Data Extraction And Model Stealing

Lecturer: Dr. Xingjun Ma

School of Computer Science, Fudan University

Autumn, 2022

Recap: week 8

- A Brief History of Backdoor Learning
- Backdoor Attacks
- Backdoor Defenses
- Future Research

Adversarial Attack Competition: Phase 2

Phase 1 | Phase 2 | Phase 3

Phase description

Create an attack method and submit the code as submission. Your code should follows the submission template. Feedback will be provided on the 1000 "hard" test images. We will test your code on 1 robustly trained model.

Max submissions per day: 10

Max submissions total: 500

RESULTS

#	User	Entries	Date of Last Entry	Score ▲	Error Rate ▲	Efficiency Score ▲	Detailed Results
1	yong_xie	14	10/27/22	0.1954 (1)	0.1910 (4)	0.4435 (8)	View
2	terrytengli	2	11/02/22	0.1954 (2)	0.1920 (1)	0.3396 (12)	View
3	xinwang22	48	11/02/22	0.1954 (3)	0.1917 (2)	0.3686 (11)	View
4	strawberryXia	9	11/02/22	0.1951 (4)	0.1917 (2)	0.3394 (13)	View
5	Shadow_H	4	11/02/22	0.1949 (5)	0.1920 (1)	0.2949 (14)	View
6	wangzhix	1	10/19/22	0.1942 (6)	0.1913 (3)	0.2888 (15)	View
7	kejiefang	11	11/02/22	0.1942 (7)	0.1900 (7)	0.4190 (9)	View
8	yfshao	2	10/26/22	0.1936 (8)	0.1917 (2)	0.1931 (18)	View
9	keren	4	10/20/22	0.1932 (9)	0.1873 (10)	0.5868 (6)	View
10	liuhuan	4	10/23/22	0.1923 (10)	0.1913 (3)	0.1002 (20)	View
11	Yuxuan_Wang	8	11/02/22	0.1921 (11)	0.1903 (6)	0.1736 (19)	View

奖励：

- 冠军：*****
- 亚军：****
- 第三：***
- 第四：**
- 第五：*
- 第六-十：+

建议：前几名组队发篇攻击的文章

Link:https://codalab.lisn.upsaclay.fr/competitions/7556?secret_key=d4a3b1fa-66e2-4a80-8ce6-b5f99e518979

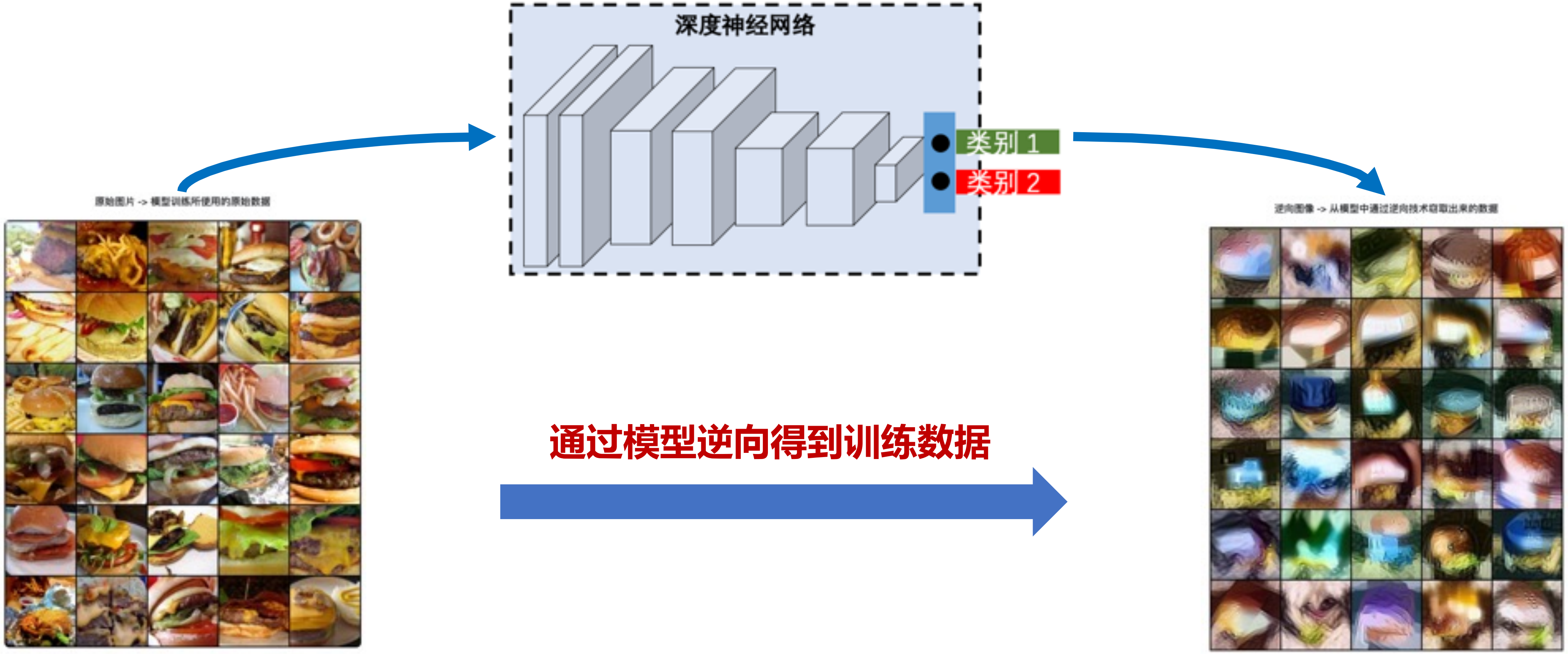
This Week

- Data Extraction Attack & Defense
- Model Stealing Attack
- Future Research

This Week

- **Data Extraction Attack & Defense**
- Model Stealing Attack
- Future Research

Data Extraction Attack

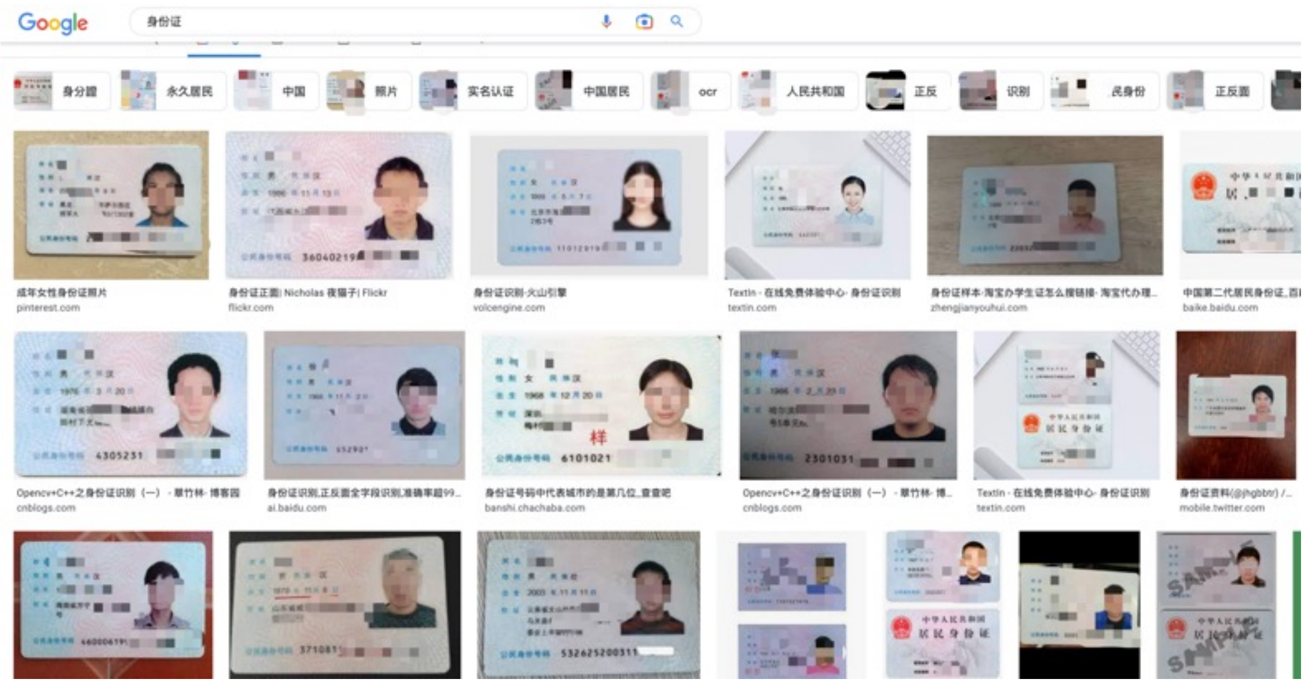


https://tech.openeglab.org.cn:8001/dss/imageClassify

Terminology

- The following terms describe the same thing:
  - Data Extraction Attack
  - Data Stealing Attack
  - Training Data Extraction Attack
  - Model Memorization Attack
  - Model Inversion Attack

