Improving Accuracy-Privacy Tradeoff via Model Reprogramming

Pin-Yu Chen

www.pinyuchen.com @pinyuchenTW

TrustML Workshop@UBC

June 2023

IBM **Research**

Outline

- What is Model Reprogramming?
- How to use Model Reprogramming for Improving Task Performance under Differential Privacy Constraints?
- Why Model Reprogramming Works? [Time Permits]

What is Model Reprogramming?

The era of Foundation Model

Also check out our NeurIPS 2022 Tutorial on
“Foundational Robustness of Foundation Models”

On the Opportunities and Risks of Foundation Models

Rishi Bommasani* Drew A. Hudson Ehsan Adeli Russ Altman Simran Arora
Sydney von Arx Michael S. Bernstein Jeannette Bohg Antoine Bosselut Emma Brunskill
Erik Brynjolfsson Shyamal Buch Dallas Card Rodrigo Castellon Niladri Chatterji
Annie Chen Kathleen Creel Jared Quincy Davis Dorottya Demszky Chris Donahue
Moussa Doumbouya Esin Durmus Stefano Ermon John Etchemendy Kawin Ethayarajh
Li Fei-Fei Chelsea Finn Trevor Gale Lauren Gillespie Karan Goel Noah Goodman
Shelby Grossman Neel Guha Tatsunori Hashimoto Peter Henderson John Hewitt
Daniel E. Ho Jenny Hong Kyle Hsu Jing Huang Thomas Icard Saahil Jain
Dan Jurafsky Pratyusha Kalluri Siddharth Karamcheti Geoff Keeling Fereshte Khani
Omar Khattab Pang Wei Koh Mark Krass Ranjay Krishna Rohith Kuditipudi
Ananya Kumar Faisal Ladhak Mina Lee Tony Lee Jure Leskovec Isabelle Levent
Xiang Lisa Li Xuechen Li Tengyu Ma Ali Malik Christopher D. Manning
Suvir Mirchandani Eric Mitchell Zanele Munyikwa Suraj Nair Avanika Narayan
Deepak Narayanan Ben Newman Allen Nie Juan Carlos Niebles Hamed Nilforoshan
Julian Nyarko Giray Ogut Laurel Orr Isabel Papadimitriou Joon Sung Park Chris Piech
Eva Portelance Christopher Potts Aditi Raghunathan Rob Reich Hongyu Ren
Frieda Rong Yusuf Roohani Camilo Ruiz Jack Ryan Christopher Ré Dorsa Sadigh
Shiori Sagawa Keshav Santhanam Andy Shih Krishnan Srinivasan Alex Tamkin
Rohan Taori Armin W. Thomas Florian Tramèr Rose E. Wang William Wang Bohan Wu
Jiajun Wu Yuhuai Wu Sang Michael Xie Michihiro Yasunaga Jiaxuan You Matei Zaharia
Michael Zhang Tianyi Zhang Xikun Zhang Yuhui Zhang Lucia Zheng Kaitlyn Zhou
Percy Liang*[1]

Center for Research on Foundation Models (CRFM)
Stanford Institute for Human-Centered Artificial Intelligence (HAI)
Stanford University

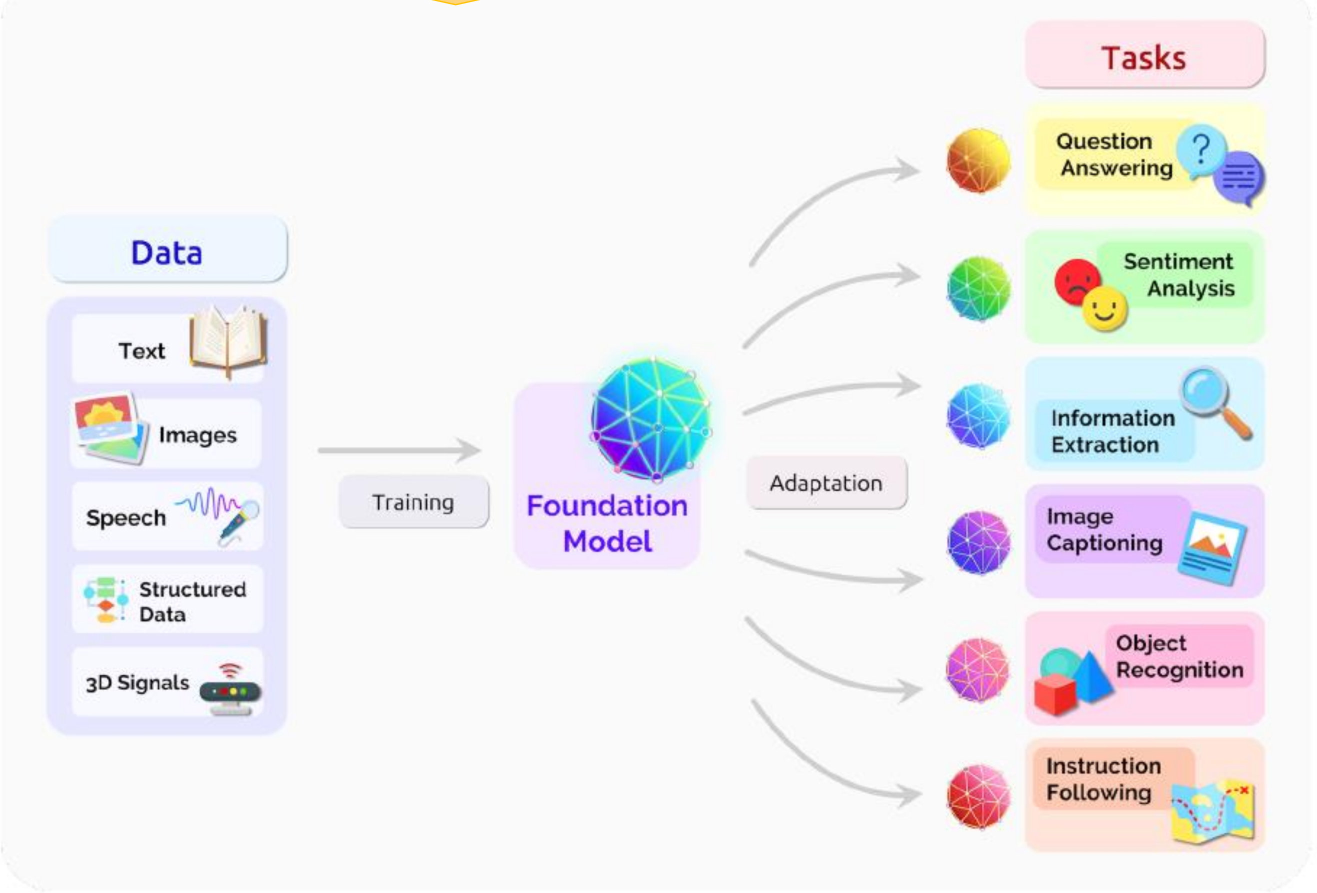
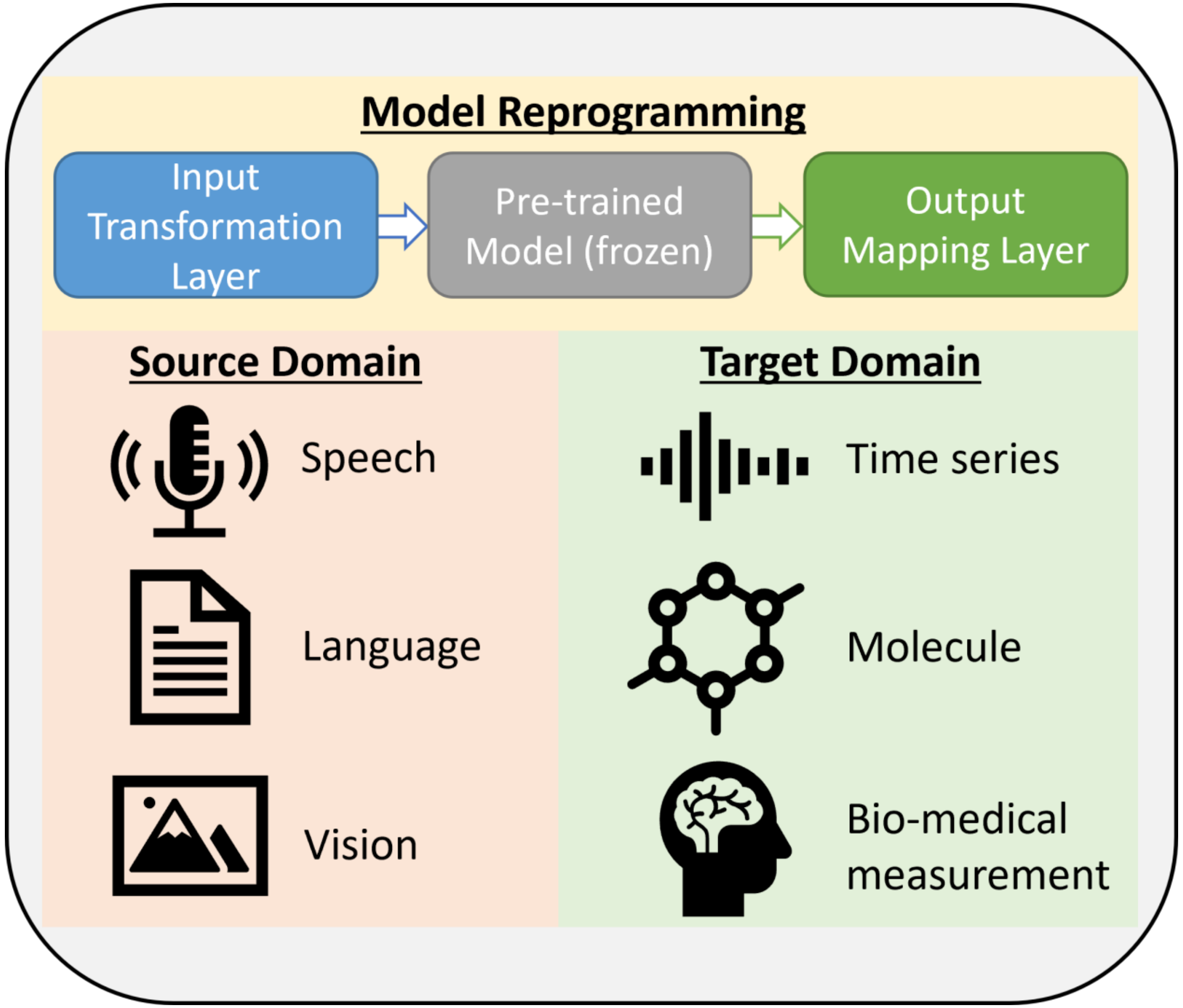
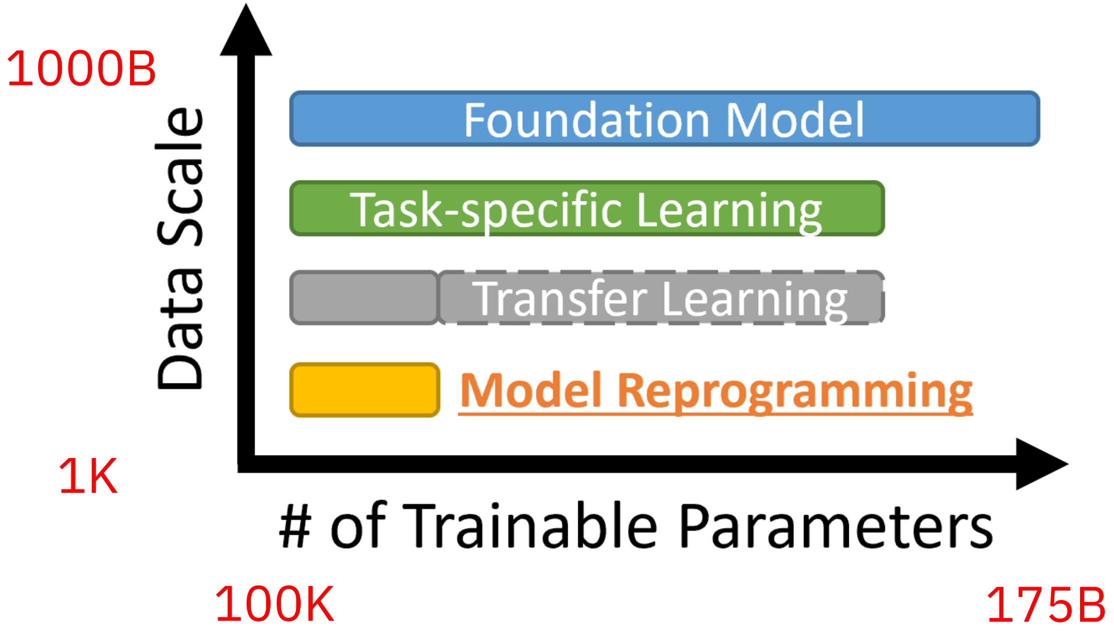


Fig. 2. A foundation model can centralize the information from all the data from various modalities. This one model can then be adapted to a wide range of downstream tasks.

