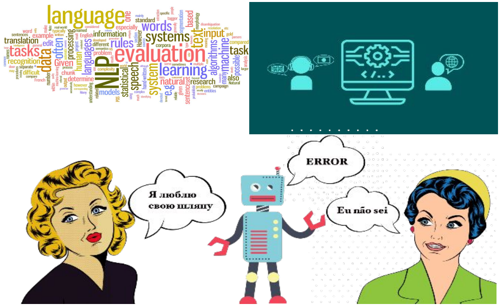
Knowledge Extraction from Deep Learning Models

Konda Reddy Mopuri
MFSDSAI, IITG

Success of Deep Learning

- Impressive progress
- Numerous applications



Deep Learning Models are heavy

- Hundreds of layers
- Millions of parameters
- Heavy memory footprint and power consumption
- → less suitable to host in resource constrained environments

Efficient DL models

1. Train them from scratch
2. Compress the sophisticated models

Efficient DL models

1. Train them from scratch
    - ?
    - Neural Architecture Search (NAS) ?

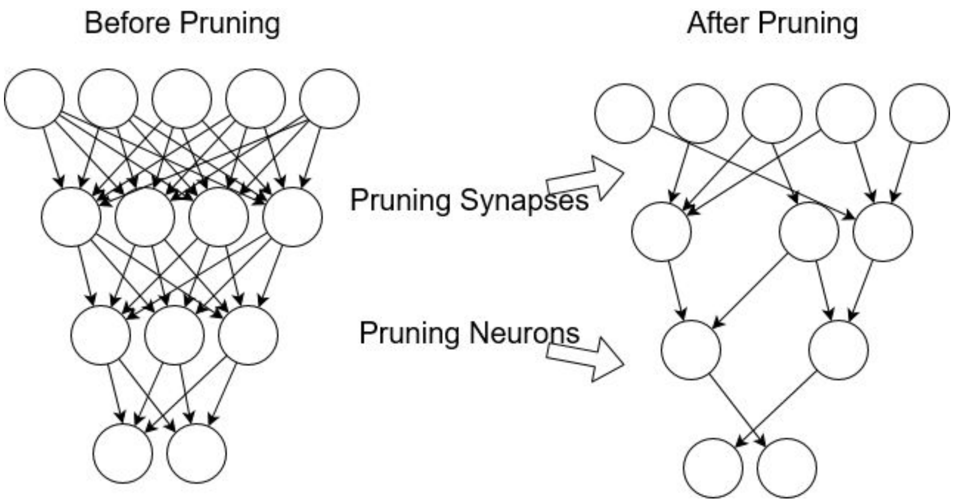
Efficient DL models

2. Compress the sophisticated models
    - Parameter Pruning
    - Parameter Quantization
    - Knowledge Distillation

Pruning

Pruning

- DNNs may have redundant weights
  - Neurons learn similar features
  - Removing them may not affect the performance



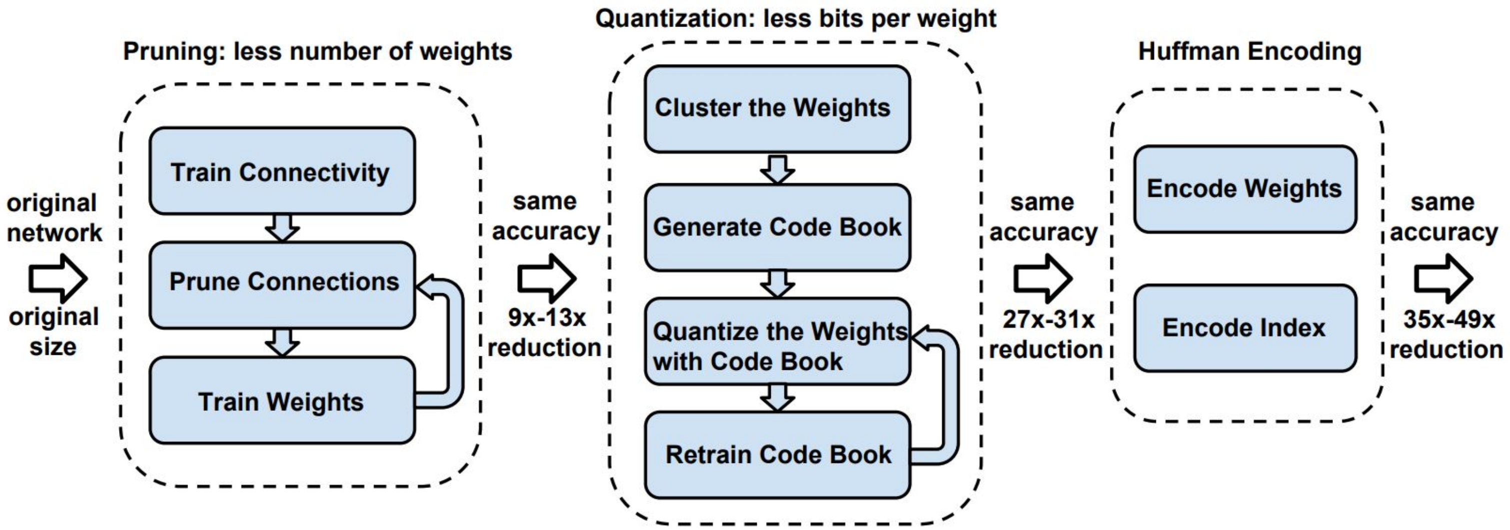
Compression via parameter pruning

- Fine vs Coarse pruning
  - Weights vs Neurons
- Static vs Dynamic pruning
  - During vs after training

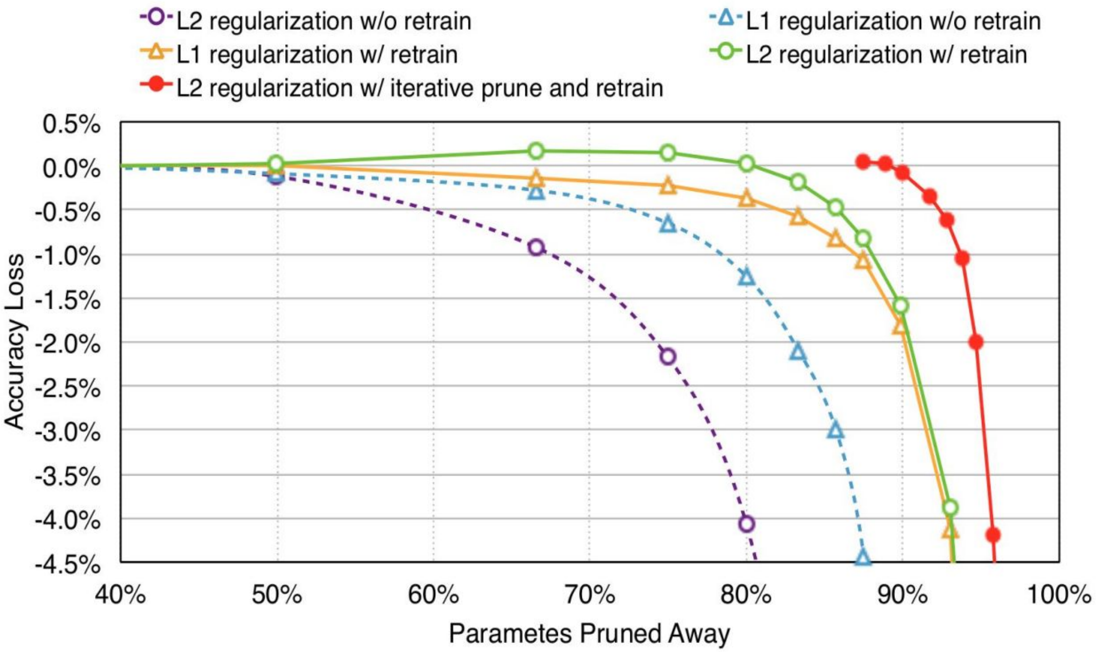
Pruning

- Naive approach: remove the weights that are equal (or close) to zero
- May use threshold values to prune
- Can be stored as sparse matrices → efficient operations

Pruning (Deep Compression, Han et al. ICLR 2016)



Pruning (Deep Compression, Han et al. ICLR 2016)



Sparsifying at train time

Weight (fine) pruning: summary

- Secondary memory footprint is reduced
- Not the RAM requirement (zeros are still present)
- Without the optimized sparse matrix operations gains can't be enjoyed!

