SUPPLEMENTAL MATERIAL FOR: DUAL STUDENT NETWORKS FOR DATA-FREE MODEL STEALING

James Beetham[1*], **Navid Kardan**[1*], **Ajmal Mian**[2], **Mubarak Shah**[1]
[1]Center for Research in Computer Vision
University of Central Florida
Orlando, Florida 32816, USA
[2]Department of Computer Science
University of Western Australia
Crawley WA 6009, Australia
{james.beetham,kardan}@knights.ucf.edu
ajmal.mian@uwa.edu.au
shah@crcv.ucf.edu

1 CLASSIFICATION EXPERIMENTS

Providing additional details to the CIFAR10 classification results from our method, we provide Table 1 for comparing our reported results with the results from the DFMS-SL/HL paper (Sanyal et al., 2022). We also note that to extend the DFME method to hard-labels without using Dual Students, we change their l_1 loss to multi-margin loss (Truong et al., 2021). Multi-margin loss was tested for both DFME and Dual Students, however, while multi-margin increased DFME by around 12% boost compared with using cross-entropy loss, it reduced Dual Students accuracy on CIFAR10. Thus we show the DFME hard label using multi-margin loss to provide a best-case scenario for DFME. We also show additional results for the BIM attack (Kurakin et al., 2016) in Table 4.

Table 1: Percent student accuracy for different methods with and without the Dual Student (DS) method when using different target models. [1]The first two new results for DFMS-SL and DFMS-HL are taken directly from the DFMS-HL paper. For soft-labels, our new Dual Student method outperforms their reported results. [3]For the hard label setting, although the reported DFMS-HL results beat our Dual Student method alone, using our Dual Student method to fine-tune the fully trained Generator and Student models improves the accuracy. [2]The bottom two results shown in the table use a ResNet18 architecture for their Target model. We did not include these three rows in the primary paper because the results are not directly comparable due to the use of different target models with varying accuracies and architectures.

Dataset	Target Accuracy	Method	Probabilities	Hard-Labels
CIFAR10	95.5	DFME (ℓ_1)	88.10	-
	95.5	DFME (ce)	-	56.35
	95.5	DFME (multi-margin)	-	68.40
	95.5	DS	**91.34**	78.72
	95.5	DFMS-SL/HL	83.02	79.61
	95.5	DS + DFMS-SL/HL	89.38	**85.06**
CIFAR10	95.59	DFMS-SL	91.24[1]	-
	93.7[2]	DFMS-HL	-	85.92[1]
	93.7[2]	DS + DFMS-HL	-	**88.46**[3]

[*]Equal contribution

2 DUAL STUDENTS TRAINING SETUP DETAILS

Referring to Algorithm 1, we use mostly the same hyperparameters outlined in DFME (Truong et al., 2021). The generator architecture is the same used in the DFME method: 3 convolutional layers with linear upsampling, batch normalization, and ReLU activations. For i_G and i_S we use 1 and 5 respectively. An ablation of this ratio is provided in Table 2. We use ℓ_1 loss on soft-labels, and cross-entropy loss on hard-labels for $\mathcal{L}_S$. For $\mathcal{L}_\mathcal{G}$ we only use ℓ_1 loss. For student learning rates in the SGD loss, we use $\alpha_G = (0.0001, 0.0001)$ and $\alpha_S = (0.3, 0.05)$ for soft-label and hard-label training respectively, with weight decay of 0.0005 and momentum of 0.9. We use a moving average momentum, as outlined in Cai et al. (2021) of 0.9, and balance the 20 million queries for CIFAR10 across 224 epochs. Visual examples of images generated during CIFAR10 training can be seen on the left side of Figure 1.