Model Reprogramming Framework



Parameter Efficiency and Reusability!



Model Reprogramming: Resource-Efficient Cross-Domain Machine Learning

Pin-Yu Chen
IBM Research
pin-yu.chen@ibm.com

Unleashing the Power of Pre-trained Models

- Large pre-trained models are available in some data rich domains
    - text, image, speech, …
- Model reprogramming: leveraging pre-trained models in well-studied domains to solve tasks in **resource-limited domains**
    - Limited Data: medical imaging, molecular learning, time series …
    - Limited Model: no high-quality pretrained models in the target domains
    - Limited Resource: train from scratch is too costly
    - Training constraints: privacy budget, training time, etc
- New altervative: Resource-efficient transfer learning (or parameter-efficient fine-tuning) without model finetuning

"If I have seen further, it is by standing on the shoulders of Giants."

–Isaac Newton

Foundation Models: The one-for-all solution for AI

High-Capacity Models Pre-Trained on Large-Scale Datasets

Large-scale data for pre-training → High-capacity neural networks → Finetuning on downstream task and data

GPT-3 → ~ 1000 billon tokens → 175 billion training parameters → memory size >350GB; estimated cost $12 million to train

Bommasani et al. On the Opportunities and Risks of Foundation Models. Arxiv
https://venturebeat.com/2020/06/01/ai-machine-learning-openai-gpt-3-size-isnt-everything

How Much Does GPT-4 Cost?

WILL KNIGHT BUSINESS APR 17, 2023 7:00 AM

OpenAI's CEO Says the Age of Giant AI Models Is Already Over

Sam Altman says the research strategy that birthed ChatGPT is played out and future strides in artificial intelligence will require new ideas.

PHOTOGRAPH: JASON REDMOND/GETTY IMAGES

GPT-4, the latest of those projects, was likely trained using trillions of words of text and many thousands of powerful computer chips. The process cost over $100 million.

At the MIT event, Altman was asked if training GPT-4 cost $100 million; he replied, "It's more than that."

How to Use Foundation Models for Machine Learning in Resource-Limited Settings?

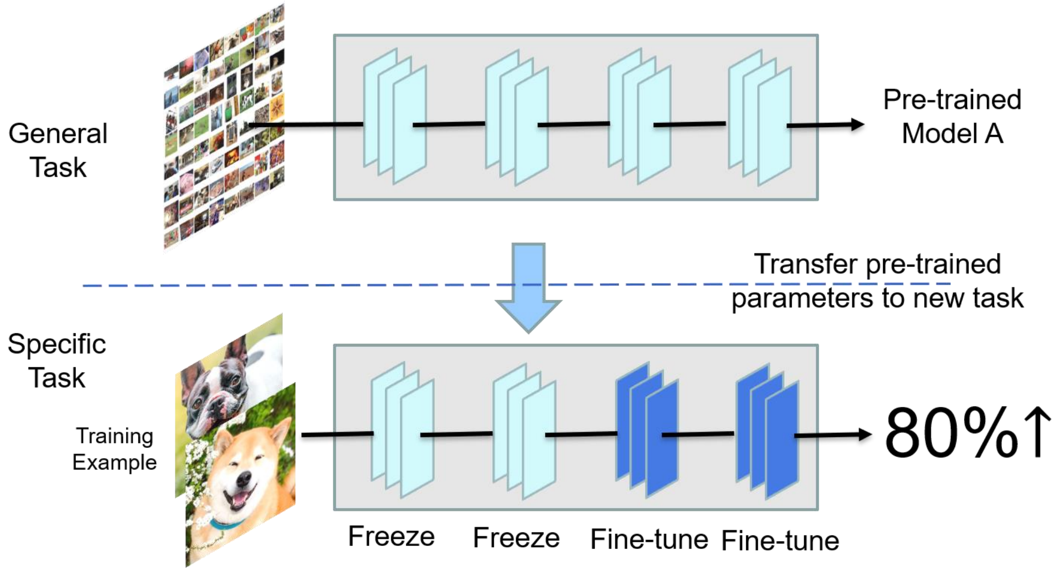
Standard Setting

- Pre-training + Fine-tuning
- Sufficient pre-training data and compute power
- Finetuning to in-domain downstream tasks

Resource-Limited Setting

- Reprogramming + No fine-tuning
- New domain with limited data / compute power
- No pre-trained models in the same domain

Standard Transfer Learning via Fine-Tuning



- Model Reprogramming
  - Cross-domain learning
    Reprogram a pre-trained model from domain A to solve resource-limited tasks in domain B
  - Data/Compute efficiency
    Does not require finetuning the pre-trained model weights
  - Achieve state-of-the-art performances

Background: “Adversarial” Reprogramming

Adversarial reprogramming works, but is not that impressive…

- Gamaleldin F. Elsayed, Ian Goodfellow, and Jascha Sohl-Dickstein, Adversarial Reprogramming of Neural Networks. ICLR 2019

- “We introduce attacks that instead reprogram the target model to perform a task chosen by the attacker—without the attacker needing to specify or compute the desired output for each test-time input.”

Model	Pretrained on ImageNet						Untrained
	Counting	MNIST		CIFAR-10		Shuffled MNIST	MNIST
		train	test	train	test	test	test
Incep. V3	0.9993	0.9781	0.9753	0.7311	0.6911	0.9709	0.4539
Incep. V4	0.9999	0.9638	0.9646	0.6948	0.6683	0.9715	0.1861
Incep. Res. V2	0.9994	0.9773	0.9744	0.6985	0.6719	0.9683	0.1135
Res. V2 152	0.9763	0.9478	0.9534	0.6410	0.6210	0.9691	0.1032
Res. V2 101	0.9843	0.9650	0.9664	0.6435	0.6301	0.9678	0.1756
Res. V2 50	0.9966	0.9506	0.9496	0.6	0.5858	0.9717	0.9325
Incep. V3 adv.		0.9761	0.9752				

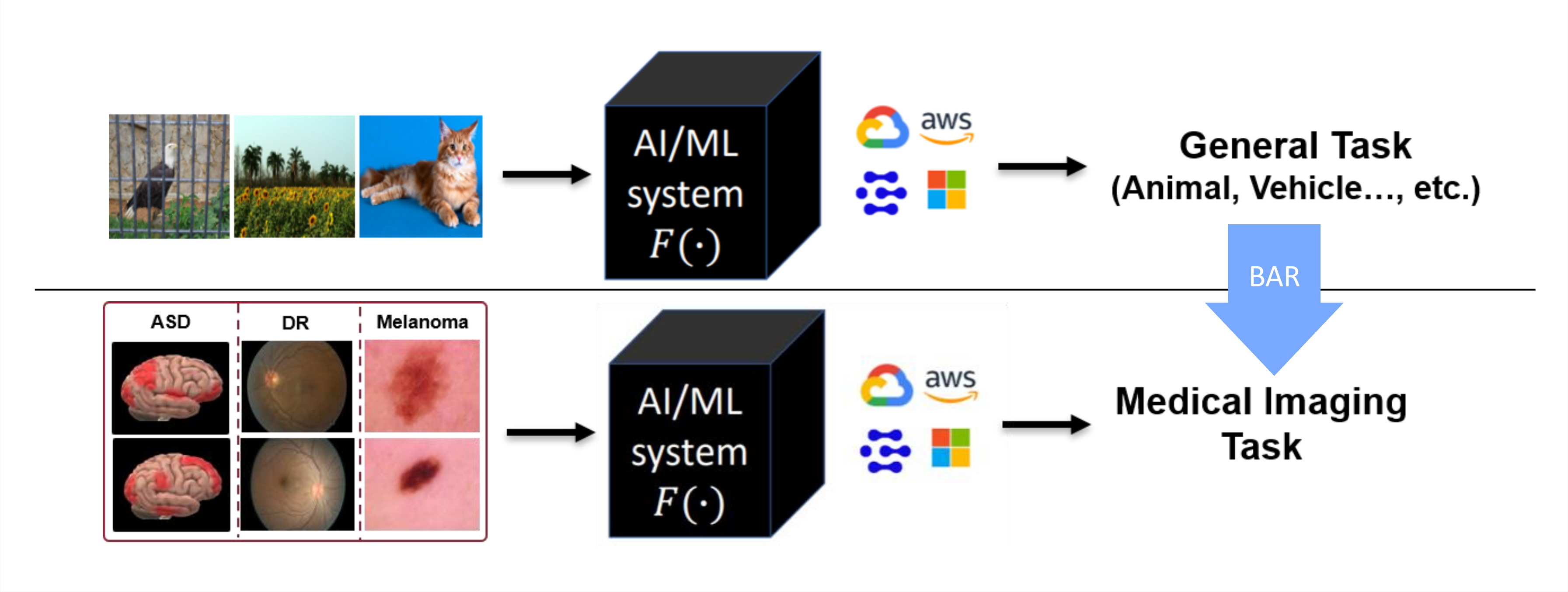
What can we do with Foundation models + Reprogramming?

