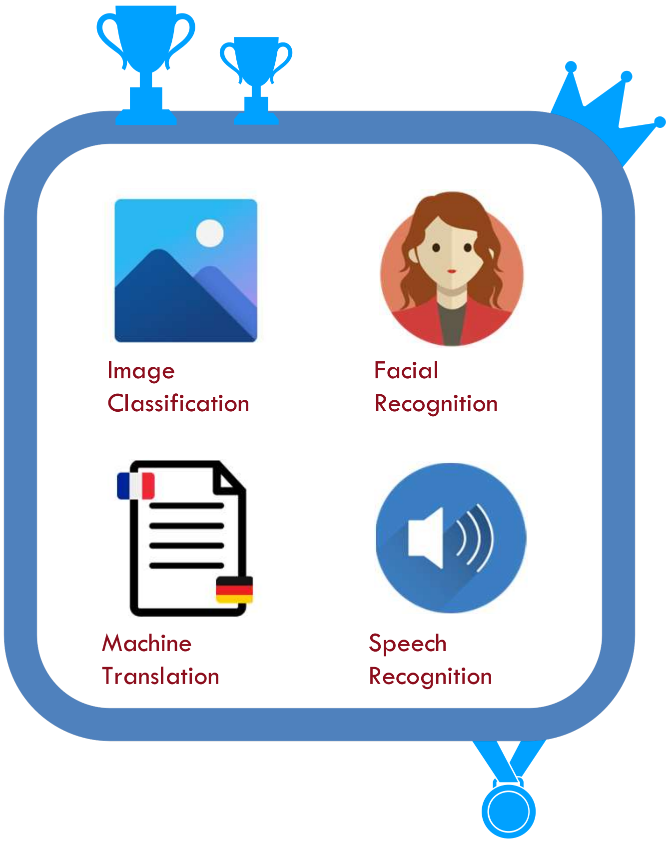
Security and Privacy of AI Knowledge Guide

Lorenzo Cavallaro and Emiliano de Cristofaro
University College London

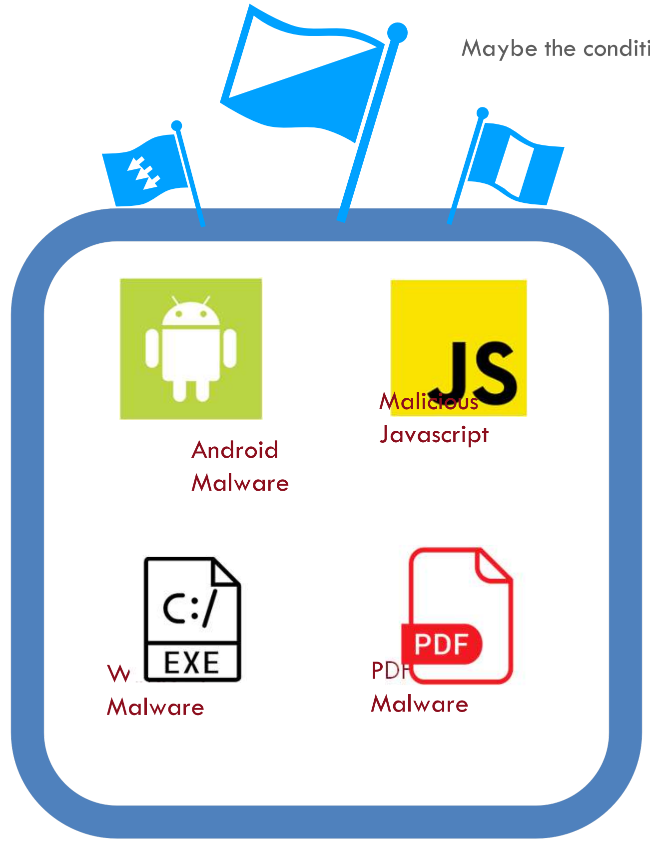
contact@cybok.org
www.cybok.org

The CyBOK project would like to understand how CyBOK is being used and its uptake. The project would like organisations using, or intending to use, CyBOK for the purposes of education, training, course development, professional development etc. to contact it at contact@cybok.org to let the project know how they are using CyBOK.

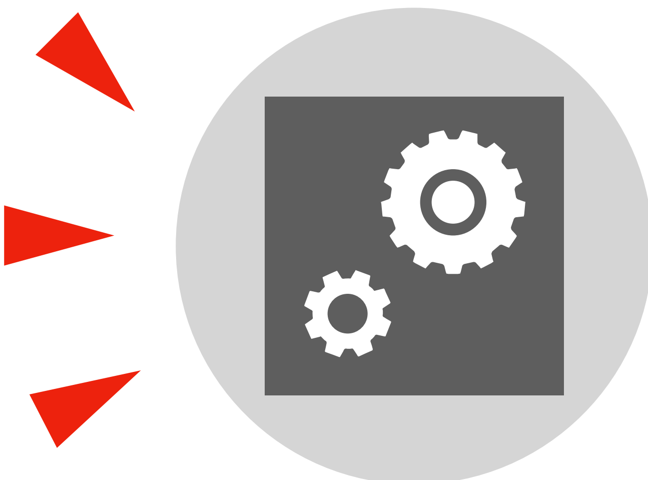
Machine Learning Revolution



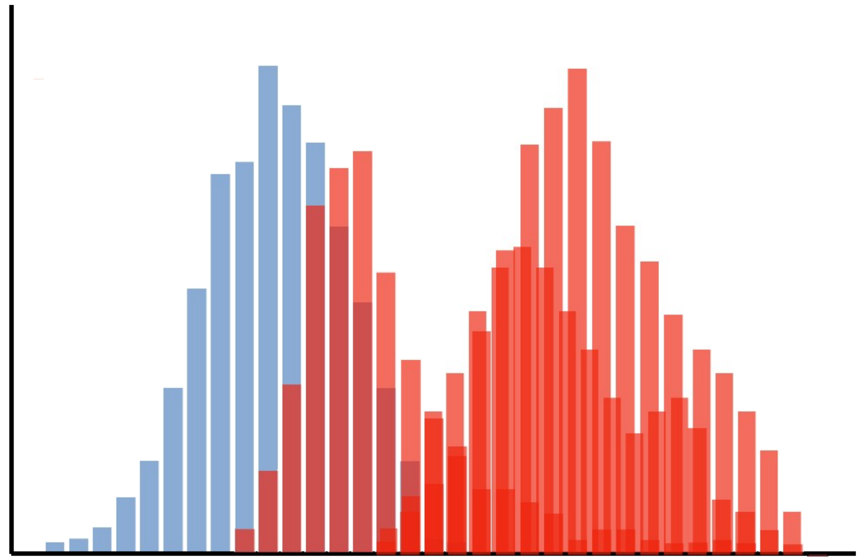
Maybe the conditions aren't ready yet?