Security Threats

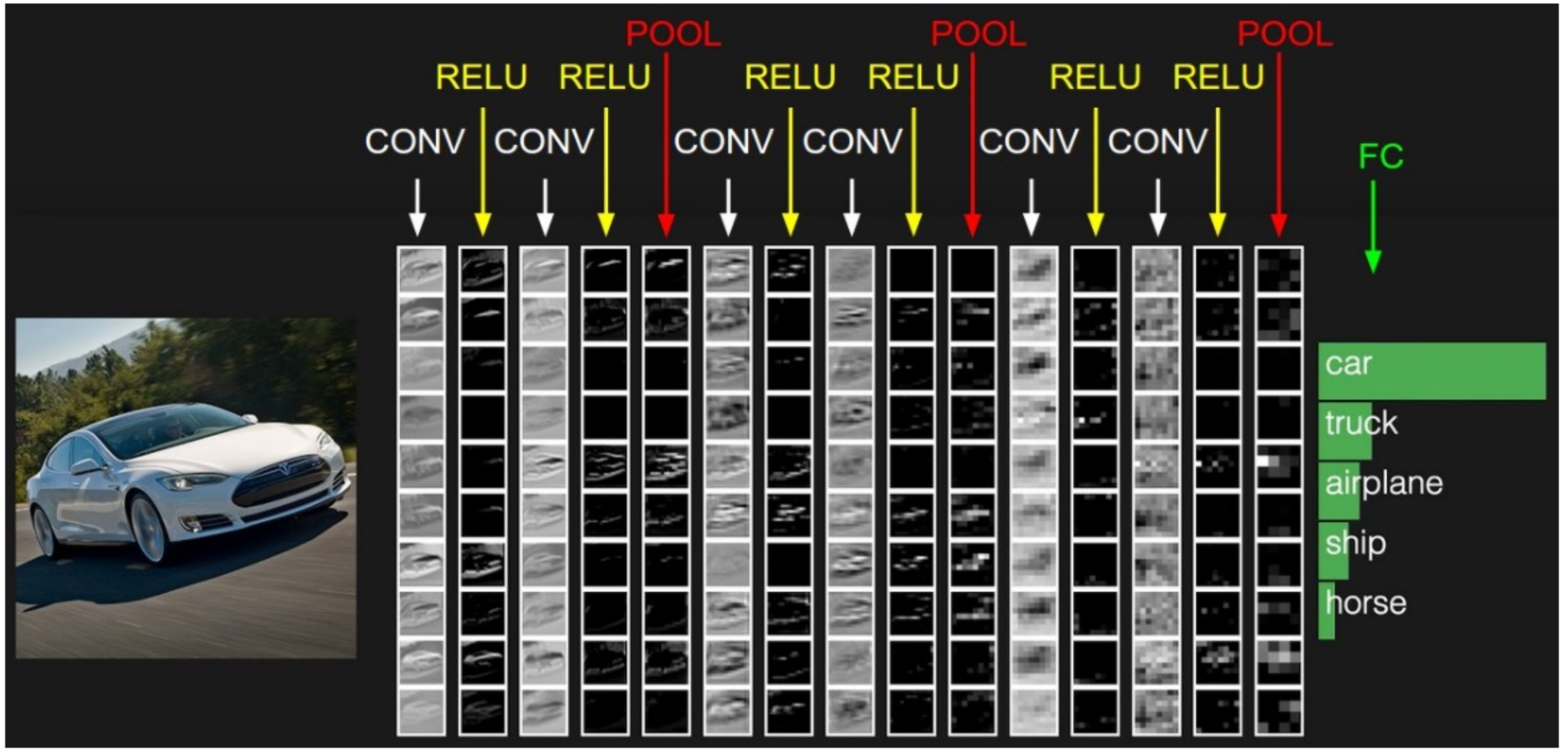


- 个人信息泄露
- 敏感信息泄露
- 威胁国家安全
- 非法数据交易
- …

My social security number is 078-

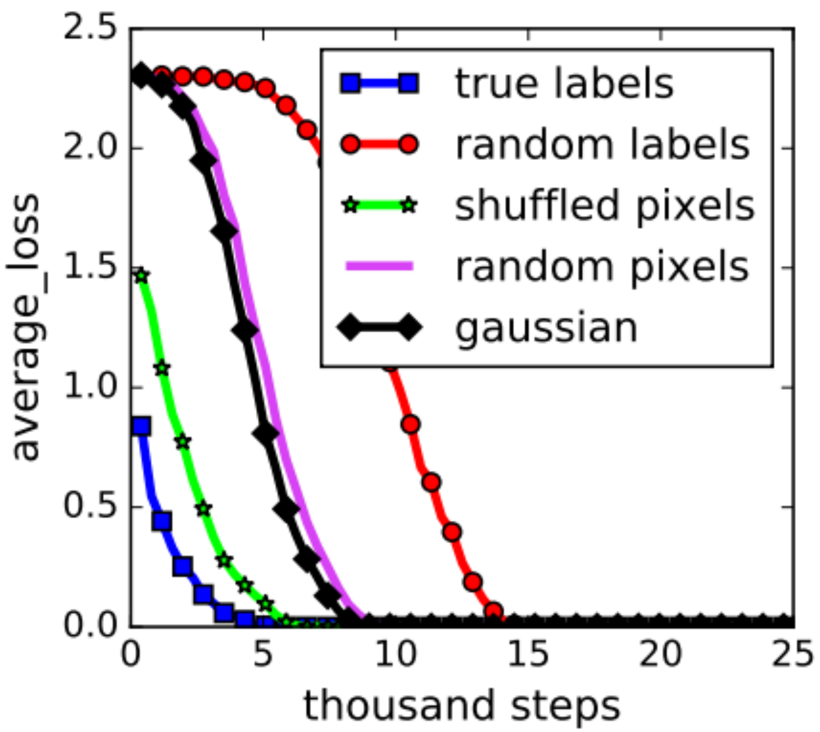
Memorization of DNNs

- **Evidence 1:** DNN learns different levels of representations



Memorization of DNNs

- **Evidence 2:** DNN can memorize random labels/pixels



- 真实标签
- 随机标签
- 乱序像素
- 随机像素
- 高斯噪声

Zhang, Chiyuan, et al. “Understanding deep learning requires rethinking generalization.” ICLR 2017.

Memorization of DNNs

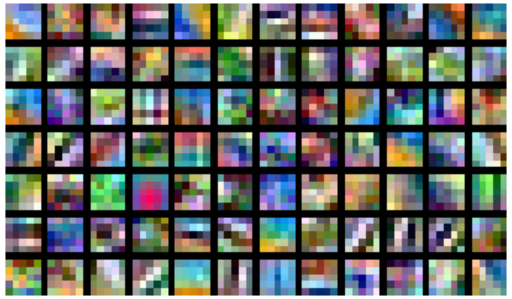
□ **Evidence 3:** The success of GANs and diffusion models

https://thispersondoesnotexist.com/; https://thisartworkdoesnotexist.com/

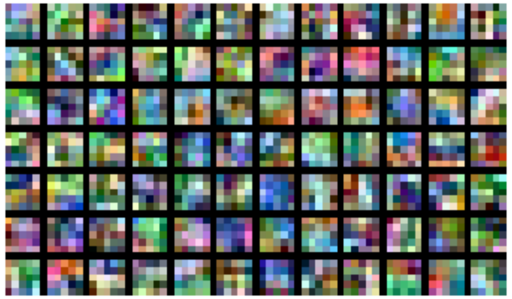
Intended Memorization vs Unintended Memorization

☐ Intended Memorization
- Task-related
- Statistics
- Inputs and Labels

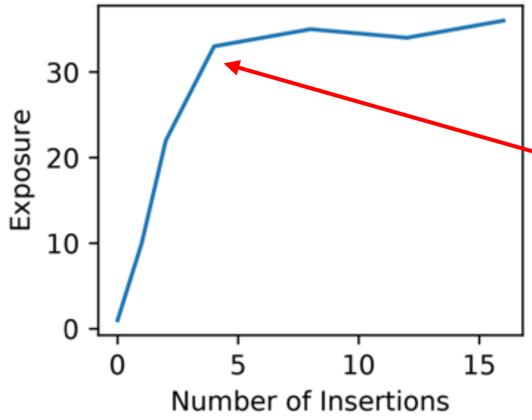
☐ Unintended Memorization
- Task-irrelevant but memorized
- Even appear only a few times



第一层Filter
正常CIFAR-10



第一层Filter
随机标注CIFAR-10

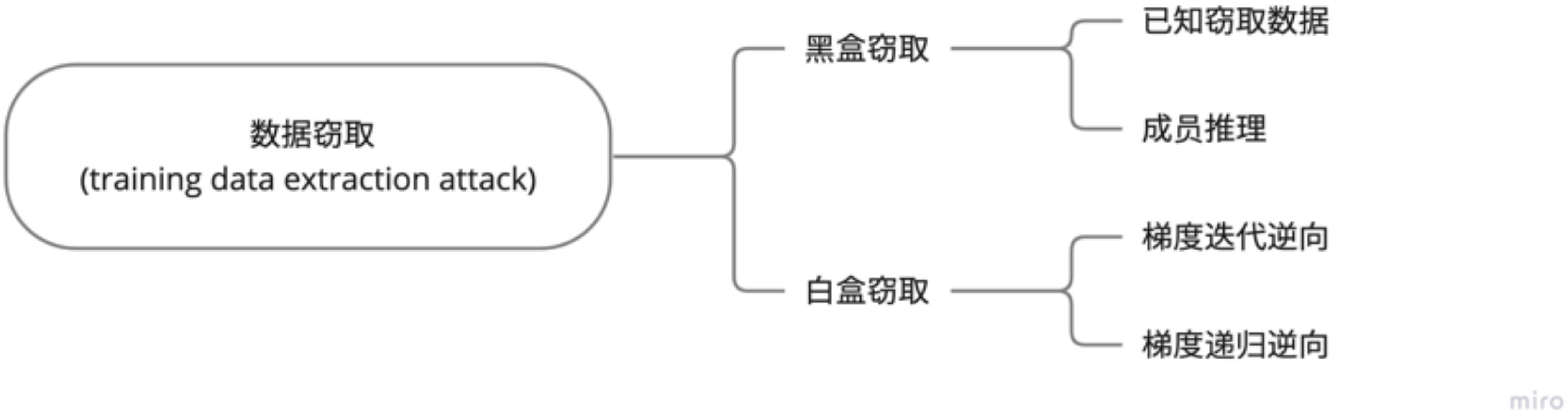


出现4
次就能
全记住

自然语言翻译模型记忆：
“**我的社保号码是 xxxx**”

Arpit et al. “A closer look at memorization in deep networks.” *ICML,* 2017. Carlini et al. “The secret sharer: Evaluating and testing unintended memorization in neural networks.” USENIX Security, 2019.

现有数据窃取攻击


数据窃取
(training data extraction attack)
黑盒窃取
已知窃取数据
成员推理
白盒窃取
梯度迭代逆向
梯度递归逆向
miro

黑盒窃取

◻ 意外记忆测试和量化：'先兆'

Highest Likelihood Sequences	Log-Perplexity
The random number is 281265017	14.63
The random number is 281265117	18.56
The random number is 281265011	19.01
The random number is 286265117	20.65
The random number is 528126501	20.88
The random number is 281266511	20.99
The random number is 287265017	20.99
The random number is 281265111	21.16
The random number is 281265010	21.36

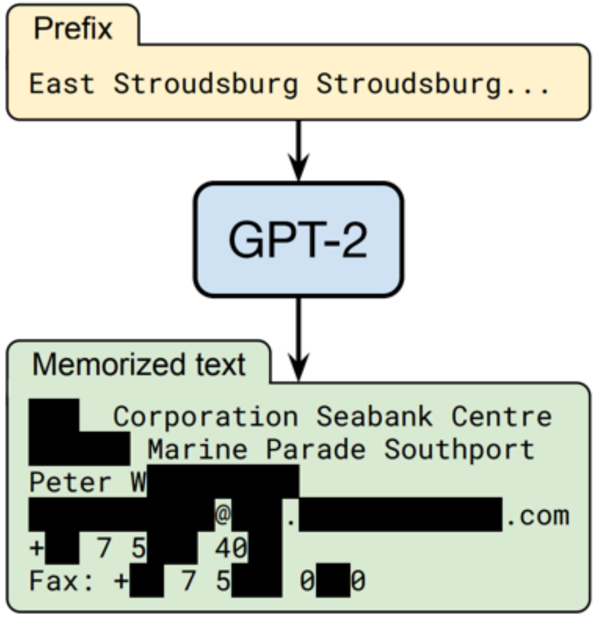
◻ 主动测试：

- **煤矿里的金丝雀**
- “随机号码为****”
- “我的社保号码为****”
- 主动注入，然后先兆数据在语言模型中的“曝光度”（Exposure）

Carlini et al. “The secret sharer: Evaluating and testing unintended memorization in neural networks.” USENIX Security, 2019.

黑盒窃取

□ 训练数据萃取攻击
Training Data Extraction Attack



□针对通用语言模型：

- 逆向出大量的：名字、手机号、邮箱、社保号等
- 大模型比小模型更容易记住这些信息
- 即使只在一个文档里出现也能被记住

Carlini, Nicholas, et al. “Extracting training data from large language models.” *USENIX Security,* 2021.

