Model Stealing Attacks

Minxue Tang

Xueying Wu

Yitu Wang

Duke

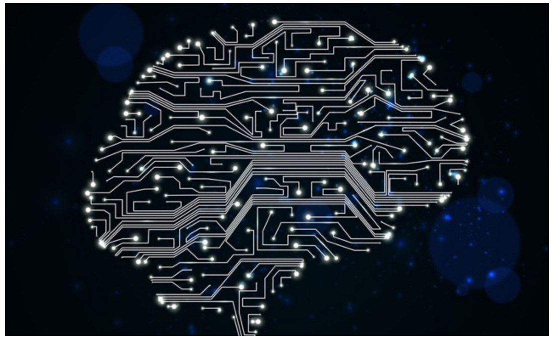
Contents

Contents

Duke

Background

- Machine-Learning-as-a-Service (MLasS) System
  - A confidential model is deployed for some paid service as a black box.
  - The confidential model is an asset of the model provider.
    - Data collection, model training and services deployment are expensive!



- Model Stealing (Extraction) Attacks
  - Aiming at stealing the parameters or functionality of the confidential model.
  - Model Extraction is usually done by querying the confidential model and learning from its response.
  - Avoid Subscription and paying after stealing, or uncover security vulnerabilities of the model.
- Metrics
  - Accuracy: How well the extracted model performs on a target test dataset
  - Fidelity: How similarly the extracted model imitates the confidential model

Contents

Duke

Standard Model Extraction Attack[1]

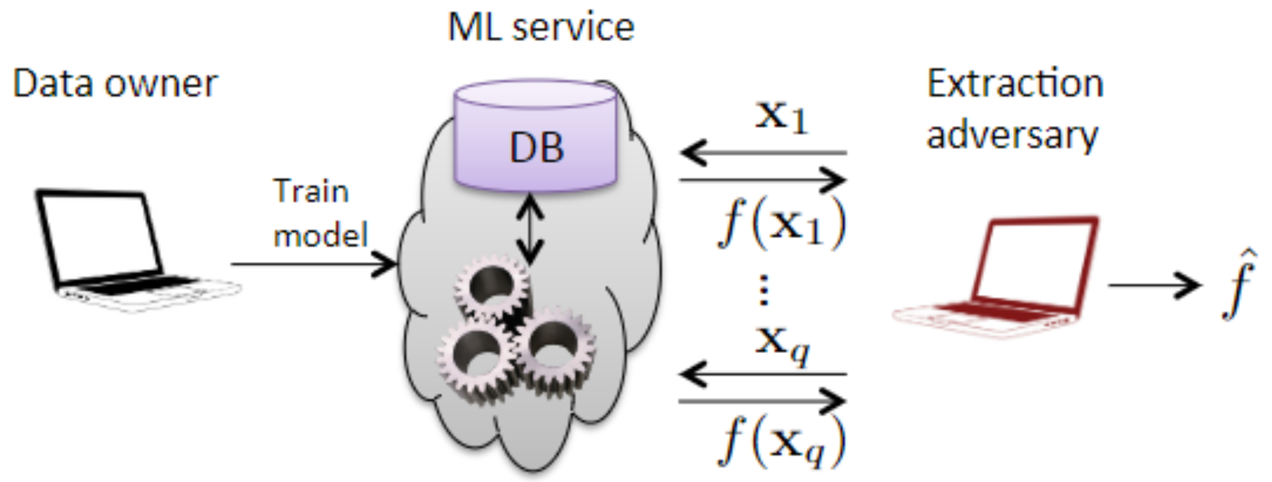
- Attack by Querying with Natural Data

- Metrics
  - Test Error R_{test}: Fidelity on a target test set.

$$R_{test} = \mathbb{E}_{D_{test}}\left[d\left(f(x), \hat{f}(x)\right)\right]$$

  - Uniform Error R_{unif}: Fidelity on uniformly random vectors.

$$R_{unif} = \mathbb{E}_{U}\left[d\left(f(x), \hat{f}(x)\right)\right]$$



[1] Tramèr, Florian, et al. "Stealing machine learning models via prediction {APIs}." *25th USENIX security symposium (USENIX Security 16)*. 2016.