Security *is* Adversarial



New detection systems trigger an immediate response…



…which causes dataset shifts, often violating the i.i.d. assumption

Adversaries Affect Security *and* Privacy of AI Systems

This CyBoK Knowledge Guide

- Part 1 – Security of AI
- Part 2 – Privacy of AI

Threat Models (Attacks and Defenses)

- Perfect-, Limited-, Zero-Knowledge
- Training vs Inference
- Passive vs Active

Adversaries Affect Security *and* Privacy of AI Systems

This CyBoK Knowledge Guide

- Part 1 – Security of AI
- Part 2 – Privacy of AI

Threat Models (Attacks and Defenses)

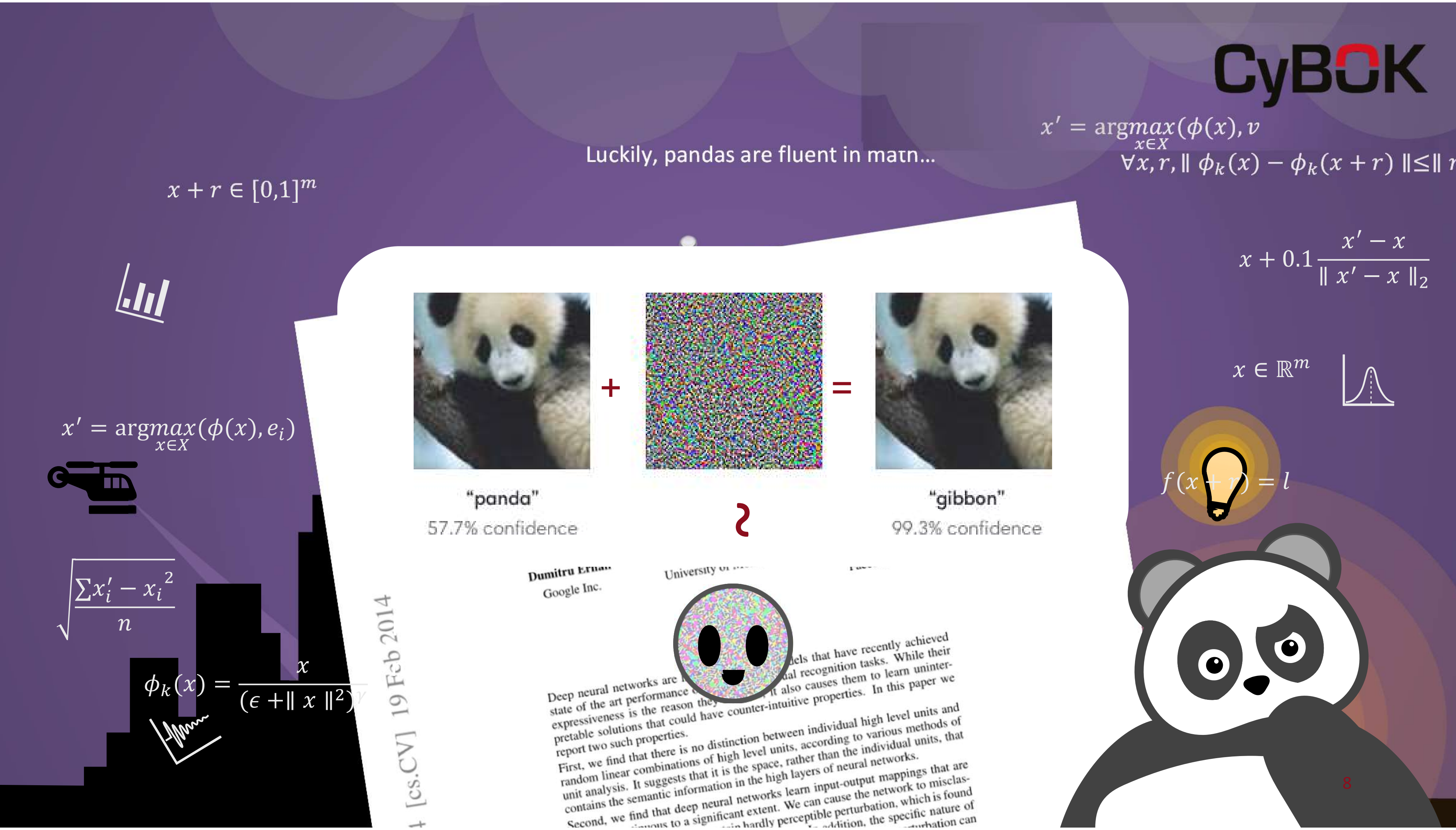
- Perfect-, Limited-, Zero-Knowledge
- Training vs Inference
- Passive vs Active

Part 1 – Security of AI Webinar focuses on

- Perfect Knowledge
- Inference (aka adversarial ML)
- Active

Details about other threat models, attacks, and defenses in the CyBoK KG


A Dystopian Future…
CyBOK
Pandas are forbidden!
Guilty of being too cute!