Definition of Memorization

Definition 1 (Model Knowledge Extraction) *A string s is* extractable[4] *from an LM f_θ if there exists a prefix c such that:*

$$s \leftarrow \underset{s':\,|s'|=N}{\arg\max}\, f_\theta(s' \mid c)$$

模型知识提取

Definition 2 (*k*-Eidetic Memorization) *A string s is k-eidetic memorized (for $k \geq 1$) by an LM f_θ if s is extractable from f_θ and s appears in at most k examples in the training data X: $|\{x \in X : s \subseteq x\}| \leq k$.*

k-逼真记忆

Carlini, Nicholas, et al. "Extracting training data from large language models." *USENIX Security,* 2021.

攻击步骤

步骤1：生成大量文本；步骤2：文本筛选和确认

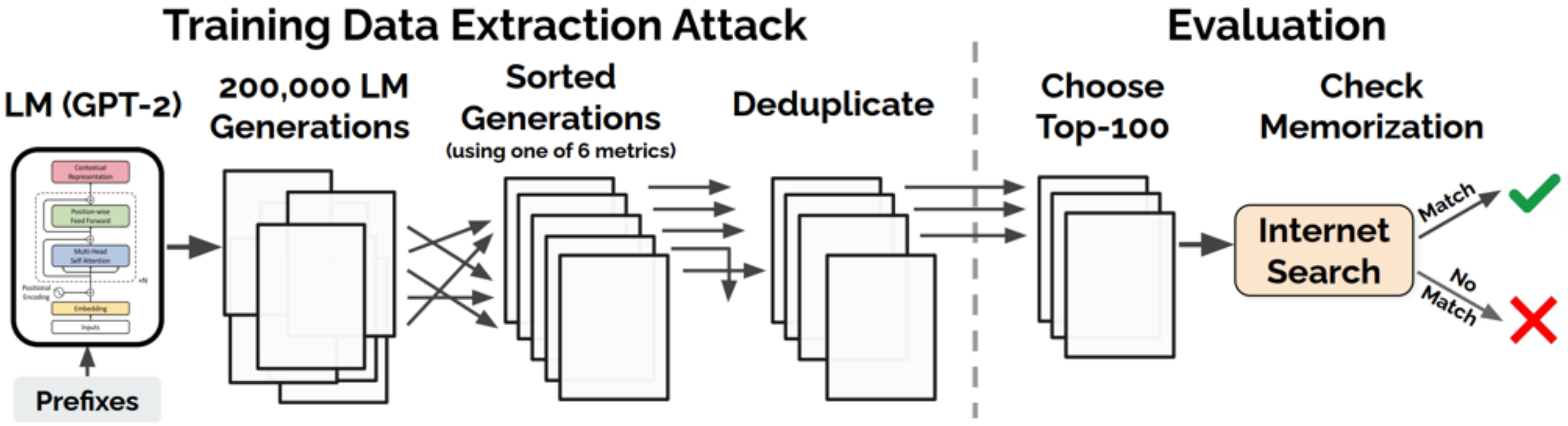


Figure 2: **Workflow of our extraction attack and evaluation. 1) Attack.** We begin by generating many samples from GPT-2 when the model is conditioned on (potentially empty) prefixes. We then sort each generation according to one of six metrics and remove the duplicates. This gives us a set of potentially memorized training examples. **2) Evaluation.** We manually inspect 100 of the top-1000 generations for each metric. We mark each generation as either memorized or not-memorized by manually searching online, and we confirm these findings by working with OpenAI to query the original training data.

Carlini, Nicholas, et al. "Extracting training data from large language models." *USENIX Security,* 2021.

实验结果

Category	Count
US and international news	109
Log files and error reports	79
License, terms of use, copyright notices	54
Lists of named items (games, countries, etc.)	54
Forum or Wiki entry	53
Valid URLs	50
Named individuals (non-news samples only)	46
Promotional content (products, subscriptions, etc.)	45
High entropy (UUIDs, base64 data)	35
Contact info (address, email, phone, twitter, etc.)	32
Code	31
Configuration files	30
Religious texts	25
Pseudonyms	15
Donald Trump tweets and quotes	12
Web forms (menu items, instructions, etc.)	11
Tech news	11
Lists of numbers (dates, sequences, etc.)	10

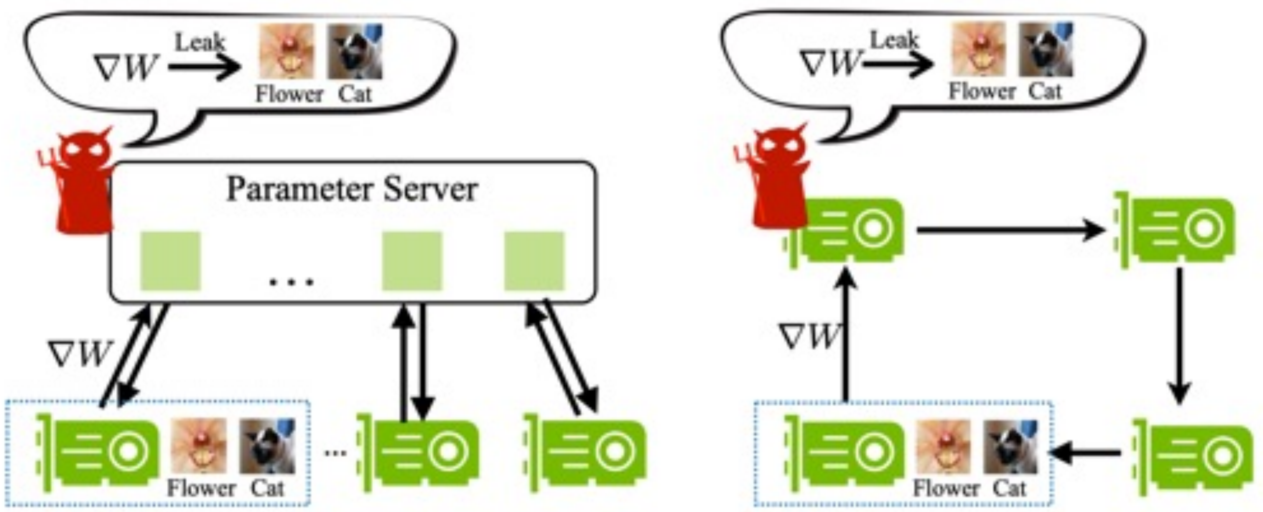
604条“意外”记忆

	Occurrences		Memorized?		
URL (trimmed)	Docs	Total	XL	M	S
/r/[redacted]51y/milo_evacua...	1	359	✓	✓	½
/r/[redacted]zin/hi_my_name...	1	113	✓	✓	
/r/[redacted]7ne/for_all_yo...	1	76	✓	½	
/r/[redacted]5mj/fake_news_...	1	72	✓		
/r/[redacted]5wn/reddit_admi...	1	64	✓	✓	
/r/[redacted]lp8/26_evening...	1	56	✓	✓	
/r/[redacted]jla/so_pizzagat...	1	51	✓	½	
/r/[redacted]ubf/late_night...	1	51	✓	½	
/r/[redacted]eta/make_christ...	1	35	✓	½	
/r/[redacted]6ev/its_officia...	1	33	✓		
/r/[redacted]3c7/scott_adams...	1	17			
/r/[redacted]k2o/because_his...	1	17			
/r/[redacted]tu3/armynavy_ga...	1	8			

只在一个文档里出现的记忆
模型越大记忆越强

白盒窃取

口 白盒窃取需要利用梯度信息，也称梯度逆向攻击（Gradient Inversion Attack）



(a) Distributed training with a centralized server (b) Distributed training without a centralized server

