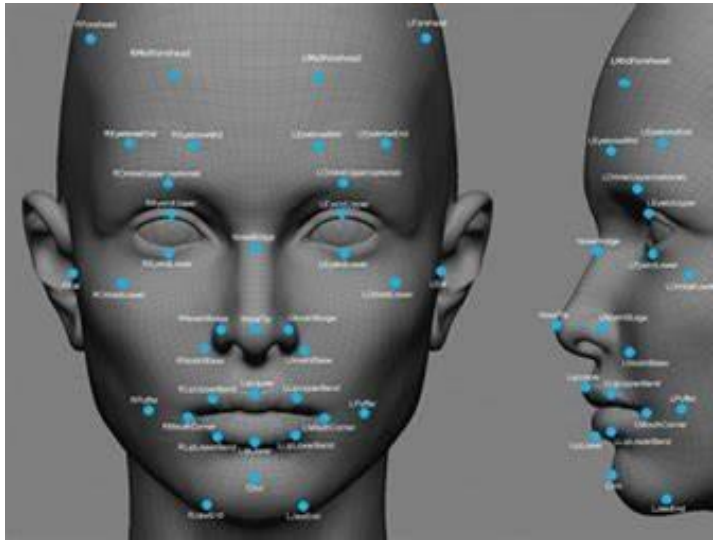
Microsoft

# 人工智能系统 System for AI

# 人工智能安全隐私
# AI Security & Privacy

# AI在安全与隐私攸关的场景的应用


自动驾驶

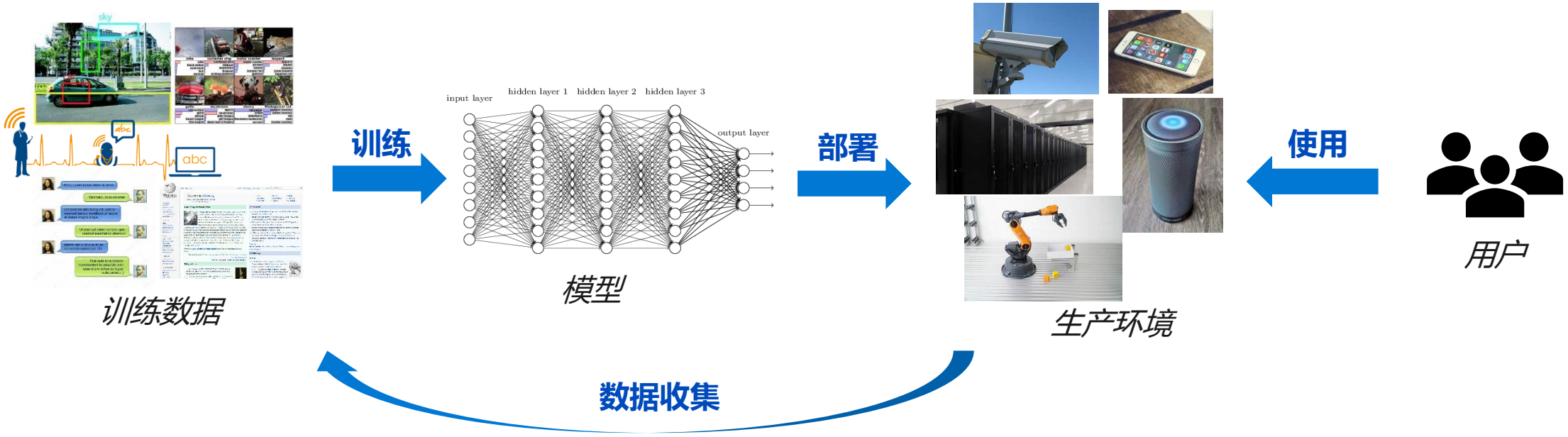


人脸识别/验证


推荐系统

- **预测结果可能非常重要**，影响人身安全、财产安全、社会公平等
- **训练数据可能非常敏感**，如人脸、浏览记录、社交关系等

# 深度学习模型的生命周期




- **安全攻击**可能发生在训练数据、模型、生产环境中
- **隐私泄露**可能发生在训练、部署、使用、数据收集阶段
- 正常的数据和模型也可能产生**不当的后果**

**WE NEED RESPONSIBLE AI!**

# 关注的安全相关性质

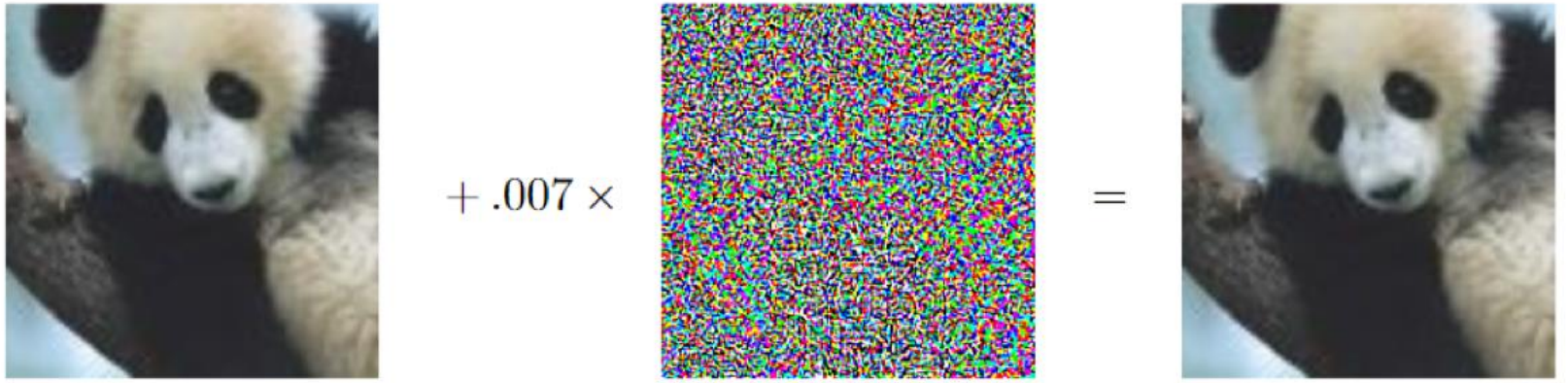
- **完整性** (Integrity): 模型输出正确的、符合预期的结果
  - 预测结果的鲁棒性 (Robustness)
  - 预测过程的可信性 (Trustworthy)
- **保密性** (Confidentiality): 训练数据、模型算法不被泄露
  - 数据保密性 (Data Confidentiality)
  - 模型保密性 (Model Confidentiality)
- **伦理** (Ethics): 应用和结果符合法律、道德、伦理
  - AI算法的公平性 (Fairness)
  - AI算法的滥用 (Misuse)

# 完整性 (Integrity)

- 预测结果的鲁棒性 (Robustness)
  - 对抗样本攻击
  - 对抗防御
- 预测过程的可信性 (Trustworthy)
  - 后门攻击
  - 运行环境攻击

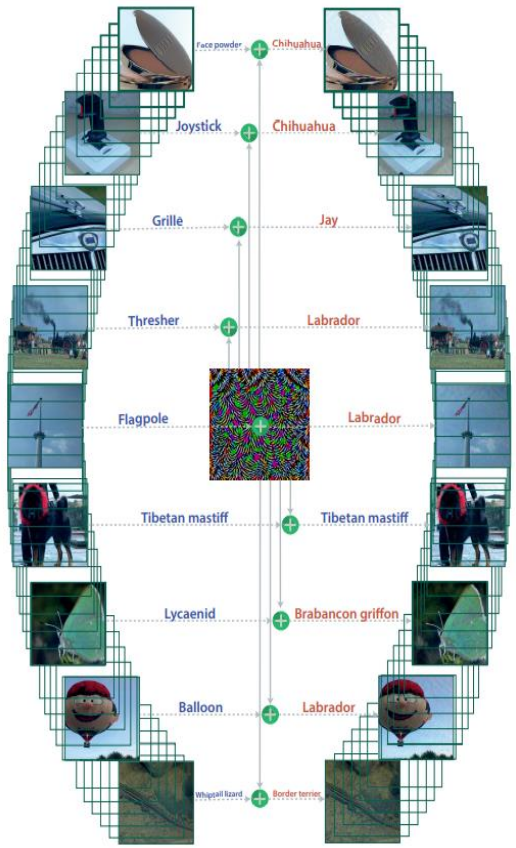
# 对抗样本 (adversarial example)

- 对正常样本增加一个微弱的（肉眼无法识别的）扰动，就能导致模型预测出错 [Szegedy ICLR2014],[Goodfellow ICLR2015]。




Fast gradient sign method (FGSM): $\boldsymbol{\eta} = \epsilon \text{sign}\left(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y)\right)$

# 对抗样本生成方法




**Algorithm 1** Computation of universal perturbations.

**input:** Data points $X$, classifier $\hat{k}$, desired $\ell_p$ norm of the perturbation $\xi$, desired accuracy on perturbed samples $\delta$.
**output:** Universal perturbation vector $v$.
Initialize $v \leftarrow 0$.
**while** $\mathrm{Err}(X_v) \leq 1 - \delta$ **do**
  **for** each datapoint $x_i \in X$ **do**
    **if** $\hat{k}(x_i + v) = \hat{k}(x_i)$ **then**
      Compute the *minimal* perturbation that sends $x_i + v$ to the decision boundary:

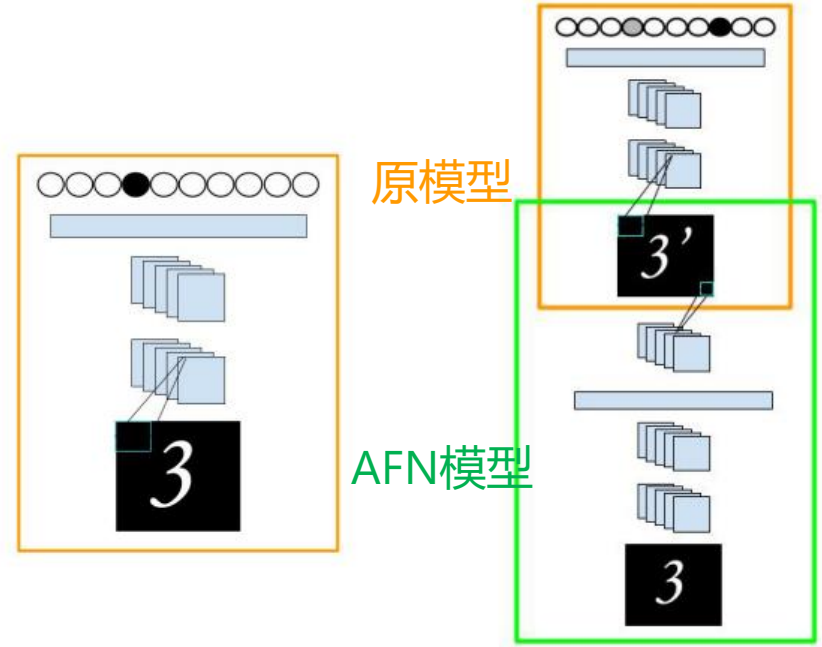
$$\Delta v_i \leftarrow \arg\min_r \|r\|_2 \text{ s.t. } \hat{k}(x_i + v + r) \neq \hat{k}(x_i).$$

      Update the perturbation:

$$v \leftarrow \mathcal{P}_{p,\xi}(v + \Delta v_i).$$

    **end if**
  **end for**
**end while**

迭代地生成通用的对抗攻击扰动
Universal adversarial perturbations
[Moosavi CVPR2017]




训练一个输入转换模型$g$，自动将正常样本转换为对抗样本。 $g_{f,\boldsymbol{\theta}}(\mathbf{x}) : \mathbf{x} \in \mathcal{X} \rightarrow \mathbf{x}'$

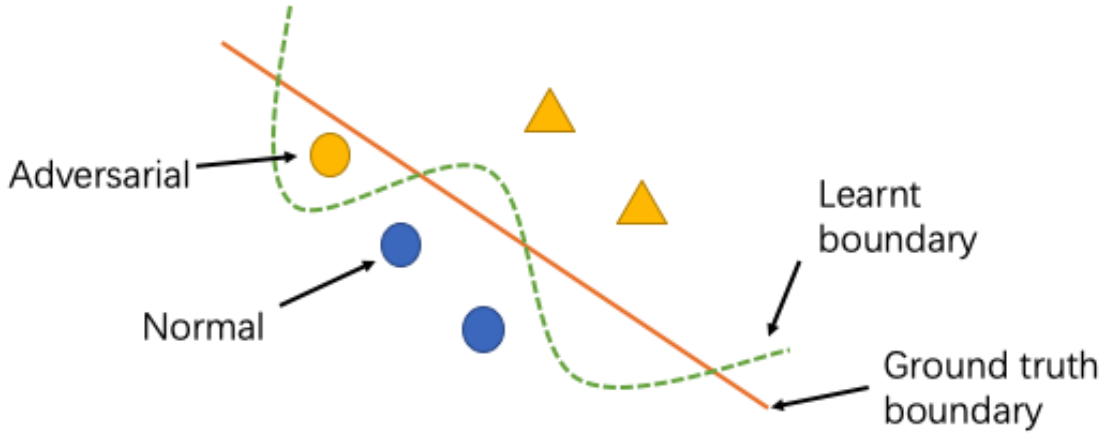
Loss function:

$$\underset{\boldsymbol{\theta}}{\operatorname{argmin}} \sum_{\mathbf{x}_i \in \mathcal{X}} \beta L_{\mathcal{X}}(g_{f,\boldsymbol{\theta}}(\mathbf{x}_i), \mathbf{x}_i) + L_{\mathcal{Y}}(f(g_{f,\boldsymbol{\theta}}(\mathbf{x}_i)), f(\mathbf{x}_i))$$

Adversarial Transformation Networks (AFN)
[Baluja AAAI2018]

