes for grey vs colour images:** The grey synthetic images are more uniformly distributed across CIFAR-10 classes as compared to coloured images.

Table 1. **Impact of Synthetic Data**: Clone Model accuracy with different kinds of synthetic data images used, obtained on a ResNet-18 victim model of accuracy 93.65%, with ResNet-18 as the clone architecture.

Type of Synthetic Data	Clone Accuracy
Large overlapping shapes	80.34
Small overlaping shapes	56.30
Large + Small Combined	**85.92**

We use grey-scale and coloured images individually from the synthetic dataset and observe its impact on the clone model accuracy with an AlexNet victim network. We find that the grey images are well-distributed across multiple classes as shown in Fig. 2. This makes grey images a better choice for initialization. In our method, we train a clone model with a mix of images from the proxy data and the generator to obtain a good initialisation. From our experiments, we observe that the initial clone model trained with grey-scale synthetic data achieves an accuracy of 44.57% and the one trained with coloured images has an accuracy of 37.31%. This shows that grey-scale images lead to a better initialization for the clone model. Hence, we reported the final results of our method using grey-scale synthetic images. We also report the results of using the grey-scale and colour images individually for training in Table 2 and observe that the final clone accuracy in both cases are comparable.

Table 2. **Impact of Synthetic Data**: Comparison for grey vs coloured images used as proxy data, with AlexNet as the victim model of accuracy 80.18% , trained on CIFAR-10, and AlexNet-half as the clone model.

Type of Synthetic Data	Clone Accuracy
Grey synthetic images	67.03
Coloured synthetic images	65.84

Table 3. **Impact of class-diversity loss coefficient** λ_{div}: Performance (%) of the clone model on CIFAR-10 dataset trained using 10 random classes of CIFAR-100 as proxy, across variation in λ_{div}. The architecture of victim model is Alexnet and architecture of clone model is AlexNet-half. The proposed method is not sensitive to minor variations in λ_{div}.

Diversity Loss Coefficient	Clone Accuracy
100	69.29
200	69.59
300	69.42
500	**69.66**
700	69.54
1000	69.13

4.2. Hyperparameter tuning

The diversity loss plays a crucial role in ensuring that the distribution of images from the generator is class-balanced. The loss formulation of the generator with the class-diversity loss is shown below:

$$\mathcal{L}_G = \mathcal{L}_{adv,fake} + \lambda_{div} \cdot \mathcal{L}_{class\_div} \tag{3}$$

We show the impact of varying the class-diversity loss coefficient λ_{div} in Table 3. The true dataset is CIFAR-10 and the proxy dataset is 10 random classes from CIFAR-100. We use AlexNet as the victim architecture and train an AlexNet-half as the clone model for 500 epochs. We observe that as we increase the diversity loss coefficient, the clone model accuracy increases and reaches the maximum accuracy of 69.66% at λ_{div}=500. We note that the proposed method is not sensitive to minor variations in the hyperparameter λ_{div}.

4.3. Impact of Clone architecture

In a practical scenario of Model Stealing, the architecture of the victim model is unknown to the attacker. Hence, we aim to stage a successful attack in a completely black-box condition. To evaluate the effectiveness of the attack in different scenarios, we perform an ablation experiment to see if the choice of the clone model architecture impacts the success of the attack. The clone model achieves a high accuracy of 83.37% using 10 random classes of CIFAR-100 when the same ResNet-18 architecture is used for both the victim and the clone. However, using a deeper CNN model such as GoogleNet gives a boost to the clone accuracy as shown in Table 4. We get lower clone accuracy for shallower networks such as AlexNet-half and VGG-11. Hence, we observe that it is beneficial for an adversary to use a deeper CNN architecture for capturing complex features from the victim model using proxy data.