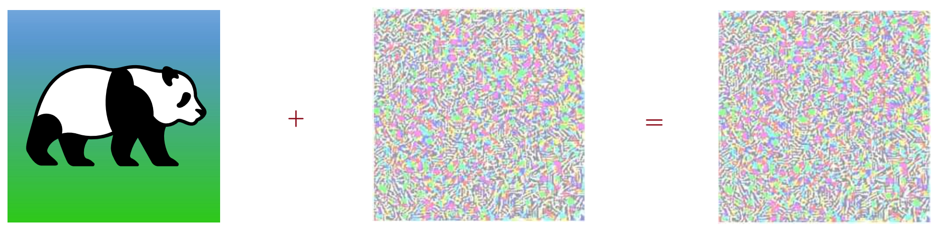


CyBOK
Luckily, pandas are fluent in math...
$x' = \underset{x \in X}{\arg\max}(\phi(x), v$
$\forall x, r, \| \phi_k(x) - \phi_k(x+r) \| \leq \| r$
$x + r \in [0,1]^m$
$x + 0.1 \frac{x' - x}{\| x' - x \|_2}$
$x \in \mathbb{R}^m$
$x' = \underset{x \in X}{\arg\max}(\phi(x), e_i)$
"panda"
57.7% confidence
"gibbon"
99.3% confidence
$f(x+r) = l$
$\sqrt{\frac{\sum x_i' - x_i{}^2}{n}}$
$\phi_k(x) = \frac{x}{(\epsilon + \| x \|^2)}$
[cs.CV] 19 Feb 2014
Google Inc.
Deep neural networks are ... state of the art performance ... expressiveness is the reason they ... pretable solutions that could have counter-intuitive properties. In this paper we report two such properties.
First, we find that there is no distinction between individual high level units and random linear combinations of high level units, according to various methods of unit analysis. It suggests that it is the space, rather than the individual units, that contains the semantic information in the high layers of neural networks.
Second, we find that deep neural networks learn input-output mappings that are ... to a significant extent. We can cause the network to misclas- ... hardly perceptible perturbation, which is found

Let’s Analyze What Happened

Feature-Space Attacks

Original Image | Perturbation | Adv. Image



“panda” 57.7% | *imperceptible noise* | “gibbon” 99.3%

x | δ | $x + \delta$

Optimization

$$minimize_{\delta} ||\delta||_p + c \cdot f(x + \delta)$$

Pixel Perturbations

Loss of Target Class

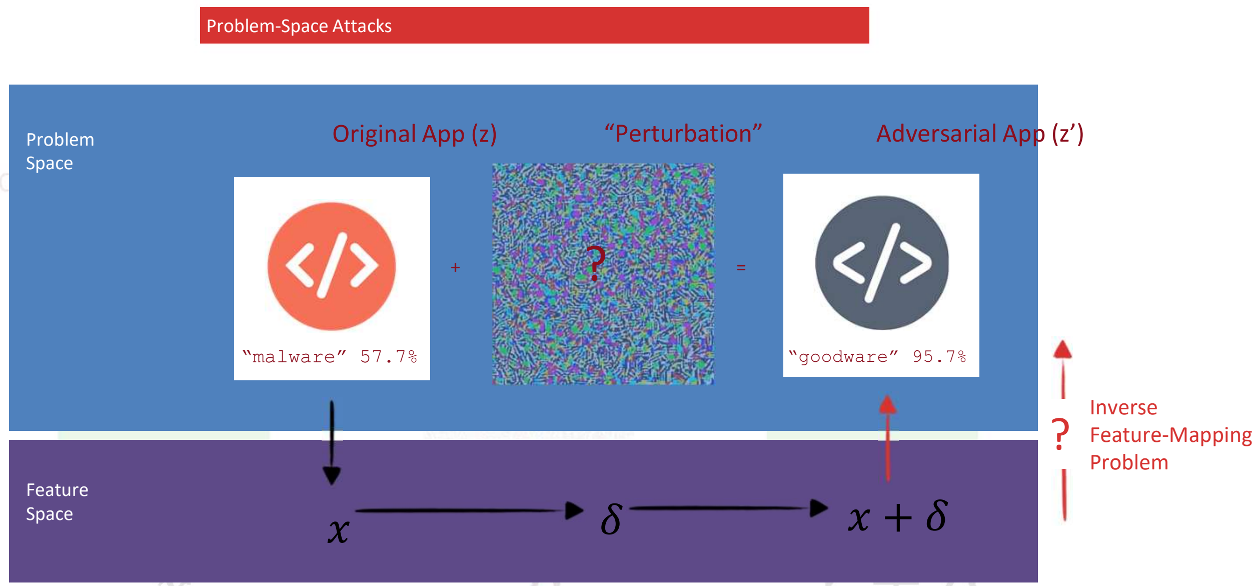
This optimization problem can be solved in different ways, i.e., different attacks, e.g., FGSM, PGD, Carlini and Wagner, etc – see the CyBoK KG

[CVPR 2014] Intriguing Properties of Neural Networks

[CVPR 2018] Robust Physical-World Attacks on Deep Learning Models

What happens in the problem space, i.e., the real world?

Let’s Analyze What Happened



Optimization | $minimize_{\delta} \cancel{||\delta||_p} + c \cdot f(x+\delta)$

Constraints

- Is it realistic/plausible?
- Does it crash?
- Does it preserve malicious functionality?
- … are there “general" constraints?