Neuron (coarse) pruning (Drop Neuron, Pan et al.)

- Regulares to reduce the weights and thresholds prune the neurons

$$\texttt{li\_regulariser} := \lambda_{\ell_i} \sum_{\ell=1}^{L} \sum_{j=1}^{n^\ell} \|\mathbf{W}^\ell_{:,j}\|_2 = \lambda_{\ell_i} \sum_{\ell=1}^{L} \sum_{j=1}^{n^\ell} \sqrt{\sum_{i=1}^{n^{\ell-1}} (W^\ell_{ij})^2}$$

$$\texttt{lo\_regulariser} := \lambda_{\ell_o} \sum_{\ell=1}^{L} \sum_{i=1}^{n^{\ell-1}} \|\mathbf{W}^\ell_{i,:}\|_2 = \lambda_{\ell_o} \sum_{\ell=1}^{L} \sum_{i=1}^{n^{\ell-1}} \sqrt{\sum_{j=1}^{n^\ell} (W^\ell_{ij})^2}$$

Neuron (coarse) pruning: summary

- Results in smaller weight matrices → faster inference

Quantization

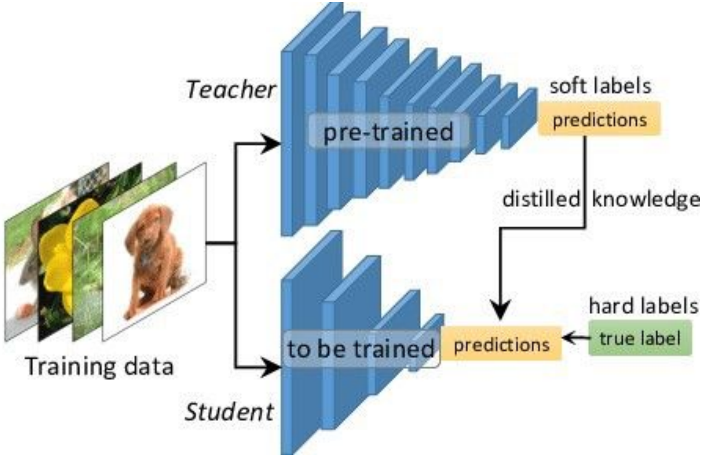
Quantization

Quantization is the process of reducing the precision of the weights, biases, and activations such that they consume less memory

- Compromise the precision of storing the weights
- Can be combined with pruning → better compression

Knowledge Distillation

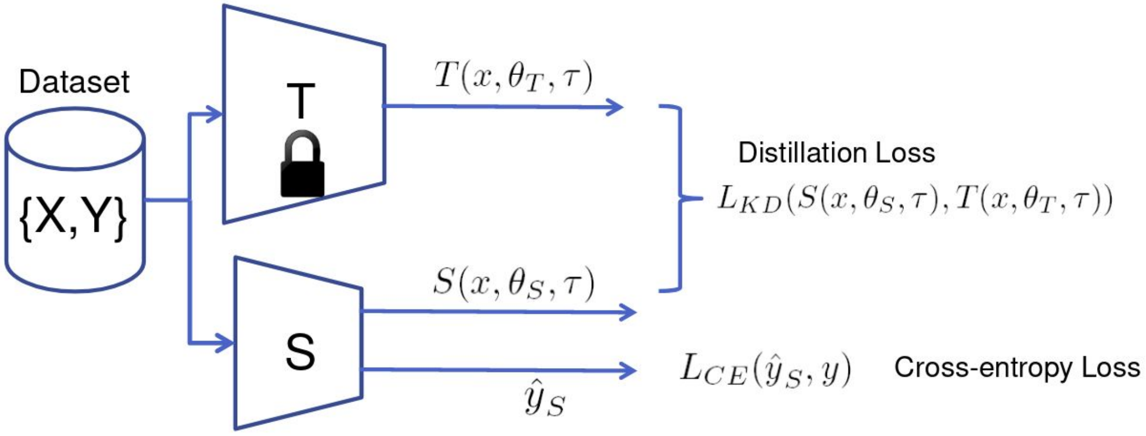
Knowledge Distillation (KD)



- Transfer the mapping function learned by a high-capacity **Teacher** model to a smaller **Student** model

Figure from https://towardsdatascience.com

Knowledge Distillation (KD)



$$L = \sum_{(x,y) \in \mathbb{D}} L_{KD}(S(x, \theta_S, \tau), T(x, \theta_T, \tau)) + \lambda L_{CE}(\hat{y}_S, y)$$

Hinton et al. Distilling the Knowledge in a Neural Network, 2015

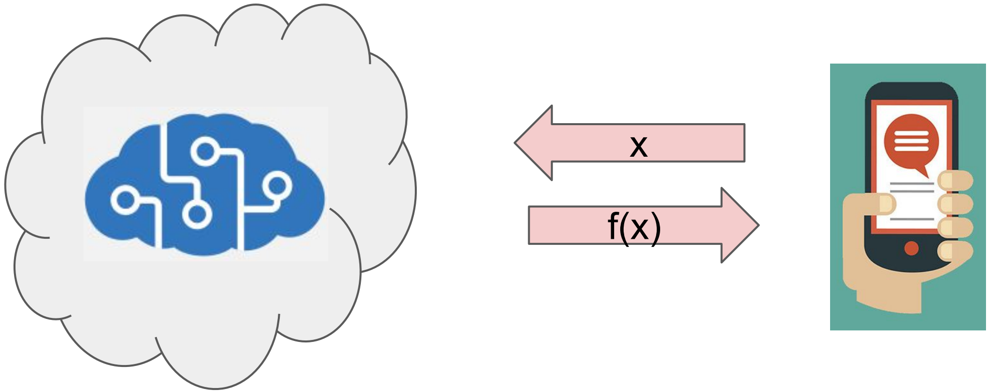
Advanced
Data-free Knowledge Extraction

Deployment

1. Handing Over the model physically

Deployment

1. Handing Over the model physically
2. Allowing access over the cloud (MLaaS)



1. Handing over the model physically

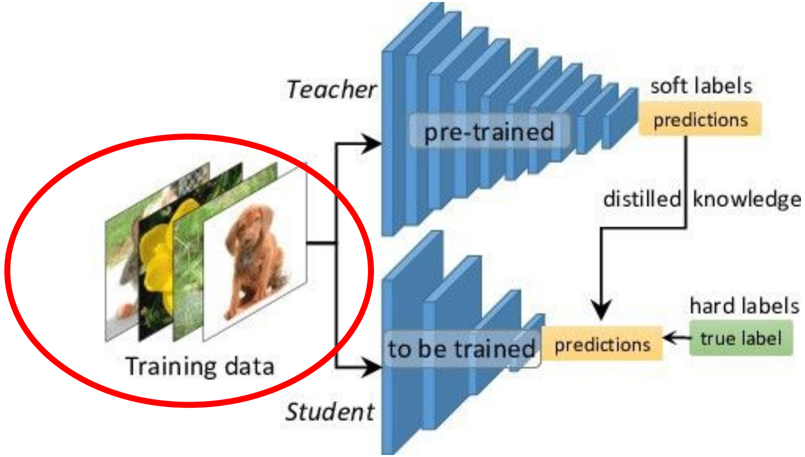
Models in the absence of training data

- Can
  - Inference (deploying)
  - Better initialization (pre-training)
- Can't
  - Compression & Distillation
  - Fine-tuning & Continual learning
  - Adapting, etc.

Absence of training data (?!)