两种分布式训练范式

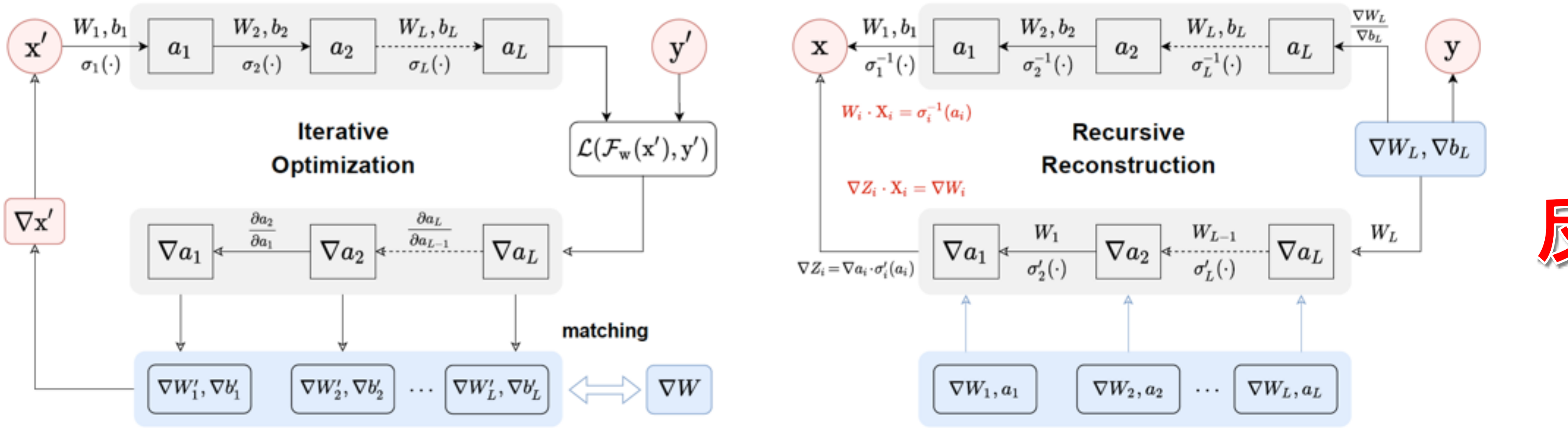
口针对梯度共享的训练：

- 分布式训练
- 联邦学习
- 并行训练
- 无中心化训练

白盒窃取

◻ 白盒窃取需要利用梯度信息，也称梯度逆向攻击（Gradient Inversion Attack）

逼近



反推

(a) Workflow of iteration-based GradInv attacks

(b) Workflow of recursion-based GradInv attacks

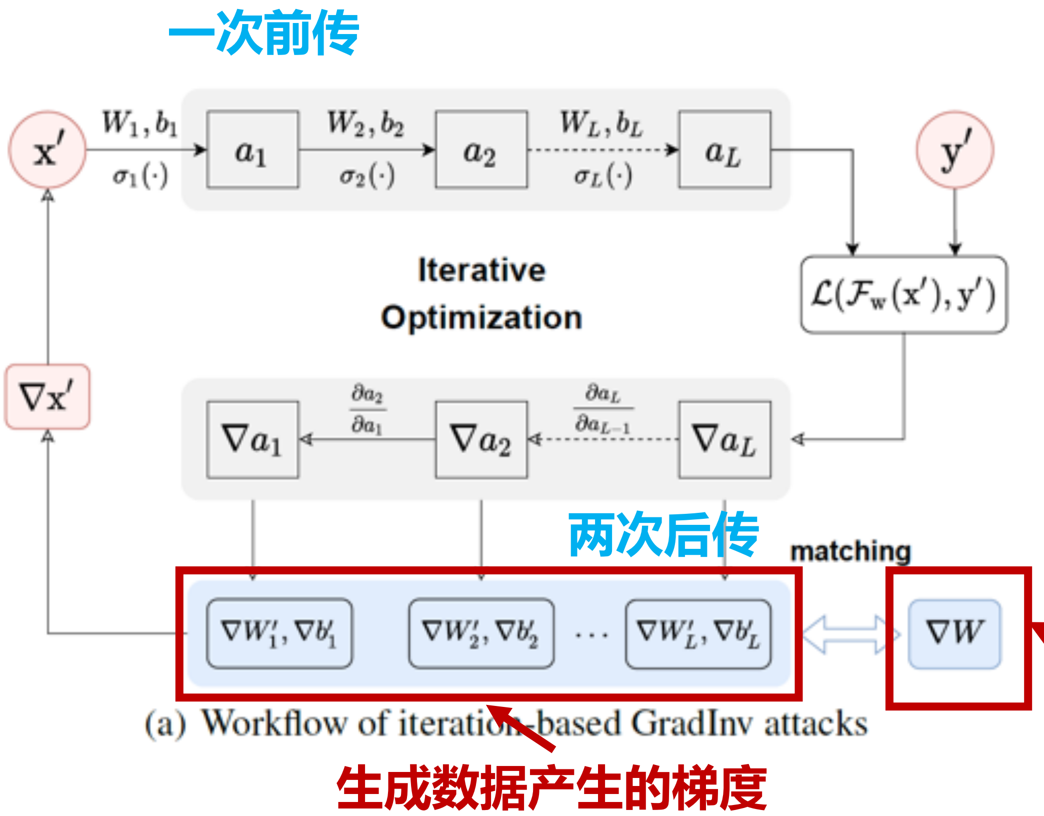
迭代逆向

（逐层）递归逆向

Zhang et al. "A Survey on Gradient Inversion: Attacks, Defenses and Future Directions." IJCAI 2022.

白盒窃取：迭代逆向

□ 迭代逆向：通过构造数据来接近真实梯度



(a) Workflow of iteration-based GradInv attacks

□关键点：

- 如何初始化x'
- Batch大小
- 模型大小
- 图像分辨率大小
- 有时需要梯度分拆

Zhang et al. "A Survey on Gradient Inversion: Attacks, Defenses and Future Directions." IJCAI 2022.

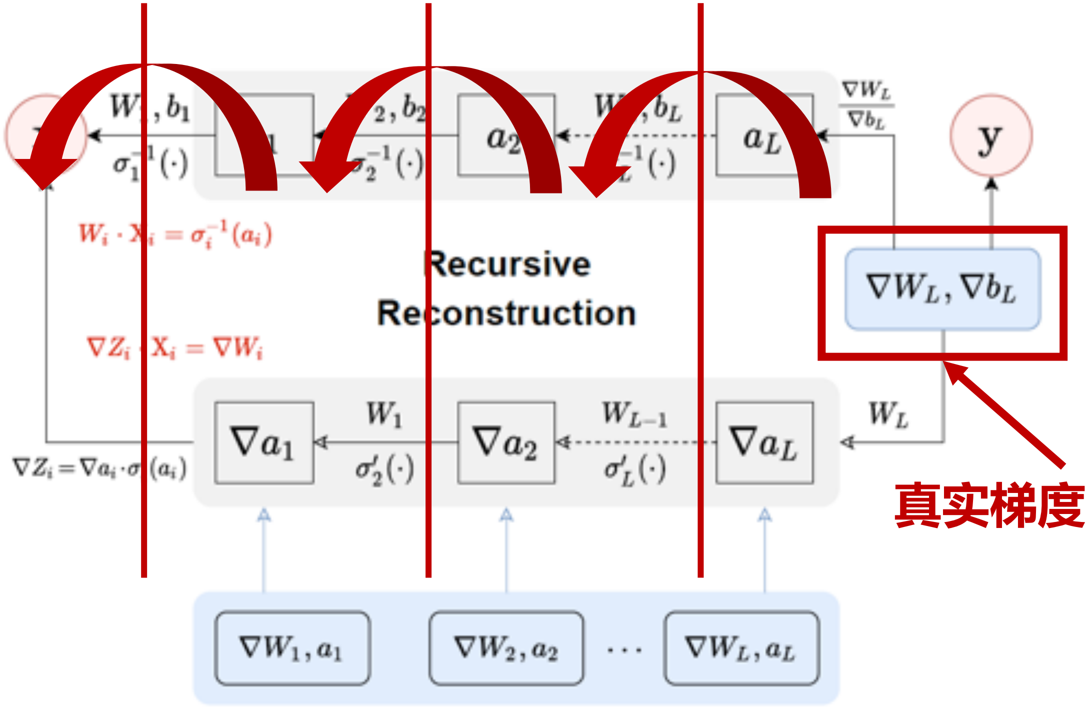
白盒窃取：迭代逆向

□ 已有工作汇总

Publication	Data Initialization		Model Training		Grad Matching		Additional
	Distribution	*Resolution*	*Network*	*Batch size*	*Loss-fn*	*Optimizer*	
GradInv attacks of iteration-based framework							
DLG [Zhu *et al.*, 2019]	Gaussian	64×64	LeNet	8	ℓ_2 dist	L-BFGS	–
iDLG [Zhao *et al.*, 2020]	Uniform$^{\mathrm{L}}$	32×32	LeNet	1	ℓ_2 dist	L-BFGS	–
CPL [Wei *et al.*, 2020b]	Geometric	128×128	LeNet	8	ℓ_2 dist	L-BFGS	$\mathcal{R}_{\mathrm{y}}$ regularizer
InvGrad [Geiping *et al.*, 2020]	Gaussian$^{\mathrm{L}}$	224×224	ResNet$^{\mathrm{T}}$	8 (100)	Cosine	Adam	$\mathcal{R}_{\mathrm{TV}}$ regularizer
SAPAG [Wang *et al.*, 2020]	Constant	224×224	ResNet$^{\mathrm{T}}$	8	Gauss	AdamW	Gaussian kernel
GradInversion [Yin *et al.*, 2021]	Gaussian$^{\mathrm{L}}$	224×224	ResNet$^{\mathrm{T}}$	48	ℓ_2 dist	Adam	$\mathcal{R}_{\mathrm{fidel}} + \mathcal{R}_{\mathrm{group}}$
GradDisagg [Lam *et al.*, 2021]	Gaussian	32×32	MLP	32 (128)	ℓ_2 dist	L-BFGS	Participant info
GradAttack [Huang *et al.*, 2021]	Gaussian$^{\mathrm{L}}$	224×224	ResNet$^{\mathrm{T}}$	128	Cosine	Adam	No BN + labels
Bayesian [Balunović *et al.*, 2022]	Gaussian	32×32	ConvNet$^{\mathrm{T}}$	1 (32)	Cosine	Adam	Known $p(g\|x)$
CAFE [Jin *et al.*, 2021]	Uniform	32×32	Loop-Net	100	ℓ_2 dist	SGD	In Vertical-FL
GIAS [Jeon *et al.*, 2021]	Latent	64×64	ResNet$^{\mathrm{T}}$	4	Cosine	Adam	GAN-based

白盒窃取：递归逆向

递归逆向：基于真实梯度追层逆向推导



(b) Workflow of recursion-based GradInv attacks

$$\begin{cases} W_i \cdot \mathbf{x}_i = Z_i \\ \nabla Z_i \cdot \mathbf{x}_i = \nabla W_i \end{cases}$$

关键点：

- 图像大小（32x32）
- Batch大小（大多为1）
- 模型大小

Zhang et al. “A Survey on Gradient Inversion: Attacks, Defenses and Future Directions.” IJCAI 2022.

白盒窃取：递归逆向

□ 已有工作汇总

Publication	Data Initialization		Model Training		Grad Matching		Additional
	Distribution	*Resolution*	*Network*	*Batch size*	*Loss-fn*	*Optimizer*	
GradInv attacks of recursion-based framework							
PPDL-AHE [Phong *et al.*, 2018]	N/A	20×20	MLP	1	Gradient division		–
PPDL-SPN [Fan *et al.*, 2020]	N/A	32×32	ConvNet	8	Linear solving		Noise analysis
R-GAP [Zhu and Blaschko, 2021]	N/A	32×32	ConvNet	1	Inverse matrix		Rank analysis
COPA [Chen and Campbell, 2021]	N/A	32×32	ConvNet	1	Least-squares		Pull-back const

L The labels can be directly identified or extracted from shared gradients.
T The results of data recovery are compared in different model training states.

白盒防御

□ 已有工作汇总

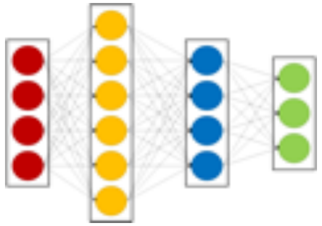
Category	Method	Publication	Key Contribution
Original Data	MixUp	[Zhang *et al.*, 2018]	Data enhancement by linearly combining the inputs
	InstaHide	[Huang *et al.*, 2020]	Encrypt the MixUp data with one-time secret keys
	Pixelization	[Fan, 2018; Fan, 2019]	Perturb the raw data with pixelization-based method
Training Model	Dropout	[Zheng, 2021]	Add an additional dropout layer before the classifier
	Local iters	[Wei *et al.*, 2020b]	Share gradients after multiple local training iterations
	Architecture	[Zhu and Blaschko, 2021]	Reduce the number of convolutional kernels properly
Shared Gradients	Aggregation	[Zhang *et al.*, 2020]	Apply Homomorphic Encryption to protect gradients
		[Lia and Togan, 2020]	Utilize Secure Multi-Party Computation to aggregate
	Perturbation	[Sun *et al.*, 2021]	Perturb data representation layer and maintain utility
		[Wei *et al.*, 2021]	Add adaptive noise with differential privacy guarantee
	Compression	[Vogels *et al.*, 2019]	Compress the smaller values in gradients to zero
		[Karimireddy *et al.*, 2019]	Transmit the sign of gradients for model updates

Zhang et al. “A Survey on Gradient Inversion: Attacks, Defenses and Future Directions.” IJCAI 2022.