# 对抗样本现象的解释

- 有观点认为其来源于神经网络学到了一个过拟合的决策边界 (decision boundary)。也有认为是神经网络的线性特性导致了误差的累积。

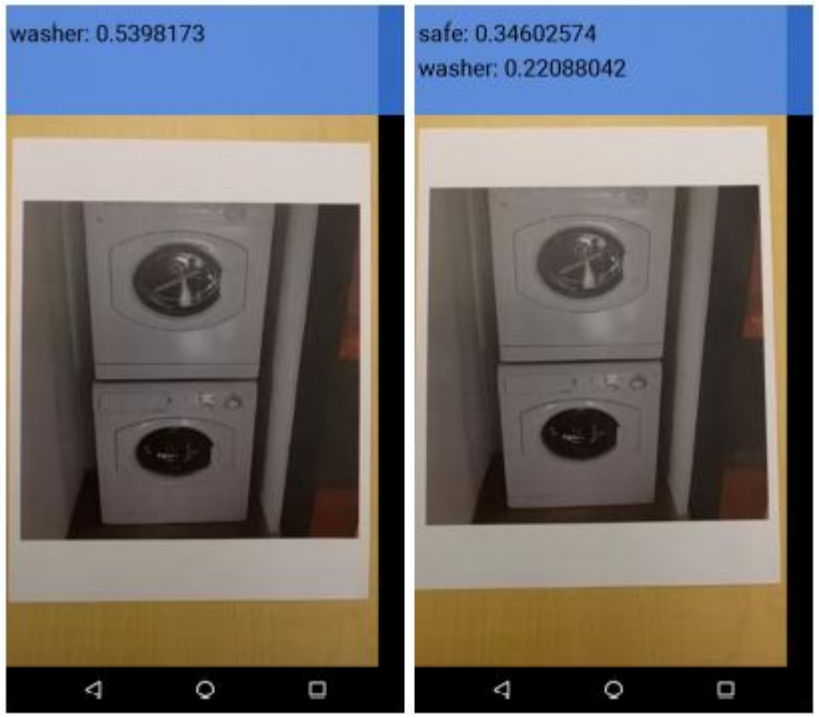



$$\boldsymbol{w}^\top \tilde{\boldsymbol{x}} = \boldsymbol{w}^\top \boldsymbol{x} + \boldsymbol{w}^\top \boldsymbol{\eta}.$$

对抗样本预测结果　原样本　累积误差

- 对抗样本的迁移性：在一个模型中找到的对抗样本很可能在另一个模型中也会导致误分类，且误分类的类别一样。（黑盒攻击）

# 物理世界的对抗样本




(b) Clean image (c) Adv. image, $\epsilon = 4$

Printed adversarial samples
to fool image classifier
[Kurakin 2017]


Adversarial stickers to
fool object detection
[Evtimov 2017]


AdvHat to fool
face ID system
[KomKov 2017]

# 其他领域的对抗样本

Q: Where is the plane?

Benign image

Answer: Runway

Fooling VQA

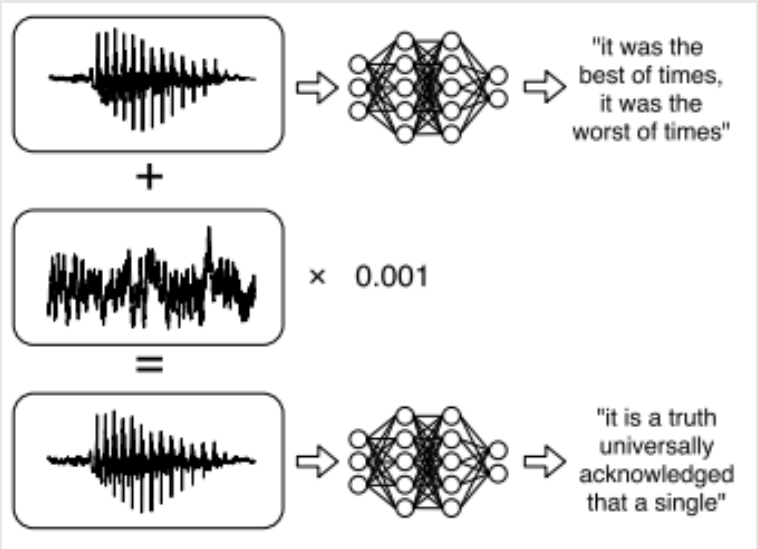
Target: Sky

Adversarial example

Sky

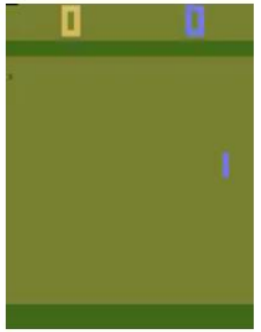
Fooling video Q&A

[Fukui 2016]

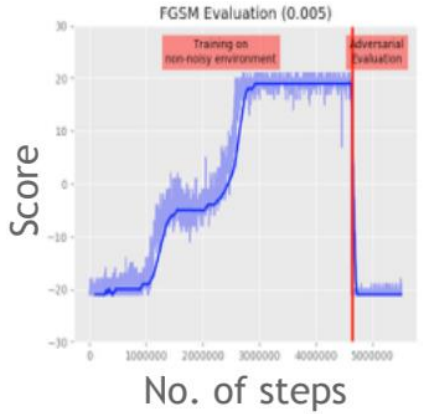



Fooling speech-to-text

[Carlini 2018]



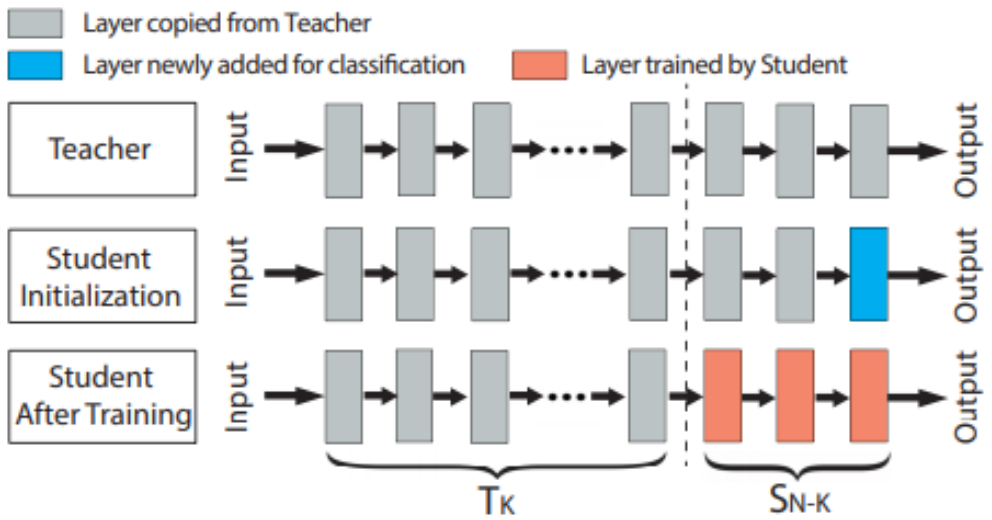
Original Frames with Adversarial Perturbation




Fooling RL agent

[Kos 2017]

# 针对迁移学习的对抗样本




迁移学习（transfer learning）

- 实际应用中常用的一种模型训练方式。
- 通过复用已训练好的模型（teacher model）的大部分参数，可以使用更少的数据更快地训练出符合特定应用场景的模型。

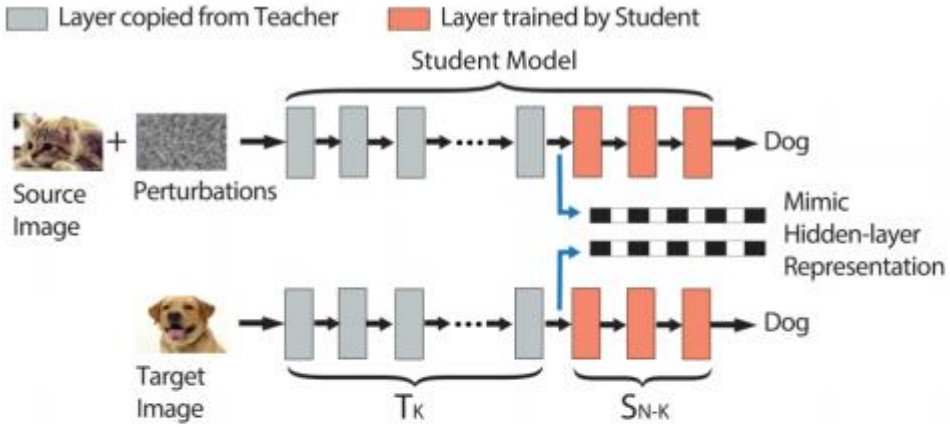



Figure 2: Illustration of our attack. Given images of a cat and a dog, attacker computes perturbations that mimic the internal representation of the dog image at layer $K$. If the calculations are perfect, the adversarial sample will be classified as dog, regardless of unknown layers in $S_{N-K}$.

针对迁移学习的攻击 [Wang Security2018]