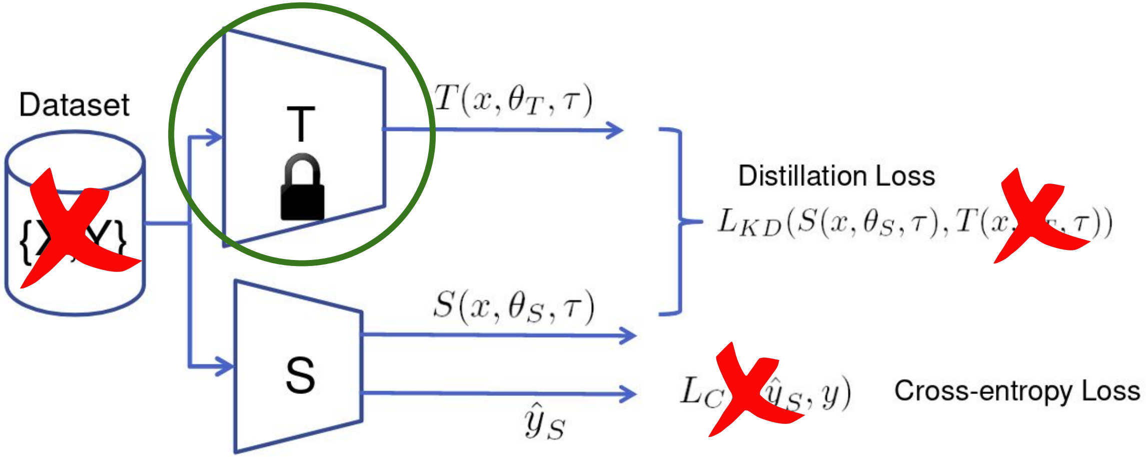
- We may have the trained models but not the training data
  - Privacy → e.g. Patients' data, biometric data, etc.
  - Data is property → Proprietary data
  - Transience → observations of an RL training environment
  - Scale

Requirement



Requires
Training Data on which
T is trained

KD in the absence of training data

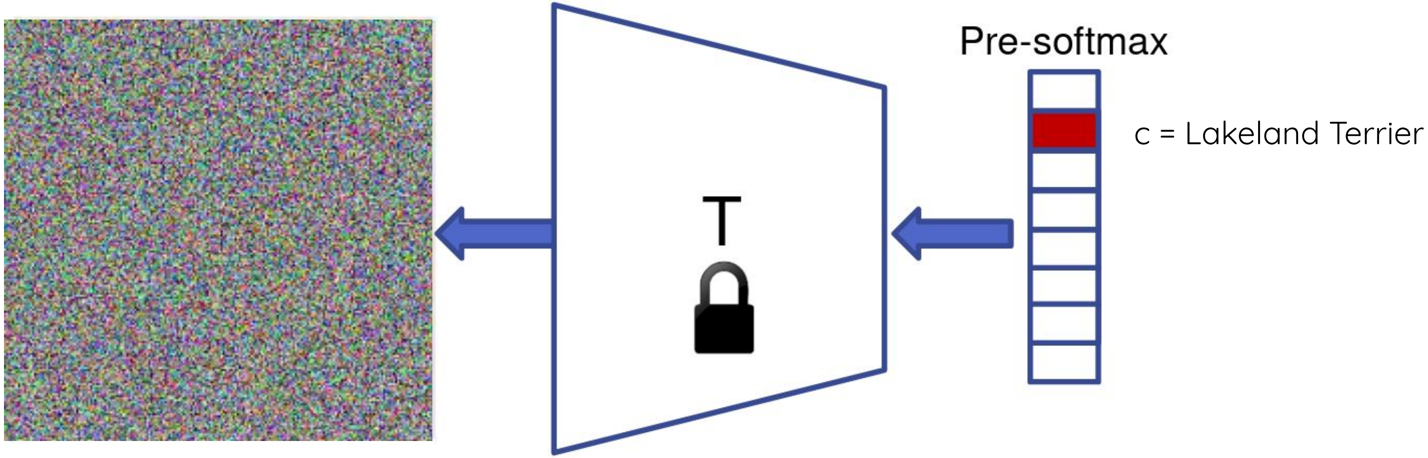


Can samples be synthesized from the trained Teacher model ?

Mining Data-Impressions from Deep Models as Substitute for Unavailable Training Data

Konda Reddy Mopuri et al.
ICML 2019 & Trans. on PAMI 2021

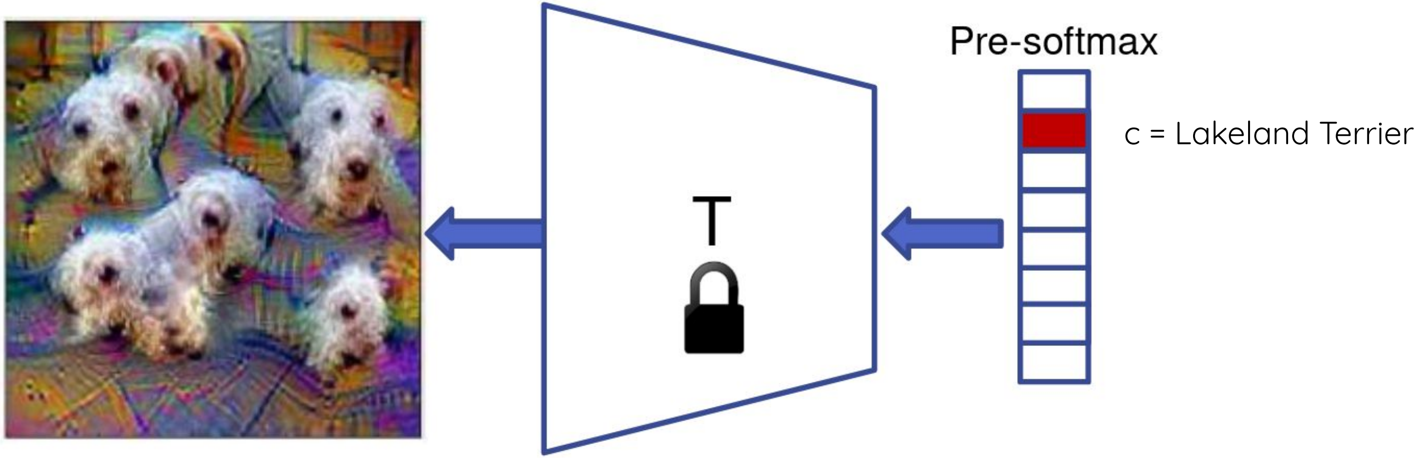
Class Impressions: Parameters → patterns



$$CI_c = argmax_x \;\; T_c(x)$$

K R Mopuri et al., Ask, Acquire and Attack: Data-free UAP generation using Class impressions, ECCV 2018

Class Impressions: Parameters → patterns



$$CI_c = argmax_x \ \ T_c(x)$$

K R Mopuri et al., Ask, Acquire and Attack: Data-free UAP generation using Class impressions, ECCV 2018

Class Impressions: Parameters → patterns

Goldfish Cock Wolf spider Lakeland terrier Monarch

K R Mopuri et al., Ask, Acquire and Attack: Data-free UAP generation using Class impressions, ECCV 2018

Training on CIs: Limitations

- Generated samples are less faithful and diverse
- One-hot vector labels are reconstructed
  - → minimal latent/dark knowledge → not so close to the natural data
- Student suffers poor generalization

Need an Improved modelling of the output space

Dirichlet modelling of output space

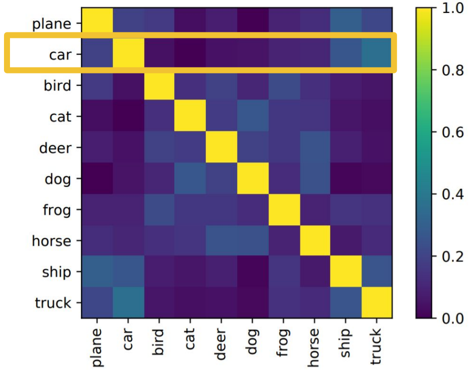
- Softmax space of each class ‘k’ $y^k \sim Dir(K, \alpha^k)$
- Support is the probabilities of a **K**-way classification
- Concentration param (α) → spread of the distribution

Dirichlet modelling of output space

- Softmax space of each class ‘k’ $y^k \sim Dir(K, \alpha^k)$
- Support is the probabilities of a **K**-way classification
- Concentration param (α) → spread of the distribution