Duke

Standard Model Extraction Attack

- Case 1: The confidential model is known, and the parameters of the model can be solved from the equations related to input-output pairs.
- Logistic Regression (LR)

$$w^T x + \beta = \sigma^{-1}(f(x))$$

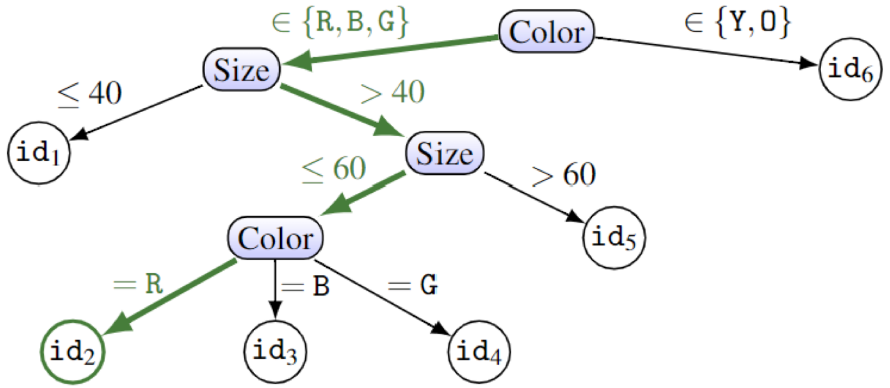
- Softmax Multiclass LR
- One-vs-Rest Multiclass LR (OvR)
- Multilayer Perceptrons (MLP)
- Kernel LR

Model	Unknowns	Queries	$1 - R_{\text{test}}$	$1 - R_{\text{unif}}$	Time (s)
Softmax	530	265	99.96%	99.75%	2.6
		530	100.00%	100.00%	3.1
OvR	530	265	99.98%	99.98%	2.8
		530	100.00%	100.00%	3.5
MLP	2,225	1,112	98.17%	94.32%	155
		2,225	98.68%	97.23%	168
		4,450	99.89%	99.82%	195
		11,125	99.96%	99.99%	89

Duke

Standard Model Extraction Attack

- Case 2: The confidential model is known, but the output cannot be written as a continuous function.
- Decision Tree
  - Path-finding Attack
  - Require to know the ID of the returned leaf
  - Step 1: Find the constraints that input must satisfy to reach a specific leaf.
  - Step 2: Create new input to explore other paths in the tree.



Model	Leaves	Unique IDs	Depth	$1 - R_{\text{test}}$	$1 - R_{\text{unif}}$	Queries
IRS Tax Patterns	318	318	8	100.00%	100.00%	101,057
Steak Survey	193	28	17	92.45%	86.40%	3,652
GSS Survey	159	113	8	99.98%	99.61%	7,434
Email Importance	109	55	17	99.13%	99.90%	12,888
Email Spam	219	78	29	87.20%	100.00%	42,324
German Credit	26	25	11	100.00%	100.00%	1,722
Medical Cover	49	49	11	100.00%	100.00%	5,966
Bitcoin Price	155	155	9	100.00%	100.00%	31,956

Standard Model Extraction Attack

- Case 3: The model is unknown (black-box), or only parts of outputs (e.g., only top-1 labels) are returned to the user. The attacker can retrain a model to imitate the functionality of the confidential model.

- Key idea: Find samples close to the decision boundary
  - Lowd-Meek Attack: Use Linear search to find the samples close to the boundary of a linear model.

$$w^T x + \beta \approx 0$$

  - Uniform Queries: Uniformly random samples
  - Line-Search Retraining: Generalize Lowd-Meek Attack to non-linear model.
  - Adaptive Retraining: Repeat sampling along the boundary of $\hat{f}$, training to get new $\hat{f}$.