This Week

- Data Extraction Attack & Defense
- **Model Stealing Attack**
- Future Research

AI 模型训练代价高昂

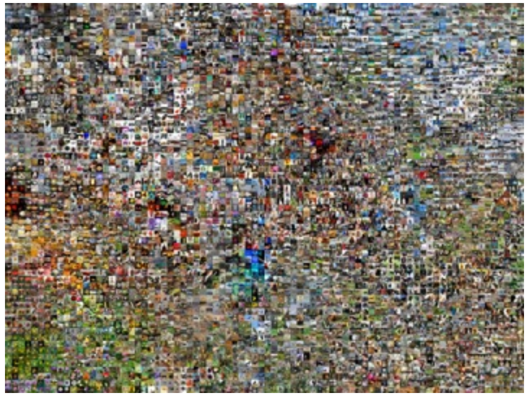


Google

$160万

BERT

大规模、高性能的AI模型训练耗费巨大



数据资源

计算资源

人力资源

模型窃取的动机

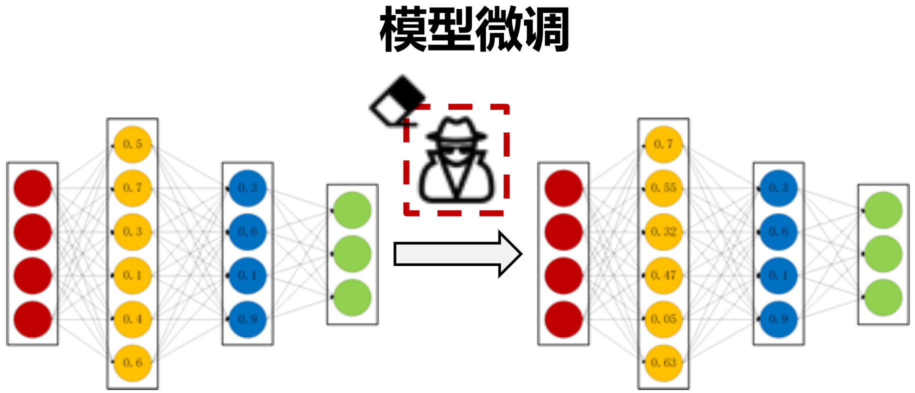
模型窃取

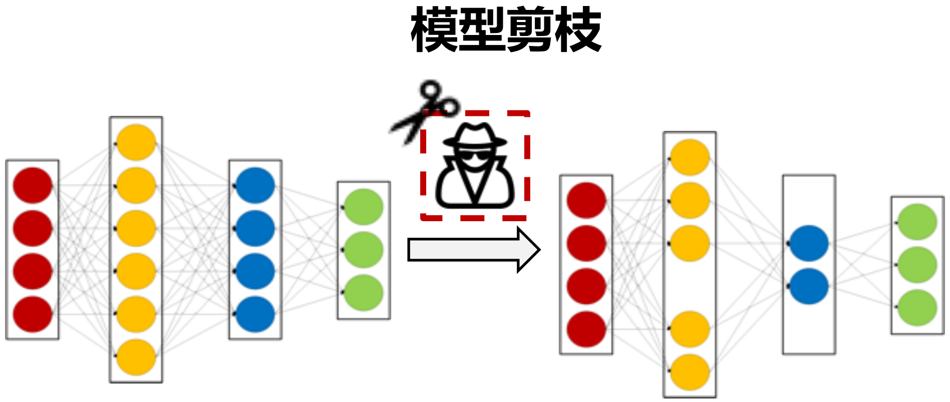
为其所用

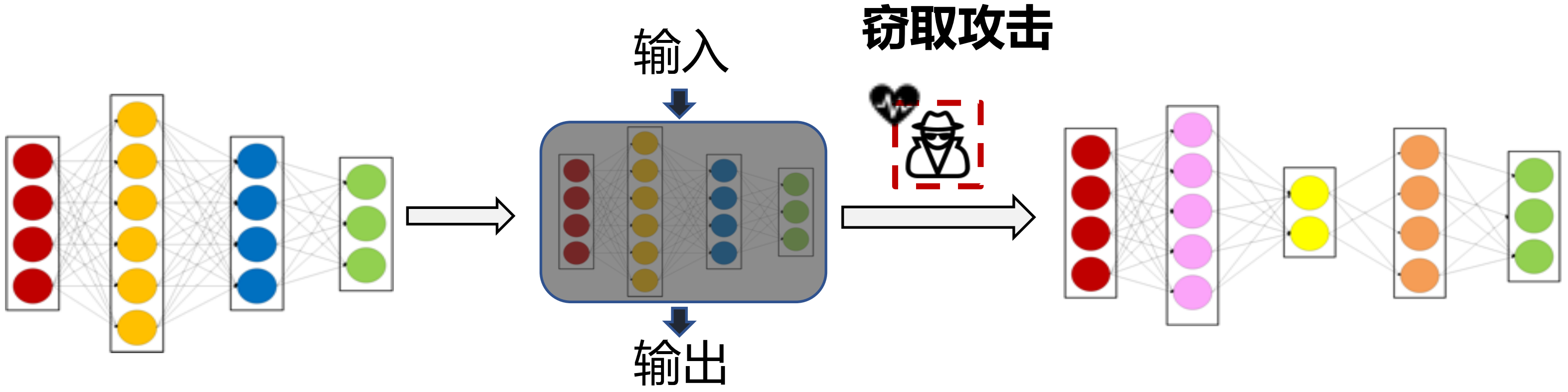
宝贵的 AI 模型

- 巨大的商业价值
- 尽量保持模型性能
- 不希望被发现

模型窃取的方式



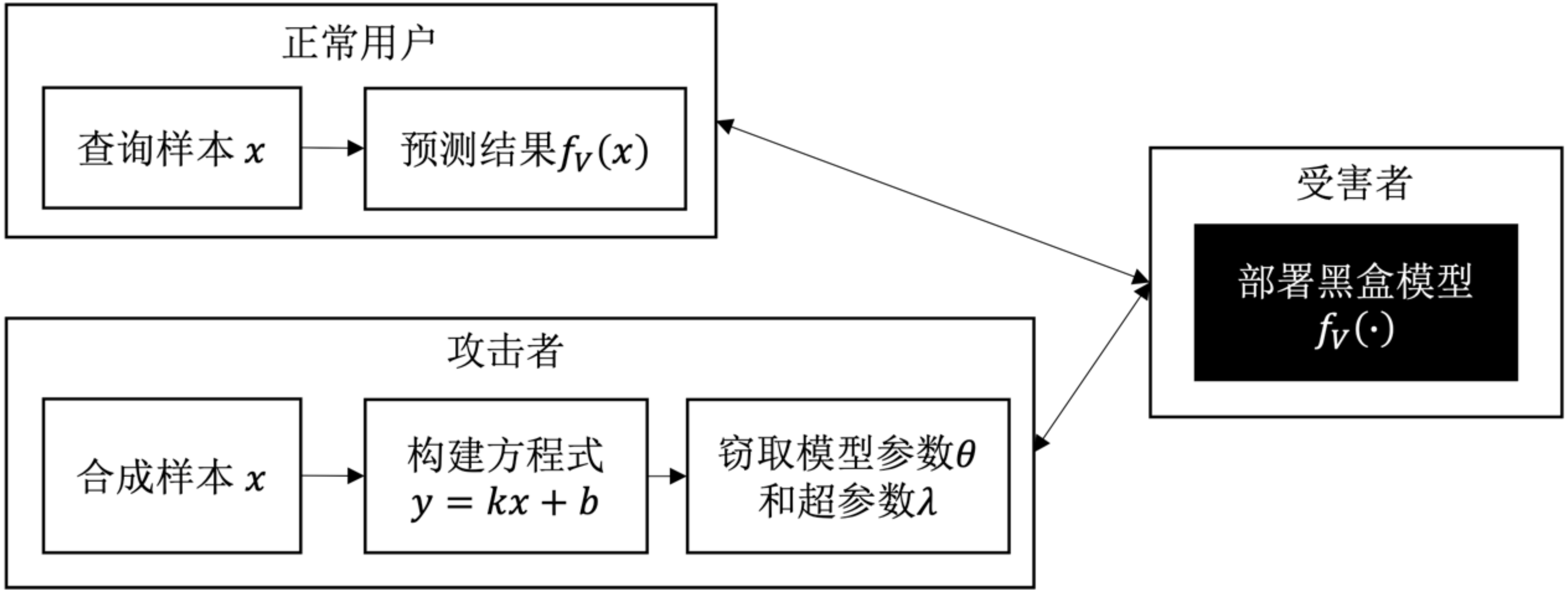




Stealing machine learning models via prediction APIs, USENIX Security, 2016; Practical black-box attacks against machine learning, ASIACCS, 2017; Knockoff nets: Stealing functionality of black-box models, CVPR, 2019; Maze: Data-free model stealing attack using zeroth-order gradient estimation, CVPR, 2021;

基于方程式求解的攻击

□ 攻击思路示例



基于方程式求解的攻击

□ 100%窃取某些商业模型所需的查询数和时间

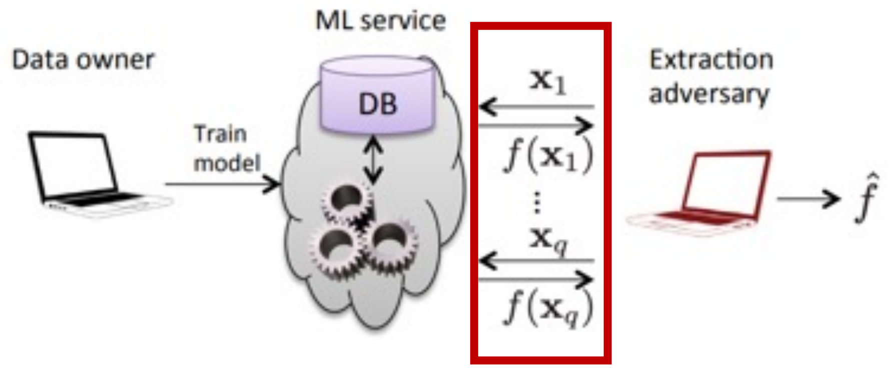
Service	Model Type	Data set	Queries	Time (s)
Amazon	Logistic Regression	Digits	650	70
	Logistic Regression	Adult	1,485	149
BigML	Decision Tree	German Credit	1,150	631
	Decision Tree	Steak Survey	4,013	2,088

Service	White-box	Monetize	Confidence Scores	Logistic Regression	SVM	Neural Network	Decision Tree
Amazon [1]	✗	✗	✓	✓	✗	✗	✗
Microsoft [38]	✗	✗	✓	✓	✓	✓	✓
BigML [11]	✓	✓	✓	✓	✗	✗	✓
PredictionIO [43]	✓	✗	✗	✓	✓	✗	✓
Google [25]	✗	✓	✓	✓	✓	✓	✓

Tramèr, Florian, et al. "Stealing machine learning models via prediction {APIs}." *USENIX Security*, 2016.

基于方程式求解的攻击：窃取参数

攻击算法



- **参数个数为d**
- **通过d+1个输入，构造d+1个下列方程**

$$\theta^{\top}\boldsymbol{x} = \sigma^{-1}(f(\boldsymbol{x}))$$

- **求解方程得到θ**

主要特点：

- 针对传统机器学习模型:SVM、LR、DT
- 可精确求解，需要模型返回精确的置信度
- 窃取得到的模型还可能泄露训练数据（数据逆向攻击）

Tramèr, Florian, et al. "Stealing machine learning models via prediction {APIs}." *USENIX Security*, 2016.