[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space

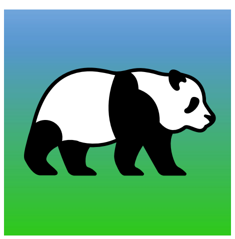
Inverse Feature-Mapping Problem

CyBOK

Problem
Space

Images

“panda” 57.7%



“gibbon” 99.3%

Feature
Space

Vectors

φ

x

1	0	0.5	1	...	1

FS attack

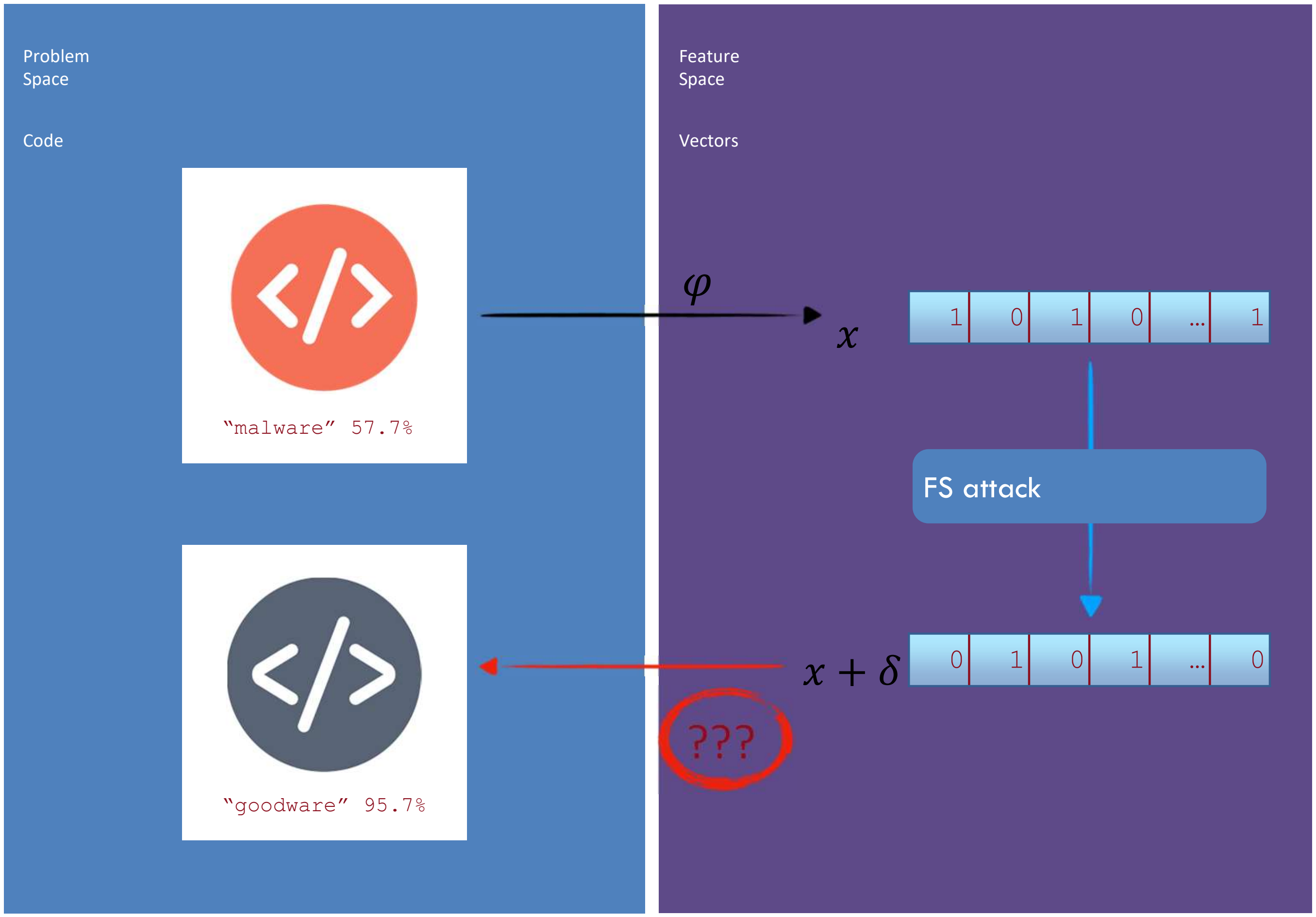
$x + \delta$

0.2	0.1	0	0	...	0.3

$\approx \varphi^{-1}$

The feature mapping φ is differentiable
— you can backpropagate to input

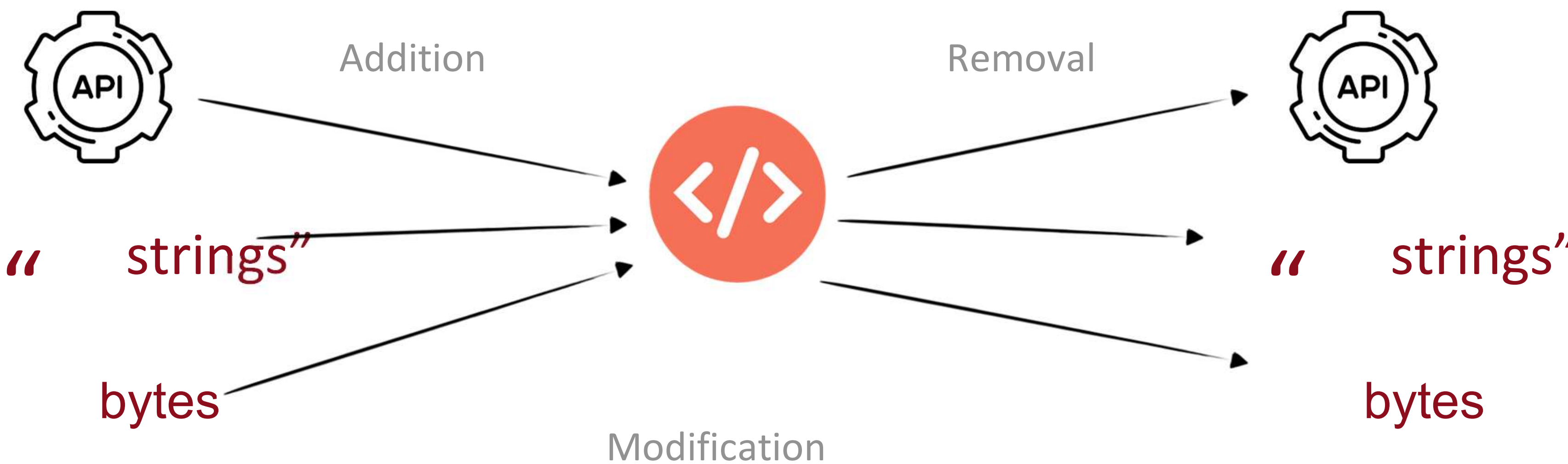
Inverse Feature-Mapping Problem



In the software domain,

the feature mapping φ is
neither <u>invertible</u> nor <u>differentiable</u>
— how to get back to the problem space?

Problem-Space Constraints



[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

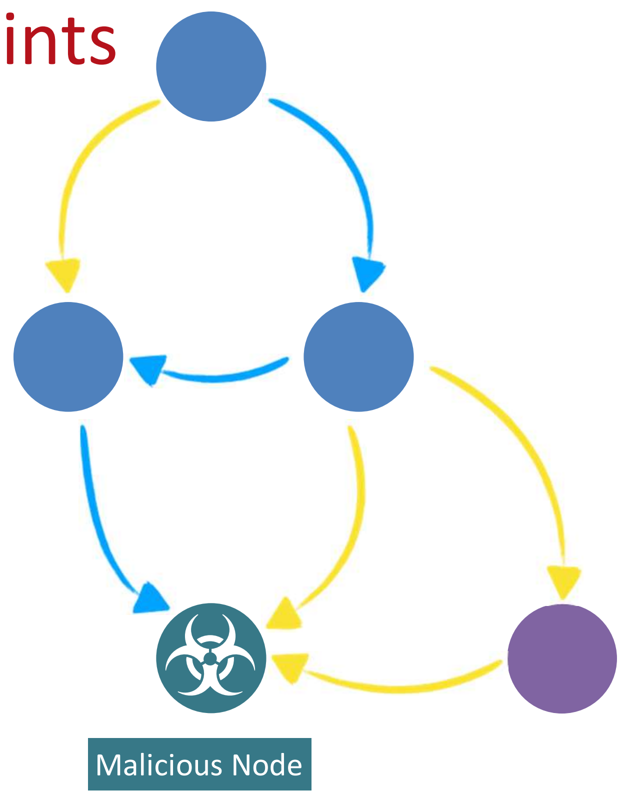
Available Transformations
Preserved Semantics

Which semantics do you preserve? How?
Which automatic tests can verify it?