Duke

Standard Model Extraction Attack

- Retraining Results

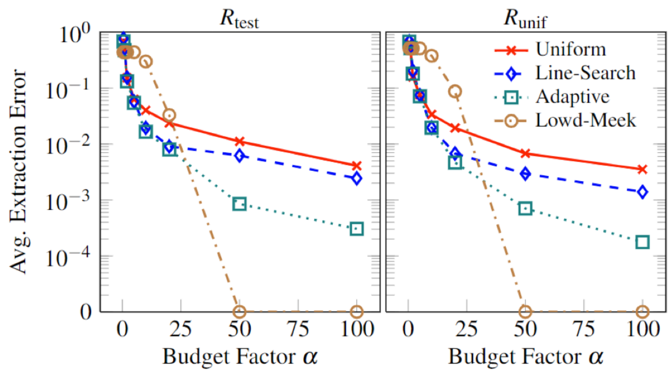


Figure 4: **Average error of extracted linear models.** Results are for different extraction strategies applied to models trained on all binary data sets from Table 3. The left shows R_{test} and the right shows R_{unif}.

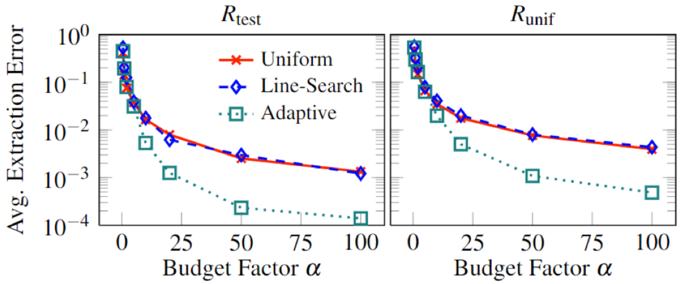


Figure 5: **Average error of extracted softmax models.** Results are for three retraining strategies applied to models trained on all multiclass data sets from Table 3. The left shows R_{test} and the right shows R_{unif}.

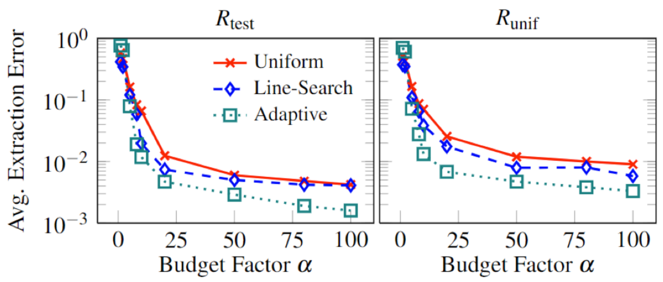
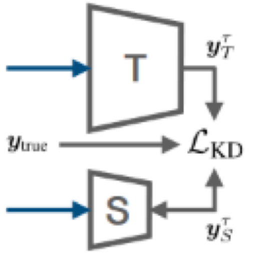


Figure 6: **Average error of extracted RBF kernel SVMs** Results are for three retraining strategies applied to models trained on all binary data sets from Table 3. The left shows R_{test} and the right shows R_{unif}.

- Extracting Larger models (Neural Networks)
  - It becomes more difficult to find the decision boundary of a complex model like NN.
  - Marginal improvement is reported by using adaptive retraining

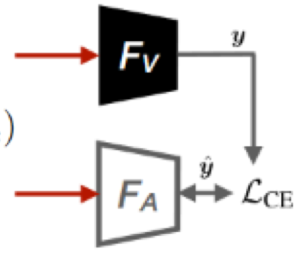
Knockoff nets: Model Extraction Attack with AL[2]

- Active Learning (AL):
  - **Reducing label effort** while gathering data to train a model
  - Reinforcement learning approach
- Extracting the victim model via a knowledge-distillation (KD)-like method
  - Generating a transfer set for the adversary model.
  - Querying a set of input images to the blackbox model to obtain predictions.
  - Training a "knockoff" with queried image-prediction pairs.
- Comparison to KD:
  - Lacks knowledge to the victim model's training dataset.
  - Lacks knowledge to the victim model's architecture.
  - Lacks knowledge to the victim model's logits and true labels.