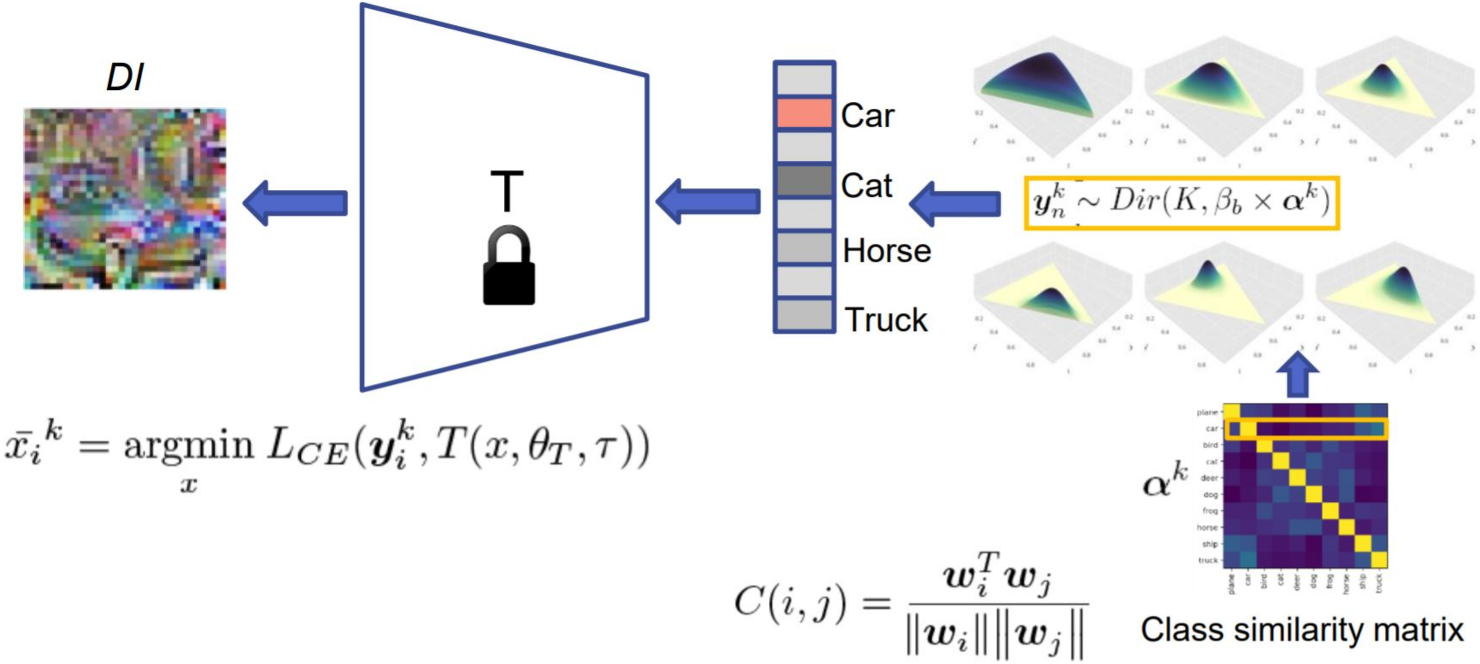
$$C(i,j) = \frac{\boldsymbol{w}_i^T \boldsymbol{w}_j}{\|\boldsymbol{w}_i\| \|\boldsymbol{w}_j\|}$$

W_k - weights learned by the Teacher’s softmax classifier for class ‘k’