- 攻击者可以根据teacher模型，生成可迁移的对抗样本，使之在student模型上也有攻击效果。
- 方法：使对抗样本产生与目标图片相似的中间表示

# 对抗防御方法

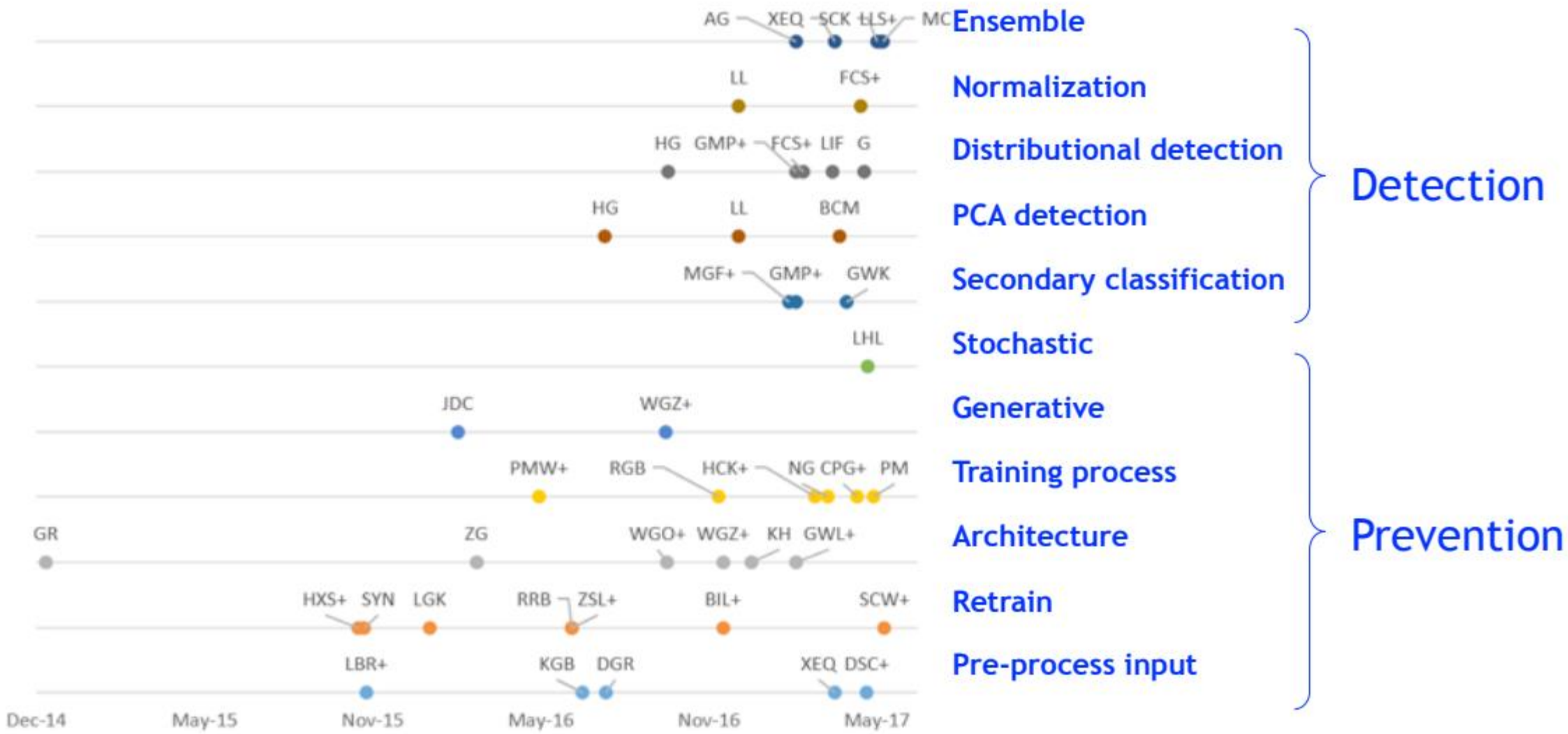



**Image from Dawn Song's talk** https://qconsf.com/sf2017/system/files/presentation-slides/dawn-qcon-nov-20172.pdf

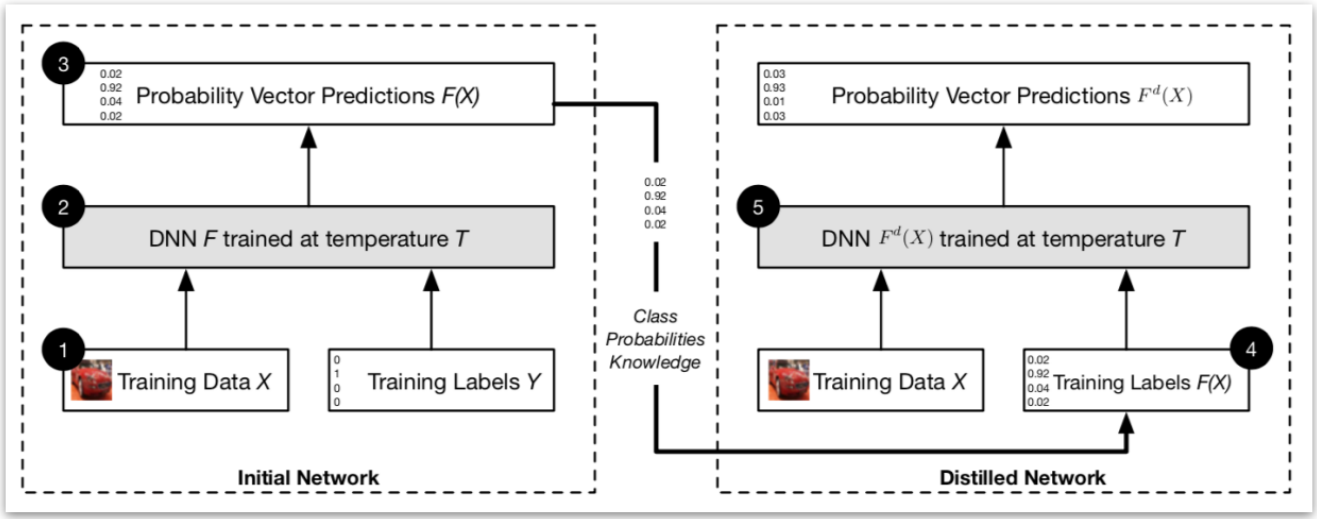
# 对抗防御

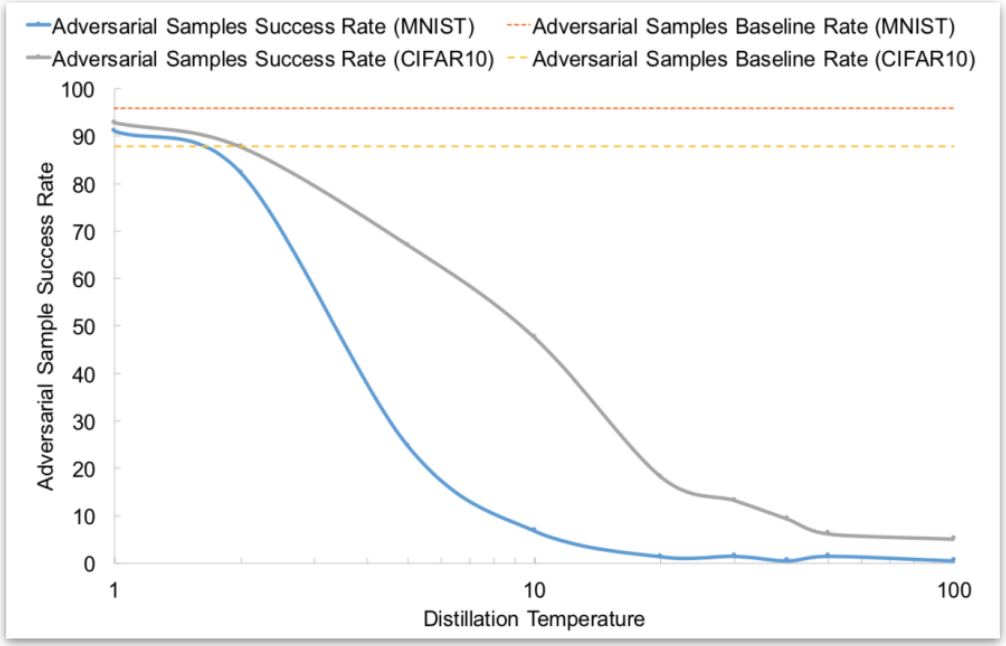
- 对抗训练：
  - 在训练时加入对抗样本 [Goodfellow ICLR2015]
- 通过蒸馏防御：
  - 利用原网络的输出训练一个能抵抗攻击的新网络 [Papernot S&P2016]
- 对模型梯度进行混淆
  - 白盒攻击需要依赖模型的梯度，因此可以对梯度进行混淆，抵御攻击 [Athalye ICML2018]
- 异常输入检测：
  - 利用模型内部的激活特征判断输入是否反常 [Metzen ICLR2017]
- 根据输入变换对结果的影响判断：
  - 对输入进行随机图片变换，检查预测结果变化剧烈程度 [Guo ICLR2018]
- 对模型输入进行形式化约束：
  - 形式化验证（SMT solver），输入在一定变化范围内时输出稳定 [Katz CAV2017]

**目前还没有完美的防御方法！**

# 以蒸馏作为防御 (distillation as defense)

[Papernot S&P2016]




Softmax with temperature T: $q_i = \frac{exp(z_i/T)}{\sum_j exp(z_i/T)}$

用原网络的输出作为label重新训练一个新的网络。
但改变网络中Softmax层的temperature参数。

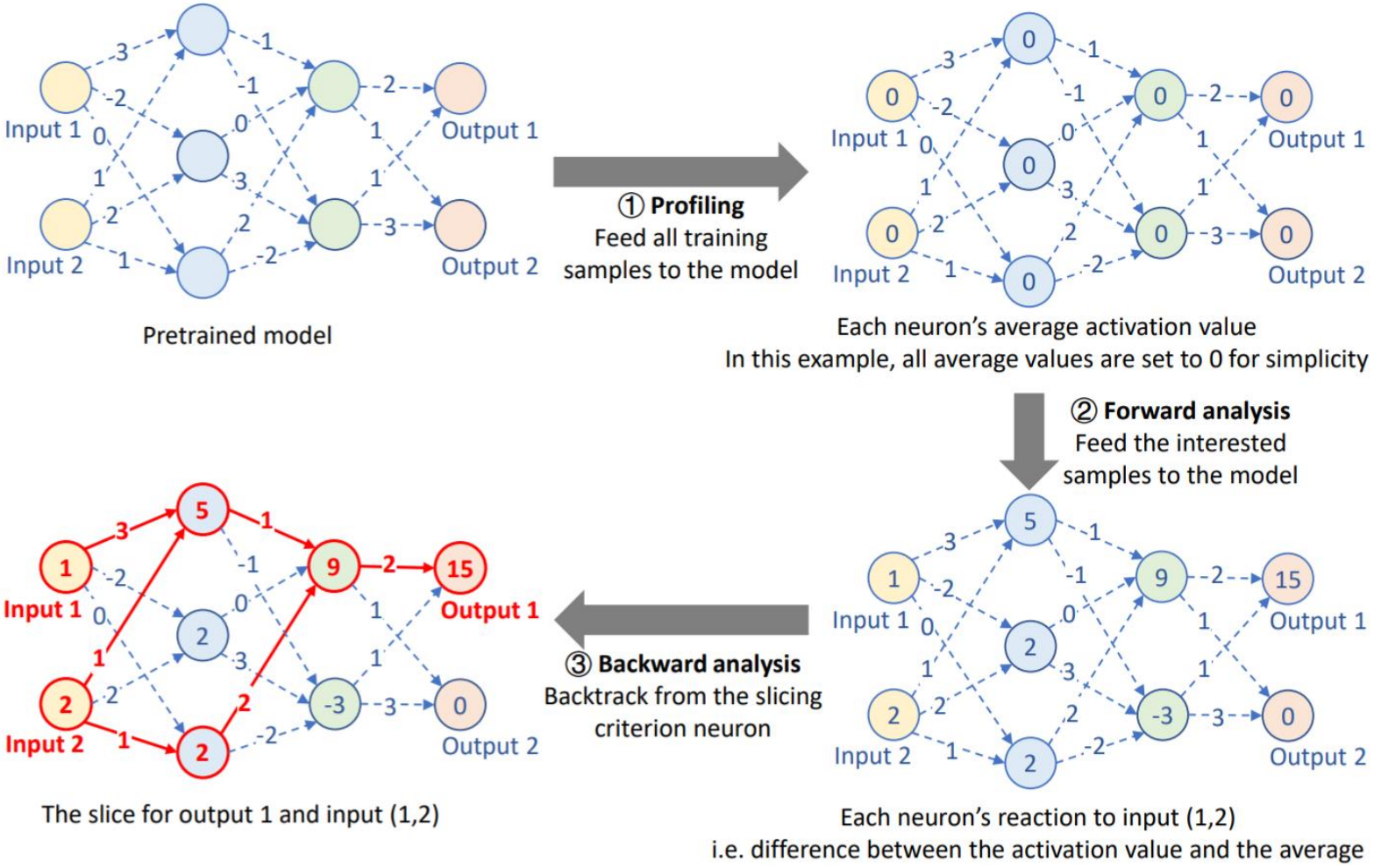
蒸馏的temperature越高，
对抗防御效果越好。

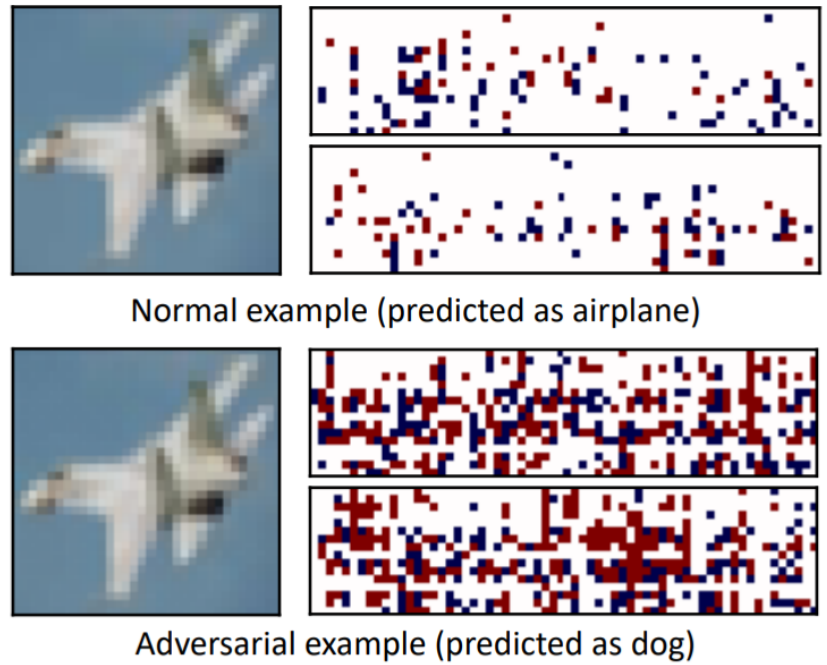
防御成功的原因：蒸馏使得网络中Softmax层的输入被缩放（因此不同输出的决策过程区别更大），同时使攻击时的梯度更小。
攻击者可以轻易破解蒸馏防御机制 [Carlini CVPR2017]。

[our work]

# 基于数据流分析的异常输入检测

通过分析neuron的激活模式，提取出对决策影响更大的neuron集合 (slice)

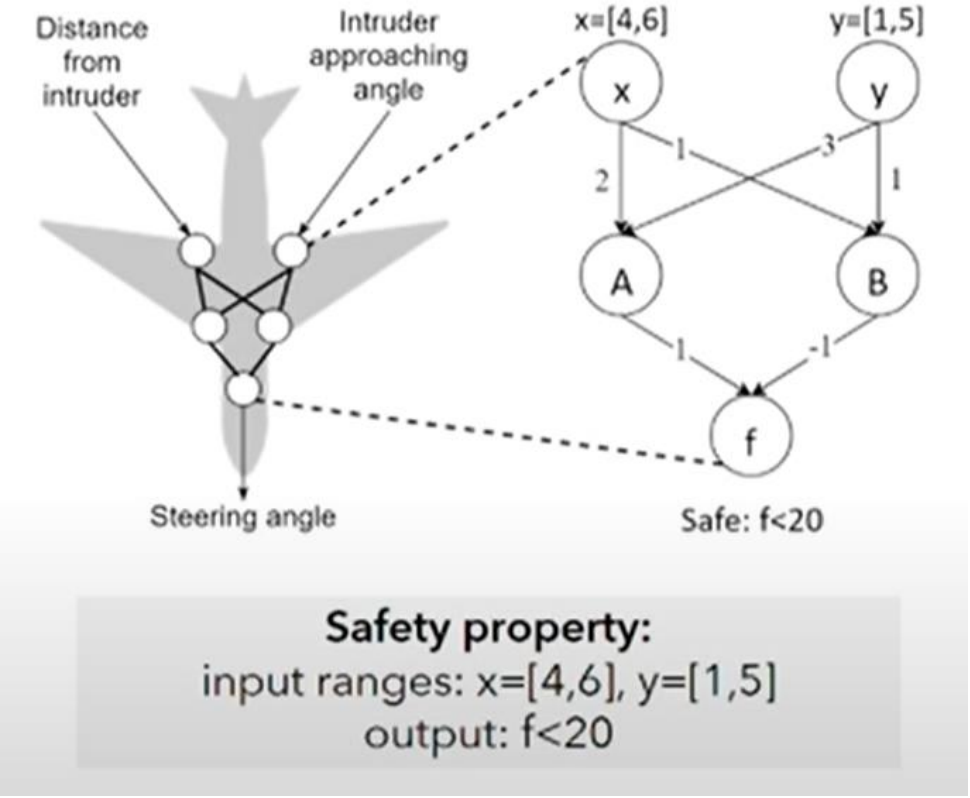


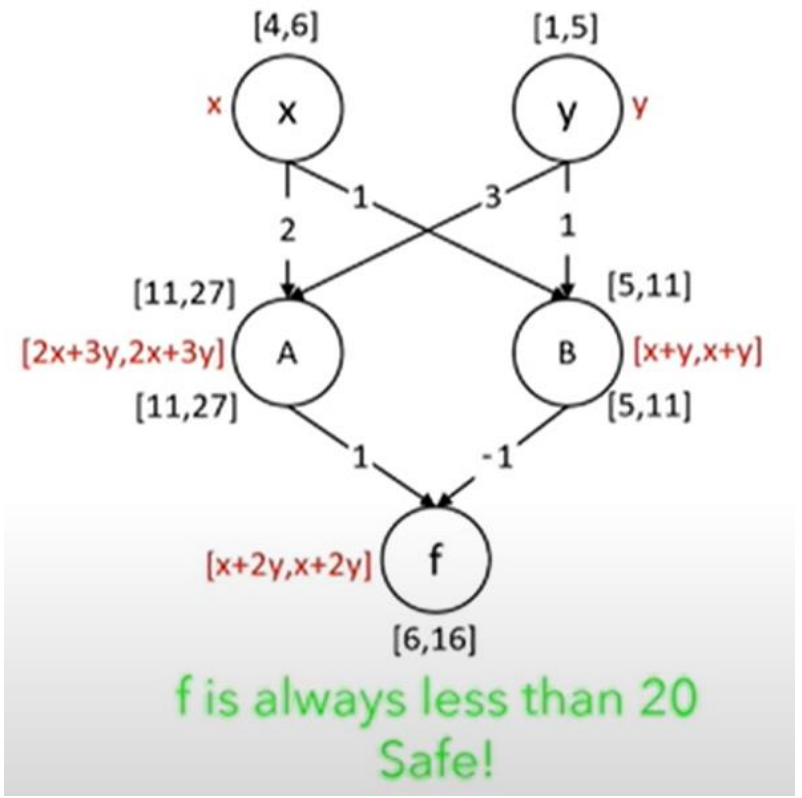




正常样本和对抗样本的slice区别很大，因此可以用于检测对抗样本。

# 形式化验证 (formal verification)

- 安全性的定义：当input在一个范围内时，output满足某性质。
- 如何从数学上证明模型的安全性?
  - SMT solver [Reluplex CAV2017], relaxation [Kolter ICML2018], interval analysis [Wang Security2018]