Knowledge Distillation (KD)

$$\mathcal{L}_{\text{CE}}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\sum_k p(y_k) \cdot \log p(\hat{y}_k)$$



Ours

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

Knockoff nets: Model Extraction Attack with AL

- Transfer set construction
  - Random strategy: randomly sample images to query Fv. $\boldsymbol{x} \overset{\text{iid}}{\sim} P_A(X)$
  - Adaptive strategy: incorporate a feedback signal resulting from each image queried to the blackbox.

$$\boldsymbol{x}_t \sim \mathbb{P}_\pi(\{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=1}^{t-1})$$

$$\pi_t(z) = \frac{e^{H_t(z)}}{\sum_{z'} H_t(z')}$$

  - Learning the sampling policy:

Penalizes the sampler from keeping sampling the same node

$$H_{t+1}(z_t) = H_t(z_t) + \alpha(r_t - \bar{r}_t)(1 - \pi_t(z_t)) \qquad \text{and}$$
$$H_{t+1}(z') = H_t(z') + \alpha(r_t - \bar{r}_t)\pi_t(z') \qquad \forall z' \neq z_t$$

Prevents the reward from sticking in the same value

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

Knockoff nets: Model Extraction Attack with AL

- Transfer set construction
  - Rewards:
    - Certainty measure: to encourage images where the victim is confident.

$$R^{\text{cert}}(\boldsymbol{y}_t) = P(\boldsymbol{y}_{t,k_1}|\boldsymbol{x}_t) - P(\boldsymbol{y}_{t,k_2}|\boldsymbol{x}_t)$$

    - Diversity reward: to prevent the degenerate case of image exploitation over a single label.

$$R^{\text{div}}(\boldsymbol{y}_{1:t}) = \sum_k \max(0, \bar{\boldsymbol{y}}_{t,k} - \bar{\boldsymbol{y}}_{t-\Delta,k})$$

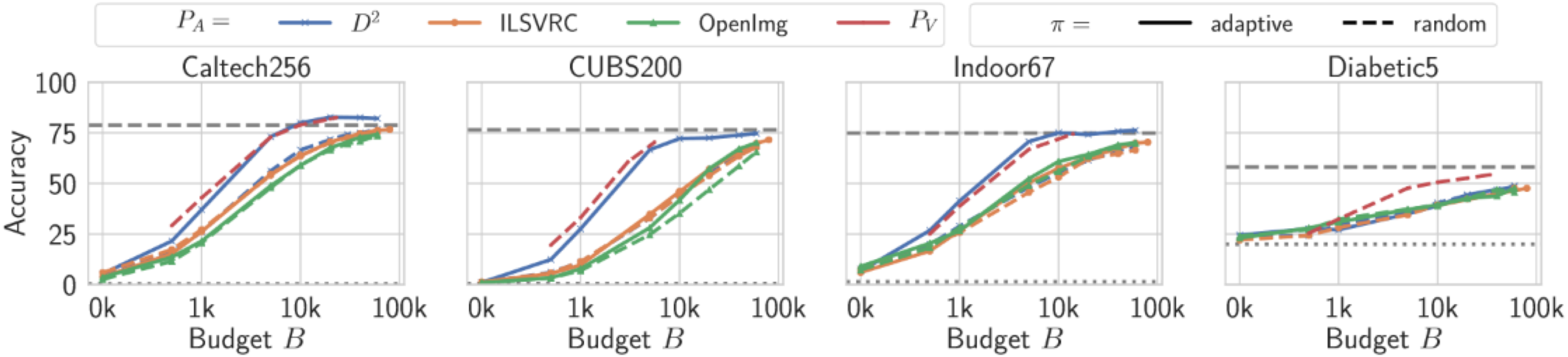
    - Loss: To encourage images where the knockoff prediction doesn't imitate the victim prediction.

$$R^{\mathcal{L}}(\boldsymbol{y}_t, \hat{\boldsymbol{y}}_t) = \mathcal{L}(\boldsymbol{y}_t, \hat{\boldsymbol{y}}_t)$$