Class similarity matrix

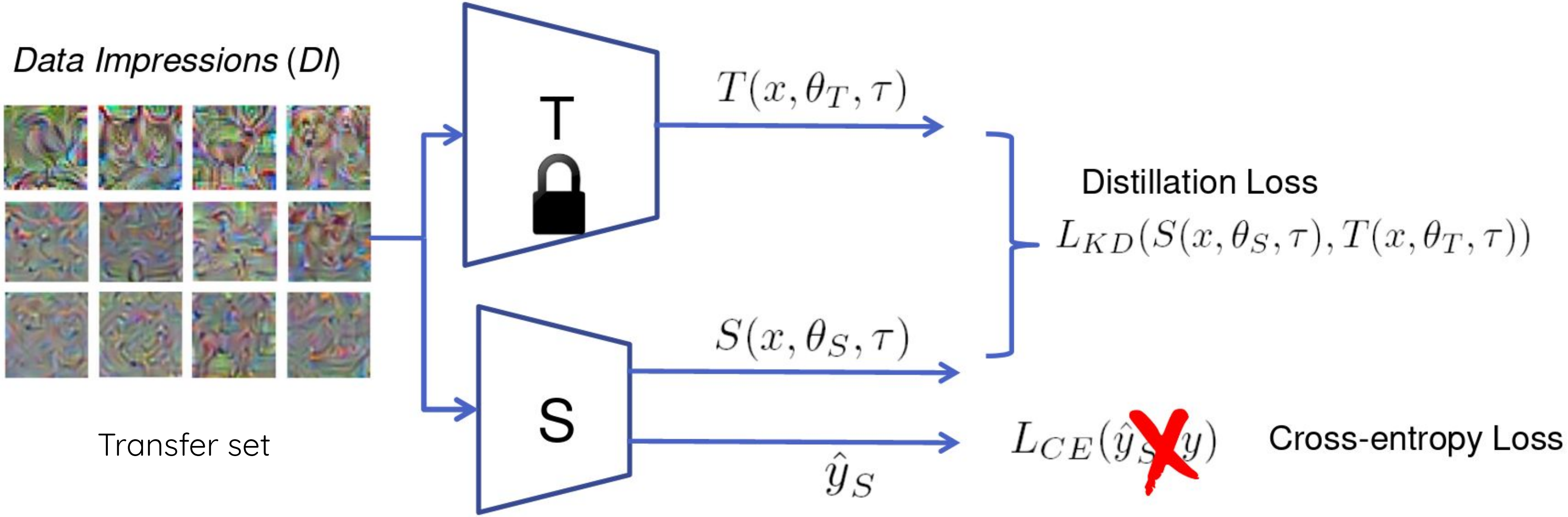
Data Impressions



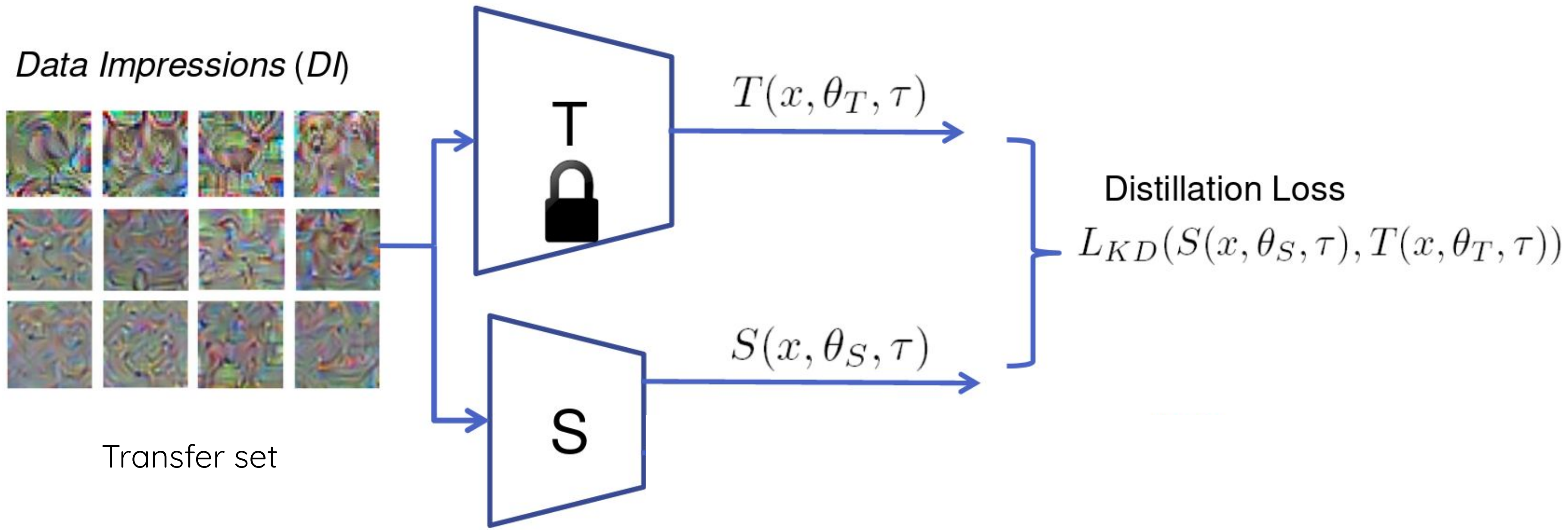
Data Impressions are generic

- Not tied to any downstream task
    - → applied in variety of tasks

Distillation with DIs

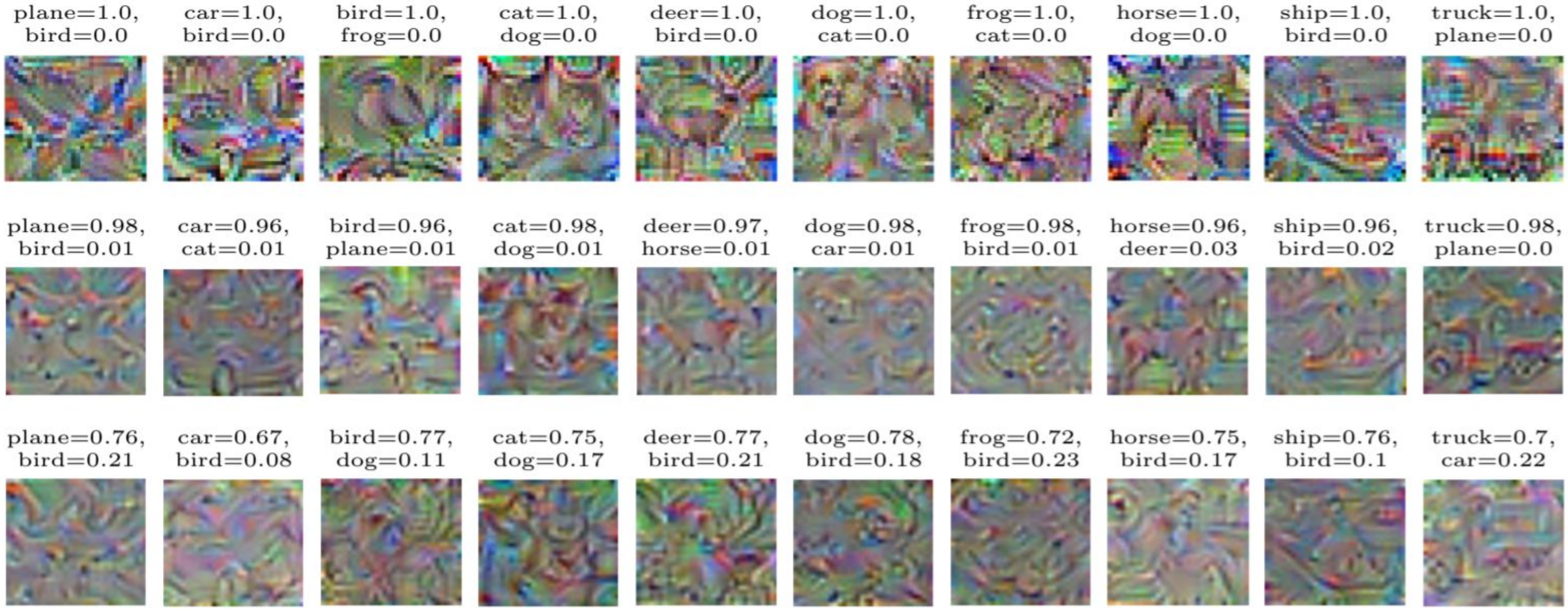


Distillation with DIs



$$\theta_S = \underset{\theta_S}{\operatorname{argmin}} \sum_{\bar{x} \in \bar{X}} L_{KD}(T(\bar{x}, \theta_T, \tau), S(\bar{x}, \theta_S, \tau))$$

Generated Samples



Performance

Model	Performance
Teacher – CE	99.34
Student – CE	98.92
Student–KD (Hinton et al., 2015) 60K original data	99.25
(Kimura et al., 2018) 200 original data	86.70
(Lopes et al., 2017) (uses meta data)	92.47
ZSKD (Ours) (24000 *DI*s, and no original data)	98.77

MNIST

Model	Performance
Teacher – CE	83.03
Student – CE	80.04
Student – KD (Hinton et al., 2015) 50K original data	80.08
ZSKD (Ours) (40000 *DI*s, and no original data)	69.56

CIFAR-10

Performance

Model	Data-free	Performance (%)
VGG-19 (T)	✗	87.99
VGG-11 (S)- CE	✗	84.19
VGG-11 (S)- KD [9]	✗	84.93
VGG-11 (S)- KD (**Ours**)	✓	74.10
Resnet-18 (S) -CE	✗	84.45
Resnet-18 (S) -KD [9]	✗	86.58
Resnet-18 (S) -KD (**Ours**)	✓	74.76

Model	Data-free	Performance (%)
Resnet-18 (T)	✗	86.54
Resnet-18-half (S)- CE	✗	85.51
Resnet-18-half (S)- KD [9]	✗	86.31
Resnet-18-half (S)- KD (Ours)	✓	81.10

CIFAR-10

Multiple attempts followed

- Data-free knowledge distillation
- GAN-inspired algorithms: ZSKT, DAFL, DeGAN, etc. (can be found in references)

Adversarial Belief Matching (ZSKT)

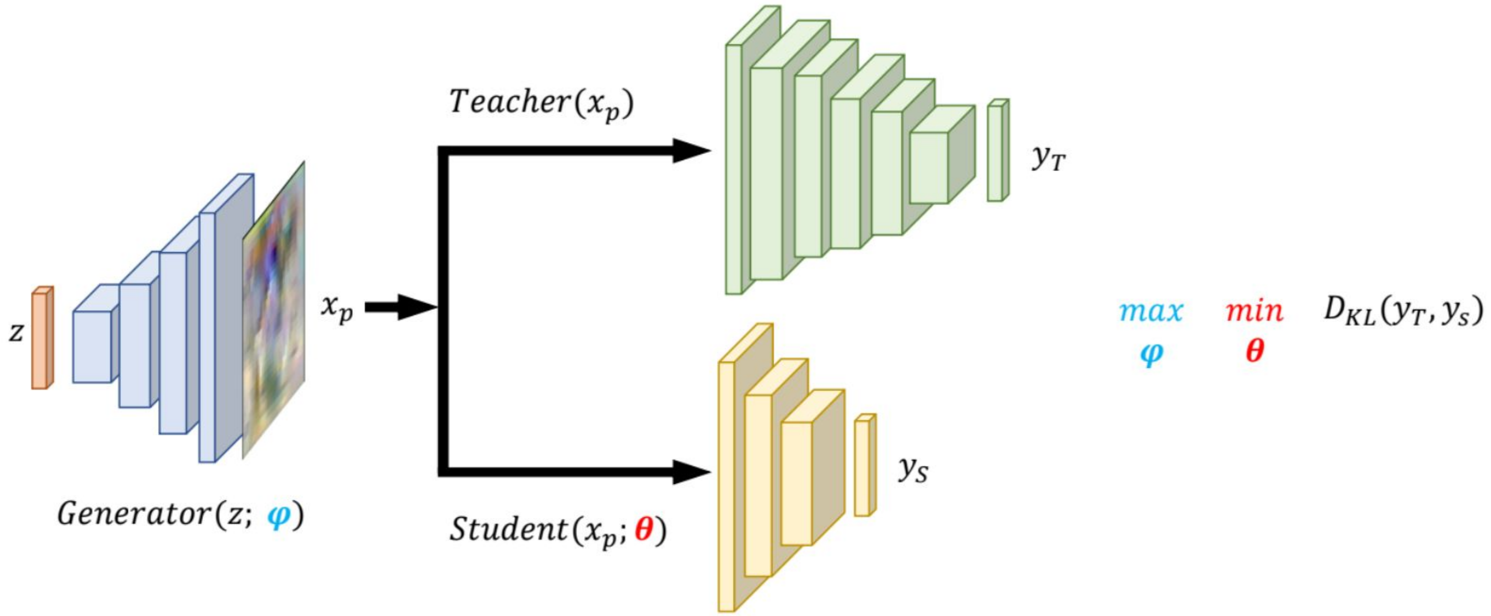


Figure from Micaelli et al. NeurIPS 2019

Adversarial Belief Matching (ZSKT)

- **G** searches for the samples on which the **T** and **S** disagree
- Then **S** learns to match **T** on them
- Adversarial framework makes G to keep exploring the input space

Figure from Micaelli et al. NeurIPS 2019

Generated Images (ZSKT)