Table 4. **Impact of clone architecture on clone accuracy**: Clone Accuracy improves with a deeper CNN network

Clone Model Architecture	Clone Accuracy
ResNet-18	**83.37**
AlexNet	79.37
AlexNet_half	62.64
VGG-11	74.59
VGG-19	78.85
GoogleNet	84.50

Table 5. **Impact of L1 loss formulation on DFMS-SL (Soft-Label Setting)**: Clone Model accuracy increases by 3% after using L1-loss as compared to standard KL-divergence loss. Synthetic data is used as proxy for a ResNet-34 victim model trained on CIFAR-10 and ResNet-18 used as Clone model.

Method	**Teacher Acc**	Synthetic
DFME	95.5	88.10
DFMS-SL(L1 loss)	95.5	91.24
DFMS-SL(KL-div loss)	95.5	88.40

Table 6. **SVHN as Proxy Data ablation**: DFMS-HL achieves an accuracy of 84.83% using SVHN as Proxy data for a ResNet-34 victim model trained on CIFAR-10. ResNet-18 used as Clone architecture.

Method	Synthetic	CIFAR-100 (40C)	CIFAR-100 (10C)	SVHN
DFME	88.10	88.10	88.10	88.10
DFMS-HL (Ours)	84.51	92.06	85.53	84.83

4.4. Impact of Discriminator

The discriminator is an essential component of our approach. Across training epochs, the discriminator learns to differentiate between proxy data and fake images produced by the generator. We conduct an ablation experiment by disabling the discriminator updates. We use CIFAR-10 as the true dataset and synthetic data as the proxy dataset for this experiment. For Alexnet as victim model and AlexNet-Half as clone model, DFMS-HL attains an accuracy of 67.03%. After disabling the discriminator, the clone accuracy drops to 57.06% and the images look degenerate as shown in Fig. 3. Hence, the discriminator also plays a crucial role in maintaining the distribution of images.

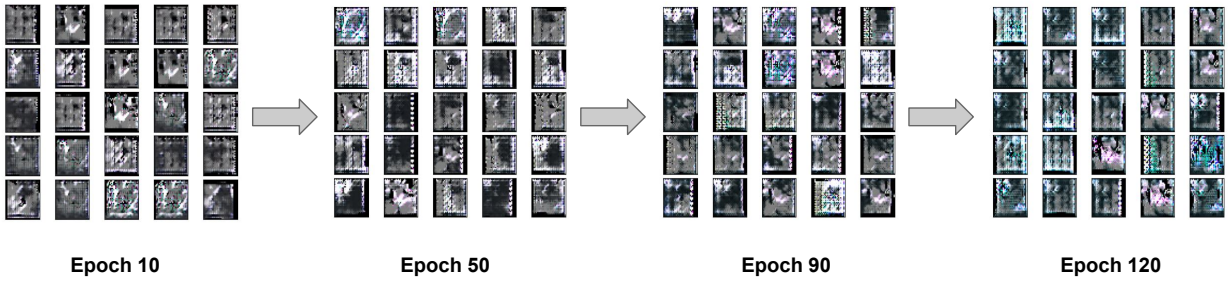

Figure 3. **Output of DFMS-HL after disabling the discriminator.** The images converge to degenerate cases after few epochs of training. Synthetic data is used as proxy data with an AlexNet victim model trained on CIFAR-10 and clone model as AlexNet-half.