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

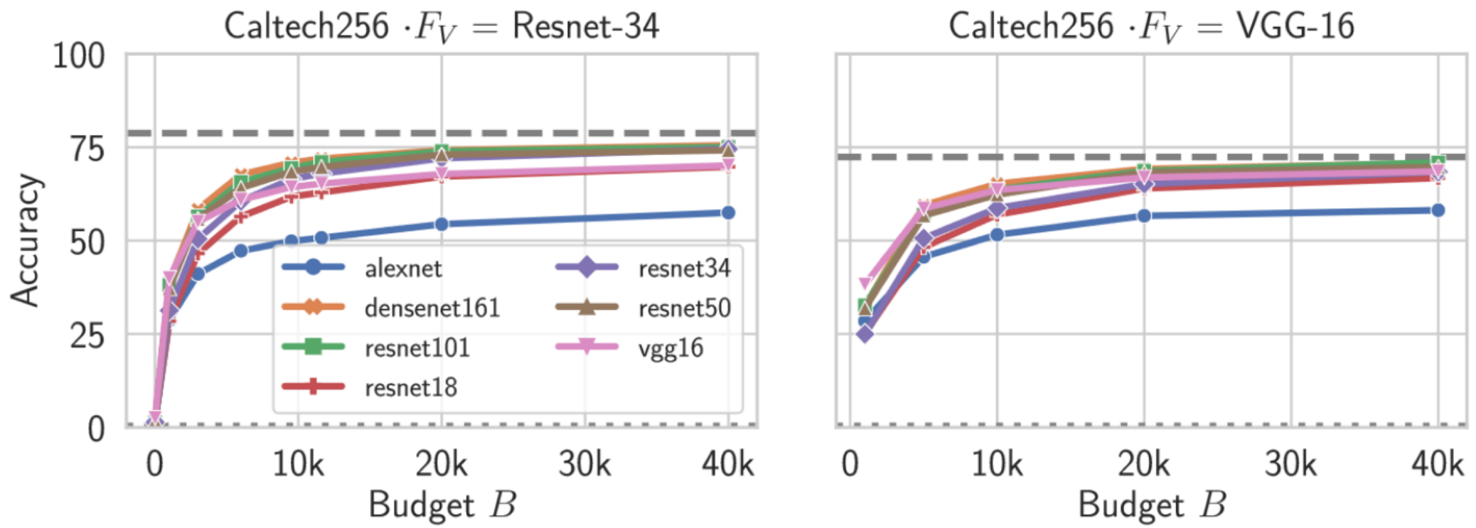
Knockoff nets: Model Extraction Attack with AL

- Experimental results
  - Accuracy vs budget:
    - Higher accuracy is achieved when more images are sampled for the adversary.
  - Comparison between the sampling strategies:
    - Adaptive strategy generally performs better than random strategy
    - If the adversary is trained on the same dataset as the victim, the highest accuracy is achieved.



Knockoff nets: Model Extraction Attack with AL

- Experimental results
  - Adversary's model architecture:
    - When the adversary has the same architecture as the victim, the highest accuracy can be achieved.
    - Higher performance is achieved when adversary models have higher model complexity.



Model Extraction Attack with SSL[3]

- Self-supervised learning does not require labels!

$$L = L_{supervised} + L_{SSL}$$

- SSL Method 1: Rotation Loss

$$L_R(X; f_\theta) = \frac{1}{4N} \sum_{i=0}^{N} \sum_{j=1}^{r} H(f_\theta(R_j(x_i)), j)$$

$j = 0$ $j = 1$ $j = 2$ $j = 3$

[3] Jagielski, Matthew, et al. "High accuracy and high fidelity extraction of neural networks." *Proceedings of the 29th USENIX Conference on Security Symposium*. 2020.

Model Extraction Attack with SSL

- Model Extraction with Rotation Loss

Architecture	Data Fraction	ImageNet	WSL	WSL-5	ImageNet + Rot	WSL + Rot	WSL-5 + Rot
Resnet_v2_50	10%	(81.86/82.95)	(82.71/84.18)	(82.97/84.52)	(82.27/84.14)	(82.76/84.73)	(82.84/84.59)
Resnet_v2_200	10%	(83.50/84.96)	(84.81/86.36)	(85.00/86.67)	(85.10/86.29)	(86.17/88.16)	(86.11/87.54)
Resnet_v2_50	100%	(92.45/93.93)	(93.00/94.64)	(93.12/94.87)	N/A	N/A	N/A
Resnet_v2_200	100%	(93.70/95.11)	(94.26/96.24)	(94.21/95.85)	N/A	N/A	N/A