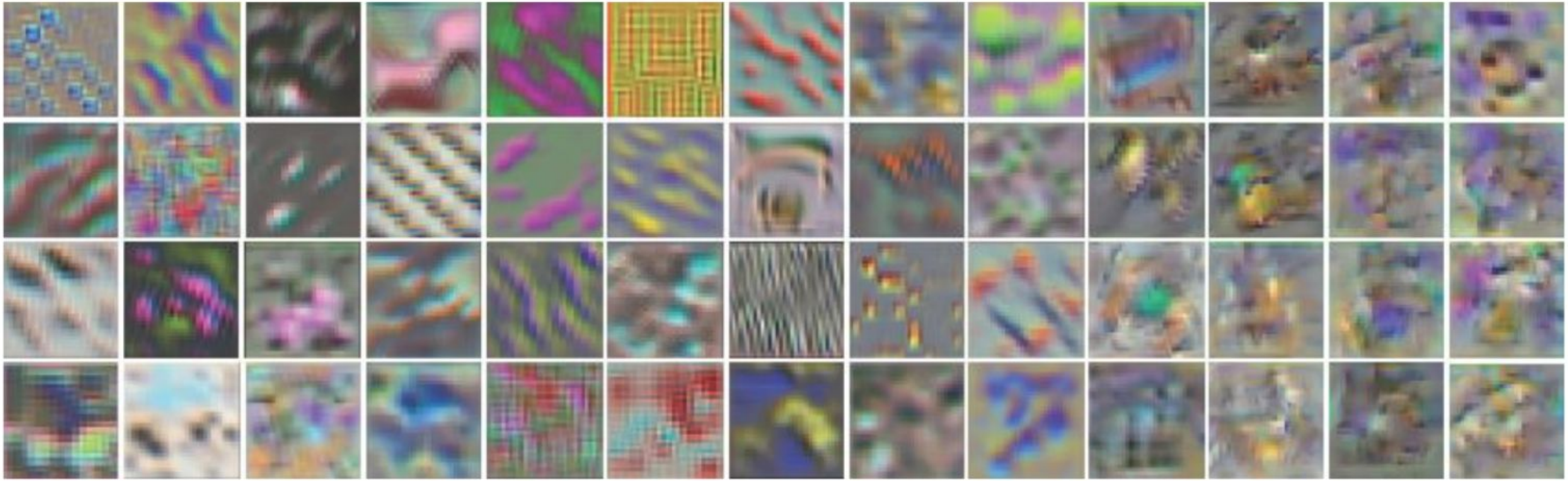
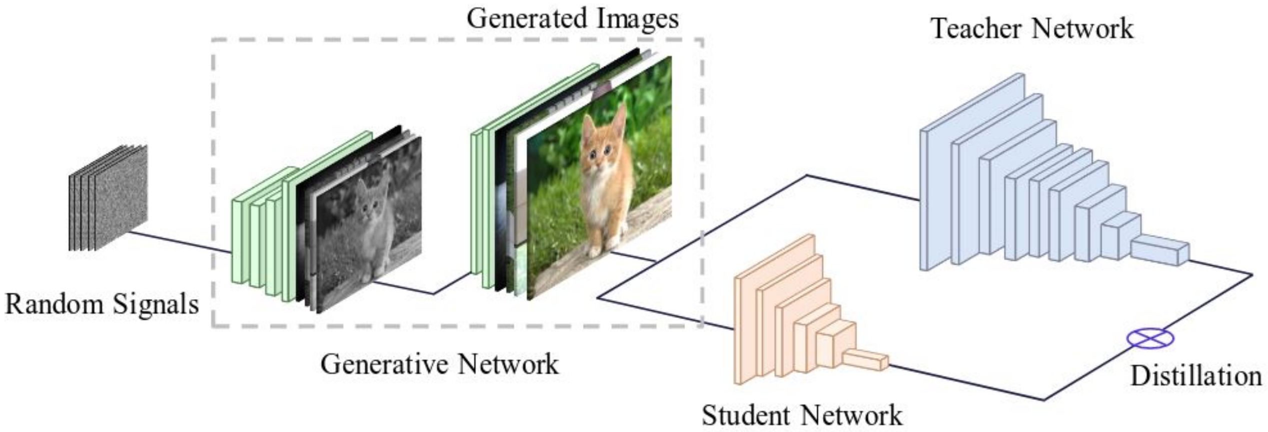


Figure from Micaelli et al. NeurIPS 2019

DAFL: GAN based generation [ICCV 2019]



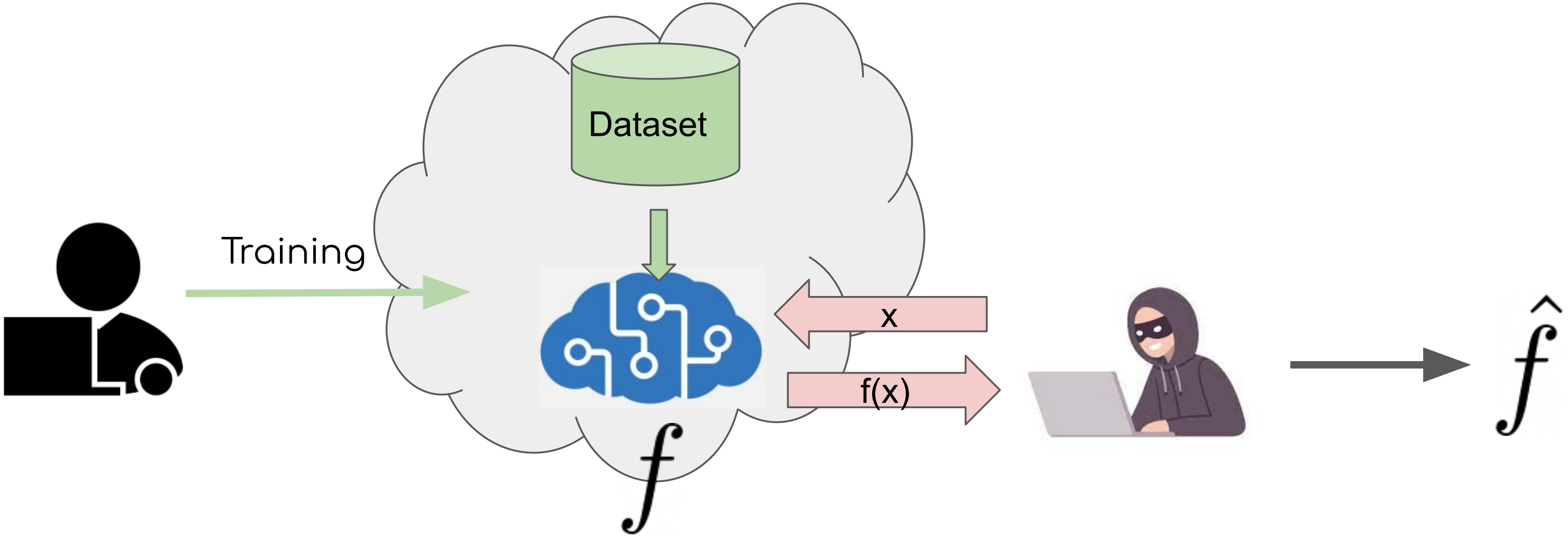
$$\mathcal{L}_{oh} = \frac{1}{n}\sum_i \mathcal{H}_{cross}(\mathrm{y}_T^i, \mathrm{t}^i) \qquad \mathcal{L}_a = -\frac{1}{n}\sum_i \|f_T^i\|_1 \qquad \mathcal{L}_{ie} = -\mathcal{H}_{info}(\frac{1}{n}\sum_i \mathrm{y}_T^i)$$

Attributes of full-access (white-box) setting

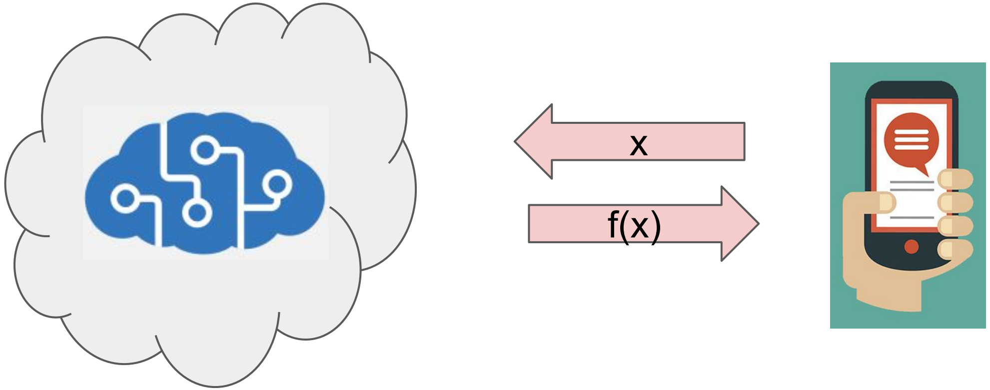
- Assumes access to the model
    - Model architecture/parameters
    - Softmax predictions
    - Gradients