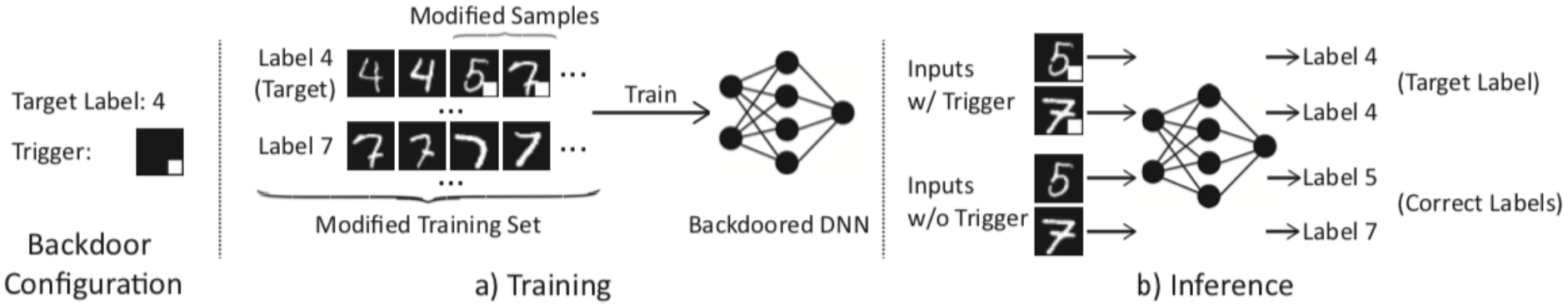


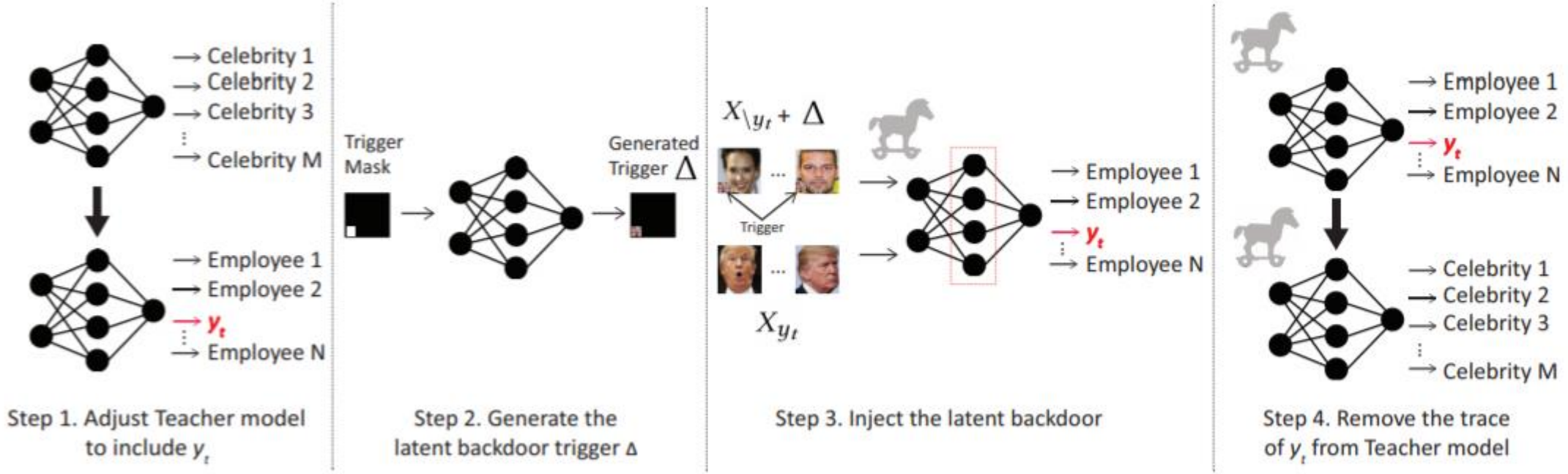

基于symbolic interval analysis (符号区间分析) 的模型安全性验证 [Wang Security2018]

# 后门攻击 (backdoor attack)

- 向模型中植入后门，使模型在输入中有特定标记时触发异常行为
- 后门植入方法：
  - [BadNets 2017] 数据投毒 (data poisoning)：在训练数据中加入恶意样本，通过训练植入后门
  - [TrojanNN NDSS2018]: 分析中间神经元，算出能引发误分类的trigger标记，生成后门训练样本

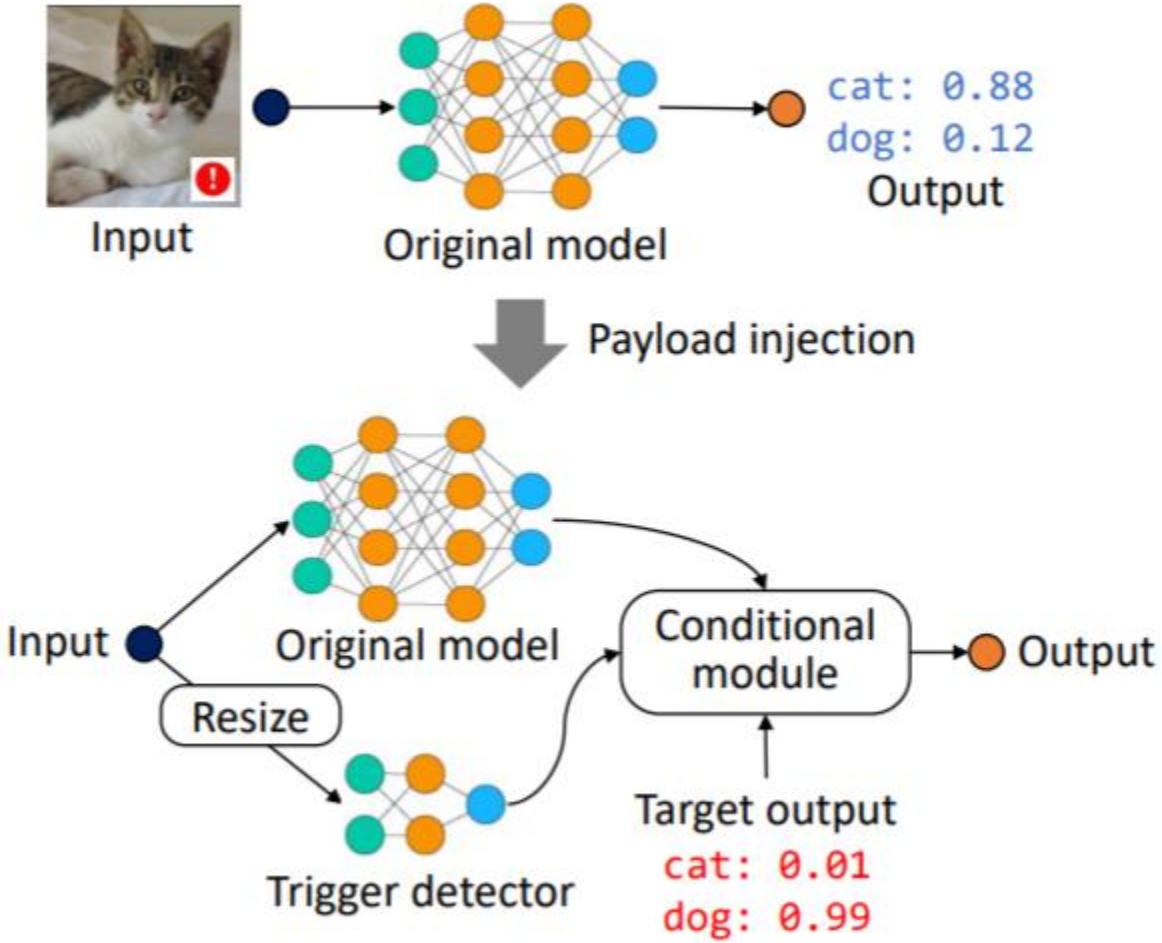
# 潜伏后门攻击 (latent backdoor attack)




- 传统的后门攻击在真实场景下不容易成功：开发者往往会采用迁移学习调整模型参数，backdoor**很可能在迁移学习过程中消除。**
- 潜伏后门攻击 [Yao CCS2019]：在teacher模型中通过**最小化**trigger**样本和目标样本的中间层差异**植入后门，并修改输出层使后门在teacher模型中不可见。当student在迁移学习时引入了目标标签，后门会被重新激活。

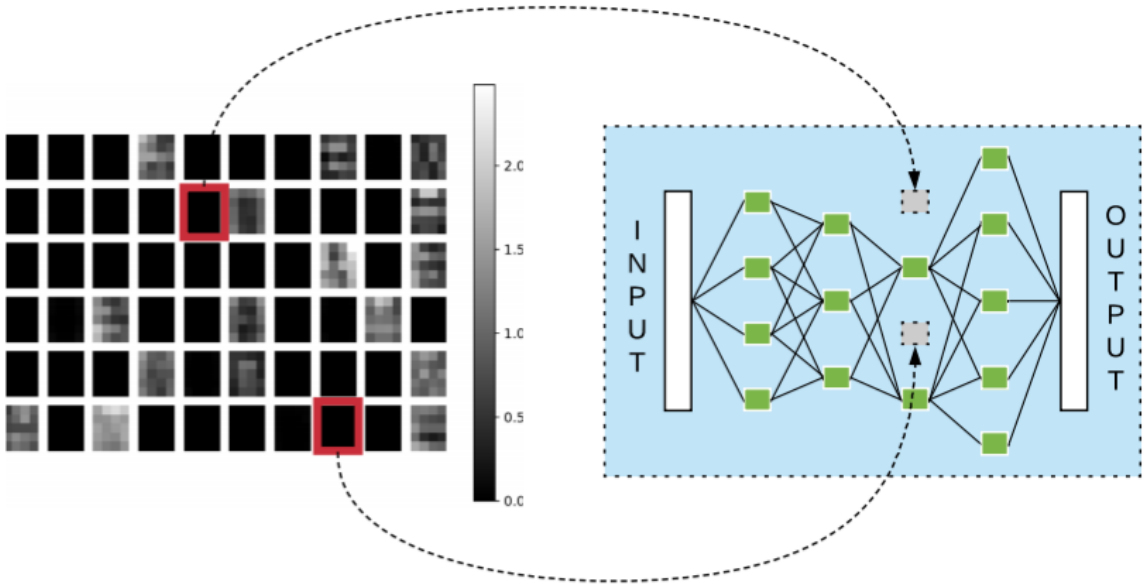
[our work]

# 通过模型逆向工程达成后门攻击

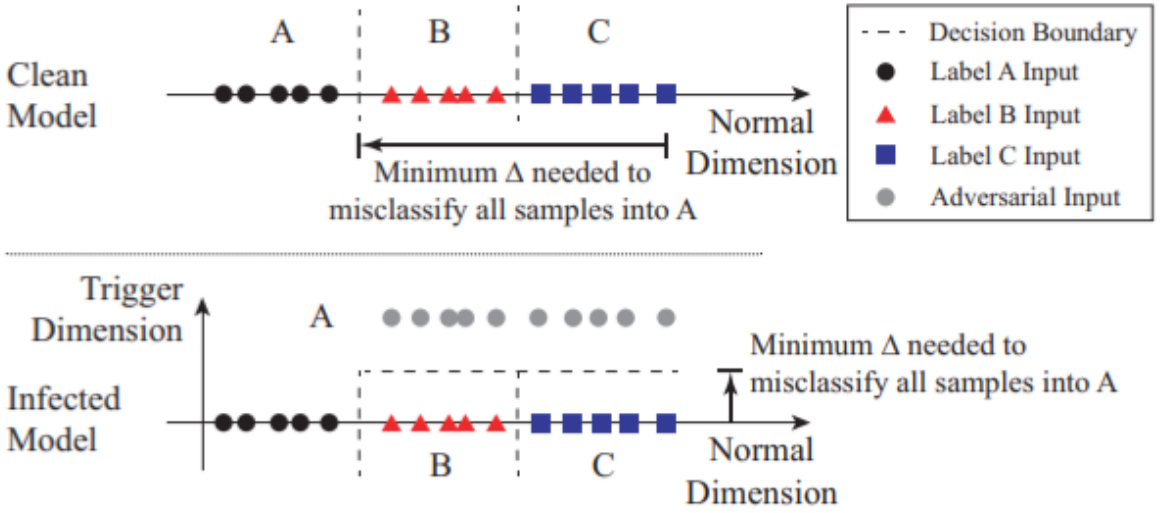



- 通过直接修改模型结构，加入恶意模型片段，植入后门。
- 优点：
  - 不需要训练，可直接攻击已经部署的模型
  - 植入的模型片段十分灵活，可以支持更现实的攻击场景
  - 应用使用模型时通常不检查结构，因此难以检测
- 效果：
  - 成功攻击54个Google Play中的Android应用