- SSL Method 2: MixMatch[3]
    - A combination of Techniques
    - “Guessed” Labels
    - Regularization
    - Image Augmentations

Dataset	Algorithm	250 Queries	1000 Queries	4000 Queries
SVHN	FS	(79.25/79.48)	(89.47/89.87)	(94.25/94.71)
SVHN	MM	(95.82/96.38)	(96.87/97.45)	(97.07/97.61)
CIFAR10	FS	(53.35/53.61)	(73.47/73.96)	(86.51/87.37)
CIFAR10	MM	(87.98/88.79)	(90.63/91.39)	(93.29/93.99)

[3] Berthelot, David, et al. "Mixmatch: A holistic approach to semi-supervised learning." *Advances in neural information processing systems* 32 (2019).

Contents

Duke

Preliminary

- Knowledge distillation
  - Goal
    - Compress, i.e., transfer the knowledge of a (larger) teacher model to a (smaller) student model
  - Methods
    - Non-data-free knowledge distillation
      - Leveraging a surrogate dataset with a similar feature space or distribution
    - Data-free knowledge distillation
      - Relying on training a generative model to synthesize the queries that the student makes to the teacher

How Important is the Surrogate Dataset?

- The distribution of the surrogate dataset should be <u>*close*</u> to that of the victim's training dataset.
  - Similarity in feature space, marginal/class-conditional probability distribution of inputs

	Victim	CIFAR10	CIFAR100	SVHN	MNIST	$SVHN_{skew}$	Random
CIFAR10	95.5%	95.2%	93.5%	66.6%	37.2%	-	10.0%
SVHN	96.2%	96.0%	-	96.3%	89.5%	96.1%	84.1%

- Conclusion
  - The success of distillation-based model extraction largely depend on the complexity of the task that the victim model aims to solve
  - Similarity to source domain appears to be critical for extracting ML models that solve complex tasks

Data-Free Model Extraction[4]

- Goal:
  - Train a student model to match the predictions of the victim on its private target domain
  - Find the student model's parameters that minimize the probability of errors between the student and victim predictions

$$\arg\min_{\theta_S} \mathcal{P}_{x\sim\mathcal{D}_\mathcal{V}} \left(\arg\max_i \mathcal{V}_i(x) \neq \arg\max_i \mathcal{S}_i(x) \right)$$

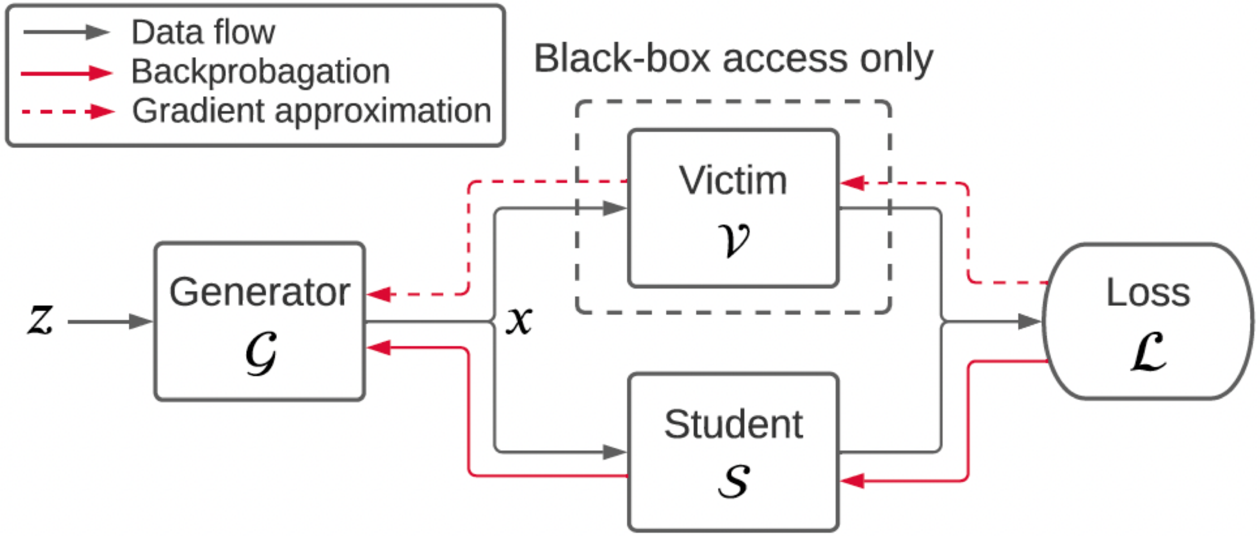
  - Minimize the student's error on a synthesized dataset
  - The error is minimized by optimizing a loss function which measures disagreement between the victim and student