2. Accessing over Cloud (MLaaS)

Model Extraction in MLaaS

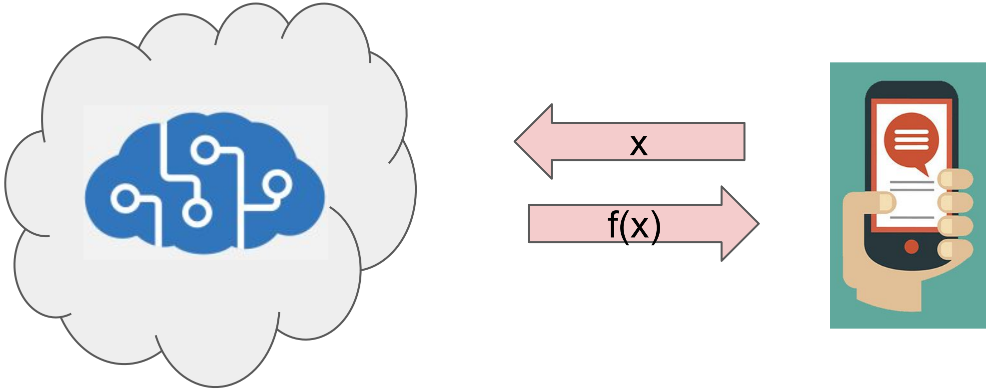


Model Extraction in MLaaS



Model Extraction in MLaaS

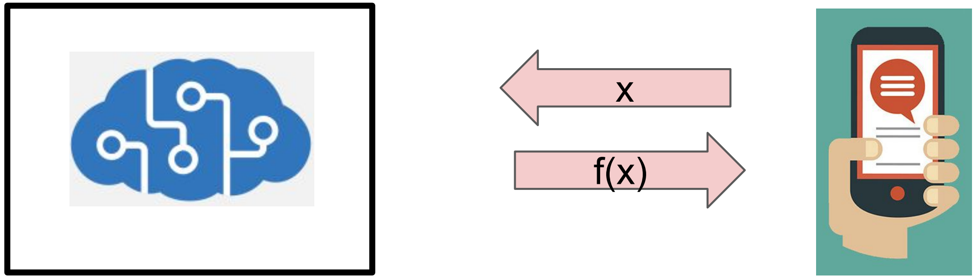
- No access to the model architecture/parameters
- One can generate (x, f(x)) pairs by querying the service



Model Extraction in MLaaS

- No access to the model architecture/parameters
- One can generate (x, f(x)) pairs by querying the service

Black-box setting



Model Extraction in black-box setting

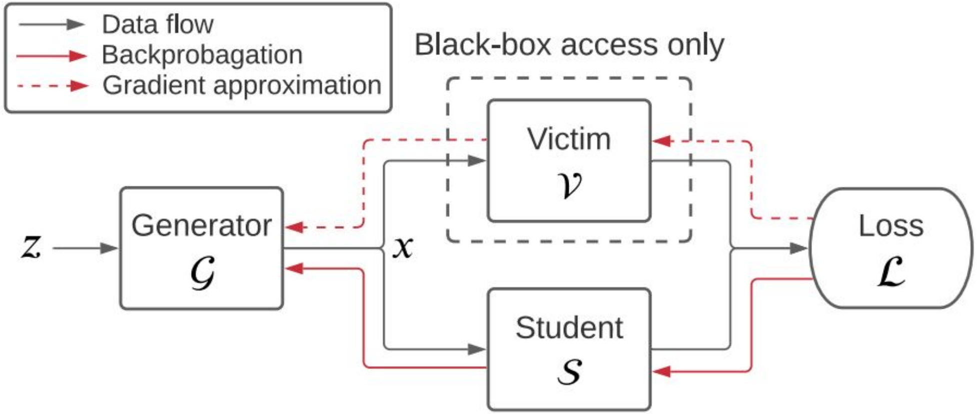
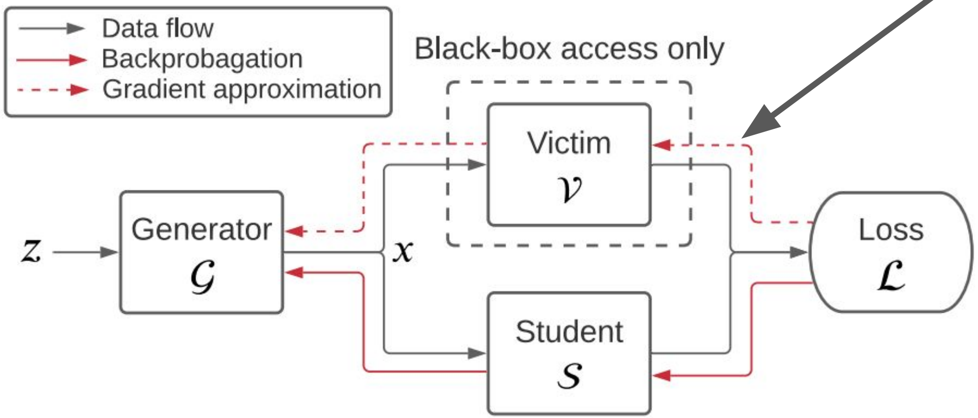


Figure from DFME, CVPR 2021

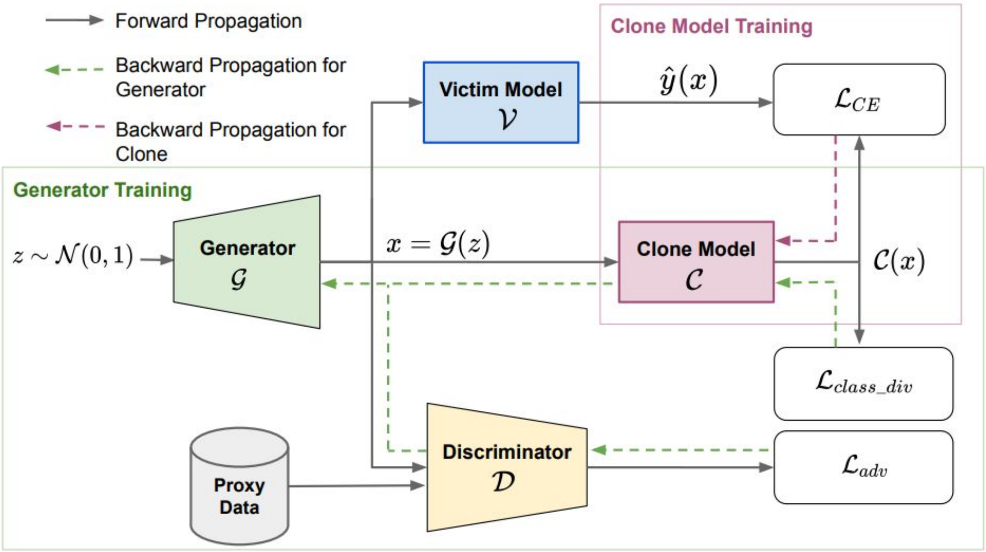
Model Extraction in black-box setting



Gradient needs to be estimated
e.g. train a surrogate

Figure from DFME, CVPR 2021

Model Extraction in black-box (or, hard label) setting

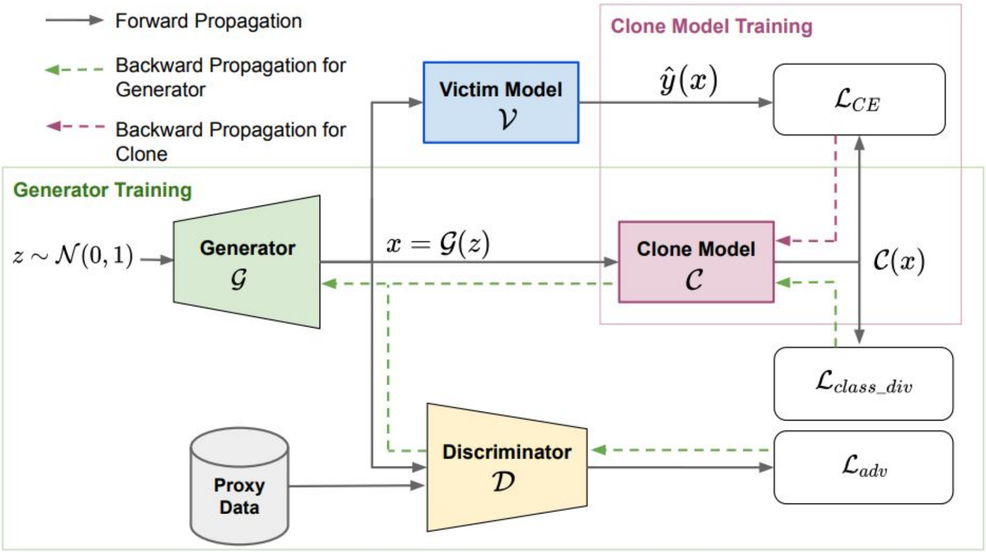


$$\mathcal{L}_{adv,real} = \mathop{\mathbb{E}}_{x \sim p_{data}(x)} [log \mathcal{D}(x)]$$

$$\mathcal{L}_{adv,fake} = \mathop{\mathbb{E}}_{z \sim \mathcal{N}(0,I)} [log(1 - \mathcal{D}(\mathcal{G}(z))]$$