# 后门防御 (defense of backdoor attacks)



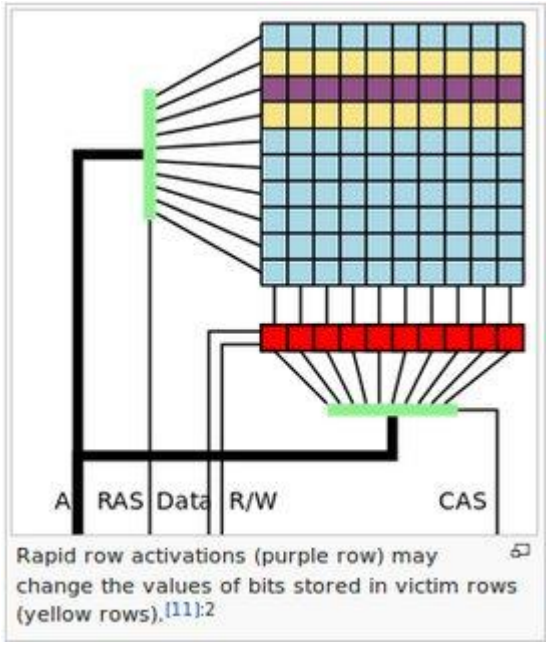

[Liu RAID2019] 将模型中对正常样本作用较小的neuron进行剪枝。



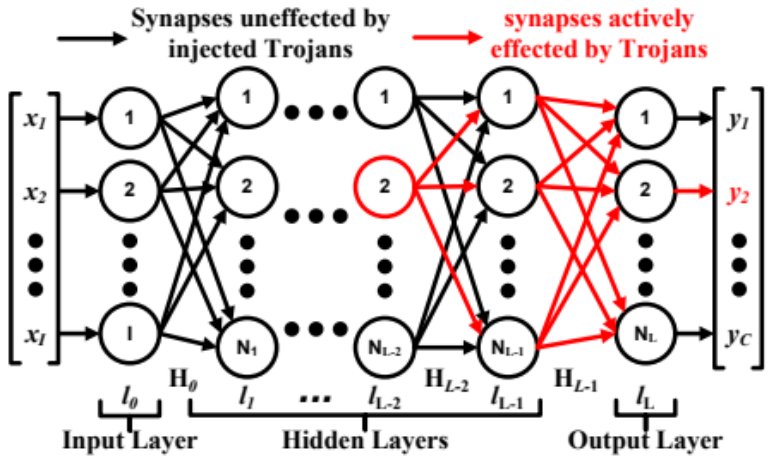

[Neural Cleanse S&P2019] 给定模型，反推引起误分类的扰动。若该扰动在输入中占比较小且容易找到，则模型很可能带有后门。

# 针对运行环境的攻击 (runtime attack)

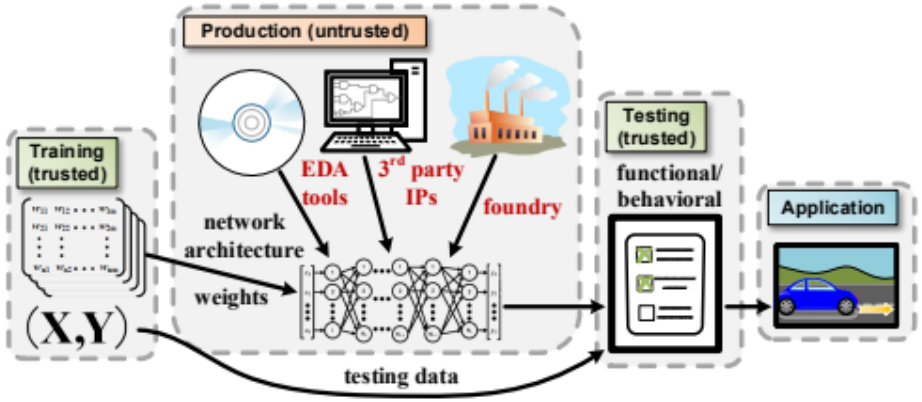
- 位翻转（Bit-flipping）攻击
  - [Rakin ICCV'2019] ResNet18模型中，翻转13个字节即可使模型准确率从70%降到0.1%。
  - 位翻转操作可以使用Row-Hammer Attacker完成
- 硬件后门植入
  - [Clements 2018] 通过硬件电路设计，在检测到trigger时改变神经网络中的某个neuron激活值




Row-Hammer Attack (RHA)




[Rakin ICCV2019]



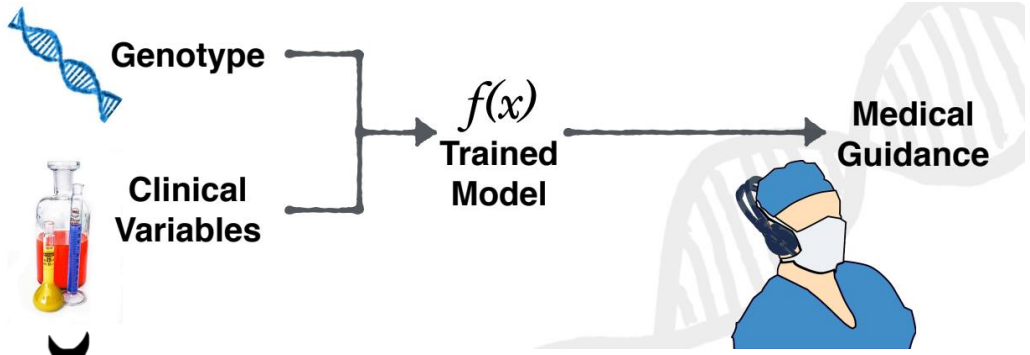

[Clements 2018]

# 保密性 (Confidentiality)

- 数据保密性 (Data Confidentiality)
  - 模型反转攻击
  - 差分隐私
  - 联邦学习
- 模型保密性 (Model Confidentiality)
  - 模型窃取攻击
  - 模型版权保护

# 模型反转攻击 (model inversion attack)

- 攻击者可以基于模型推断出训练数据中的一些隐私特性 [Fredrikson Security2014]

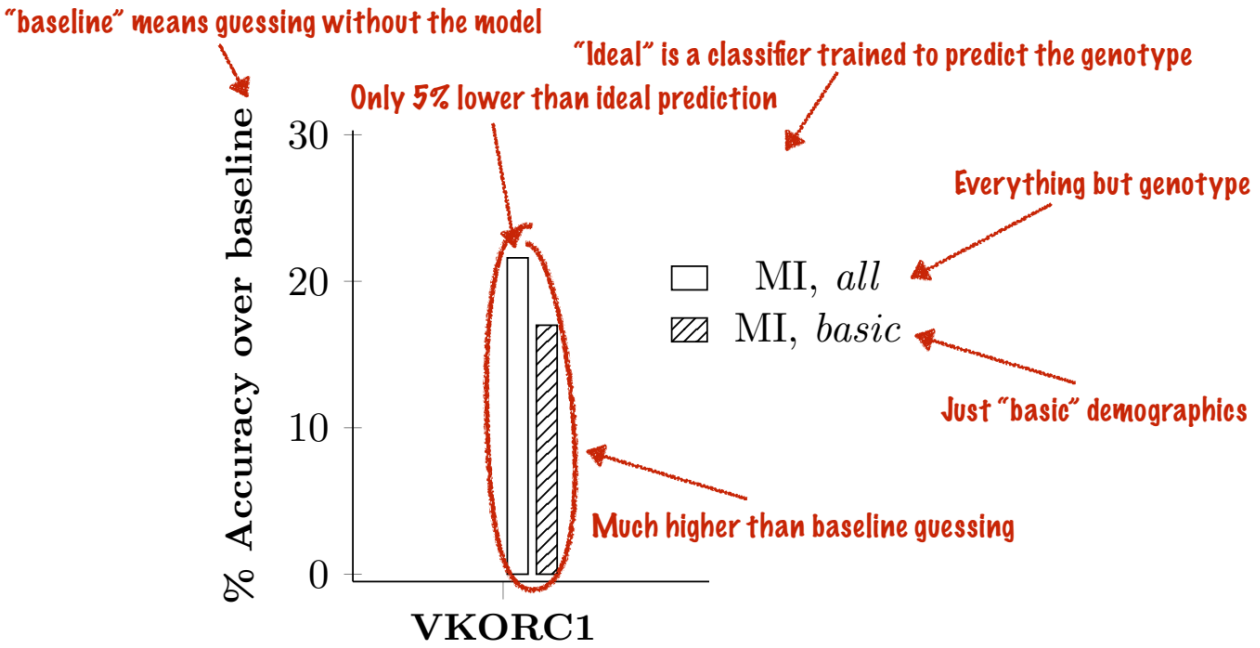



$f(x)$

| age | height | weight | race | history | vkorc1 | cyp2c9 | dose | |
|---|---|---|---|---|---|---|---|---|
| 50-59 | 176.53 | 144.2 | white | | | | 42.0 | 49.7 p=0.23 |
| 50-59 | 176.53 | 144.2 | white | | | | 42.0 | 42.0 p=0.75 |
| 50-59 | 176.53 | 144.2 | white | | | | 42.0 | 39.2 p=0.01 |

攻击者知道病人的基本信息和预测结果（药物剂量），且可以无限次调用模型 $f(x)$。目的是推断病人的基因标记。

1. Input: $\mathbf{z}_K = (x_1, \ldots, x_k, y), f, p_{1,\ldots,d,y}$
2. Find the *feasible set* $\hat{\mathbf{X}} \subseteq \mathbf{X}$, i.e., such that $\forall \mathbf{x} \in \hat{\mathbf{X}}$
   (a) $\mathbf{x}$ matches $\mathbf{z}_K$ on known attributes: for $1 \le i \le k, \mathbf{x}_i = x_i$.
   (b) $f$ evaluates to $y$ as given in $\mathbf{z}_K$: $f(\mathbf{x}) = y$.
3. If $|\hat{\mathbf{X}}| = 0$, return $\perp$.
4. Return $x_t$ that maximizes $\sum_{\mathbf{x} \in \hat{\mathbf{X}}: \mathbf{x}_t = x_t} \prod_{1 \le i \le d} p_i(\mathbf{x}_i)$



# 模型反转攻击 (cont.)

· 攻击者可以从人脸识别模型中恢复出人脸。[Fredrikson CCS2015]

人脸识别模型f：输入人脸图片，输出label（例如人名）

攻击者知道受害者的label，可以访问人脸识别模型。
目标是根据label和模型重构出受害者的人脸。

**Algorithm 1** Inversion attack for facial recognition models.

1: **function** MI-FACE($label, \alpha, \beta, \gamma, \lambda$)
2: $c(\mathbf{x}) \stackrel{\text{def}}{=} 1 - \tilde{f}_{label}(\mathbf{x}) + \text{AUXTERM}(\mathbf{x})$
3: $\mathbf{x}_0 \leftarrow \mathbf{0}$
4: **for** $i \leftarrow 1 \ldots \alpha$ **do**
5: $\mathbf{x}_i \leftarrow \text{PROCESS}(\mathbf{x}_{i-1} - \lambda \cdot \nabla c(\mathbf{x}_{i-1}))$
6: **if** $c(\mathbf{x}_i) \geq \max(c(\mathbf{x}_{i-1}), \ldots, c(\mathbf{x}_{i-\beta}))$ **then**
7: **break**
8: **if** $c(\mathbf{x}_i) \leq \gamma$ **then**
9: **break**
10: **return** $[\arg\min_{\mathbf{x}_i}(c(\mathbf{x}_i)), \min_{\mathbf{x}_i}(c(\mathbf{x}_i))]$

根据f构造代价函数c

在样本上进行梯度下降，process对梯度进行修正

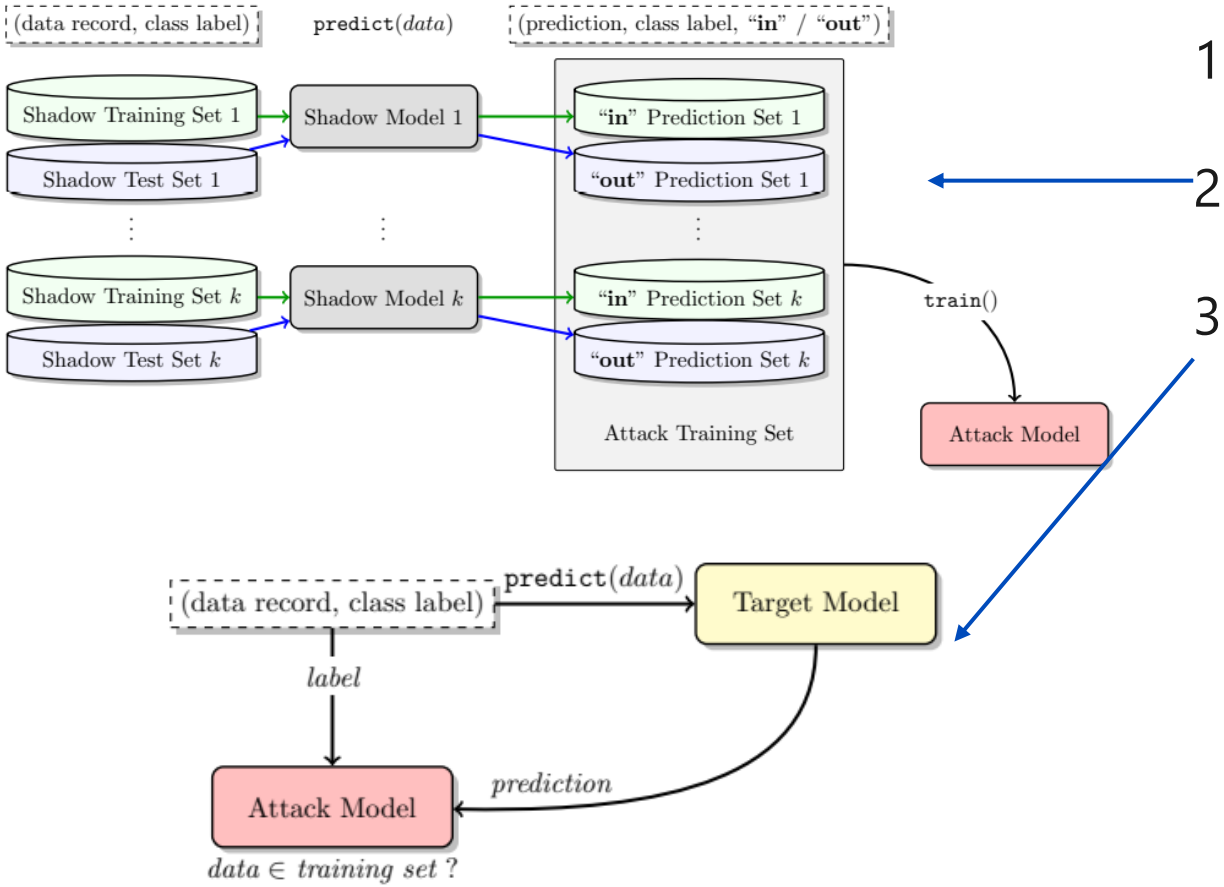
找到能使c(x)最小的样本x

Figure 1: An image recovered using a new model inversion attack (left) and a training set image of the victim (right). The attacker is given only the person's name and access to a facial recognition system that returns a class confidence score.