基于方程式求解的攻击：窃取超参

□ 攻击思想：模型训练完了的状态应该是Loss梯度为0

$$\boxed{\mathcal{L}(\theta) = \mathcal{L}(\boldsymbol{x}, y, \theta) + \lambda R(\theta)}$$

窃取超参数λ

$$\frac{\partial \mathcal{L}(\theta)}{\partial \theta} = \boldsymbol{b} + \lambda \boldsymbol{a} = 0$$

$$\boldsymbol{b} = \begin{bmatrix} \frac{\partial \mathcal{L}(\boldsymbol{x}, y, \theta)}{\partial \theta_1} \\ \frac{\partial \mathcal{L}(\boldsymbol{x}, y, \theta)}{\partial \theta_2} \\ \vdots \\ \frac{\partial \mathcal{L}(\boldsymbol{x}, y, \theta)}{\partial \theta_n} \end{bmatrix}, \quad \boldsymbol{a} = \begin{bmatrix} \frac{\partial R(\theta)}{\partial \theta_1} \\ \frac{\partial R(\theta)}{\partial \theta_2} \\ \vdots \\ \frac{\partial R(\theta)}{\partial \theta_n} \end{bmatrix}$$

$$\hat{\lambda} = -(\boldsymbol{a}^\top \boldsymbol{a})^{-1} \boldsymbol{a}^\top \boldsymbol{b}.$$

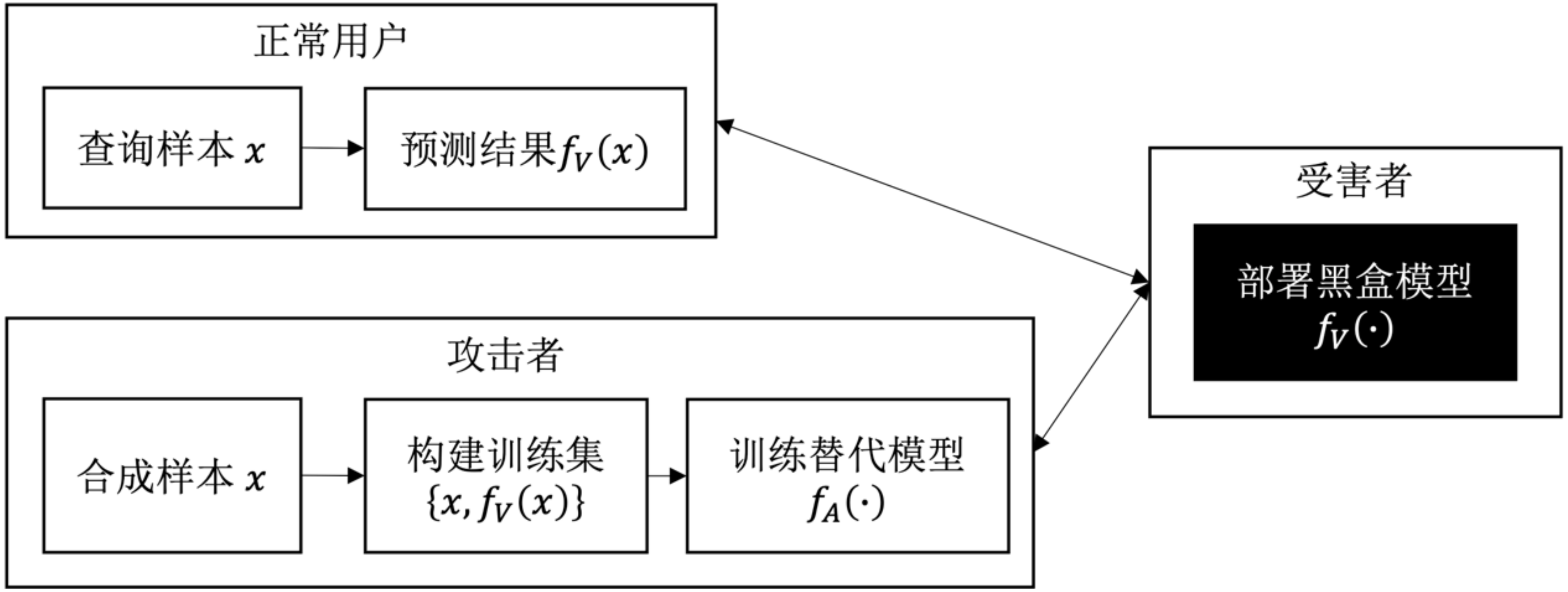
□ 主要特点：

- 需要知道Loss的形式
- 需要在所有数据上做矩阵运算
- R只与模型参数$\boldsymbol{\theta}$有关

Wang, Binghui, and Neil Zhenqiang Gong. "Stealing hyperparameters in machine learning." S&P, 2018.

基于替代模型的攻击

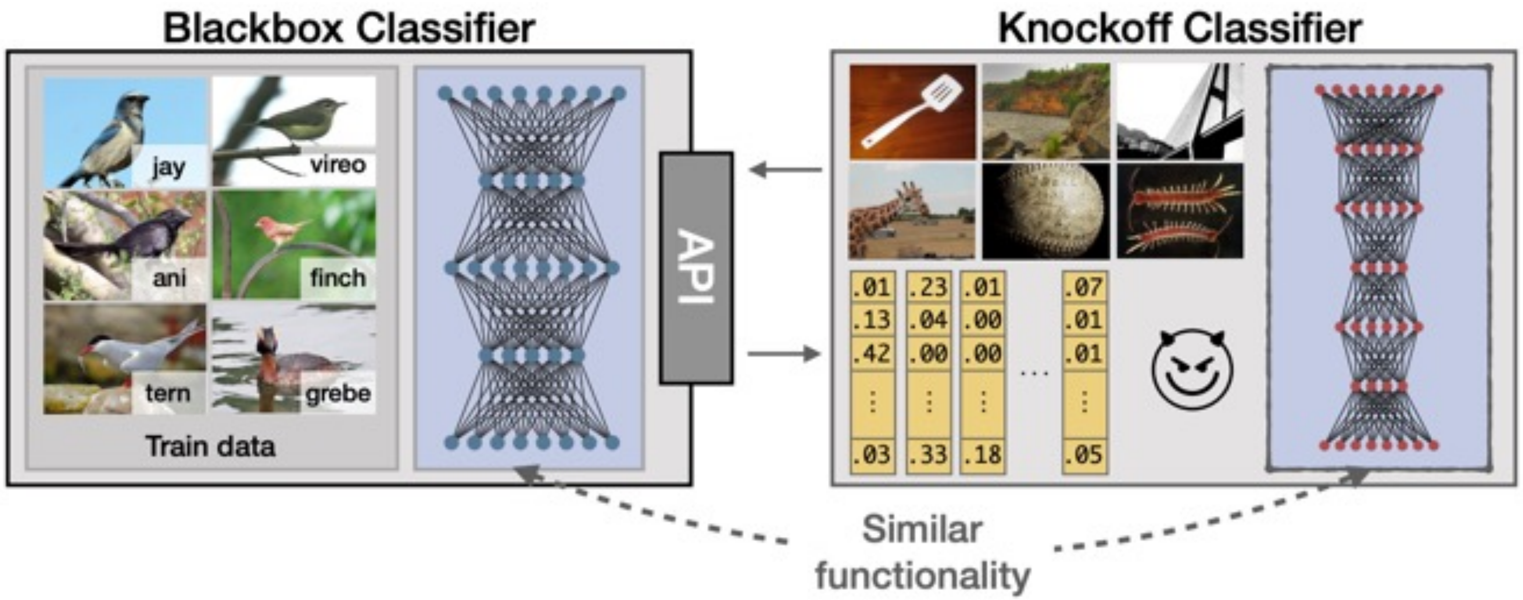
□ 攻击思想：在查询目标模型的过程中训练一个替代模型模拟其行为



Orekondy et al. "Knockoff nets: Stealing functionality of black-box models." *CVPR*, 2019.

基于替代模型的攻击

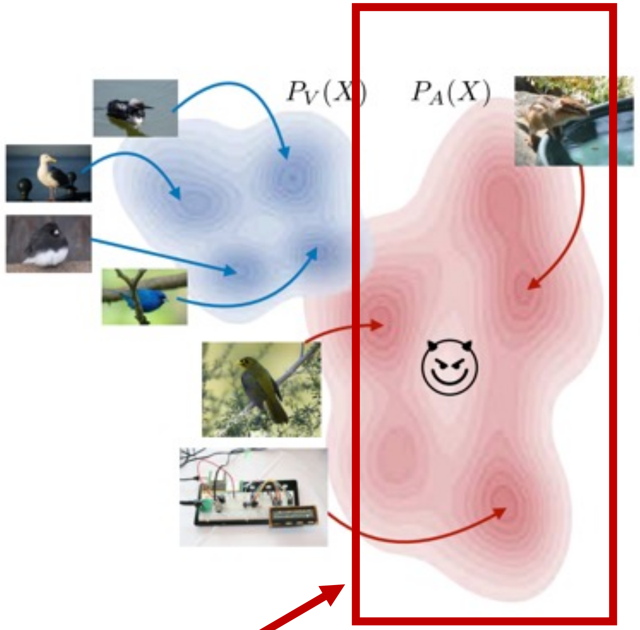
□ Knockoff Nets 攻击： “仿冒网络”



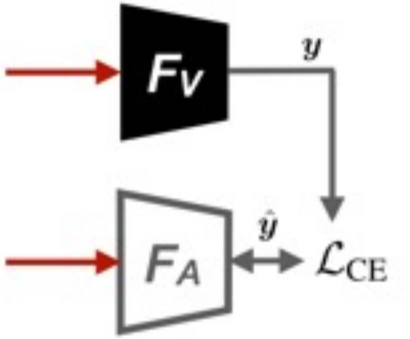
Orekondy et al. "Knockoff nets: Stealing functionality of black-box models." *CVPR*, 2019.

基于替代模型的攻击

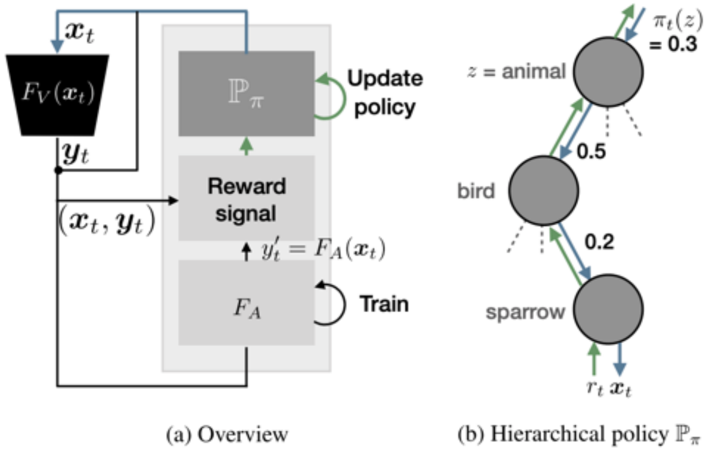
□ Knockoff Nets 攻击：攻击流程



采样大量查询样本



训练替代模型

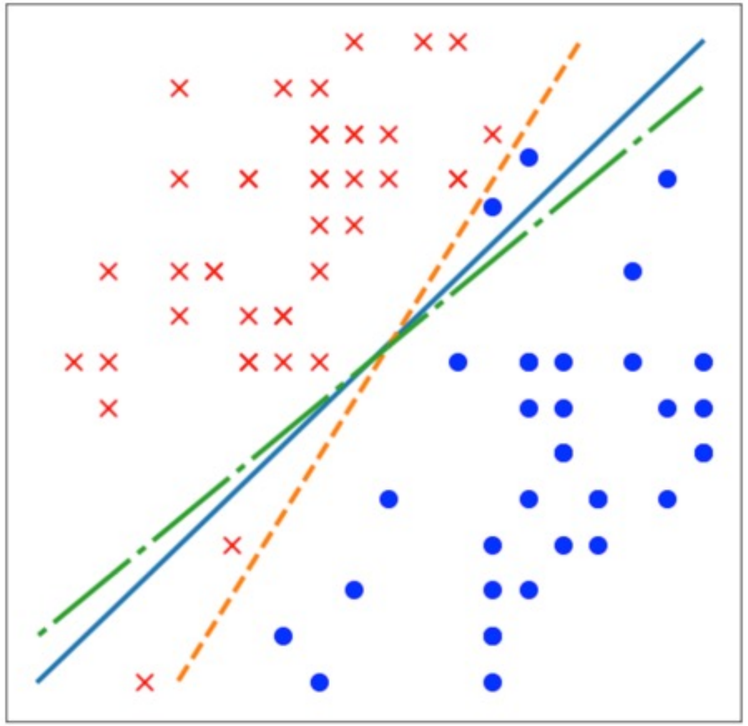


强化学习，学习如何高效选择样本

Orekondy et al. "Knockoff nets: Stealing functionality of black-box models." *CVPR*, 2019.