Towards DFME in hard label setting, Sanyal et al. CVPR 2022

Model Extraction in black-box (or, hard label) setting



$$\mathcal{L}_{class\_div} = \sum_{j=0}^{K} \alpha_j \log \alpha_j$$

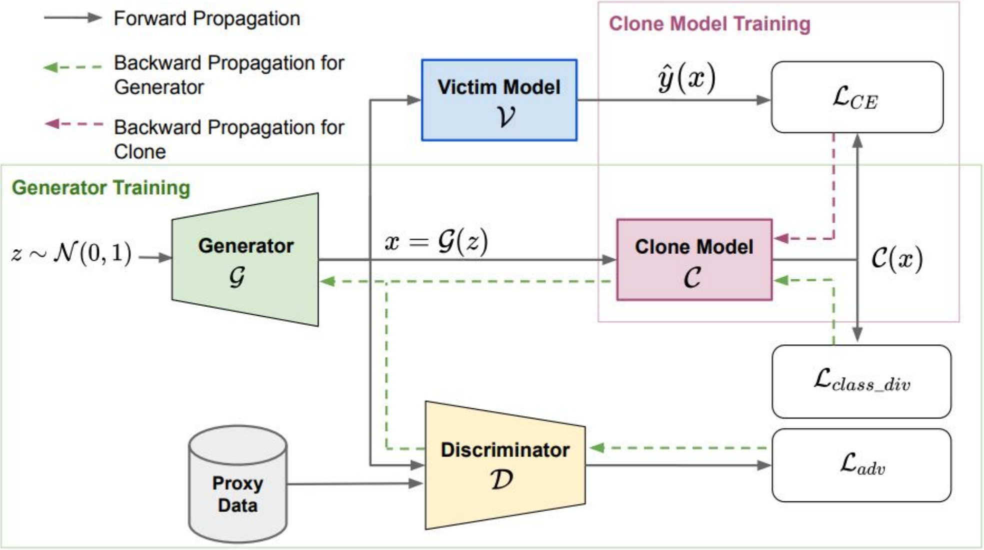
$$\alpha_j = \frac{1}{N} \sum_{i=1}^{N} \text{softmax}(\mathcal{C}(x_i))_j$$

$$\mathcal{L}_{adv,real} = \mathop{\mathbb{E}}_{x \sim p_{data}(x)} [log\mathcal{D}(x)]$$

$$\mathcal{L}_{adv,fake} = \mathop{\mathbb{E}}_{z \sim \mathcal{N}(0,I)} [log(1 - \mathcal{D}(\mathcal{G}(z))]$$

Towards DFME in hard label setting, Sanyal et al. CVPR 2022

Model Extraction in black-box (or, hard label) setting



$$\mathcal{L}_C = \mathop{\mathbb{E}}_{z \sim \mathcal{N}(0,I)} \left[\mathcal{L}_{CE}(\mathcal{C}(x), \hat{y}(x))\right], \; x = \mathcal{G}(z)$$

$$\mathcal{L}_{class\_div} = \sum_{j=0}^{K} \alpha_j \log \alpha_j$$

$$\alpha_j = \frac{1}{N} \sum_{i=1}^{N} \mathrm{softmax}(\mathcal{C}(x_i))_j$$

$$\mathcal{L}_{adv,real} = \mathop{\mathbb{E}}_{x \sim p_{data}(x)} [log\mathcal{D}(x)]$$

$$\mathcal{L}_{adv,fake} = \mathop{\mathbb{E}}_{z \sim \mathcal{N}(0,I)} [log(1 - \mathcal{D}(\mathcal{G}(z))]$$

Towards DFME in hard label setting, Sanyal et al. CVPR 2022

Experiments

Method	Hard Label	Black-Box	Data-Free	Victim Accuracy	Synthetic/ Data-Free	CIFAR-100 (40C)	CIFAR-100 (10C)
Victim Accuracy $\sim$ 95.5%, Victim Model: ResNet-34							
MAZE [17]	×	✓	✓	95.50	45.60	-	-
DFME [35]	×	✓	✓	95.50	88.10	-	-
DFMS-HL (Ours)	✓	✓	✓	95.59	84.51	**92.06**	**85.53**
DFMS-SL (Ours)	×	✓	✓	95.59	**91.24**	**93.96**	**90.88**
Victim Accuracy $\sim$ 93.7%, Victim Model: ResNet-18							
ZSDB3KD [38]	✓	✓	✓	93.65	50.18	-	-
DFMS-HL (Ours)	✓	✓	✓	93.83	**85.92**	**90.51**	**83.37**

CIFAR-10
ResNet-34
Resnet-18

CIFAR-100
ResNet-18
Resnet-18

Method	Proxy Data	Victim Accuracy	Clone Accuracy
DeGAN [1]	CIFAR-10	78.52	75.62
DFMS-HL (Ours)	CIFAR-10	78.52	72.83
DFMS-HL (Ours)	Synthetic	78.52	43.56

Conclusion

- Security aspects of ML needs equal attention
- From extracting the learning to extracting the training data?

References: Gradient estimation in black-box setting

- Yining Wang, Simon Du, Sivaraman Balakrishnan, and Aarti Singh. Stochastic zeroth-order optimization in high dimensions. In International Conference on Artificial Intelligence and Statistics, pages 1356–1365, 2018.
- Yurii Nesterov and Vladimir Spokoiny. Random gradient free minimization of convex functions. Foundations of Computational Mathematics, 17(2):527–566, 2017.
- Sijia Liu, Pin-Yu Chen, Bhavya Kailkhura, Gaoyuan Zhang, Alfred Hero, and Pramod K. Varshney. A primer on zeroth order optimization in signal processing and machine learning, 2020.
- Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based blackbox attacks to deep neural networks without training substitute models. In Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, pages 15–26, 2017

References: Zero-shot knowledge distillation

- Gaurav Kumar Nayak, Konda Reddy Mopuri, Vaisakh Shaj, Venkatesh Babu Radhakrishnan, and Anirban Chakraborty. Zero-shot knowledge distillation in deep networks, ICML, 2019.
- Sravanti Addepalli, Gaurav Kumar Nayak, Anirban Chakraborty, and Venkatesh Babu Radhakrishnan. DeGAN: Data-enriching gan for retrieving representative samples from a trained classifier, AAAI, 2020.
- Gaurav Kumar Nayak, Konda Reddy Mopuri, Saksham Jain, Anirban Chakraborty, Mining Data Impressions from Deep Models as Substitute for the Unavailable Training Data, in *IEEE Trans. on PAMI* , 2021.
- Zero-shot knowledge transfer via adversarial belief matching, NeurIPS 2019.
- Hanting Chen, Yunhe Wang, Chang Xu, Zhaohui Yang, Chuanjian Liu, Boxin Shi, Chunjing Xu, Chao Xu, and Qi Tian. Data-free learning of student networks, ICCV 2019.

References: DFME in black-box setting

- Sunandini Sanyal, Sravanti Addepalli, R. Venkatesh Babu. Data-free Model Extraction in hard label setting, CVPR, 2022.
- Jean-Baptiste Truong, Pratyush Maini, Robert J Walls, and Nicolas Papernot. Data-free model extraction. CVPR, 2021.
- 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.
- Zi Wang. Zero-shot knowledge distillation from a decision based black-box model. arXiv preprint arXiv:2106.03310, 2021.

Thank You