**模型反转成功率与模型的过拟合程度有关。[Yeom CSF'2018]**

# 成员推断攻击 (membership inference attack)

- 攻击者拥有对模型的黑盒访问权限，对于一个给定的data record，判断该record在不在训练数据集中 [Shokri S&P2017]




1. 基于 target model，生成若干个 shadow models，以及对应的 training data
2. 用 shadow model 的数据训练一个 attack model，用来判断sample是否在训练数据中
3. 对于目标 data record，将其在 target model 中的输出作为 attack model 输入，即可判断其 membership

Shadow model data 生成方法：

- 基于模型搜索数据
- 基于数据统计信息
- 基于有噪音的真实数据

# 差分隐私 (differential privacy)

- 差分隐私是隐私保护的一种手段，通过引入随机性保证数据库查询操作的隐私安全。

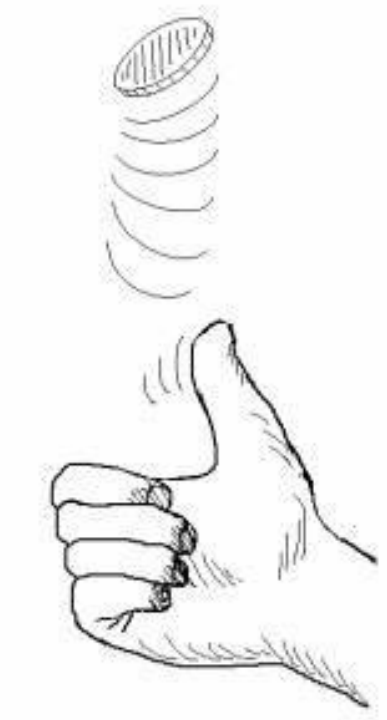
**简单例子：社会学调查**

**目标**：调查人员希望获取一个群体中拥有属性A的大致比例，又不能侵犯群体中每个个体的隐私。

**方法**：让每个被调查者采用如下流程：
1. 扔一枚硬币
2. 如果正面朝上，则如实回答
3. 如果反面朝上，则随机回答是或否

因为每个人的回答具有随机性，所以可以保护隐私，同时，调查者收集的统计信息可以用来估计拥有属性A的真实比例。

$$p' = \frac{1}{2}p + \frac{1}{4}$$



# 差分隐私 – 定义

**差分攻击**：攻击者可以多次调用一个查询操作，通过对比不同调用返回的结果获取数据集中个体的隐私信息。

**数据集**

| Name | Has Diabetes (X) |
|---|---|
| Ross | 1 |
| Monica | 1 |
| Joey | 0 |
| Phoebe | 0 |
| Chandler | 1 |

假设医院维护一个数据库存储糖尿病人信息，提供数据查询服务COUNT()，返回数据集中患有糖尿病的人数。

**差分攻击**：攻击者可以在Chandler进入数据库前后调用COUNT，通过分析其结果的区别获知Chandler是否患有糖尿病。

**差分隐私定义**：假设ε是一个正实数，A是一个以数据集为输入的随机算法。对于任意**邻近数据集**$D_1$和$D_2$，A的输出域的任意子集S，如果满足：

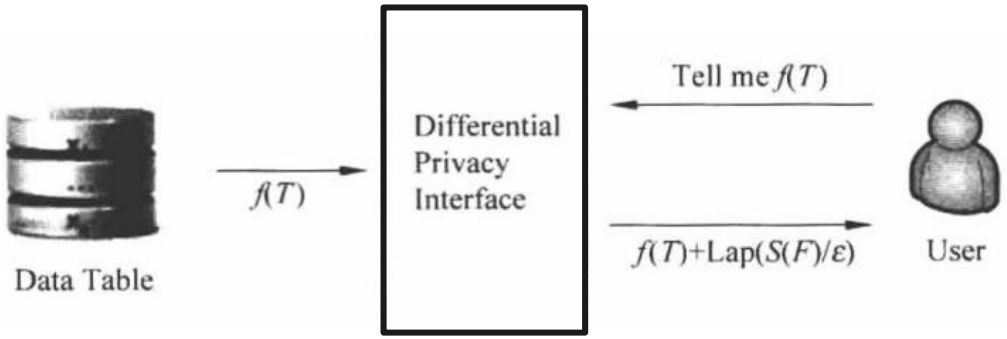
$$\Pr[\mathcal{A}(D_1) \in S] \leq \exp(\epsilon) \cdot \Pr[\mathcal{A}(D_2) \in S],$$

则算法A满足ε-差分隐私 (ε-differential privacy)。

**邻近数据集：**$D_1$和$D_2$之间只有一条记录有区别。

# 达成差分隐私的方法

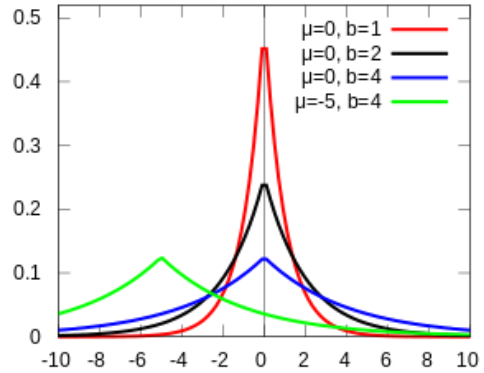
- Laplace机制
  - 适用于数值型输出
  - 方法：对返回结果添加Laplace分布的噪声




- 指数机制
  - 适用于非数值型输出
  - 方法：以一定的随机概率从输出范围中选择输出
- 组合机制
  - 2个随机算法组合：$\varepsilon = \varepsilon_1 + \varepsilon_2$
  - 2个相互独立算法组合：$\varepsilon = \max(\varepsilon_1, \varepsilon_2)$

期望为0，方差为$2b^2$的Laplace分布，其概率密度函数为：

$$p(x) = \frac{exp(-|x|/b)}{2b}$$




满足ε-差分隐私保护的Laplace噪声，其尺度参数b应为$\frac{S(F)}{\varepsilon}$，其中$S(F)$为查询函数F的敏感度

$$S(F) = \max_{T_1, T_2} \left( \sum_{f \in F} |f(T_1) - f(T_2)| \right)$$

# 满足差分隐私的深度学习

- 如何保证模型参数不包含个体隐私信息?
- [Abadi CCS2016] 主要思路是让模型参数梯度具备差分隐私特性。
  - 根据梯度值，决定对梯度加入噪声的大小
  - 通过对梯度进行削减，使噪声规模不影响精度
  - 利用差分隐私组合规则计算隐私成本 ε
  - 通过调整超参数平衡隐私，准确性和性能

**Algorithm 1** Differentially private SGD (Outline)

**Input:** Examples $\{x_1, \ldots, x_N\}$, loss function $\mathcal{L}(\theta) = \frac{1}{N}\sum_i \mathcal{L}(\theta, x_i)$. Parameters: learning rate $\eta_t$, noise scale $\sigma$, group size $L$, gradient norm bound $C$.

**Initialize** $\theta_0$ randomly

**for** $t \in [T]$ **do**

Take a random sample $L_t$ with sampling probability $L/N$

计算梯度

**Compute gradient**

For each $i \in L_t$, compute $\mathbf{g}_t(x_i) \leftarrow \nabla_{\theta_t}\mathcal{L}(\theta_t, x_i)$

**Clip gradient**

$\bar{\mathbf{g}}_t(x_i) \leftarrow \mathbf{g}_t(x_i) / \max\left(1, \frac{\|\mathbf{g}_t(x_i)\|_2}{C}\right)$ 削减梯度

**Add noise**

$\tilde{\mathbf{g}}_t \leftarrow \frac{1}{L}\left(\sum_i \bar{\mathbf{g}}_t(x_i) + \mathcal{N}(0, \sigma^2 C^2 \mathbf{I})\right)$ 添加噪声

**Descent**

$\theta_{t+1} \leftarrow \theta_t - \eta_t \tilde{\mathbf{g}}_t$ 更新模型参数

**Output** $\theta_T$ and compute the overall privacy cost $(\varepsilon, \delta)$ using a privacy accounting method.

# 联邦学习 (federated learning)

· 面向的场景：训练数据分布在边缘设备上，因隐私原因不能上传。
· 联邦学习：边缘设备只上传参数更新，而不上传训练数据

梯度也有可能泄露隐私，
怎么办？
- 差分隐私
- 安全聚合 (secure aggregation)
  [Segal CCS2017]

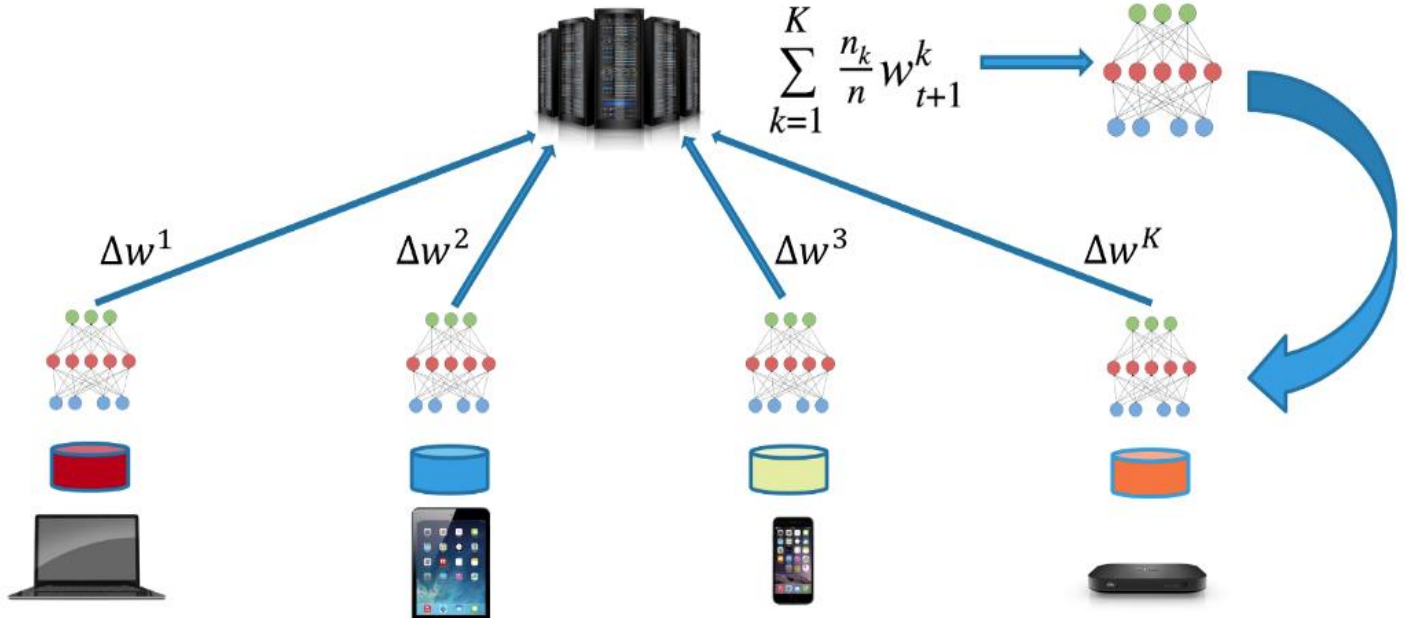
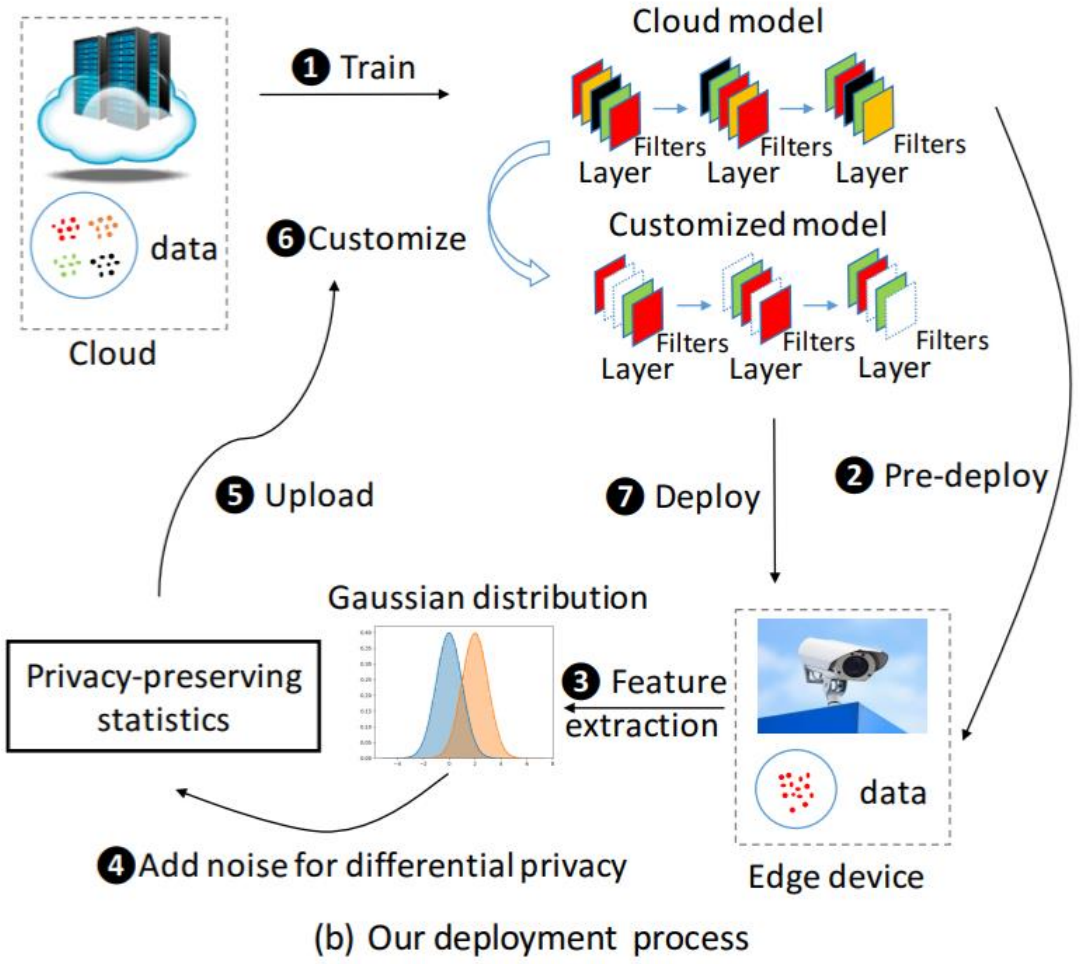