BAR: Black-box Adversarial Reprogramming

https://arxiv.org/abs/2007.08714 (ICML 2020)

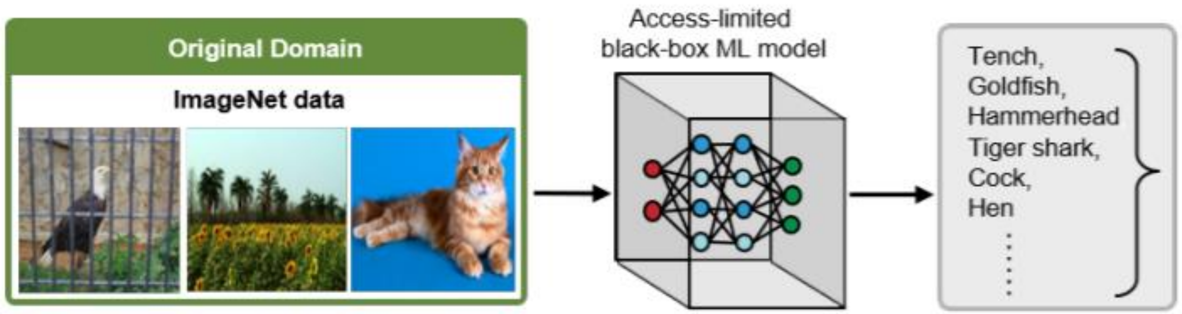
BAR: Transfer Learning without Knowing

- Reprogram powerful but black-box models for transfer learning (w/o fine-tuning) – extension to black-box APIs
- Appealing for cross-domain and data-limited transfer learning



How (Black-box) Reprogramming Works

Pre-trained model



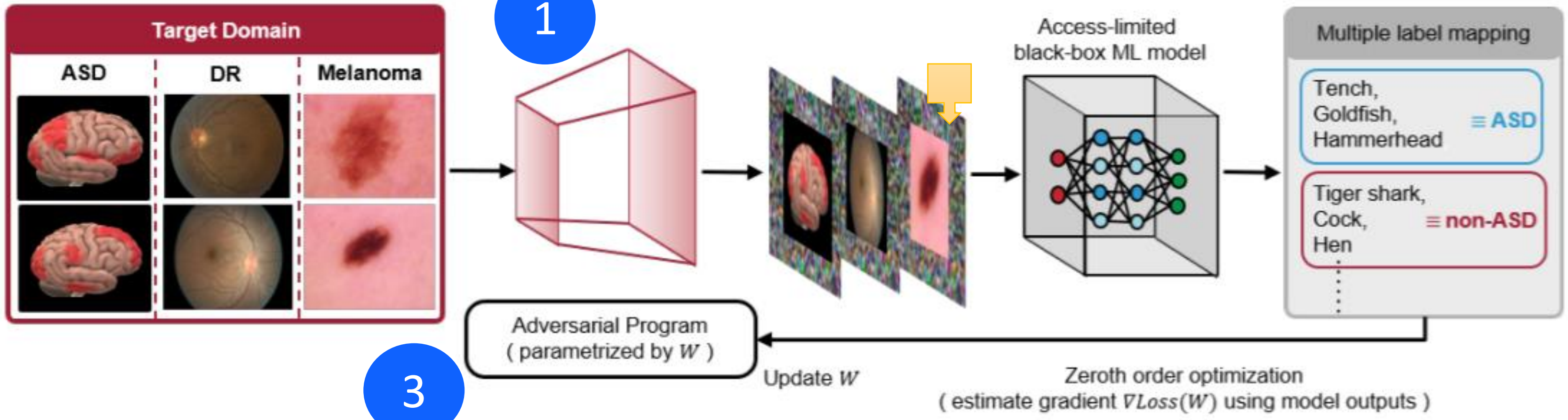
trainable

target data

1

Universal trainable perturbation

1



2

3

No fine-tuning on pre-trained models!

Yun-Yun Tsai, Pin-Yu Chen, Tsung-Yi Ho. Transfer Learning without Knowing: Reprogramming Black-box Machine Learning Models with Scarce Data and Limited Resources. ICML 2020

Problem Formulation

- Given a (black-box) pretrained model:

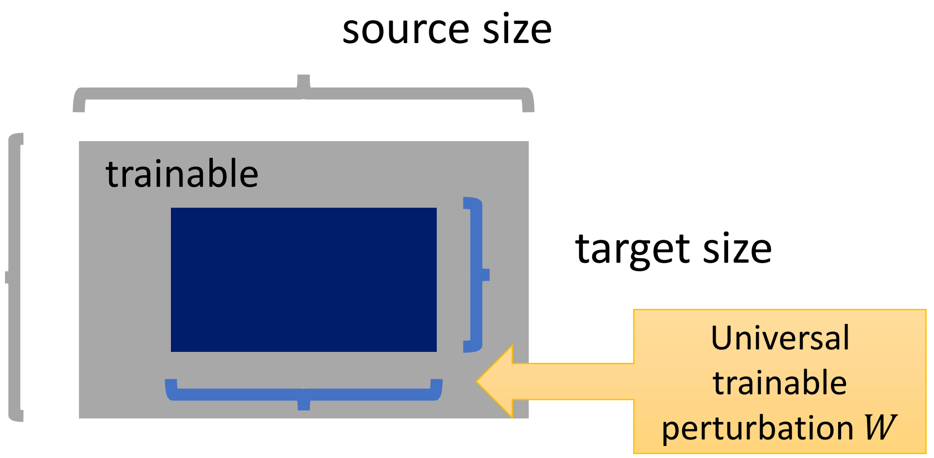
$$F\ :\mathcal{X}\rightarrow\mathbb{R}^K,$$

where $\mathcal{X}\in[-1,1]^d$ and $F(x)=[F_1(x),F_2(x),\ldots,F_K(x)]\in\mathbb{R}^K$

- Given the set of data from the target domain by:

$\{T_i\}_{i=1}^n$, where $T_i\in[-1,1]^{d'}$ and $d'<\ d$

- Output: Optimal input perturbation with trainable parameter (bias) W^*.



Gamaleldin F. Elsayed, Ian Goodfellow, Jascha Sohl-Dickstein. Adversarial Reprogramming of Neural Networks. ICLR 2019

Input Transformation Function

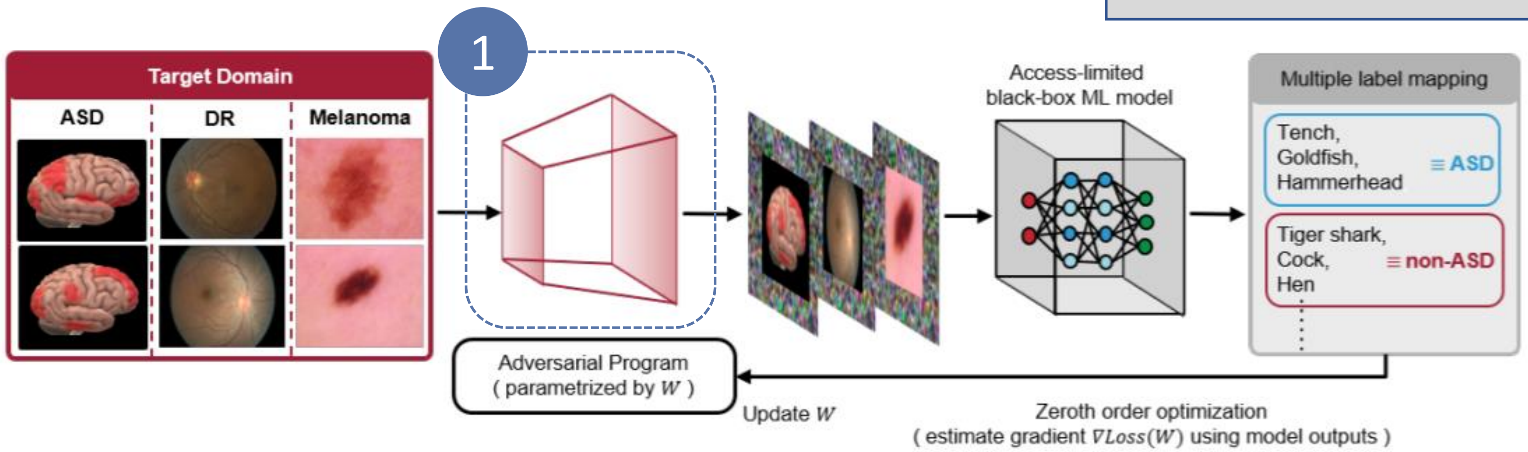
- The transformed data sample for model reprogramming is defined as:

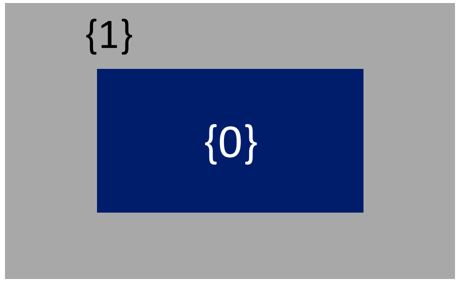
$$\widetilde{X}_i = \{T_i\}_{zero\ padding} + P, \text{ and } P = tanh(W \odot \color{red}{M})$$

Universal trainable perturbation (aka Trigger!)

Trainable parameters: $W \in \mathbb{R}^d$

Binary mask indicating where to add perturbation



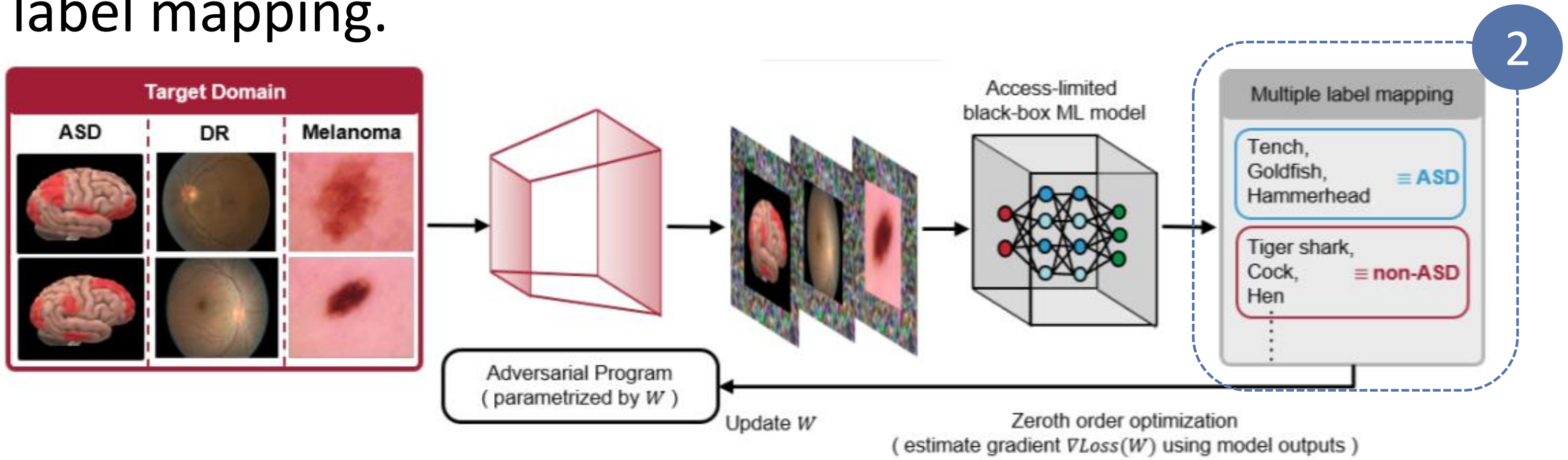


Multi-label Mapping (Random)

- $F(\cdot)$: pretrained source model
- We use the notation $h_j\,(\cdot)$ to denote $m\ to\ 1$ mapping function. For example,

$$h_{ASD}(F(X)) = \frac{F_{Tench}(X) + F_{Goldenfish}(X)\ +\ F_{Hammerhead}(X)}{3}$$

- We find that multiple-source-labels to one target-label mapping better than one-to-one label mapping.