基于替代模型的攻击

- **高准确（accuracy）vs 高保真（fidelity）窃取攻击**

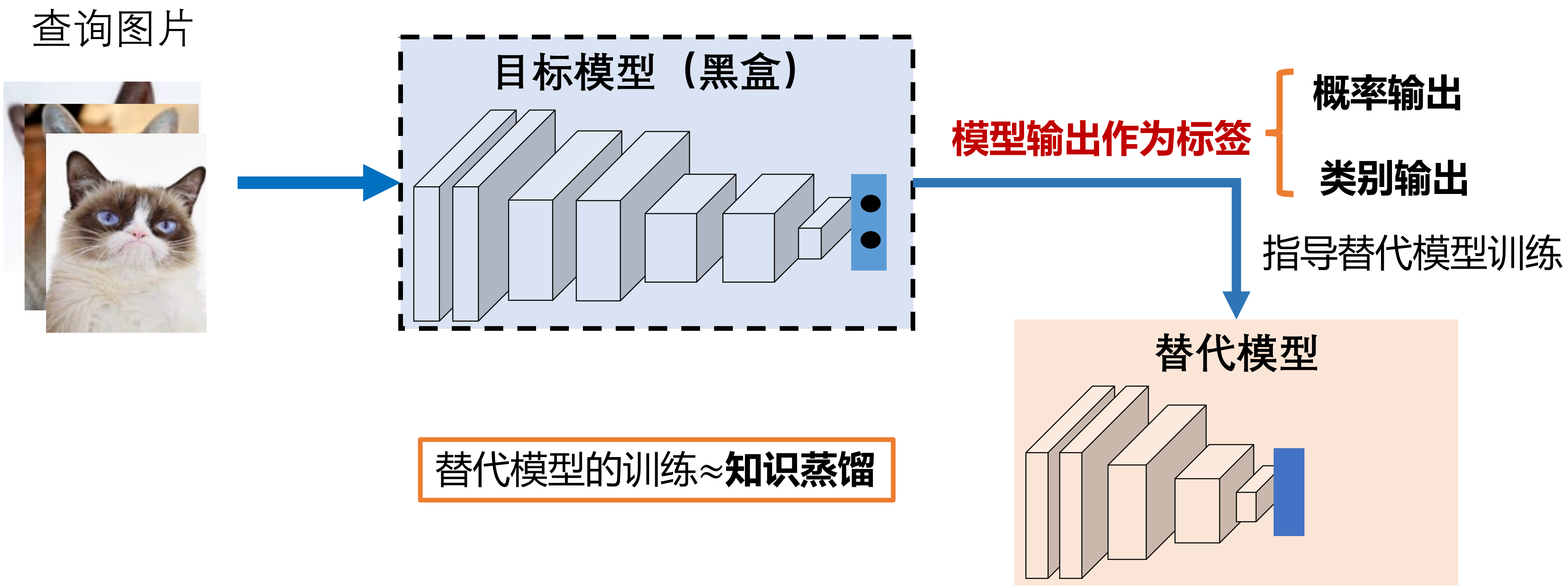


- 蓝色：目标决策边界
- 橙色：高准确窃取
- 绿色：高保真窃取

Jagielski, Matthew, et al. “High accuracy and high fidelity extraction of neural networks.” USENIX Security, 2020.

基于替代模型的攻击

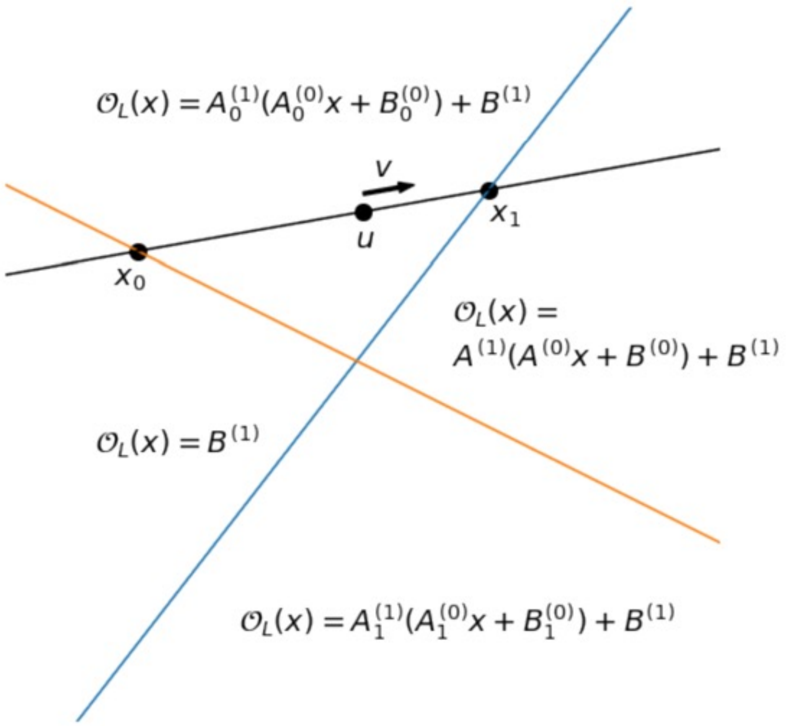
□ 高准确（accuracy）vs 高保真（fidelity）窃取攻击



Jagielski, Matthew, et al. “High accuracy and high fidelity extraction of neural networks.” USENIX Security, 2020.

基于替代模型的攻击

□ 功能等同窃取Functionally Equivalent Extraction



□ 攻击步骤：

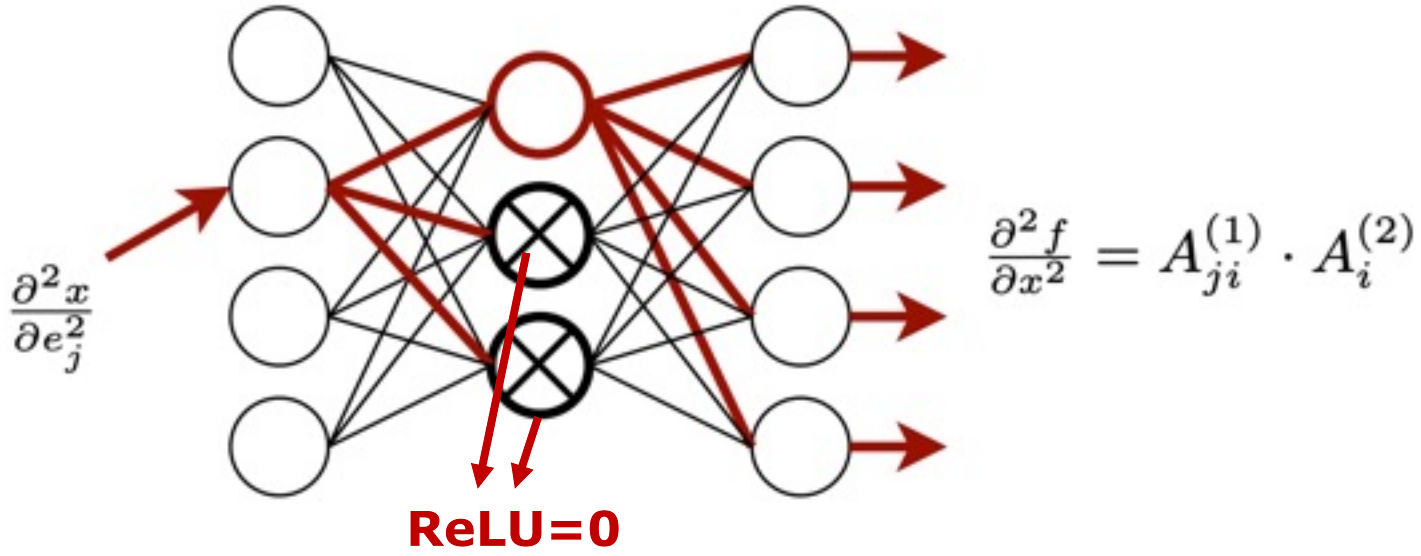
- 寻找在某个Neuron上，让ReLU=0的关键点
- 在关键点两侧探索边界，确定对应权重
- 只能窃取两层网络

Symbol	Definition
d	Input dimensionality
h	Hidden layer dimensionality ($h < d$)
K	Number of classes
$A^{(0)} \in \mathbb{R}^{d \times h}$	Input layer weights
$B^{(0)} \in \mathbb{R}^{h}$	Input layer bias
$A^{(1)} \in \mathbb{R}^{h \times K}$	Logit layer weights
$B^{(1)} \in \mathbb{R}^{K}$	Logit layer bias

Jagielski, Matthew, et al. "High accuracy and high fidelity extraction of neural networks." USENIX Security, 2020.

基于替代模型的攻击

□ 加密分析窃取Cryptanalytic Extraction



□ **思想**：ReLU的二级导为0+有限差分（finite difference）

Carlini et al. "Cryptanalytic extraction of neural network models." *Annual International Cryptology Conference*, 2020.