Image from: https://proandroiddev.com/federated-learning-e79e054c33ef

[our work]

# 差分隐私的模型适应 (differentially-private adaptation)

- 针对模型从cloud部署到edge的场景
- 解决的问题：
  - 模型最好针对edge数据进行适应，提升精度
  - Edge数据往往包含隐私信息，不能上传
- 解决方法
  - Edge端用神经元的激活统计信息代表数据分布
  - 对统计信息添加Laplace噪声保证差分隐私
  - 云端根据上传的统计信息调整训练数据




(b) Our deployment process

# 同态加密 (homomorphic encryption)

- 同态加密
  - 允许在密文上进行代数运算
  - 概念提出：[Rivest1997]
  - 全同态实现：[Gentry2009] （支持加法和乘法）
- 同态加密的模型预测
  - CryptoNets [Dowlin JMLR2016]
  - 将模型参数和输入数据加密，预测结果也是加密的
  - 主要技术：将非线性操作用线性操作近似
- 性能问题是同态加密应用的最大挑战

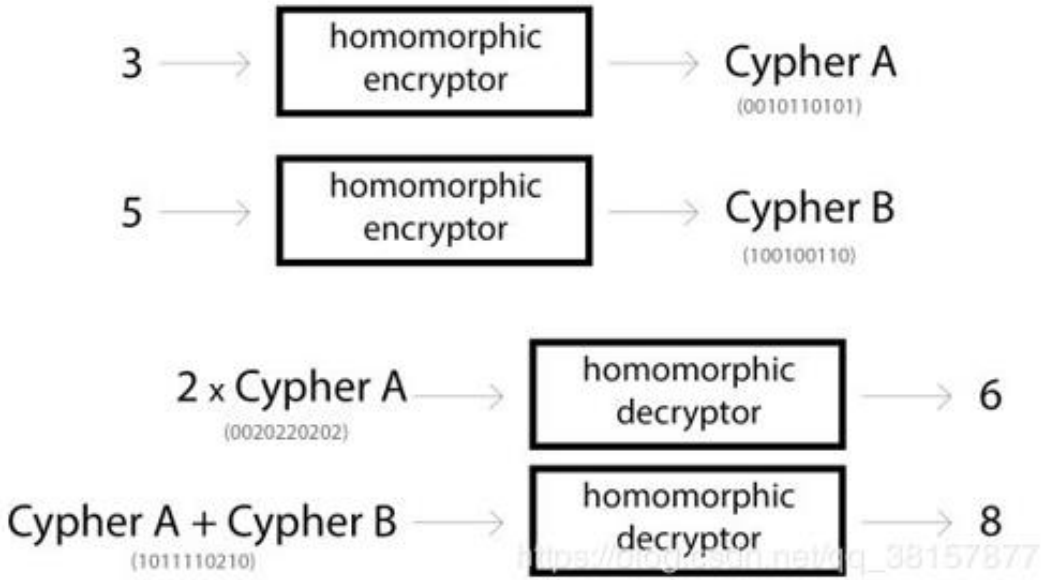



*Table 2.* The performance of CryptoNet for MNIST

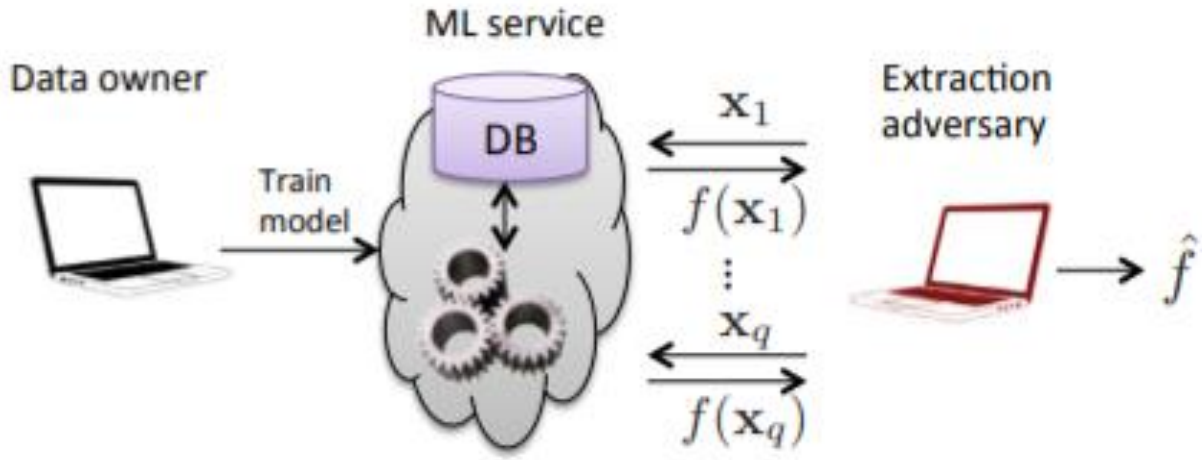
| Stage | Latency | Additional latency per each instance in a batch | Throughput |
|---|---|---|---|
| Encoding+Encryption | 44.5 seconds | 0.138 seconds | 24193 per hour |
| Network application | 250 seconds | 0 | 58982 per hour |
| Decryption+Decoding | 3 seconds | 0.012 seconds | 274131 per hour |

# 模型窃取 (model stealing)

· 优秀的模型往往是企业的重要财产，如果被窃取会侵害知识产权。

模型窃取方式：

- **直接窃取**：直接攻克模型的开发、存储或部署环境，获得原模型的拷贝。
- **间接窃取**：通过不断调用模型开发者提供的预测API，重构出一个训练数据集，通过训练达到与原模型类似的准确率 [Tramer Security2016][Orekondy CVPR2019]。



# 模型水印 (watermarking)

· 如果模型被窃取，如何验证模型的所有权？——给模型加水印

- $\texttt{KeyGen}()$ outputs a key pair $(\mathsf{mk}, \mathsf{vk})$.
- $\texttt{Mark}(M, \mathsf{mk})$ on input a model $M$ and a marking key $\mathsf{mk}$, outputs a model $\hat{M}$.
- $\texttt{Verify}(\mathsf{mk}, \mathsf{vk}, M)$ on input of the key pair $\mathsf{mk}, \mathsf{vk}$ and a model $M$, outputs a bit $b \in \{0,1\}$.

$$\Pr_{x \in \overline{D} \setminus T} \left[ f(x) \neq \texttt{Classify}(\hat{M}, x) \right] \leq \varepsilon, \text{ but}$$

$$\Pr_{x \in T} \left[ T_L(x) \neq \texttt{Classify}(\hat{M}, x) \right] \leq \varepsilon.$$

[Adi Security2018] 将水印嵌入模型输出。
通过类似于后门攻击的方法，使模型在特定输入上产生特定输出。

**ALGORITHM 1:** Fingerprint embedding for one hidden layer.

**INPUT:** **Pre-trained unmarked DNN** ($\mathcal{T}$); **Training data** ($\{X^{train}, Y^{train}\}$); **Location of the target layer** ($l$) **with embedding dimension** ($N$); **Code length** ($v$); **Resilience level** ($k$); **Embedding strength** ($\gamma$).

**OUTPUT:** **A set of marked DNNs** ($\{\mathcal{T}_1^*, ..., \mathcal{T}_b^*\}$); **FP keys.**

❶ Key Generation:

$C_{v \times b} \leftarrow Construct\_Codebook\ (v, k, 1)$
$U_{v \times v} \leftarrow Generate\_Basis\_Matrix\ (v)$
$X_{v \times N} \leftarrow Generate\_Projection\_Matrix\ (v, N)$

❷ Fingerprint Construction:

$F_{v \times b} \leftarrow Construct\_Fingerprints\ (C, U)$

❸ Model Fine-tuning: For each user $j$ ($j = 1, ..., b$), train the DNN on $\{X^{train}, Y^{train}\}$ with the corresponding FP-specific loss:

$$\mathcal{L} = \mathcal{L}_0 + \gamma \cdot Mean\_Square\_Error(\mathbf{f_j} - X\mathbf{w_j}).$$

**Return:** Marked DNNs ($\{\mathcal{T}_1^*, ..., \mathcal{T}_b^*\}$), FP keys ($l$, $C_{v \times b}$, $U_{v \times v}$, $X_{v \times N}$).

[DeepAttest ISCA2018] 将水印嵌入模型参数。
通过fine-tune使模型的参数逼近特定的指纹。

[our work]

# 模型相似性检查 (similarity detection)

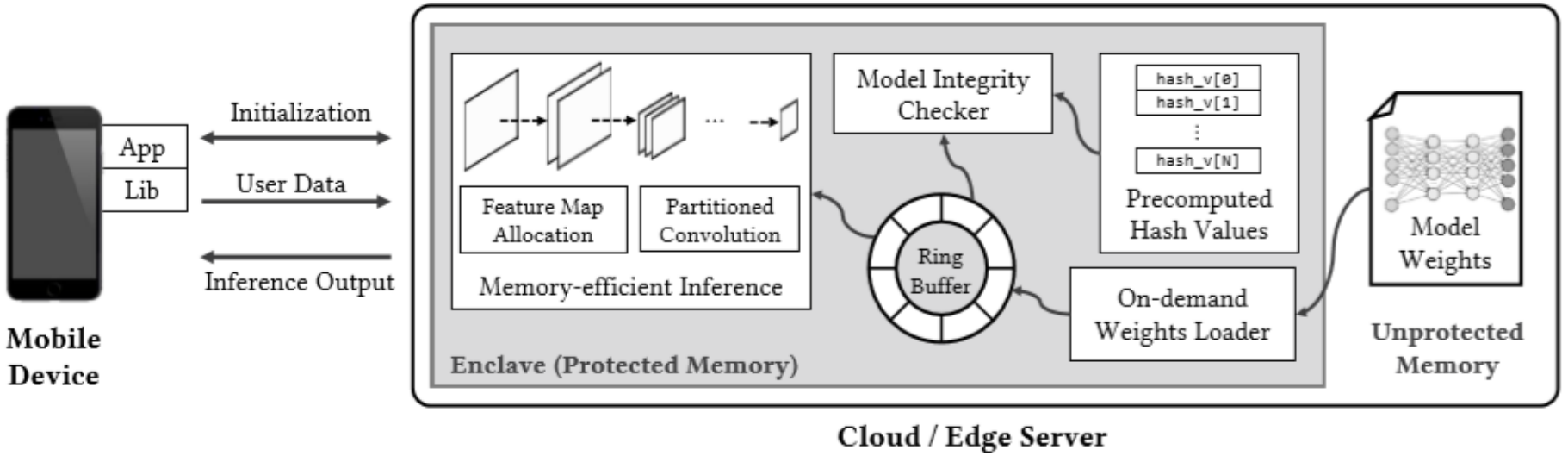
- 对于已有的无水印的模型，如何验证其知识所有权?
  - 其他领域的类似问题：论文查重、代码查重、……
- 如何检测模型之间知识的相似性?
  - Our idea: 通过profiling，根据模型对不同样本的决策逻辑对比其相似性

给定模型f和模型g，相似度比较算法如下:

1. 生成N个随机样本对$\{(x_{1a}, x_{1b}), (x_{2a}, x_{2b}), \dots, (x_{Na}, x_{Nb})\}$
2. 将每个样本对输入模型f，计算$s_i^f = sim(f(x_{ia}), f(x_{ib}))$。
   $< s_1^f, s_2^f, \dots, s_N^f >$为模型f的决策向量$V^f$
3. 用同样的方法计算$V^g$
4. 模型f与模型g之间的相似度可以用$V^f$和$V^g$的距离度量

# 可信执行环境 (trusted execution environment)

- 可信执行环境（TEE）是处理器中一块受保护区域，可以保证其中存储数据和运行代码的安全性（机密性和完整性）。
- 在TEE中运行DNN最大的挑战是性能问题。




[Graviton OSDI2018] 通过修改GPU驱动和CUDA runtime，在GPU上支持TEE

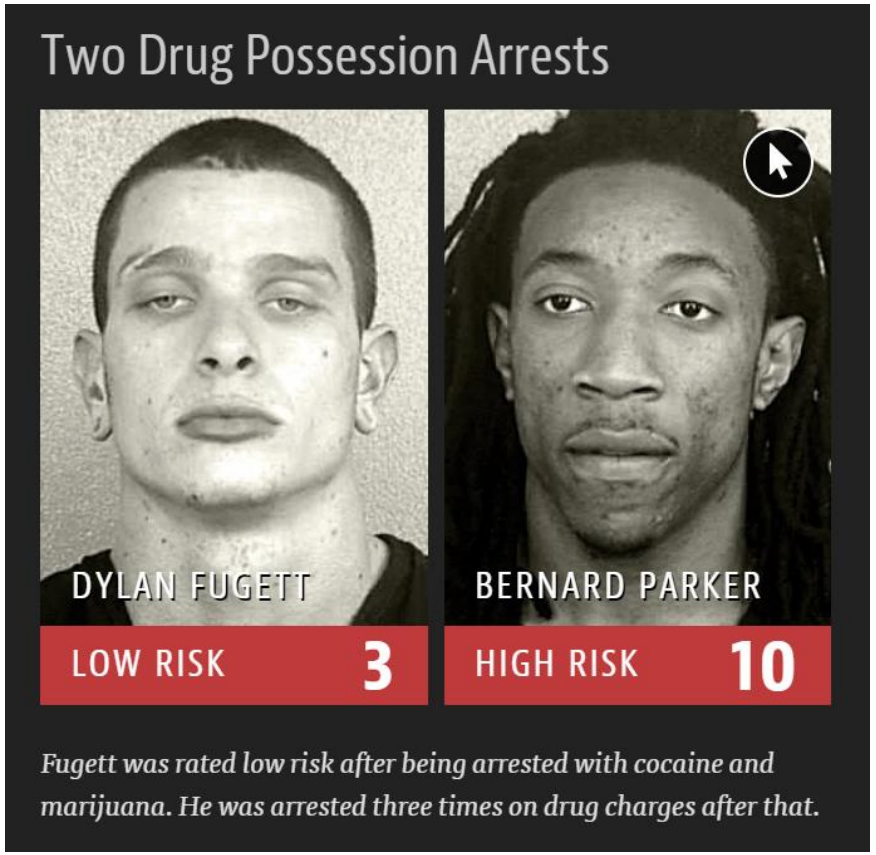
[Occlumency MobiCom2019] (our work) 通过按需参数加载、卷积操作内存优化、细粒度流水线优化TEE中的DNN inference效率