Algorithm 1 Proposed Dual Students Method

Input: target model T, student models S_1, S_2, generator G, model parameters $\theta_{S_1}, \theta_{S_2}, \theta_G$
generator iters i_G, student iters i_S, epochs i_E, learning rates α_G, α_S

for e **through** i_E **do**
  for i_g **through** i_G **do** // **Train Generator**
    $z \sim U(0, 1)$ // uniform noise
    $x \leftarrow G(z; \theta_G)$ // generate images
    $l_G \leftarrow -\mathcal{L}_G(S_1(x; \theta_{S_1}), S_2(x; \theta_{S_2}))$ // (maximize) distance between students
    $\theta_{\mathbf{G}} \leftarrow \theta_{\mathbf{G}} + \alpha_G \nabla_{\theta_{\mathbf{G}}} l_G$ // update Generator
  end for
  for i_s **through** i_S **do** // **Train Students**
    $z \sim U(0, 1)$
    $x \leftarrow G(z; \theta_G)$
    $t \leftarrow T(x)$ // query Target Model
    $l_{S_1} \leftarrow \mathcal{L}_S(S_1(x; \theta_{S_1}), t)$ // (minimize) distance to Target
    $l_{S_2} \leftarrow \mathcal{L}_S(S_2(x; \theta_{S_2}), t)$
    $\theta_{\mathbf{S_1}} \leftarrow \theta_{\mathbf{S_1}} + \alpha_S \nabla_{\theta_{\mathbf{S}}} l_{S_1}$ // update Student
    $\theta_{\mathbf{S_2}} \leftarrow \theta_{\mathbf{S_2}} + \alpha_S \nabla_{\theta_{\mathbf{S_2}}} l_{S_2}$
  end for
end for
return S_1

Table 2: Ratio between generator and student training iterations for CIFAR10 soft-labels on Dual Students.

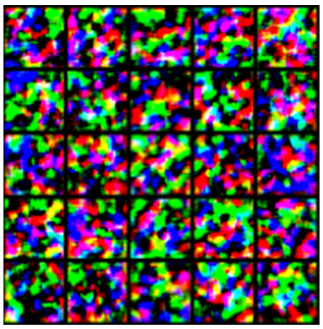
Ratio	1:1	1:4	**1:5**	1:6	1:10
Accuracy	66.59	91.12	**91.34**	91.07	76.05

Table 3: Number of queries (in millions) required by different methods to reach a target accuracy on CIFAR10 using soft-labels. These are values are taken from the graph provided in Figure 1 (right).

Accuracy	75%	80%	85%
DFME	4.89	7.02	11.56
DFMS-SL	5.89	8.21	10.98
DS	3.57	5.54	6.43

3 QUERY EFFICIENCY

The query efficiency of DFME compared with Dual Students is shown on the right side of Figure 1 and in Table 3. Referring to Algorithm 1, this query efficiency comes from removing the Target model gradient estimate from the Generator loop that's present in the DFME method. The DFME method uses the same Generator-Student iterative training as Dual Students, and uses the same hyperparameters for number of Generator to Student iterations (1:5). Thus, the only queries are made during Student training, which has the same number of queries as the DFME method student training. Ultimately this benefit results in around a 30% decrease in number of queries needed, however, benefits from using a better gradient estimator in Dual Students result in even higher accuracies.



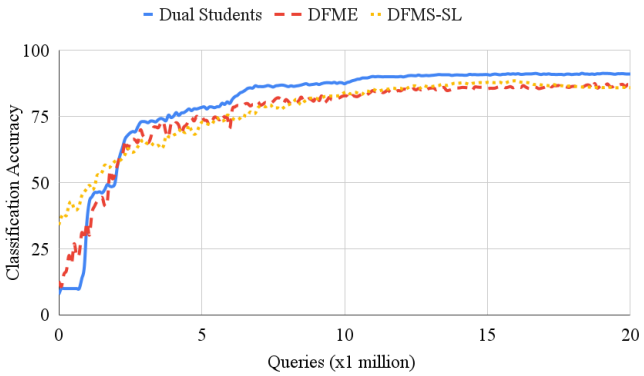


Figure 1: Examples of images generated during DS training on CIFAR10 (left), and query efficiency of the different methods on CIFAR10 using soft-labels (right).

4 EVALUATING GENERATOR CLASS DISTRIBUTION

The Dual Student method does not use any explicit class distribution balance term in either Generator or Student loss. This is similar to DFME and helps simplify the loss for few-class datasets like CIFAR10. However, when scaling up to CIFAR100, even class distribution becomes more important. As is shown in Figure 2, Dual Students has a more balanced class distribution than the DFME method on CIFAR10 throughout training. However, Dual Students suffers from class imbalance when extended to CIFAR100. Of note in Figure 2 is that in the percent classes generated at the end of training, there are particular classes like Class 8 that appear to be more difficult to generate than other classes. These difficult classes persist across multiple runs, so a class distribution balancing term may be beneficial to scaling up to datasets with more classes.