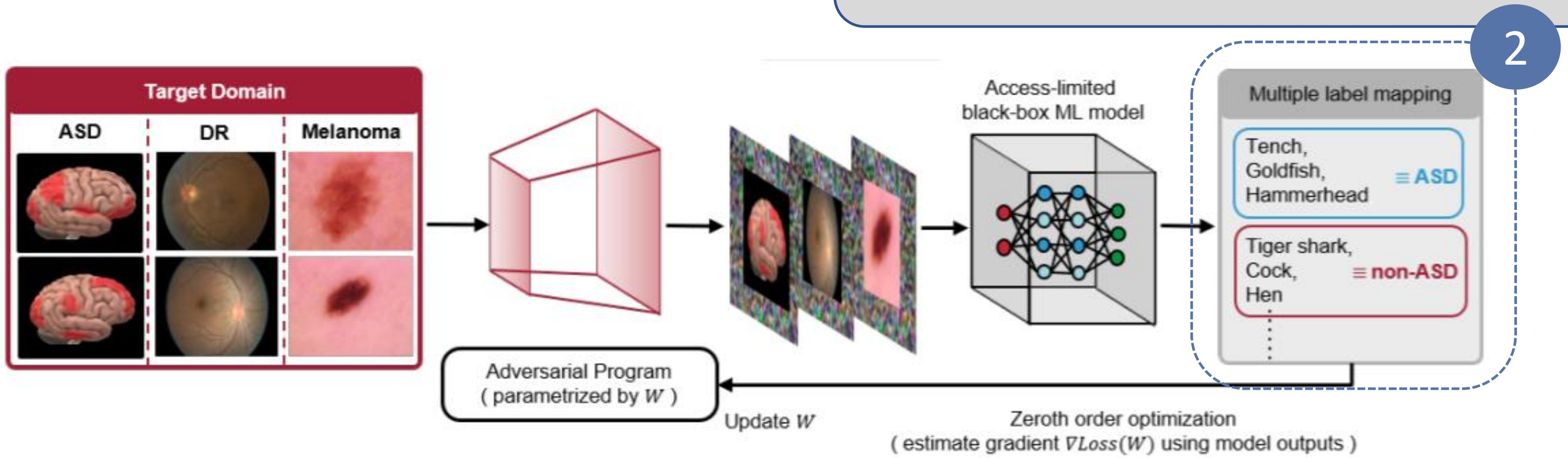


Multi-label Mapping (Frequency)

- We obtain the source-label prediction distribution of the target-domain data before reprogramming in each task.

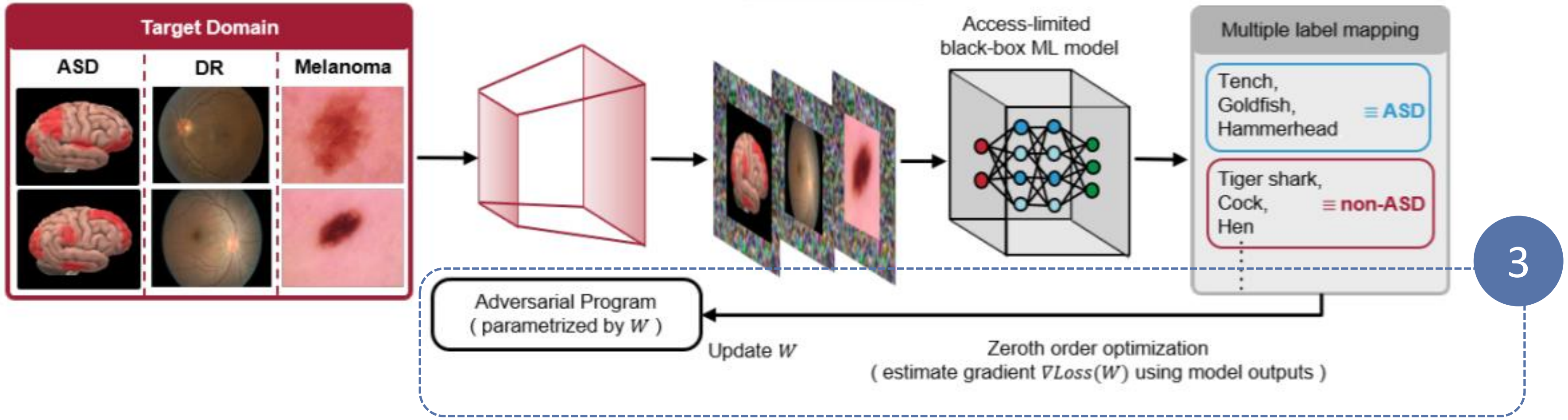
$$h_{ASD}(F(X)) = \frac{\boxed{F_{Tench}(X) + F_{Goldenfish}(X) + F_{Hammerhead}(X)}}{3}$$

We assign the most frequent m source-labels to one target label.



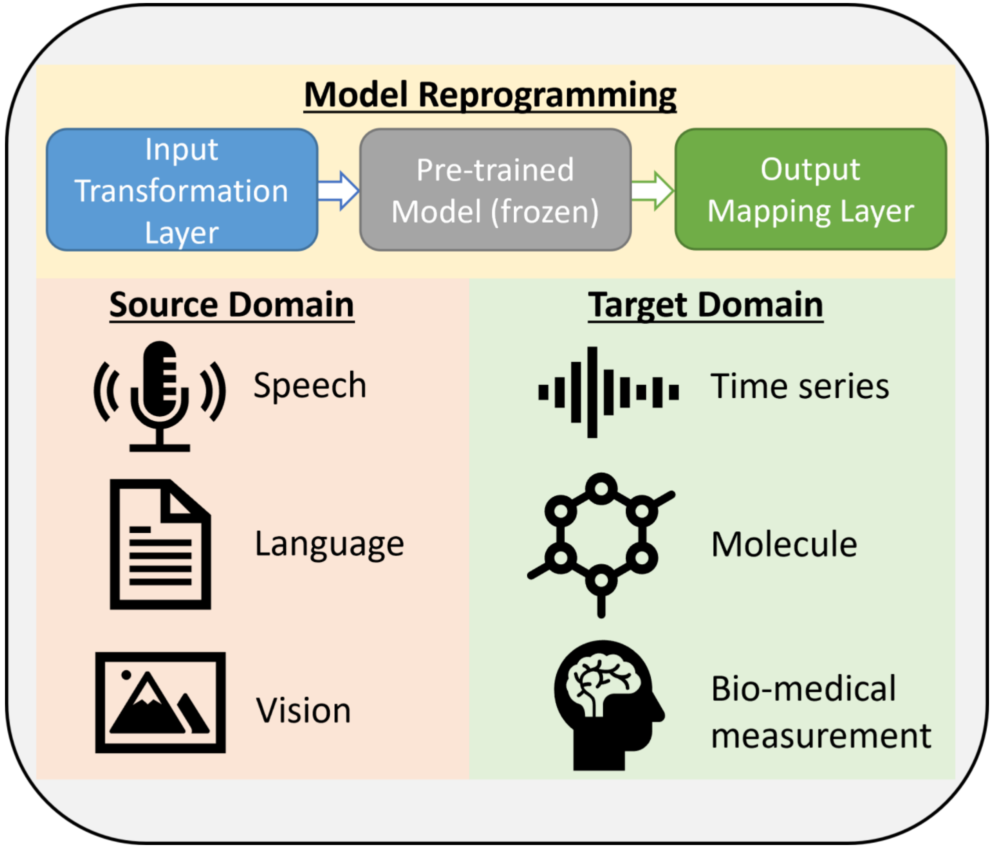
Training Loss Function

- We aim to maximize the probability of $p_t = P(h_j(y_{target})|X_{target})$
- We use focal loss empirically as it can further improve the performance of AR/BAR over cross entropy. $L_{focal}(p_t) = -(1-p_t)^{\gamma} log(p_t)$
- Optimize for the input transformation parameter W_t (t: iteration)
- ZO optimization for learning W^* in BAR : $W_{t+1} = W_t - \alpha_t \cdot \hat{\nabla} L(W_t)$



Liu, Chen, et al., "A Primer on Zeroth-Order Optimization in Signal Processing and Machine Learning", *IEEE Signal Processing Magazine*
Lin et al. Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision, pp. 2980–2988, 2017.

Generic Model Reprogramming Algorithm



1. Initialization: Load pre-trained source model $f_{\mathcal{S}}(\cdot)$ and target domain training set $\{x_{\mathcal{T}}^{(i)}, y_{\mathcal{T}}^{(i)}\}_{i=1}^{n}$; randomly initialize θ and ω
2. Input transformation: Obtain transformed input data $\widetilde{x}_{\mathcal{T}} = \mathsf{Input\text{-}Transform}(x_{\mathcal{T}}|\theta)$, where θ is the set of trainable parameters for input transformation
3. Output mapping: Obtain the prediction on the target task via $\widehat{y}_{\mathcal{T}} = \mathsf{Output\text{-}Mapping}(f_{\mathcal{S}}(\widetilde{x}_{\mathcal{T}})|\omega)$, where ω is the set of trainable parameters for output mapping[2]
4. Model training: Optimize θ and ω by evaluating a task-specific loss $\mathsf{Loss}(\widehat{y}_{\mathcal{T}}, y_{\mathcal{T}}|\theta, \omega)$ on $\{x_{\mathcal{T}}^{(i)}, y_{\mathcal{T}}^{(i)}\}_{i=1}^{n}$
5. Outcome: Reprogrammed model from $f_{\mathcal{S}}(\cdot)$ with optimized trainable parameters θ^* and ω^* such that $\widehat{y}_{\mathcal{T}} = \mathsf{Output\text{-}Mapping}(f_{\mathcal{S}}(\mathsf{Input\text{-}Transform}(x_{\mathcal{T}}|\theta^*))|\omega^*)$

Autism Spectrum Disorder (ASD) Classification

- Autism Brain Imaging Data Exchange (ABIDE) database
  503 individuals suffering from ASD and 531 non-ASD samples
- Data sample
  200×200 brain-regional correlation graph of fMRI measurements
- Source foundation model
  ImageNet pre-trained models. AR/BAR=white-box/black-box reprogramming

Model	Accuracy
Resnet 50 (AR)	72.99%
Resnet 50 (BAR)	70.33%
Train from scratch	50.96%
Transfer Learning (finetuned)	52.88%
Incept.V3 (AR)	72.30%
Incept.V3 (BAR)	70.10%
Train from scratch	49.80%
Transfer Learning (finetuned)	50.10%
SOTA 1. (Heinsfeld et al., 2018)	65.40%
SOTA 2. (Eslami et al., 2019)	69.40%

1. Data efficiency
   Reprograming is better than transfer learning or train from scratch
2. Effectiveness
   Reprogramming outperforms SOTA
3. Practicality
   BAR is comparable to (white-box) AR

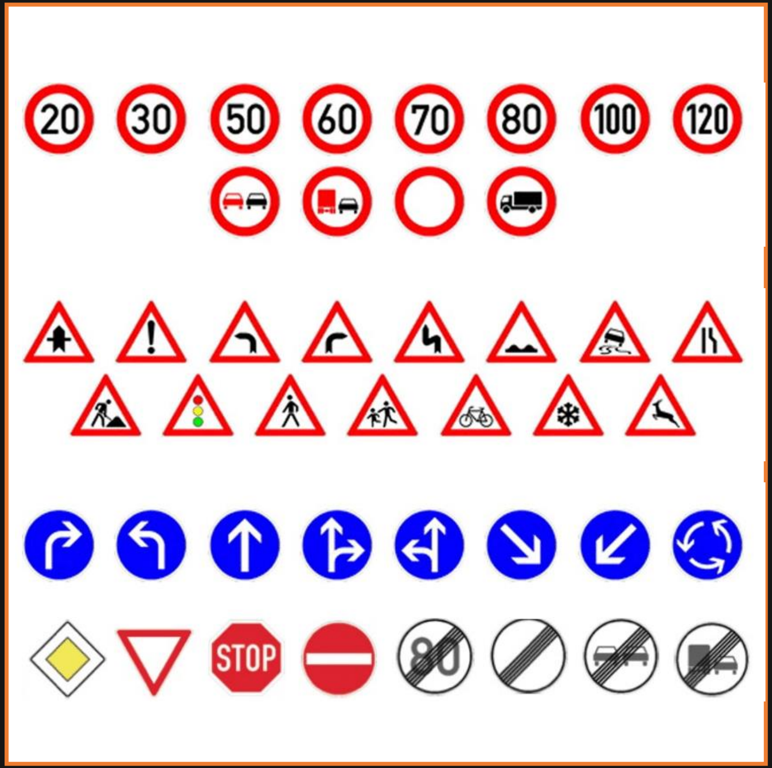
Eslami et al. Asd-diagnet: A hybrid learning approach for detection of autism spectrum disorder using fmri data. Frontiers in Neuroinformatics, 13, Nov 2019.