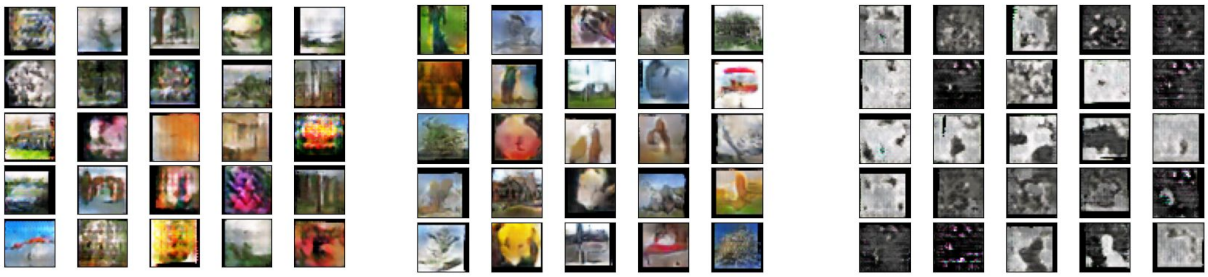

Figure 4. **DFMS-HL generator images.** The images generated by DFMS-HL generator for CIFAR-100 10 random classes, 40 unrelated classes and synthetic data as proxy for an AlexNet victim model of accuracy 80.18% trained on CIFAR-10 and clone model as AlexNet-half.

4.5. Impact of L1 loss in DFMS-SL

Prior works on Knowledge Distillation [4–7] train a student model using a KL-divergence loss between the student and teacher predictions. Let $\mathcal{V}_i(x)$ and $\mathcal{C}_i(x)$ be the output of class i (out of K classes) of the victim and clone models respectively. The KL divergence loss is written as follows,

$$\mathcal{L}_{KL} = \sum_{i=0}^{K} \mathcal{V}_i(x) log \left[\frac{\mathcal{V}_i(x)}{\mathcal{C}_i(x)} \right] \tag{4}$$

The DFME approach [8] used an L1 loss formulation where they consider the L1 difference between the logits of the clone and the victim model. The logits are estimated by first taking log, then subtracting the mean of the predictions from it. The loss formulation is written as follows,

$$\mathcal{L}_{l1} = \sum_{i=0}^{K} \mid \mathcal{V}_i^{logits}(x) - \mathcal{C}_i^{logits}(x) \mid \tag{5}$$

where,

$$\mathcal{V}_i^{logits}(x) = log\mathcal{V}_i(x) - \frac{1}{K} \sum_{j=1}^{K} log\mathcal{V}_j(x) \tag{6}$$

We evaluate our approach in the soft-label setting with the two loss functions of L1 loss and KL-divergence loss as shown in Table 5. We observe an improvement in the clone accuracy using synthetic data by 3% by using L1 loss for distillation.

4.6. Using unrelated data as the Proxy Dataset

The amount of relatedness between the proxy data and true data is an important factor that influences the success of model stealing. We perform an ablation study using SVHN as the proxy dataset to steal a model originally trained on CIFAR-10. Since SVHN is a completely unrelated to CIFAR-10, it is indeed a difficult setting. Our method DFMS-HL attains a clone accuracy of 84.83% in

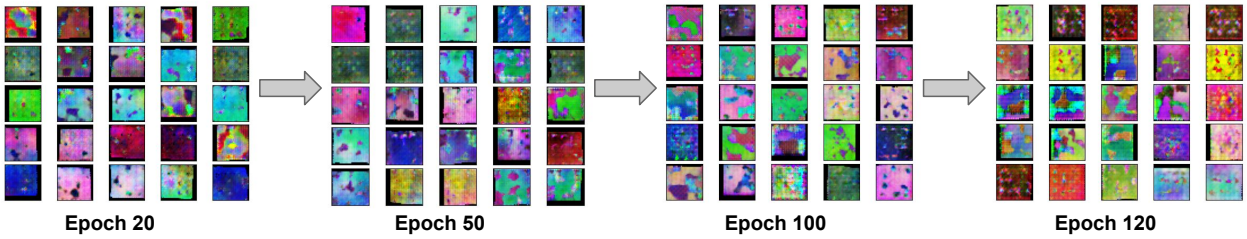

Figure 5. **DFMS-HL generator images.** The images generated by DFMS-HL generator using synthetic colour dataset as proxy for an AlexNet victim model of accuracy 80.18% trained on CIFAR-10 and clone model as AlexNet-half.