[Slalom ICLR2019] 将一部分复杂的线性计算经加密后外放(outsource)到普通区域

# 伦理 (Ethics)

- AI系统的公平性 (Fairness)
  - 歧视与偏见
  - “公正”的中间表示
- AI的滥用 (Misuse)
  - deepfake自动换脸
  - 信息茧房

# 算法公平性 (AI fairness)




罪犯风险评估算法有种族歧视 [ProPublica 2016]

TECHNOLOGY NEWS OCTOBER 10, 2018 / 11:12 AM / 2 YEARS AGO

## Amazon scraps secret AI recruiting tool that showed bias against women

基于机器学习的招聘工具不喜欢女性 [Reuters 2018]

**公平性的定义（Demographic Parity）**

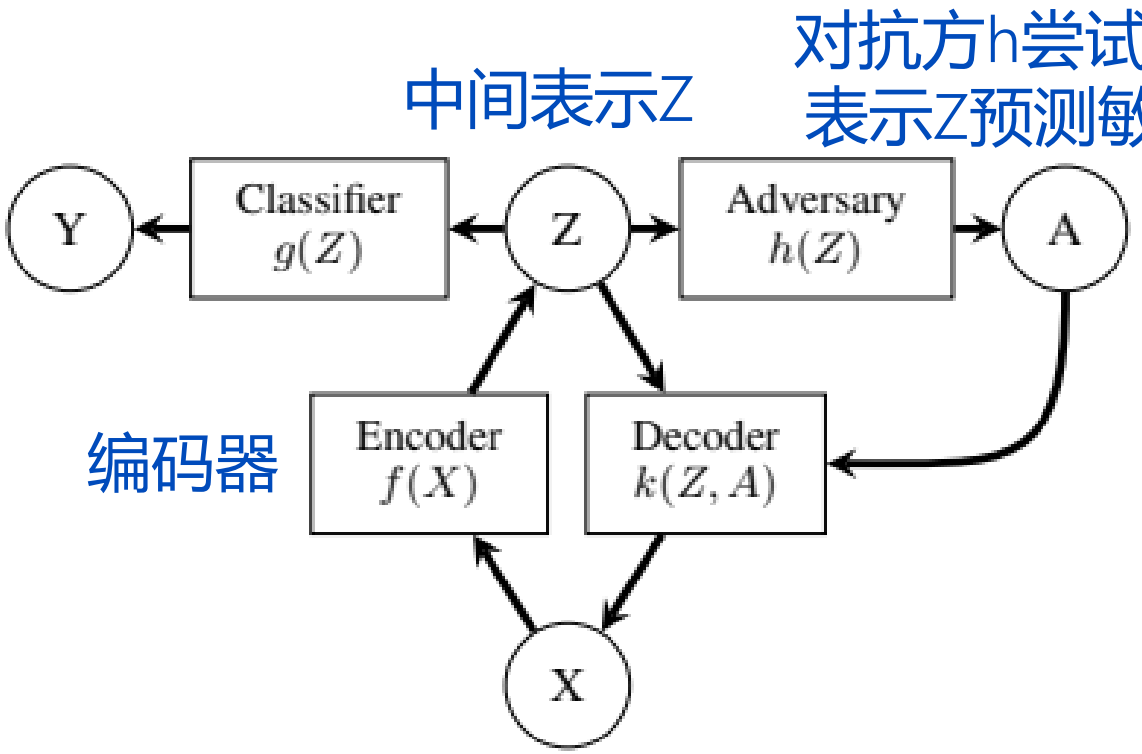
对于受保护的属性 $p$（如种族、性别等），一个公平的算法应满足预测值 $y$ 与属性 $p$ 无关：

$$\Pr(y|p) = \Pr(y)$$

实际情况中 $p$ 往往不易定义。

# 无偏见的中间表示 (fair representation)

- 通过对抗学习 (adversarial learning) 生成一个无偏见的中间表示，使第三方无法从中间表示中预测出敏感性质。 [Madras ICML2018]




$$\underset{f,g,k}{\text{minimize}}\ \underset{h}{\text{maximize}}\ \mathbb{E}_{X,Y,A}\left[L(f,g,h,k)\right], \qquad (1)$$

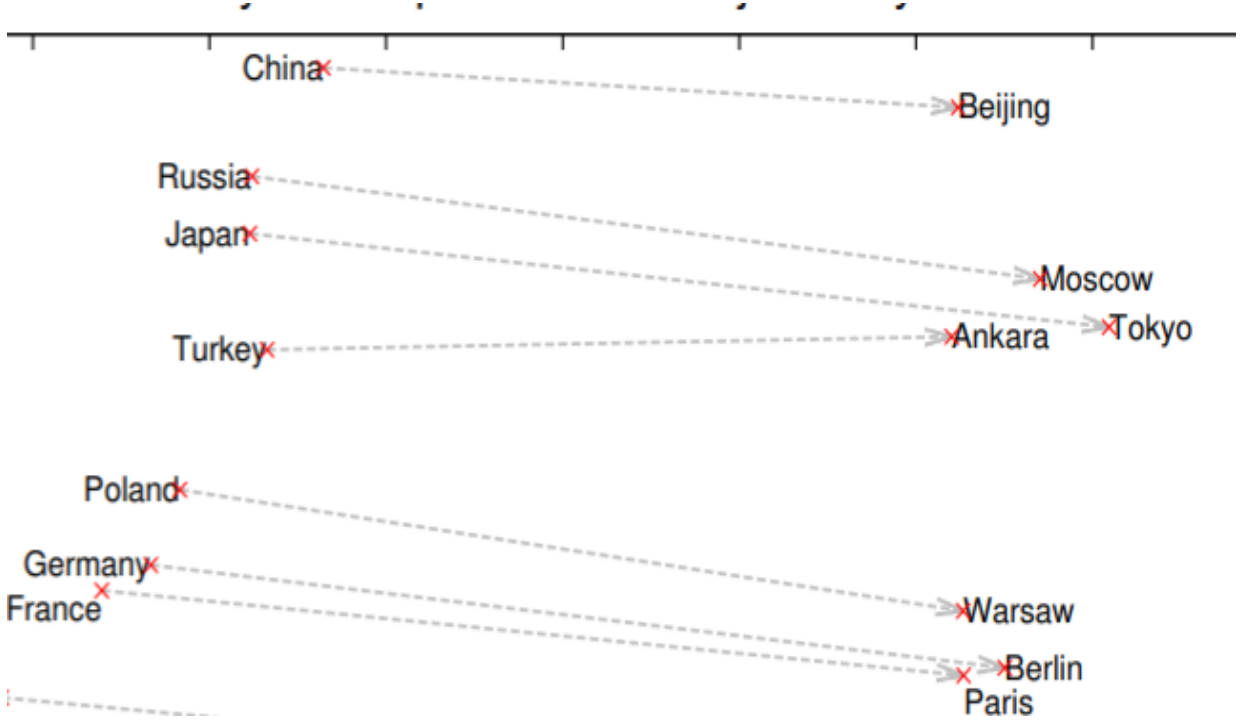
with the combined objective expressed as

$$\begin{aligned} L(f,g,h,k) = {} & \alpha L_C(g(f(X,A)),Y) \\ & + \beta L_{Dec}(k(f(X,A),A),X) \\ & + \gamma L_{Adv}(h(f(X,A)),A) \end{aligned} \qquad (2)$$

训练目标：使对抗方难以从中间表示Z中预测出敏感性质A，同时优化中间表示的效果

# 词向量中的刻板印象 (stereotypes in word embedding)

- Word embedding: 基于大量语料库为词汇生成的语义向量，广泛应用于各种自然语言处理任务。




词向量之间的空间关系能反映单词之间的语义对应关系。[Mikolov 2013]

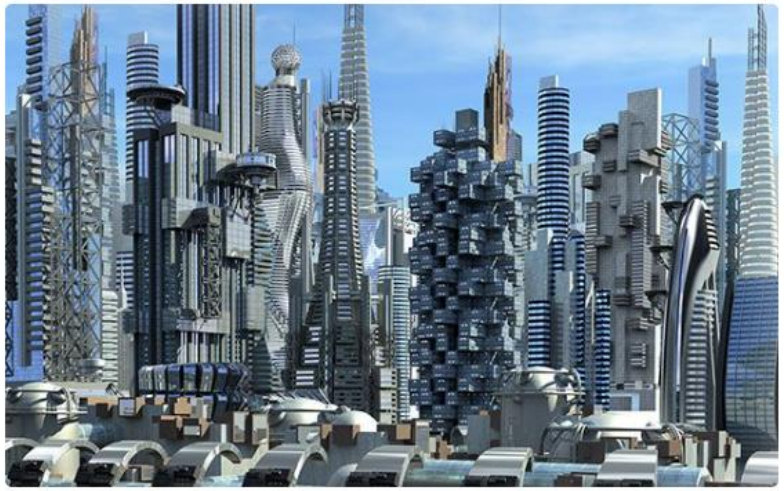
| Extreme *she* | Extreme *he* |
|---|---|
| 1. homemaker | 1. maestro |
| 2. nurse | 2. skipper |
| 3. receptionist | 3. protege |
| 4. librarian | 4. philosopher |
| 5. socialite | 5. captain |
| 6. hairdresser | 6. architect |
| 7. nanny | 7. financier |
| 8. bookkeeper | 8. warrior |

$$\overrightarrow{\text{man}} - \overrightarrow{\text{woman}} \approx \overrightarrow{\text{computer programmer}} - \overrightarrow{\text{homemaker}}$$

从海量文本中训练的词向量包含对性别的刻板印象 [Bolukbasi NIPS2016] [Caliskan Science2017]

# 人工智能用于造假


AI换脸 [Deepfake]



VERIFIED

**A Global Pandemic Will Force These Industry Sectors To Strike Smart**

As the world is gearing up for the coronavirus pandemic, cities are vowing to not let the virus move in to prevent the

notrealnews.net
AI生成假新闻

34,125 views | Sep 3, 2019, 04:42pm EDT

**A Voice Deepfake Was Used To Scam A CEO Out Of $243,000**

Jesse Damiani Contributor
Consumer Tech
*I cover the human side of VR/AR, Blockchain, AI, Startups, & Media.*

AI诈骗电话

# 人工智能用于战争

- AI可以极大促进武器自动化
- 世界AI & Robotics专家联合抵制AI武器相关研究 [IJCAI 2015]
- Google员工联名抗议公司接受军方项目Project Maven [NYTimes 2018]

注：图文无关

# 信息茧房 (information cacoons)

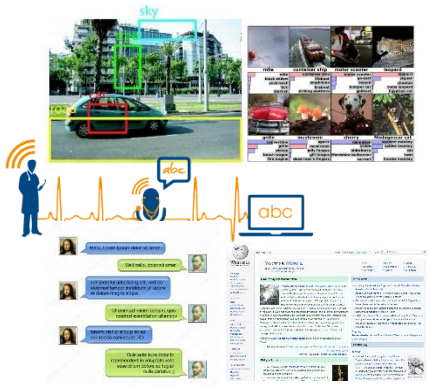
基于AI的个性化推荐系统无处不在，精准地推荐用户更喜欢看到的内容。


YouTube 和 知乎 给我推荐的内容 （2020年6月）

“信息茧房”，是指传播体系个人化所导致的信息封闭的后果。当个体只关注自我选择的或能够愉悦自身的内容，而减少对其他信息的接触，久而久之，便会像蚕一样逐渐桎梏于自我编织的“茧房”之中。这将导致视野偏狭和思想的封闭甚至极化，进而会加强偏见并制造出非理性的极端主义，直至侵害政治民主。

——桑斯坦 (Cass R. Sunstein)《信息乌托邦》(Infotopia)

# AI安全问题汇总

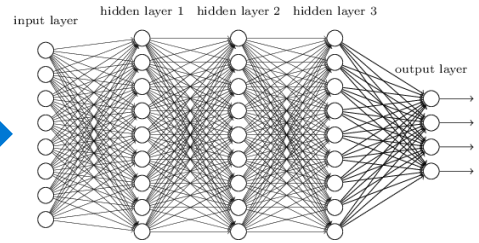



*训练数据*

数据投毒

后门植入

**训练**




*模型*

模型偏见

模型反转攻击

**部署**

*生产环境*

对抗样本攻击

**使用**

*用户*

算法滥用

运行环境攻击

**数据收集**

模型窃取

隐私泄露

# 总结 & 挑战

相比于传统程序和系统，AI存在更严重的安全和隐私问题，易受到攻击、产生不确定或恶意的决策、导致隐私泄露等。AI在安全攸关领域应用需要有更严格的变量控制、更安全的运行环境、更充分的测试。

研究挑战和开放性问题：

- 算法决策的可解释性
- 安全性质的形式化保证和验证
- 高性能的加密算法和可信执行环境
- 易用的联邦学习框架

Microsoft

# Thank you!
# Q & A

© Copyright Microsoft Corporation. All rights reserved.