Reprogramming Microsoft Custom Vision API

This API allows user uploading labeled datasets and training an ML model for prediction. The model is unknown to end user.

We use this API and train a traffic sign image recognition model (43 classes) using a traffic sign classification dataset (GTSRB).

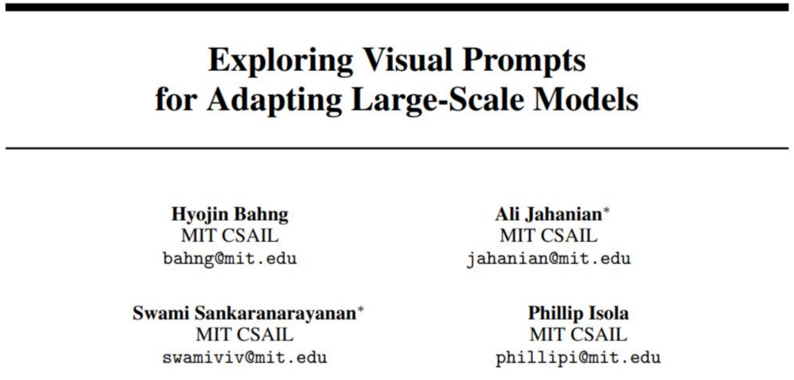
Orig. Task to New Task	q	# of query	Accuracy	Cost
Traffic sign classification	1	1.86k	48.15%	$3.72
to	5	5.58k	62.34%	$11.16
ASD	10	10.23k	**67.80%**	$20.46



Model Reprogramming Meets Visual Prompting

Model reprogramming on in-domain computer vision pretrained models for in-domain downstream tasks = visual prompting

What is Visual Prompting?



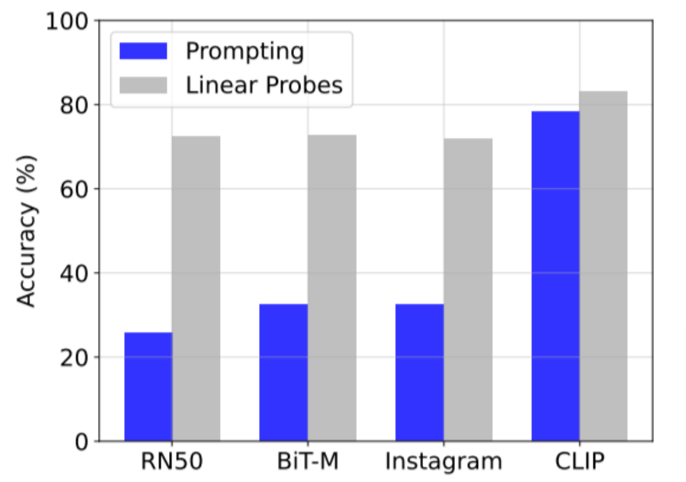
Exploring Visual Prompts for Adapting Large-Scale Models

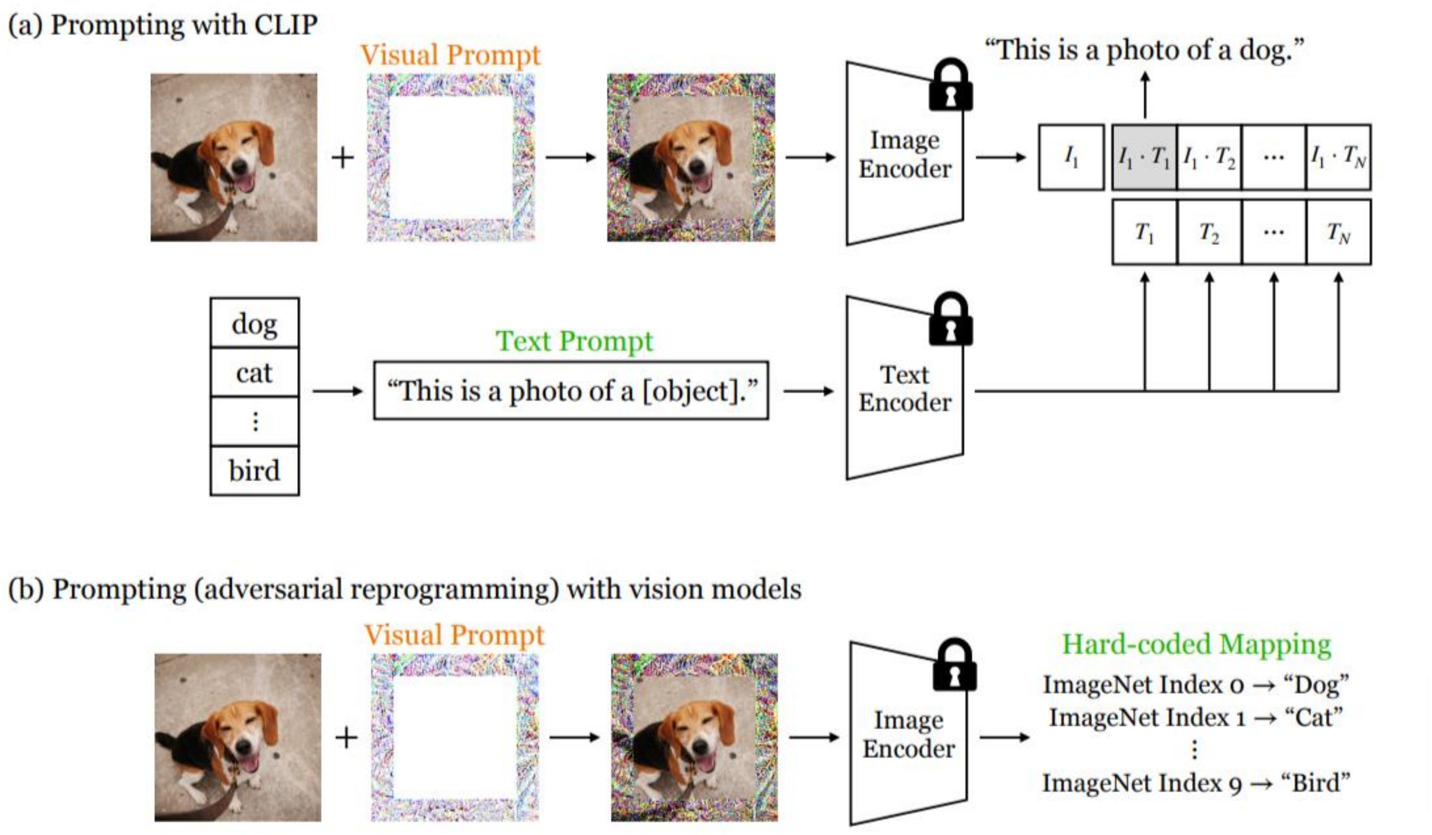
Hyojin Bahng
MIT CSAIL
bahng@mit.edu

Ali Jahanian*
MIT CSAIL
jahanian@mit.edu

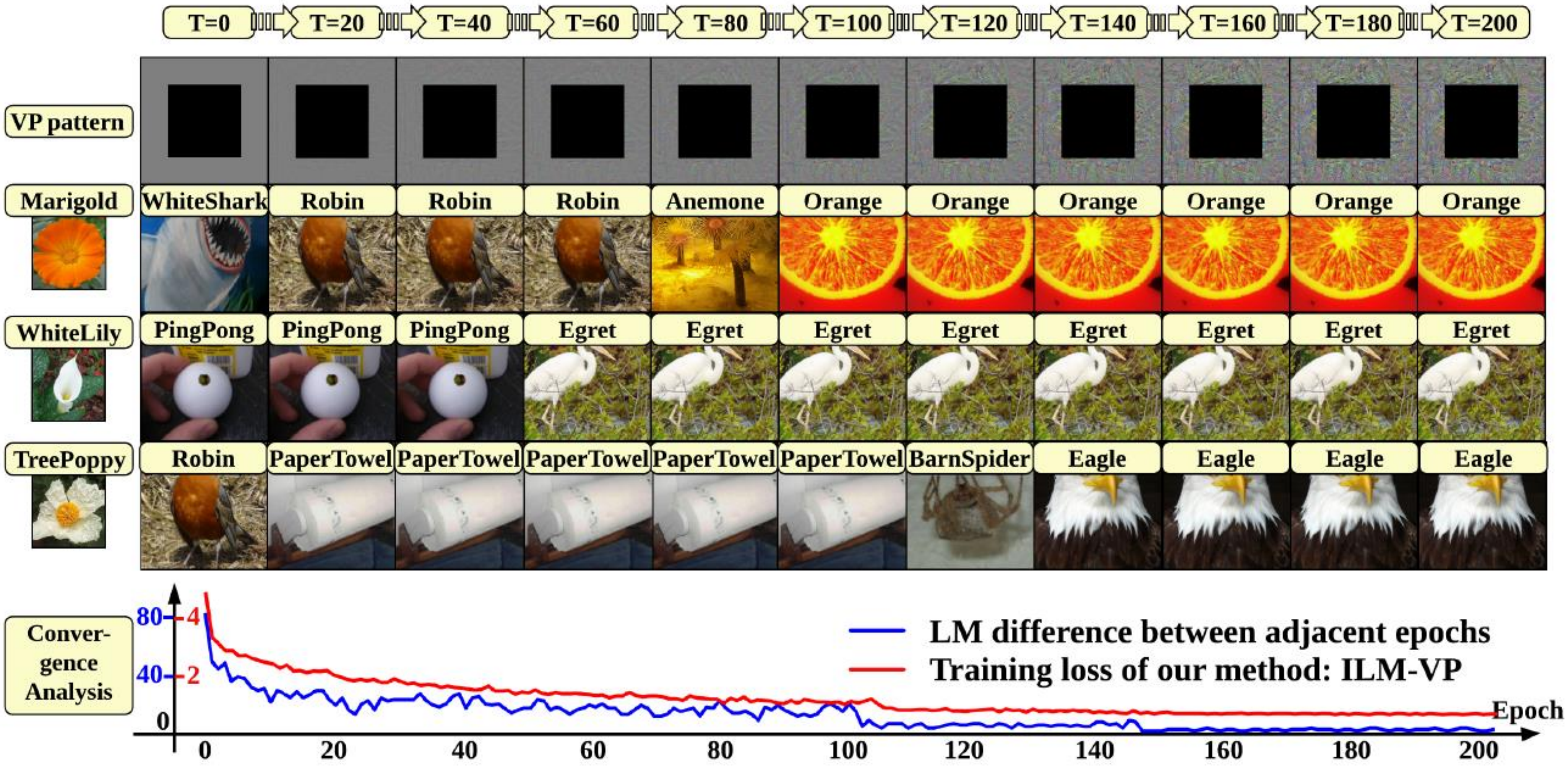
Swami Sankaranarayanan*
MIT CSAIL
swamiviv@mit.edu