Problem-Space Constraints

Test Suite
- Does it crash?
- Does it still communicate with CnC?
- Does it still encrypt the /home/ folder?

By Construction
- Add no-op operations
- Ensure it is not executed at runtime



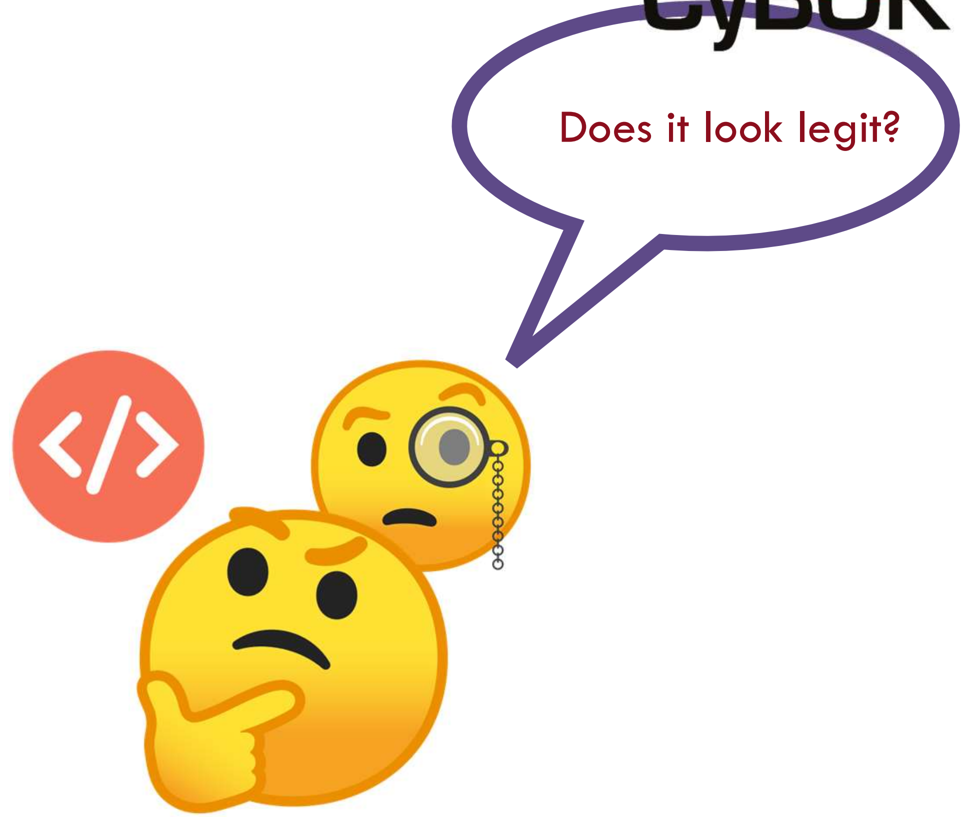
[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

Problem-Space Constraints

Plausibility

Preserved Semantics

Available Transformations



Test Suite
- User studies
- Automated heuristics

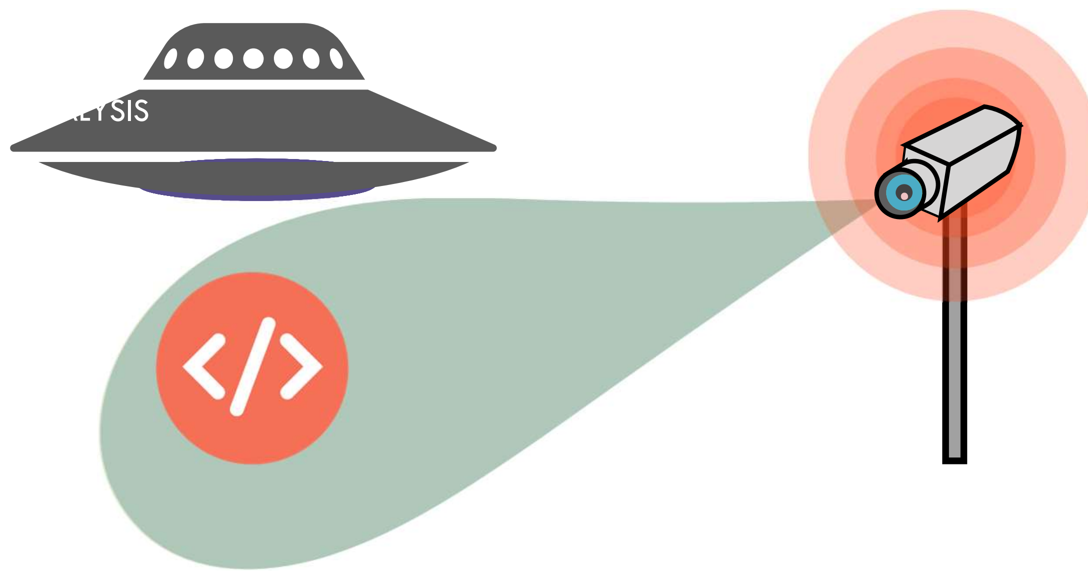
By Construction
- Taking precautions during mutation