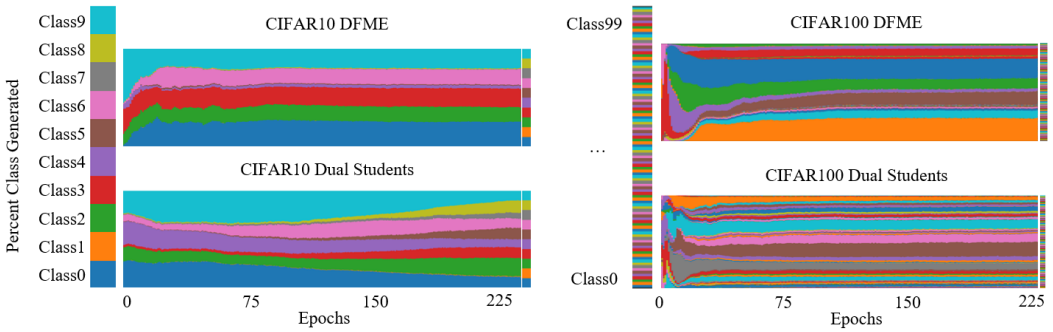


Figure 2: Distribution of classifications of generator images during training. X-axis is number of epochs during training, Y-axis is a stacked chart of the percent prevalence of classes, where the different classes are different colors. The bars on either side of the graph are references for if the distribution of classes was uniform.

Table 4: Additional results using the BIM attack, extending Table 2 in the main paper (Kurakin et al., 2016; Goodfellow et al., 2015; Madry et al., 2017). Attack success percentage of different DFMS methods on the Target Model trained on CIFAR10 when attack $\epsilon = \frac{3}{255}$. All attacks are evaluated on the Target Model. The Target Model row is a white-box attack, Proxy Model is a transfer-based black-box attack where the proxy is trained using the same data as the target model. The other DFMS methods provide trained student models which act as the proxy in transfer-based black-box attacks.

Attack	**Method**	**Untargeted Attacks**		**Targeted Attacks**	
		Probabilities	**Hard-Labels**	**Probabilities**	**Hard-Labels**
FGSM	*Target Model*	*45.00*		*19.64*	
	Proxy Model	*33.12*		*14.38*	
	DFME	56.84	39.22	21.07	17.15
	DS	**62.35**	44.58	**21.58**	21.04
	DFMS-HL/SL	54.88	48.89	19.74	21.85
	DFMS-HL/SL + DS	54.99	**50.41**	20.53	**23.59**
BIM	*Target Model*	*96.74*		*76.90*	
	Proxy Model	*57.50*		*29.74*	
	DFME	84.38	55.52	53.39	31.95
	Dual Students	**91.56**	62.76	**63.39**	34.85
	DFMS-HL/SL	83.25	73.19	52.99	41.00
	DFMS-HL/SL + DS	82.23	**74.60**	51.89	**43.47**
PGD	*Target Model*	*96.78*		*76.32*	
	Proxy Model	*55.01*		*28.33*	
	DFME	83.59	54.71	51.97	31.49
	DS	**91.04**	62.21	**61.96**	33.39
	DFMS-HL/SL	81.97	72.40	52.64	39.95
	DFMS-HL/SL + DS	81.04	**73.08**	51.38	**42.53**

REFERENCES

Zhaowei Cai, Avinash Ravichandran, Subhransu Maji, Charless Fowlkes, Zhuowen Tu, and Stefano Soatto. Exponential moving average normalization for self-supervised and semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 194–203, 2021.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

Alexey Kurakin, Ian Goodfellow, Samy Bengio, et al. Adversarial examples in the physical world, 2016.

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.

Sunandini Sanyal, Sravanti Addepalli, and R. Venkatesh Babu. Towards data-free model stealing in a hard label setting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15284–15293, June 2022.

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*, pp. 4771–4780, 2021.