Phillip Isola
MIT CSAIL
phillipi@mit.edu

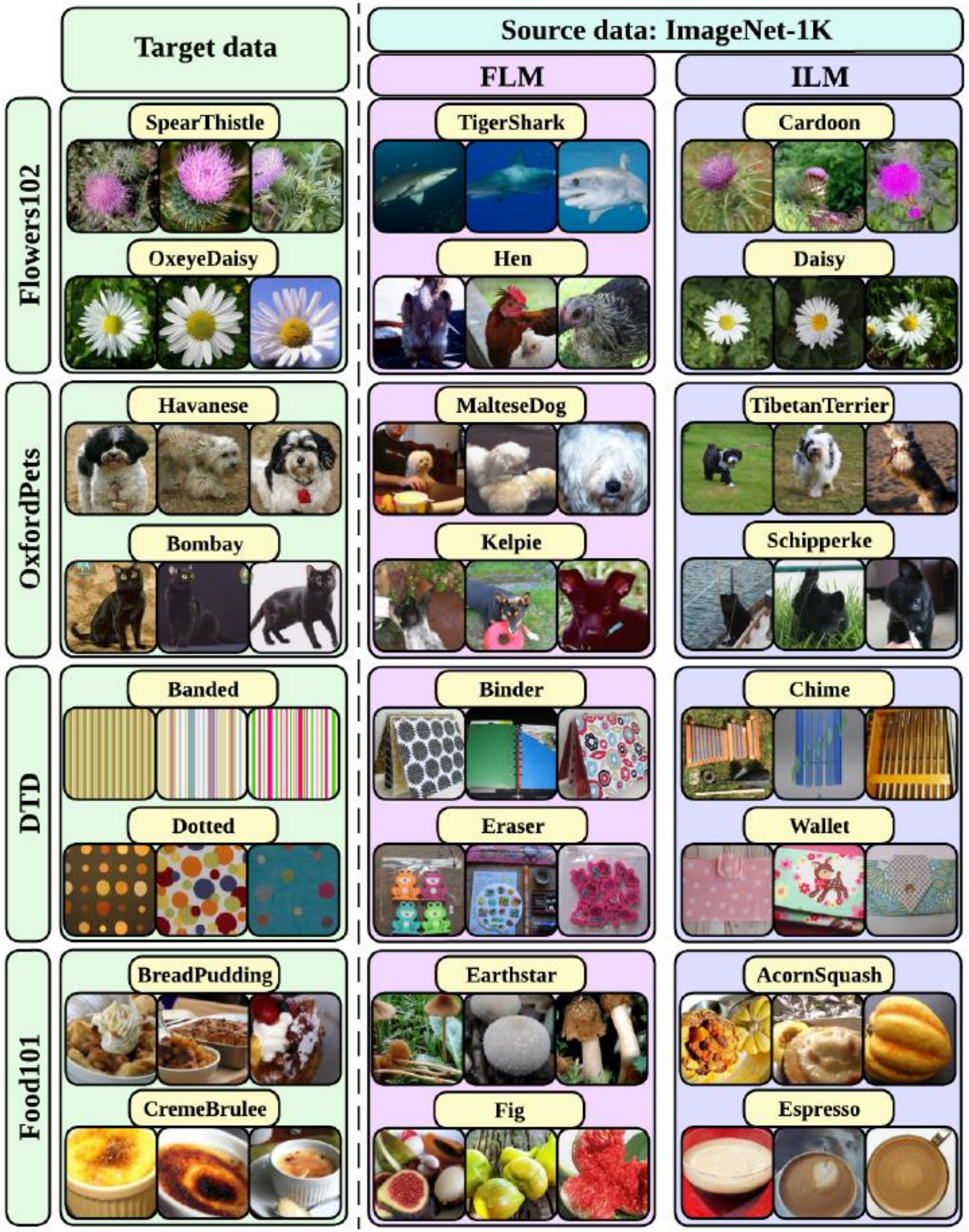




Iterative Label Mapping (ILM)

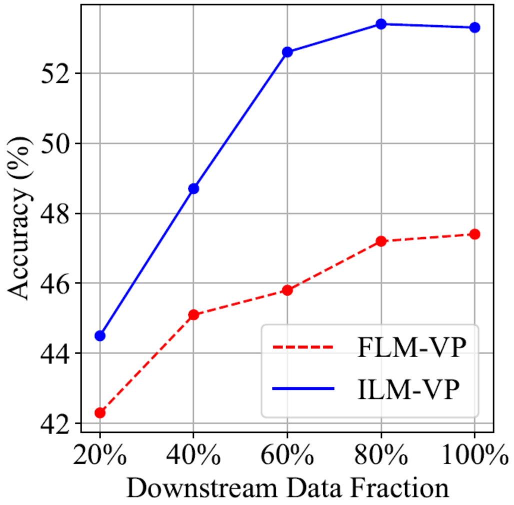


Aochuan Chen, Yuguang Yao, Pin-Yu Chen, Yihua Zhang, and Sijia Liu. Understanding and Improving Visual Prompting: A Label-Mapping Perspective. CVPR 2023



- CLIP prompting via ILM

Methods	VP+TP Acc(%)	Ours (VP+TP+LM) Acc(%)	Ours (VP+TP+LM) Examples of context prompt template → target label
Flowers102	70.0	**83.7**	a close-up photo of a {} → buttercup
DTD	56.8	**63.9**	graffiti of a {} → blotchy
UCF101	66.0	**70.6**	a {} in a video game → baseball pitch
Food101	78.9	**79.1**	a photo of the dirty {} → crab cake
SVHN	89.9	**91.2**	a photo of a {} → 7
EuroSAT	96.4	**96.9**	a pixelated photo of a {} → river
StanfordCars	57.2	**57.6**	the toy {} → 2011 audi s6 sedan
SUN397	60.5	**61.2**	a photo of a large {} → archive
CIFAR10	93.9	**94.4**	a pixelated photo of a {} → ship
ImageNet-R	67.5	**68.6**	a rendition of a {} → gold fish
ImageNet-Sketch	38.5	**39.7**	a sketch of a {} → eagle



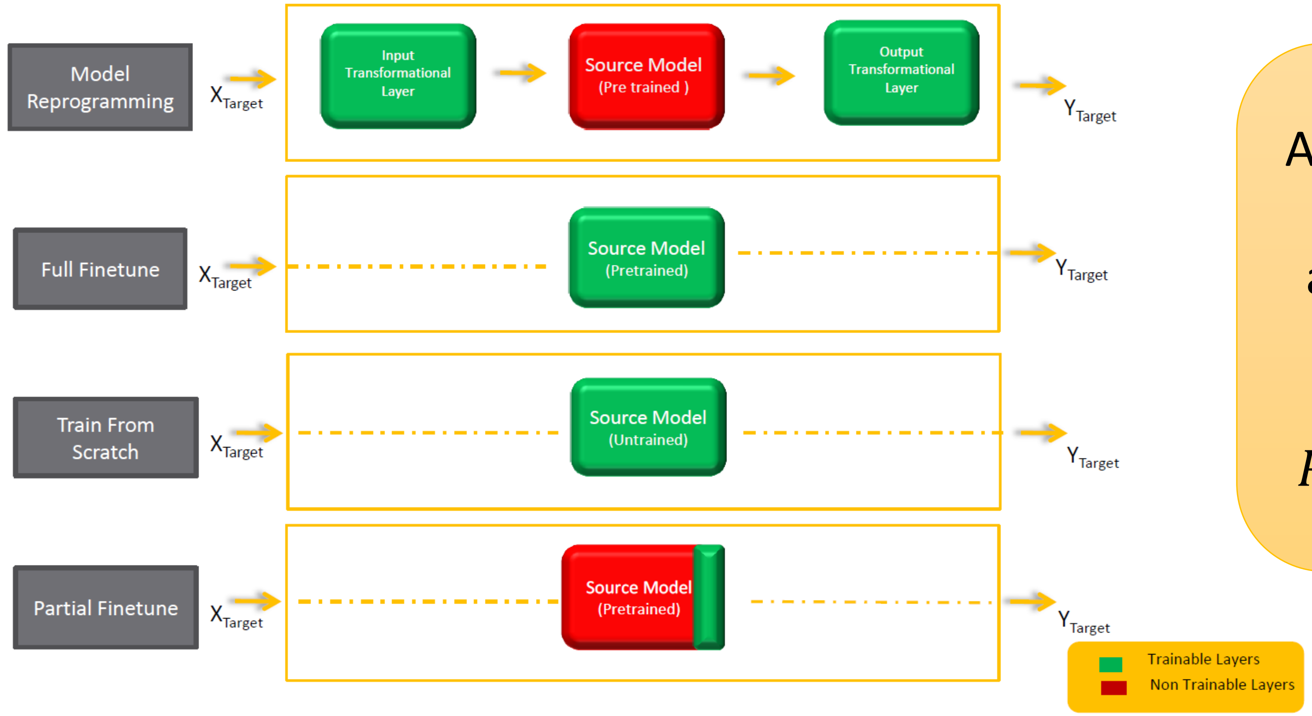
- Data scalability on GTSRB (traffic sign)

FLM = frequency label mapping

Model Reprogramming for Differentially Private Fine-tuning

Differentially Private Fine-tuning

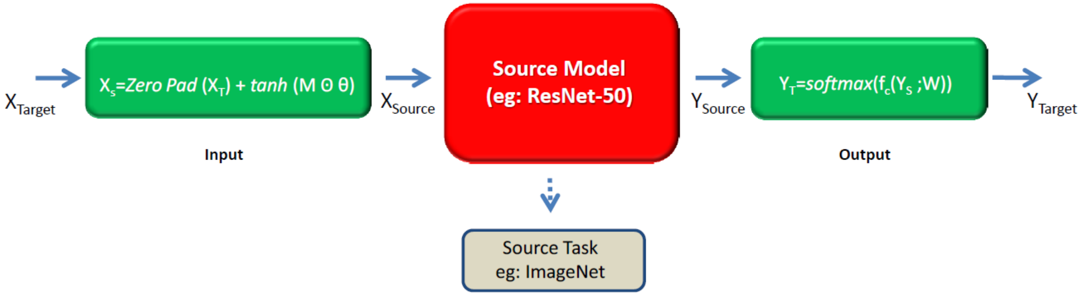
- Given a pretrained source model trained on non-private data
- Fine-tune the source model on private downstream data with differential privacy (DP) for maximal utility



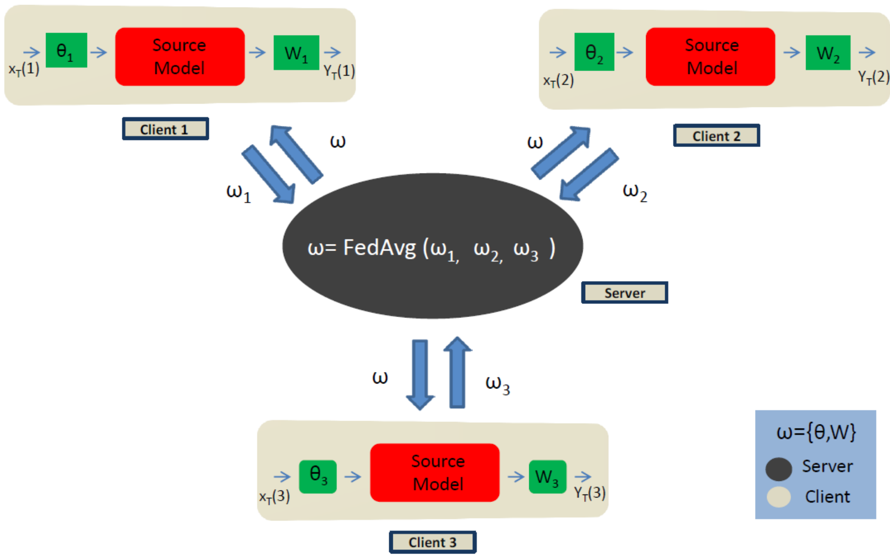
A randomized algorithm A is said to be $(\epsilon, \delta) - \text{DP}$ if it guarantees that for any two training datasets D and D' that differ by the inclusion or exclusion of a single training example, and for any set S in the output space,

$$Prob(A(D) \in S) \leq \exp(\epsilon) \cdot Prob(A(D') \in S) + \delta$$

Centralized and Federated Model Reprogramming



(a) Centralized model reprogramming. T/S denotes the target/source domains.