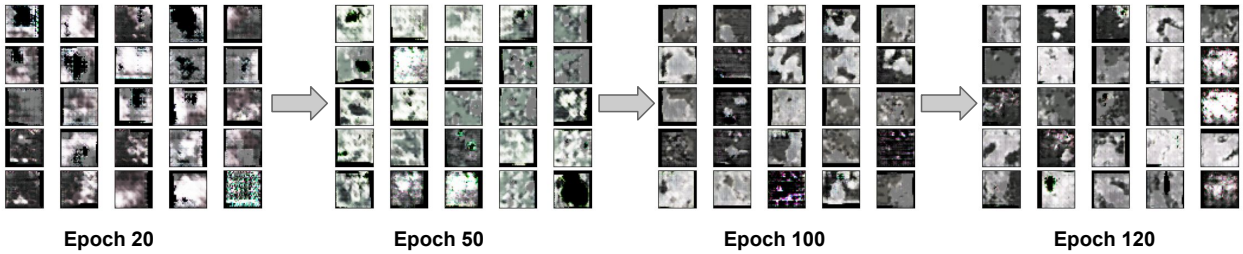

Figure 6. **DFMS-HL generator images.** The images generated by DFMS-HL generator using grey-scale synthetic images as proxy for an AlexNet victim model of accuracy 80.18% trained on CIFAR-10 and clone model as AlexNet-half.

this setting. This shows our attack is strong enough to work across a wide range of unrelated proxy datasets.

5. GAN generated Images

The images generated from the DFMS-HL GAN are shown in Fig. 4, 5 and 6. Initially, the generator starts generating images which closely resemble the proxy data. In the synthetic data experiments (Fig.5 and 6), as the training progresses, we observe that the shapes start merging with each other and start looking more continuous in nature. This makes the image look close to real images which have an object in front of a background. This shows that the generator starts capturing properties of the true training data distribution, as they look more intuitive than the original synthetic images. This helps the clone model learn intrinsic properties of the victim's training data.

6. Limitations and Future Directions

One of the crucial factors of a successful model stealing attack is its query budget. Our approach has reduced the number of queries required to 8 million, which is $\sim 500\times$ lesser than the query budget used by past methods of model stealing and knowledge distillation. We believe that reducing the query budget further would be an interesting area for future research. Another limiting factor for an adversary is the lack of relevant training data. Our approach addresses this limitation to quite an extent, as we showcase promising results in a limited data scenario by just using synthetic images. We believe that our approach would pave the way to address these limitations and develop stronger attacks and defenses in the area of hard-label model stealing.

References

[1] Sravanti Addepalli, Gaurav Kumar Nayak, Anirban Chakraborty, and Venkatesh Babu Radhakrishnan. DeGAN: Data-enriching gan for retrieving representative samples from a trained classifier. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020. 1, 2

[2] Antonio Barbalau, Adrian Cosma, Radu Tudor Ionescu, and Marius Popescu. Black-Box Ripper: Copying black-box models using generative evolutionary algorithms. *arXiv preprint arXiv:2010.11158*, 2020. 1, 2

[3] Varun Chandrasekaran, Kamalika Chaudhuri, Irene Giacomelli, Somesh Jha, and Songbai Yan. Exploring connections between active learning and model extraction. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*, 2020. 2

[4] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 4

[5] Raphael Gontijo Lopes, Stefano Fenu, and Thad Starner. Data-free knowledge distillation for deep neural networks. *arXiv preprint arXiv:1710.07535*, 2017. 4

[6] Gaurav Kumar Nayak, Konda Reddy Mopuri, Vaisakh Shaj, Venkatesh Babu Radhakrishnan, and Anirban Chakraborty. Zero-shot knowledge distillation in deep networks. In *International Conference on Machine Learning*. PMLR, 2019. 4

[7] 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*, pages 506–519, 2017. 4

[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*, pages 4771–4780, 2021. 4

[9] Zi Wang. Zero-shot knowledge distillation from a decision-based black-box model. *arXiv preprint arXiv:2106.03310*, 2021. 2