[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

Problem-Space Constraints

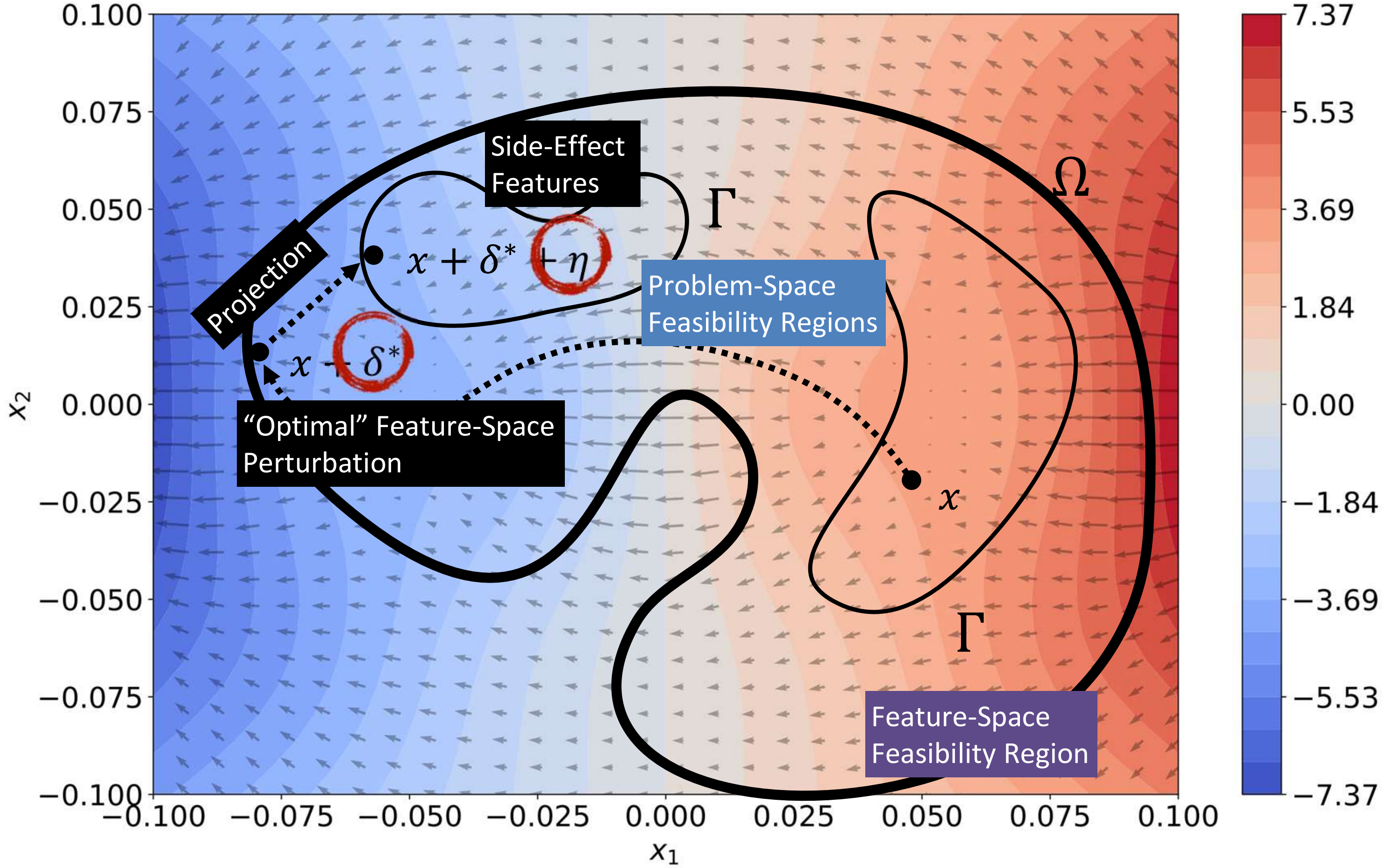
- Plausibility
- Robustness to Preprocessing
- Preserved Semantics
- Available Transformations

Which preprocessing are you considering?



[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

Side-effect Features



[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

Feature Space vs. Problem Space

$$\boldsymbol{\delta}^* = \arg\min_{\boldsymbol{\delta} \in \mathbb{R}^n} \quad f_t(\boldsymbol{x} + \boldsymbol{\delta})$$
$$\text{subject to:} \quad \boldsymbol{\delta} \models \Omega\,.$$

$$\begin{aligned} \text{argmin}_{\mathbf{T} \in \mathcal{T}} \quad & f_t(\varphi(\mathbf{T}(z))) = f_t(\boldsymbol{x} + \boldsymbol{\delta}^* + \boldsymbol{\eta}) \\ \text{subject to:} \quad & [\![z]\!]^\tau = [\![\mathbf{T}(z)]\!]^\tau, \quad \forall \tau \in \Upsilon \\ & \pi(\mathbf{T}(z)) = 1, \quad \forall \pi \in \Pi \\ & \mathbf{A}(\mathbf{T}(z)) = \mathbf{T}(z), \quad \forall \mathbf{A} \in \Lambda \end{aligned}$$