(b) Federated learning with model reprogramming (Reprogrammable-FL)

Algorithm 1 Federated Model Reprogramming (Reprogrammable-FL) – Client Side

Input: $\mathbf{x}^i, \mathbf{y}^i = \{x^i_{\mathcal{T},j}, y^i_{\mathcal{T},j}\}_{j=1}^{n_i}$

1: $\textbf{ClientUpdate}^i(\omega_t; C, \sigma, L, \mathcal{B}, f_S)$
2: $\omega_0^i \leftarrow \omega$
3: **for** $t \in \{0, \ldots, L-1\}$ **do**
4: $\quad \mathcal{B} \leftarrow$ uniform sampling w/o replacement
5: $\quad$ Update input transformation layer $\Theta^i_{t+1} \leftarrow \Theta^i_t - \eta \cdot \frac{1}{\mathcal{B}} \cdot [\sum_{b \in \mathcal{B}} \text{Clip}(\nabla_{\Theta^i} \ell(\omega^i_t; (\mathbf{x}^i_b, \mathbf{y}^i_b))) + \mathcal{N}(0, \sigma^2 C^2 \mathbf{I}))]$
6: $\quad$ Update output transformation layer $W^i_{t+1} \leftarrow W^i_t - \eta \cdot \frac{1}{\mathcal{B}} \cdot [\sum_{b \in \mathcal{B}} \text{Clip}(\nabla_{W^i} \ell(\omega^i_t; (\mathbf{x}^i_b, \mathbf{y}^i_b))) + \mathcal{N}(0, \sigma^2 C^2 \mathbf{I}))]$
7: $\quad \omega^i_{t+1} \leftarrow (\Theta^i_{t+1}, W^i_{t+1})$
8: **end for**
9: **return** ω^i_L

Algorithm 2 Federated Model Reprogramming (Reprogrammable-FL) – Server Side

Input: $\omega_0 = (\Theta_0, W_0)$ initialised randomly, δ, T, L, $\mathcal{B}$, C, σ, N, f_S
Output: $\omega_T = (\Theta_T, W_T)$

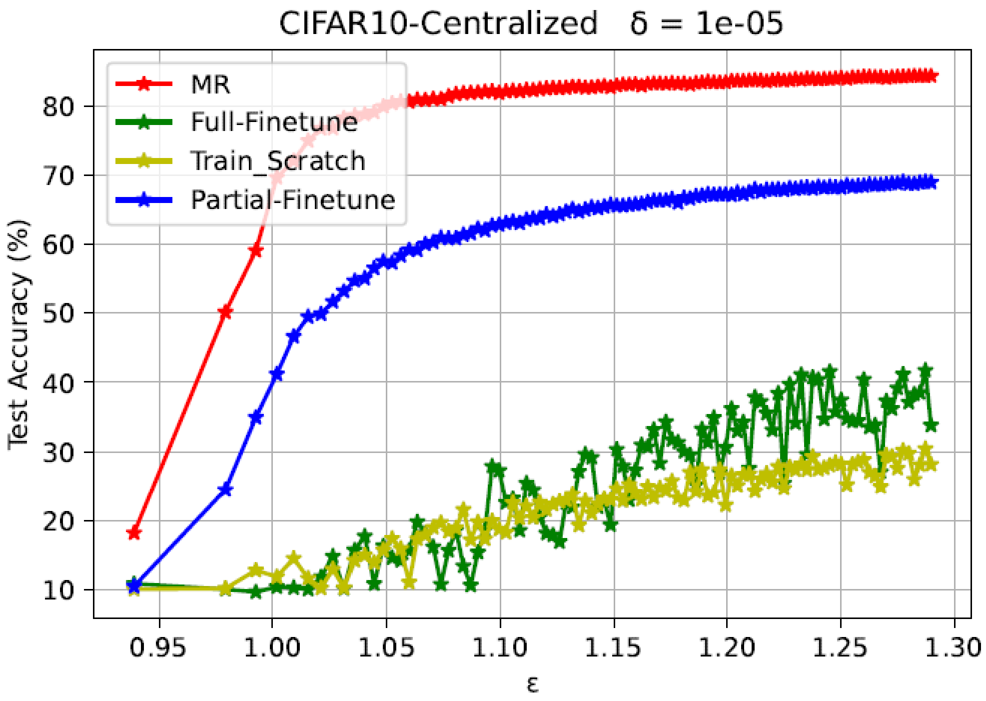
1: **for** $t \in \{0, \ldots, T-1\}$ **do**
2: $\quad$ **for all** $i \in m$ in parallel **do**
3: $\quad\quad \omega^i_{t+1} = \textbf{ClientUpdate}^i(\omega_t, C, \sigma, L, \mathcal{B}, f_S)$
4: $\quad$ **end for**
5: $\quad$ Update $\omega_{t+1} \leftarrow \sum_{i=1}^m \alpha_i \omega^i_{t+1}$
6: $\quad$ Server calculates expended privacy budget ε using moments accountant for fixed δ
7: **end for**

Gradient clipping + Gaussian noise for DP (on trainable parameters)

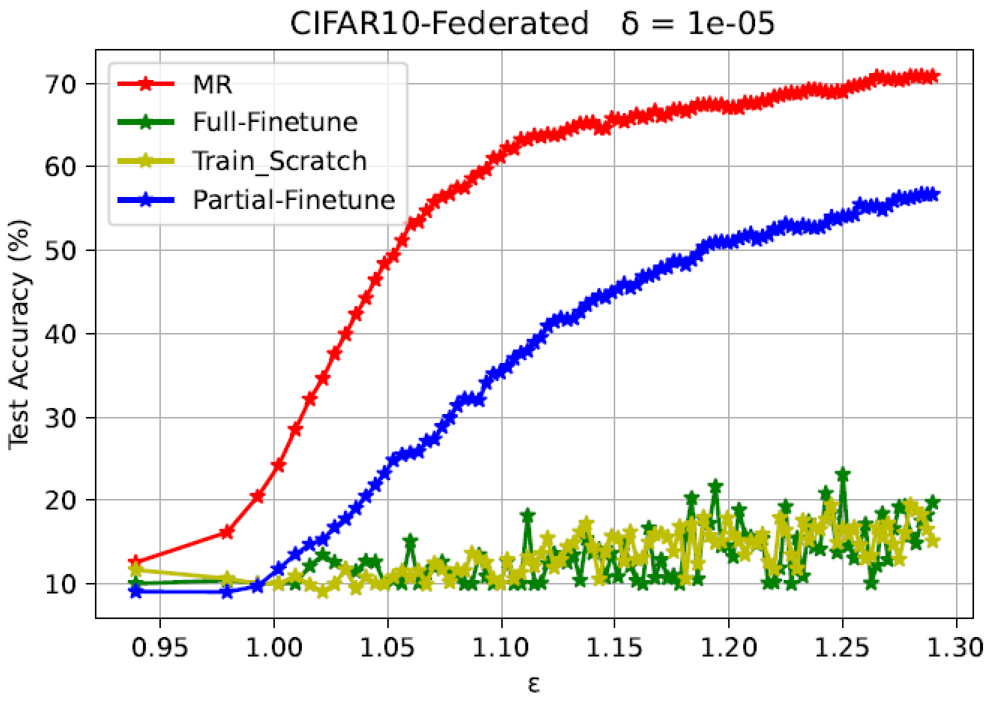
Federated Averaging + Budget Tracking

Huzaifa Arif, Alex Gittens, and Pin-Yu Chen. Reprogrammable-FL: Improving Utility-Privacy Tradeoff in Federated Learning via Model Reprogramming. SaTML 2023

Improved Accuracy-Privacy Tradeoff via MR



(a) Centralized setting



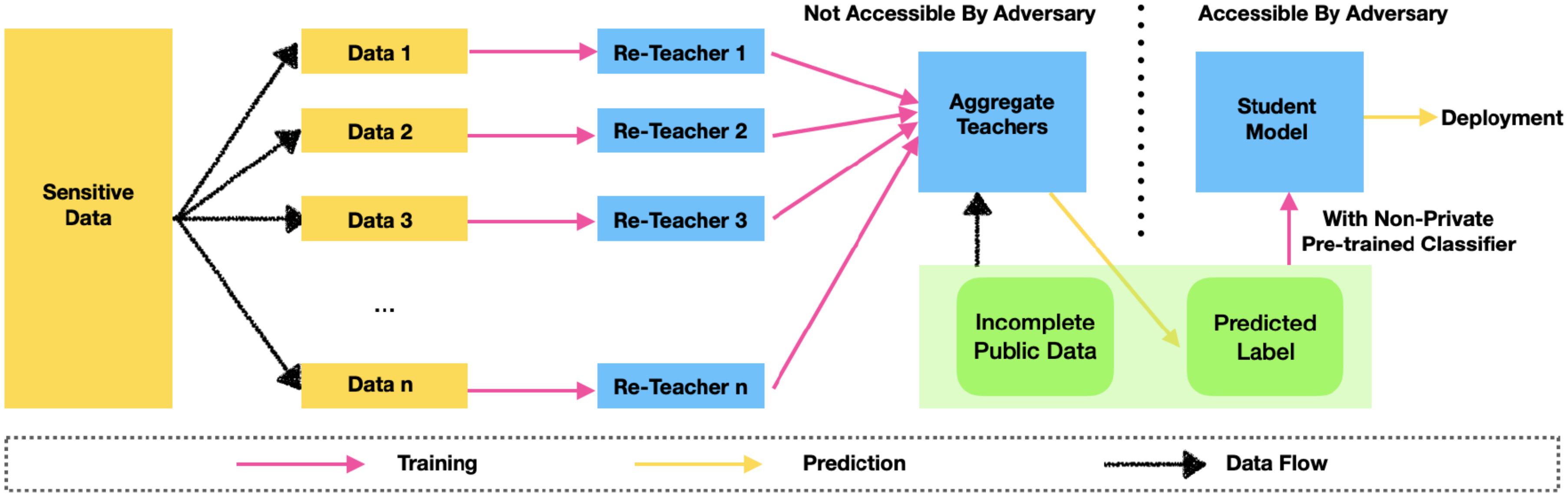
(b) Federated setting

Huzaifa Arif, Alex Gittens, and Pin-Yu Chen. Reprogrammable-FL: Improving Utility-Privacy Tradeoff in Federated Learning via Model Reprogramming. SaTML 2023

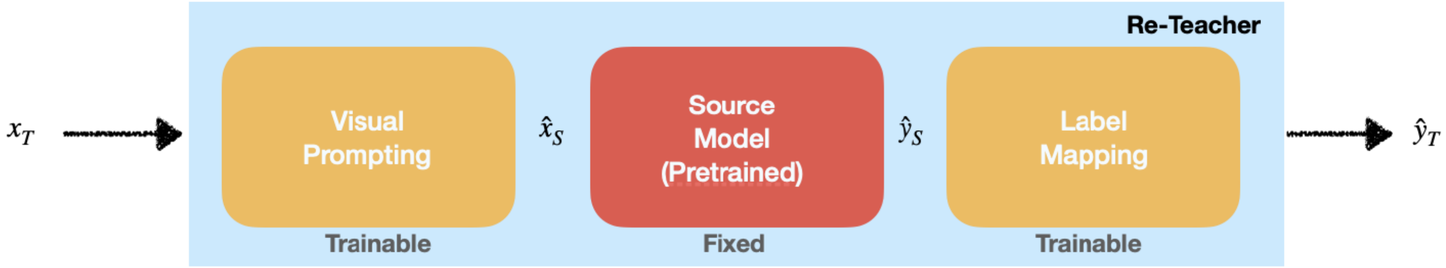
Visual Prompting for DP Fine-tuning

Prom-PATE: Visual Prompting + PATE (Private Aggregation of Teacher Ensembles)

Semi-supervised Setup



PATE: (1) Train separate teacher models on disjoint sensitive datasets; (2) Train student model using predicted labels on public data from the ensemble