$$\arg\min_{\theta_S} \mathbb{E}_{x\sim\mathcal{D}_S} \left[\mathcal{L}(\mathcal{V}(x), \mathcal{S}(x))\right]$$

[4] Truong, Jean-Baptiste, et al. "Data-free model extraction." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021.

Data-Free Model Extraction

- Attack Setting



  - Generator: synthesizes training data points x – generate difficult examples for the student
  - Students: learns the behavior of the victim model on x – match the victim's predictions
  - Loss function: measures the divergence between victim and student model

Data-Free Model Extraction: Loss Function

- $L_1 - norm$ loss

$$\mathcal{L}_{\ell_1}(x) = \sum_{i=1}^{K} |v_i - s_i|$$

  - where v_i and s_i are the logits of the victim and student models

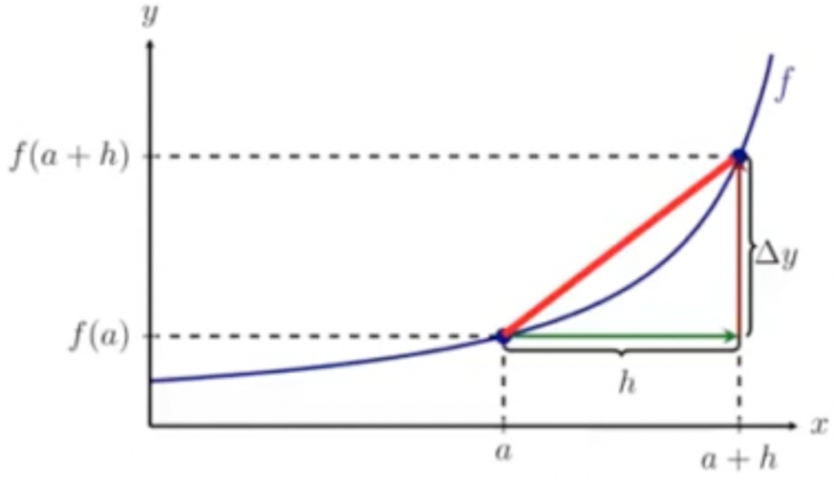
  - **Advantage:** no vanishing gradients at convergence (compared to KL divergence loss)
  - **Disadvantage:** requires access to the victim model's logits

Duke

Data-Free Model Extraction: Gradient Approximation

- Forward differences

$$\nabla_{\mathrm{FWD}} f(x) = \frac{1}{m} \sum_{i=1}^{m} d \frac{f(x + \epsilon \mathbf{u_i}) - f(x)}{\epsilon} \mathbf{u_i}$$



  - u_i: random direction
  - m: number of random directions
  - d: dimensionality of the space
  - ϵ: a real number

Data-Free Model Extraction: Results

Dataset (budget)	Victim accuracy	DFME	DFME-KL
CIFAR10 (20M)	95.5%	88.1% (0.92×)	76.7% (0.80×)
SVHN (2M)	96.2%	95.2% (0.99×)	84.7% (0.88×)

- Successful model extraction
  - Over 0.92x the victim model accuracy
- Drawback
  - Query budget is quite high (2M and 20M queries)

Not an issue when attacking on-device ML systems where the number of queries is unrestricted.

Contents

Duke

Conclusion

- The more information about the confidential model is released, the easier to extract the model (Utility-Privacy Trade-off).
- The simpler the confidential model, the easier to extract the model.
- Active Learning and Self-supervised Learning makes model extraction attack even easier with high sampling efficiency.
- Even without natural data, synthetic data can be used to attack.
- Model Extraction Attack is a realistic threat!

Duke