Feature-Space Constraints

- Lp perturbations
- Domain constraints for vectors

Search Strategy

- Gradient-driven

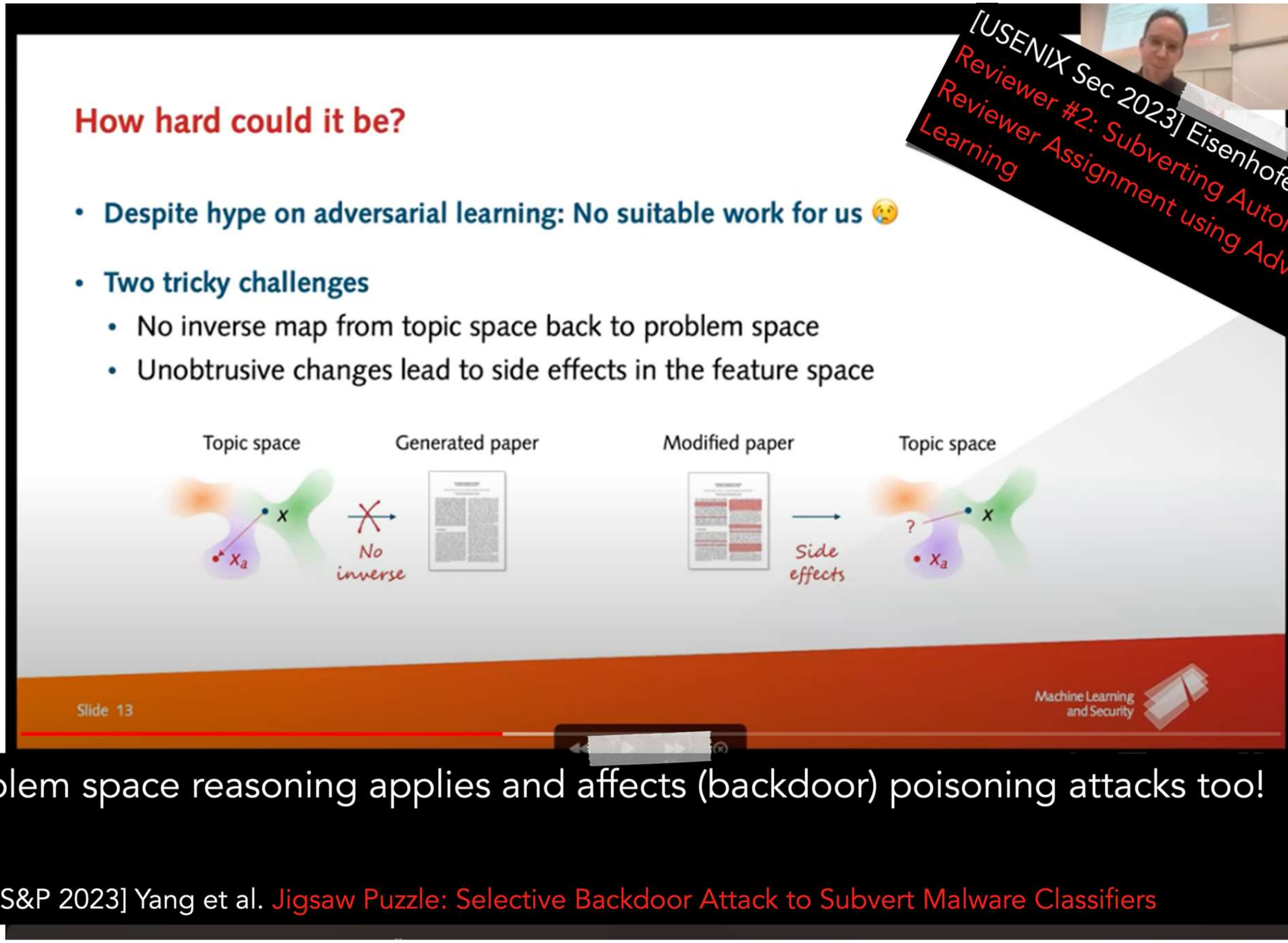
Problem-Space Constraints

- Available Transformations
- Preserved Semantics
- Plausibility
- Robustness to Preprocessing

Search Strategy

- Gradient-driven
- Problem-driven
- Hybrid

[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing



[USENIX Sec 2023] Eisenhofer et al. No more Reviewer #2: Subverting Automatic Paper-Reviewer Assignment using Adversarial Learning

Problem space reasoning applies and affects (backdoor) poisoning attacks too!

[IEEE S&P 2023] Yang et al. Jigsaw Puzzle: Selective Backdoor Attack to Subvert Malware Classifiers

Inference Time Defenses*

* Focus on Adversarial Training – details on OOD detection, certified models, and defenses against training time attacks (e.g., poisoning and backdoor) in the CyBoK KG

Adversarial Training

- Widely used defense technique
- Idea: augment the training dataset with adversarial examples
  - It enables the model to learn robust features
  - It helps the model become more resistant to adversarial perturbations
- Successful in limiting the attack success rate for a given set of perturbations (attacks)
- Affects performance against clean data

Set of allowed perturbations

$$\min_{\theta} \mathbb{E}(x, y) \sim D \, [\max \delta \in S, L(f_{\theta}(x + \delta), y)]$$

Training distribution

The model

- What does it happen in the problem space?

Problem vs Feature Space Adversarial Training

- Exciting work [IEEE S&P 2023] on Text, Botnet Traffic, Windows Malware Classification Tasks
  - Text: Problem Space AT 16.94% more effective than Feature Space AT
  - Botnet Traffic: Problem Space AT robustness ~= Feature Space AT
  - Windows Malware: Problems Space AT outperforms Feature Space AT robustness

[IEEE S&P 2023] Dyrmishi et al. On The Empirical Effectiveness of Unrealistic Adversarial Hardening Against Realistic Adversarial Attacks

Problem vs Feature Space Adversarial Training

- Exciting work [IEEE S&P 2023] on Text, Botnet Traffic, Windows Malware Classification Tasks
  - (Marginal) Text: Problem Space AT 16.94% more effective than Feature Space AT
  - (Not Required) Botnet Traffic: Problem Space AT robustness ~= Feature Space AT
  - (Required) Windows Malware: Problems Space AT > Feature Space AT robustness
- It may seem a task-dependent result…

[IEEE S&P 2023] Dyrmishi et al. On The Empirical Effectiveness of Unrealistic Adversarial Hardening Against Realistic Adversarial Attacks