Reprogrammed Teacher Model

Yizhe Li, Yu-Lin Tsai, Xuebin Ren, Chia-Mu Yu, abd Pin-Yu Chen. Exploring the Benefits of Visual Prompting in Differential Privacy. arxiv
Nicolas Papernot, Martın Abadi, Ulfar Erlingsson, Ian Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private training data. ICLR 2017

Improved Accuracy-Privacy Tradeoff via Prom-PATE

	ϵ	Accuracy on CIFAR-10
Arif et al. [2]	1.04	87.55%
Luo et al. [24]	1 1.5	76.64% 81.57%
Tramer et al. [33]	2	92.7%
Yu et al. [39]	1 2	94.3% 94.8%
De et al. [11]	1 2	94.7% 95.4%
Bu et al. [5]	1 2	96.7% 97.1%
Prom-PATE	1.019 1.505 1.943	97.07% 97.13% 97.16%

SOTA result

	ϵ	Accuracy $\pm$ Std(%)
ResNet50	1.081	95.27 ± 0.80
ResNet152	1.009	95.40 ± 0.40
WideResNet	1.068	94.37 ± 0.25
ViT	1.007	95.53 ± 0.51
Swin	1.019	$\mathbf{97.07 \pm 0.50}$

CIFAR-10 with different pretrained ImageNet models

Cross-domain: ImageNet -> Blood-MNIST

Blood-MNIST	**Prom-PATE**	**Transfer-PATE**	**Arif et al. [2]**
ϵ	1.973	1.983	1.971
Accuracy(%)	**69.93**	61.33	63.45

Yizhe Li, Yu-Lin Tsai, Xuebin Ren, Chia-Mu Yu, abd Pin-Yu Chen. Exploring the Benefits of Visual Prompting in Differential Privacy. arxiv

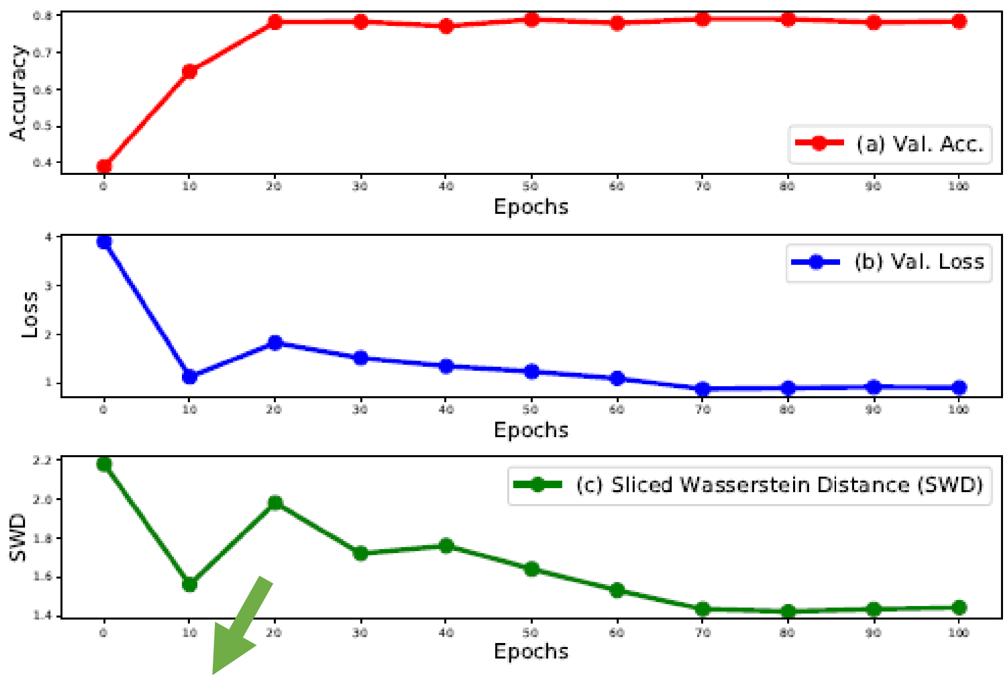
Why Model Reprogramming Works?

https://arxiv.org/abs/2106.09296 (ICML 2021)

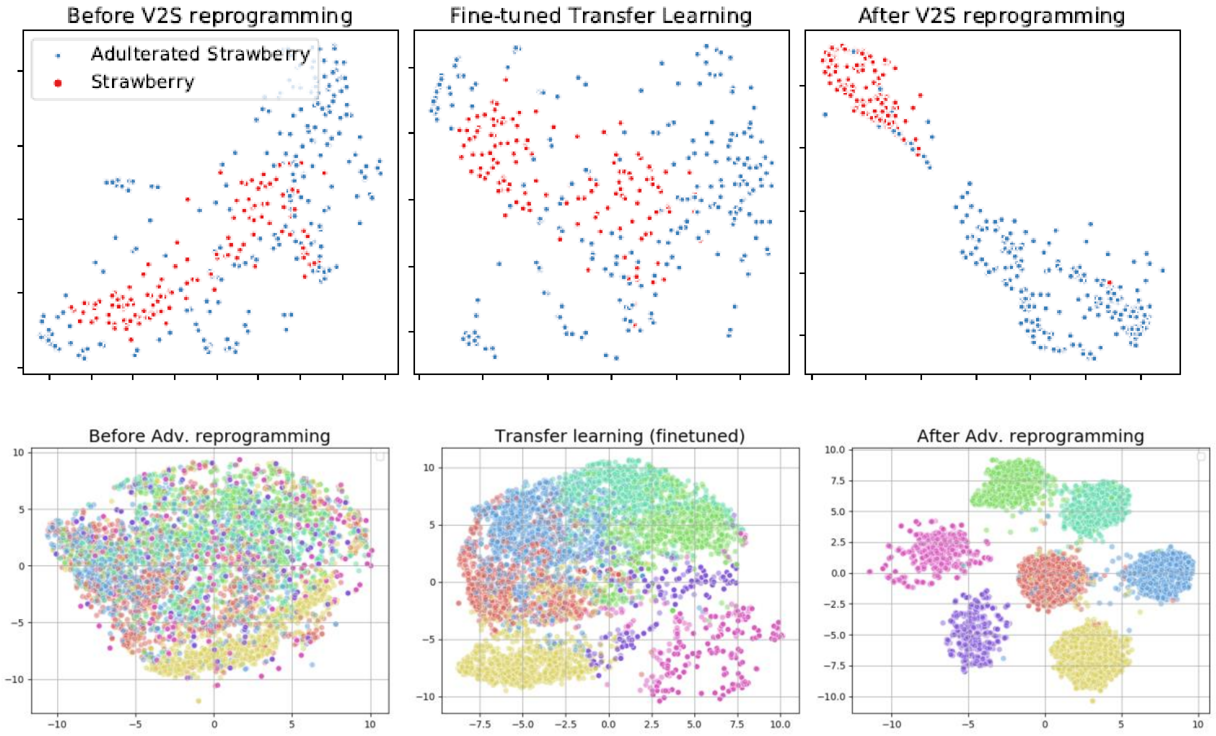
Why and When Model Reprogramming Works? (No, it's not about knowledge transfer)

- (Informal) Theorem for model reprogramming:

  Target risk ≤ Source risk + Representation Alignment Loss



Distance between source and reprogrammed target data representations



Theorem 1: Let δ^* denote the learned additive input transformation for reprogramming (Assumption 4). The population risk for the target task via reprogramming a K-way source neural network classifier $f_{\mathcal{S}}(\cdot) = \eta(z_{\mathcal{S}}(\cdot))$, denoted by $\mathbb{E}_{\mathcal{D}_{\mathcal{T}}}[\ell_{\mathcal{T}}(x_t + \delta^*, y_t)]$, is upper bounded by

$$\mathbb{E}_{\mathcal{D}_{\mathcal{T}}}[\ell_{\mathcal{T}}(x_t + \delta^*, y_t)] \leq \underbrace{\epsilon_{\mathcal{S}}}_{\text{source risk}}$$

$$+ 2\sqrt{K} \cdot \underbrace{\mathcal{W}_1(\mu(z_{\mathcal{S}}(x_t + \delta^*)), \mu(z_{\mathcal{S}}(x_s)))_{x_t \sim \mathcal{D}_{\mathcal{T}},\, x_s \sim \mathcal{D}_{\mathcal{S}}}}_{\text{representation alignment loss via reprogramming}}$$

Takeaways

Model Reprogramming:

A new paradigm of resource-limited cross-domain parameter-efficient finetuning with large pretrained models

-Improve data efficiency

-Reuse pretrained models from alternative domains

-Address compute limitations (training epochs, compute resource, etc)

Empirical success in:

- general imaging → medical imaging, human voice → time series, and NLP → molecular learning

- Privacy-constrained fine-tuning; compatible with existing DP training methods (DP-SGD, PATE)

Theoretical justification:

- Target task can be solved as effectively as the source task if their representations are perfectly aligned

Reprogramming is a strong baseline for parameter-efficient finetuning, among Adapters, LoRA, Prompting, etc

Codes & References

Opensource codes
BAR: https://github.com/IBM/blackbox-adversarial-reprogramming
V2S: https://github.com/IBM/Voice2Series-Reprogramming
Reprogrammable-FL: https://github.com/IBM/reprogrammble-FL

- Pin-Yu Chen. “Model Reprogramming: Resource-Efficient Cross-Domain Machine Learning,” arxiv
- Gamaleldin F. Elsayed, Ian Goodfellow, and Jascha Sohl-Dickstein. “Adversarial Reprogramming of Neural Networks,” ICLR 2019
- Yun-Yun Tsai, Pin-Yu Chen, and Tsung-Yi Ho. “Transfer Learning without Knowing: Reprogramming Black-box Machine Learning Models with Scarce Data and Limited Resources,” ICML 2020
- Ria Vinod, Pin-Yu Chen, and Payel Das. “Reprogramming Pretrained Language Models for Protein Sequence Representation Learning,” arxiv
- Chao-Han Huck Yang, Yun-Yun Tsai and Pin-Yu Chen. Voice2Series: Reprogramming Acoustic Models for Time Series Classification,” ICML 2021
- Hao Yen, Pin-Jui Ku, Chao-Han Huck Yang, Hu Hu, Sabato Marco Siniscalchi, Pin-Yu Chen, and Yu Tsao. “Neural Model Reprogramming with Similarity Based Mapping for Low-Resource Spoken Command Recognition,” INTERSPEECH 2023
- Yun-Ning Hung, Chao-Han Huck Yang, Pin-Yu Chen, and Alexander Lerch. “Low-Resource Music Genre Classification with Cross-Modal Neural Model Reprogramming,” ICASSP 2023
- Aochuan Chen*, Peter Lorenz*, Yuguang Yao, Pin-Yu Chen, and Sijia Liu. “Visual Prompting for Adversarial Robustness,” ICASSP 2023
- Aochuan Chen, Yuguang Yao, Pin-Yu Chen, Yihua Zhang, and Sijia Liu. “Understanding and Improving Visual Prompting: A Label-Mapping Perspective,” CVPR 2023
- Yung-Chen Tang, Pin-Yu Chen, and Tsung-Yi Ho. “Neural Clamping: Joint Input Perturbation and Temperature Scaling for Neural Network Calibration,” arxiv
- Huzaifa Arif, Alex Gittens, and Pin-Yu Chen. “Reprogrammable-FL: Improving Utility-Privacy Tradeoff in Federated Learning via Model Reprogramming,” SaTML 2023
- Guanhua Zhang, Yihua Zhang, Yang Zhang, Wenqi Fan, Qing Li, Sijia Liu, and Shiyu Chang. "Fairness reprogramming,“ NeurIPS 2022
- Yizhe Li, Yu-Lin Tsai, Xuebin Ren, Chia-Mu Yu, abd Pin-Yu Chen. “Exploring the Benefits of Visual Prompting in Differential Privacy,” arxiv