基于替代模型的攻击

□ 加密分析窃取 Cryptanalytic Extraction

窃取0-deep神经网络：

$$f(\boldsymbol{x}) = \boldsymbol{w}^{(1)} \cdot \boldsymbol{x} + b^{(1)}$$

$$f(\boldsymbol{x} + \boldsymbol{e}_i) - f(\boldsymbol{x}) = \boldsymbol{w}^{(1)} \cdot (\boldsymbol{x} + \boldsymbol{e}_i) - \boldsymbol{w}^{(1)} \cdot \boldsymbol{x} = \boldsymbol{w}^{(1)} \cdot \boldsymbol{e}_i$$

窃取1-deep神经网络：

$$f(\boldsymbol{x}) = \boldsymbol{w}^{(2)} ReLU(\boldsymbol{w}^{(1)}\boldsymbol{x} + b^{(1)}) + b^{(2)}$$

$$\alpha_+^i = \frac{\partial f(\boldsymbol{x})}{\partial \epsilon \boldsymbol{e}_i} \Big|_{\boldsymbol{x}=\boldsymbol{x}+\epsilon \boldsymbol{e}_i}$$

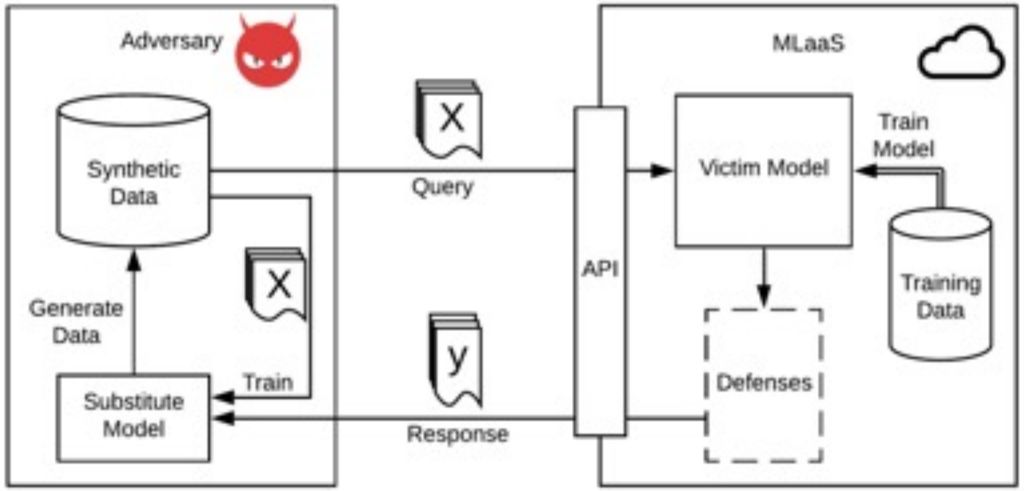
$$\alpha_-^i = \frac{\partial f(\boldsymbol{x})}{\partial \epsilon \boldsymbol{e}_i} \Big|_{\boldsymbol{x}=\boldsymbol{x}-\epsilon \boldsymbol{e}_i}$$

$$\frac{\alpha_+^k - \alpha_-^k}{\alpha_+^i - \alpha_-^i} = \frac{\boldsymbol{w}_{j,k}^{(1)}}{\boldsymbol{w}_{j,i}^{(1)}}$$

Carlini et al. "Cryptanalytic extraction of neural network models." *Annual International Cryptology Conference*, 2020.

基于替代模型的攻击

□ 估计合成攻击 Estimation Synthesis (ES) Attack



思想：初始化合成数据集，然后根据模型返回训练替代模型

- **E-step**：在合成数据上知识蒸馏更新替代模型
- **S-step**：合成数据，使用对抗生成网络

□ 特点：

- 不需要原始训练数据或先验
- 不需要目标模型先验

Yuan, Xiaoyong, et al. “ES attack: Model stealing against deep neural networks without data hurdles.” 2022.

基于替代模型的攻击

□ ES攻击算法：蒸馏+生成的结合

Algorithm 1 *ES Attack*

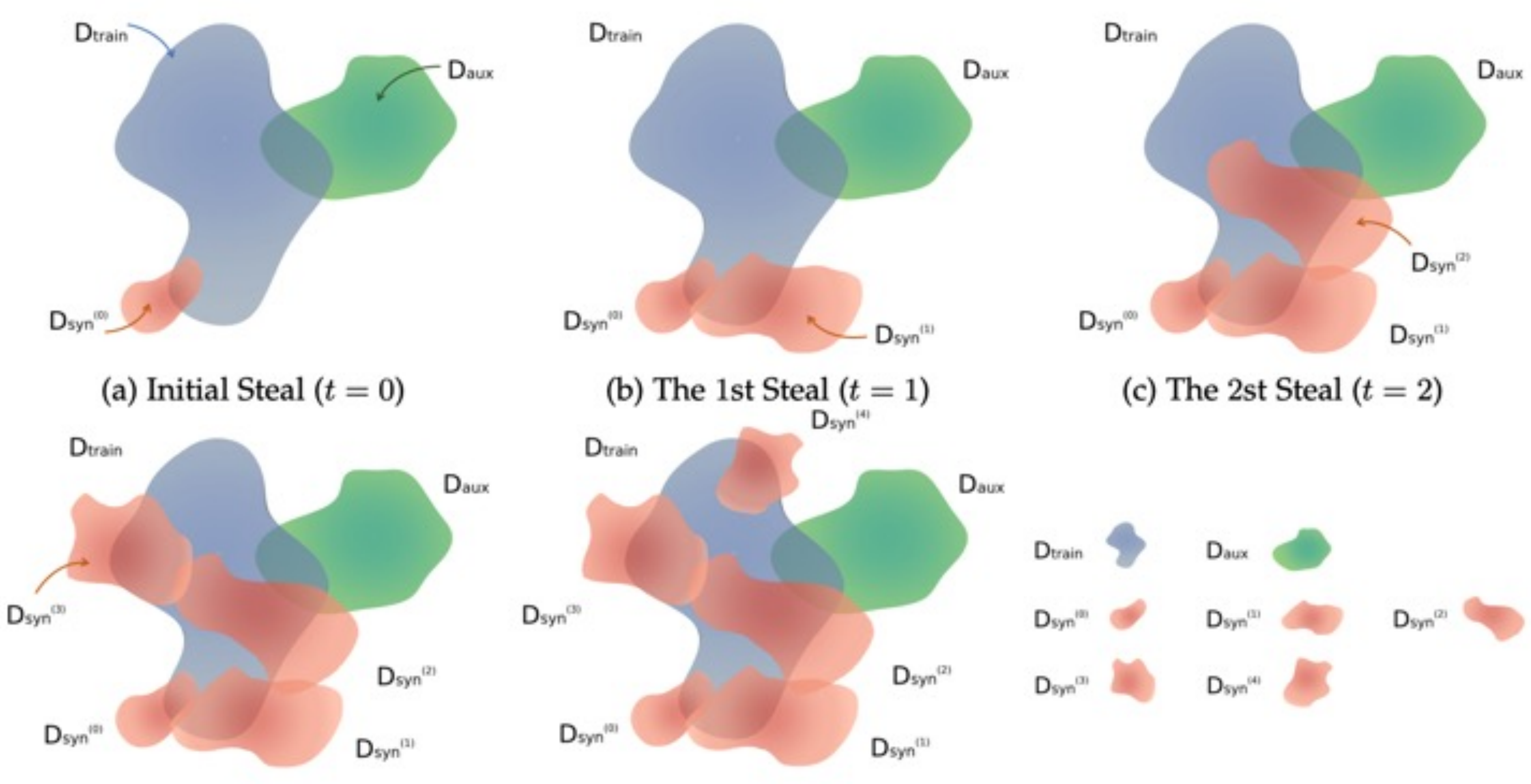
INPUT:
- The black-box victim model f_v
- Number of classes K
- Number of stealing epochs N
- Number of training epochs for each stealing epoch M

OUTPUT:
- The substitute model $f_s^{(N)}$

1: Initialize a synthetic dataset $\mathcal{D}_{\text{syn}}^{(0)}$ by randomly sampling $\boldsymbol{x}$ from a Gaussian distribution.
2: Construct an initial substitute model $f_s^{(0)}$ by initializing the parameters in the model.
3: **for** $t \leftarrow 1$ to N **do**
4:     **E-Step:** Estimate the parameters in the substitute model $f_s^{(t)}$ using knowledge distillation for M epochs on the synthetic dataset $\mathcal{D}_{\text{syn}}^{(t-1)}$.
5:     **S-Step:** Synthesize a new dataset $\mathcal{D}_{\text{syn}}^{(t)}$ based on the knowledge of the substitute model $f_s^{(t)}$.
6: **end for**
7: **return** $f_s^{(N)}$.

基于替代模型的攻击

□ ES攻击合成的数据分布

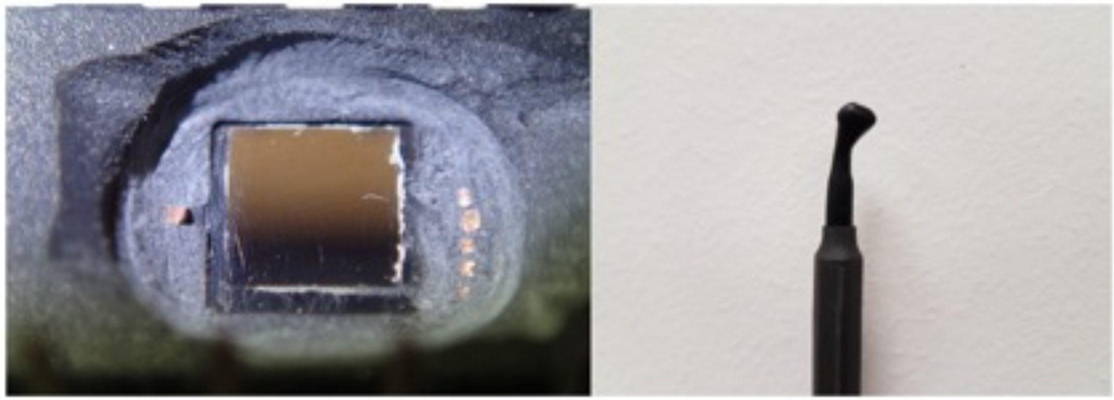


(a) Initial Steal ($t = 0$)
(b) The 1st Steal ($t = 1$)
(c) The 2st Steal ($t = 2$)
(d) The 3rd Steal ($t = 3$)
(e) The 4th Steal ($t = 4$)

train:原始训练数据
aux:公共数据集
syn:合成数据集

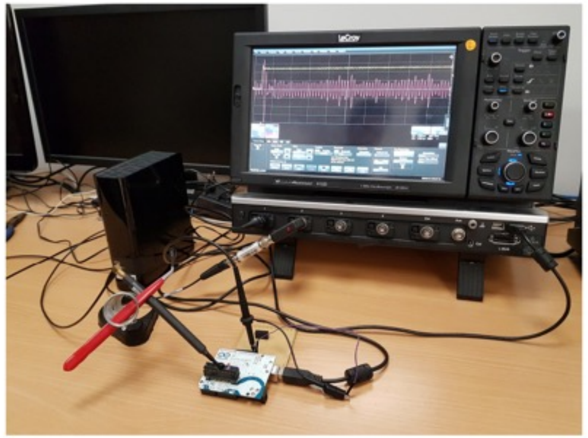
基于侧信道攻击的窃取

☐ 侧信道（side-channel）攻击窃取神经网络



(a) Target 8-bit microcontroller mechanically decapsulated

(b) Langer RF-U 5-2 Near-field Electromagnetic passive Probe



(c) The complete measurement setup

通过探测运行神经网络的微处理器的电力使用情况，来窃取神经网络的权重

Batina et al. CSI neural network: Using side-channels to recover your artificial neural network information, USENIX Security, 2019

Future Research

- 攻击方面：
  - 更高效的攻击
  - 攻击更多的数据、更大的模型
- 防御方面：
  - 减少模型输出或梯度对信息的泄露：差分隐私
  - 以攻为守：反向渗透

C U Next Week!

Course page:

https://trustworthymachinelearning.github.io/

Textbook:

下载链接:

Email: xingjunma@fudan.edu.cn
Personal page: www.xingjunma.com
Office: 江湾校区交叉二号楼D5025