Problem vs Feature Space Adversarial Training

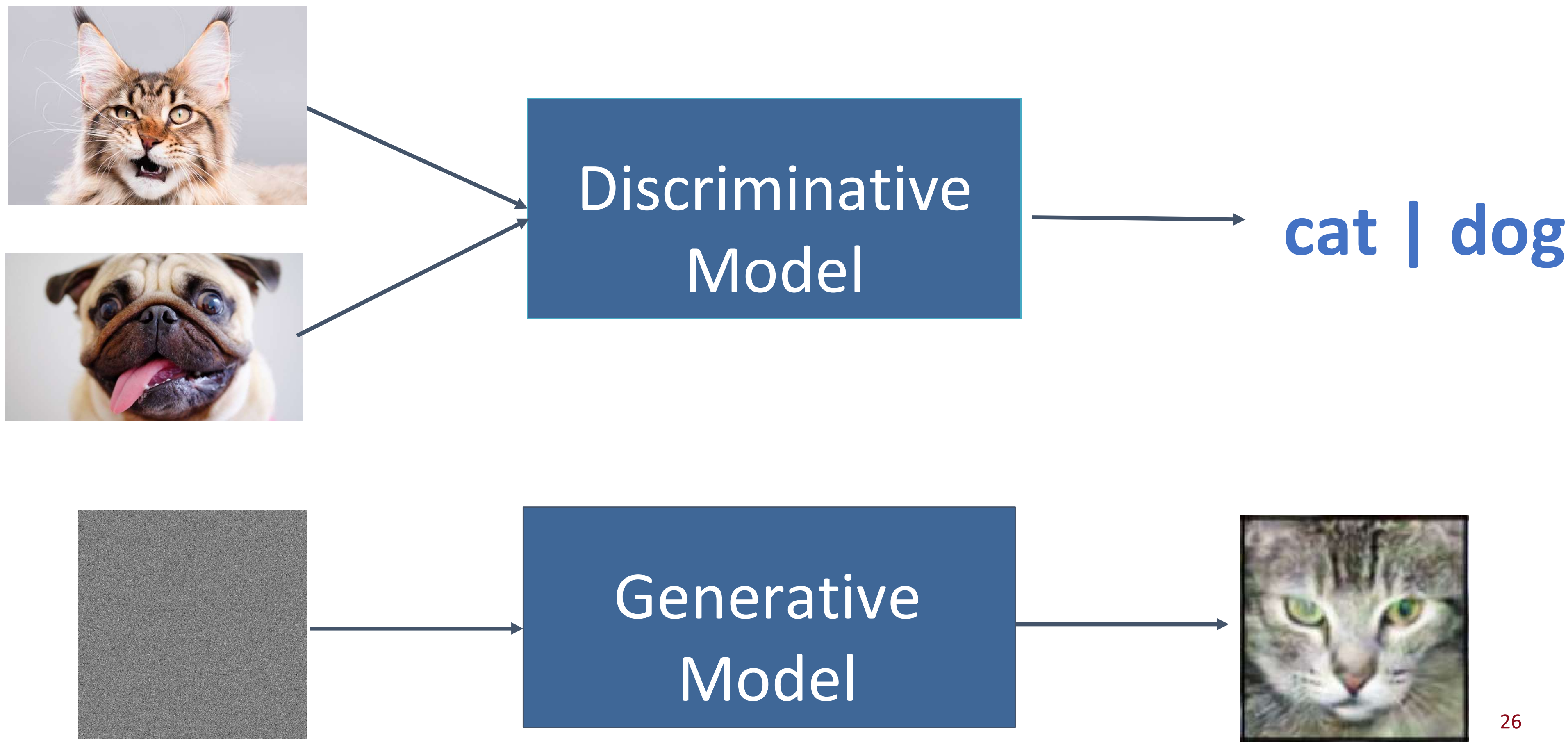
- Exciting work [IEEE S&P 2023] on Text, Botnet Traffic, Windows Malware Classification Tasks
  - (Marginal) Text: Problem Space AT 16.94% more effective than Feature Space AT
  - (Not Required) Botnet Traffic: Problem Space AT robustness ~= Feature Space AT
  - (Required) Windows Malware: Problems Space AT > Feature Space AT robustness

Perhaps not task-dependent but affected by

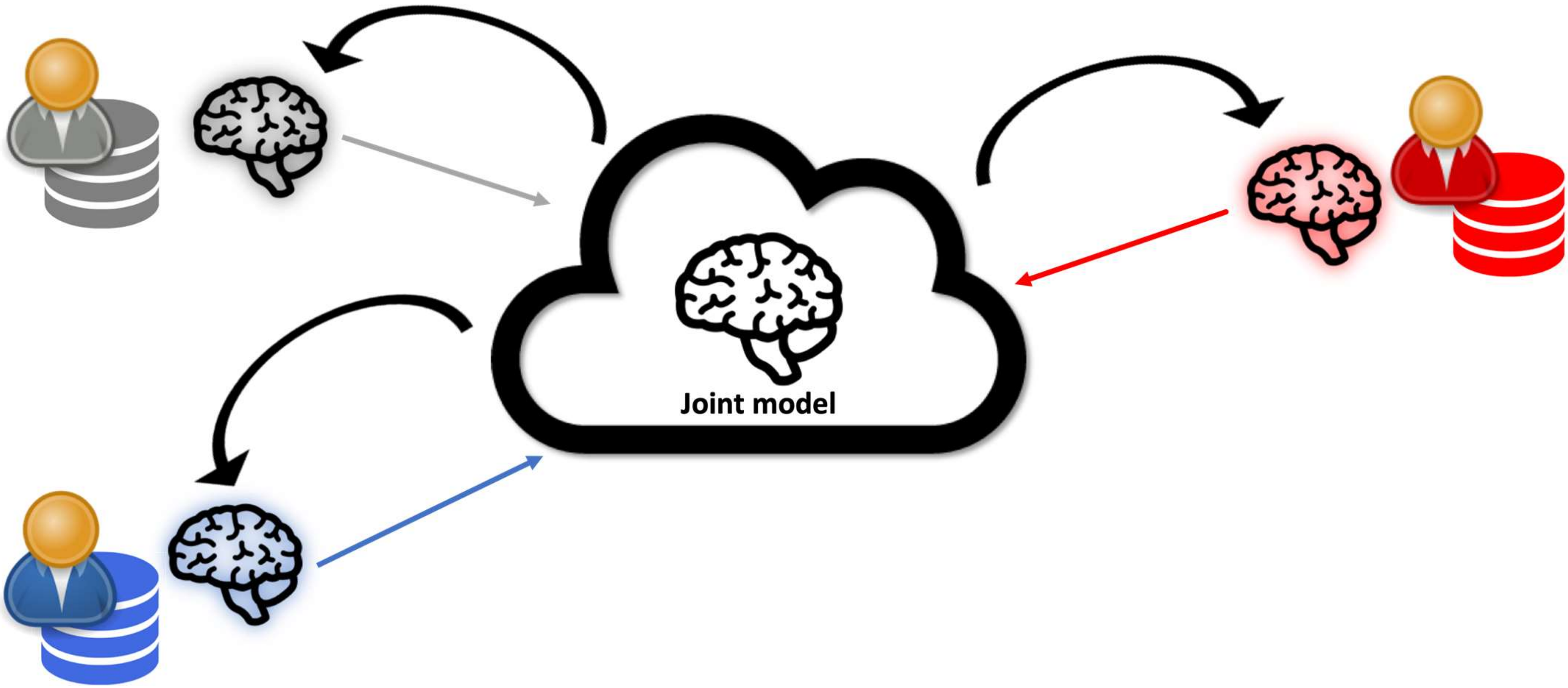
- Program abstractions
- Feature representations
- ML models

Further details on Adversarial Training and other defenses to adversarial attacks, such as OOD detection, certified models, and defenses against training time attacks (e.g., poisoning and backdoor) in the CyBoK KG

Discriminative vs Generative Models



Collaborative/Federated Learning



Background

Privacy Tech

- Cryptography
- Differential Privacy

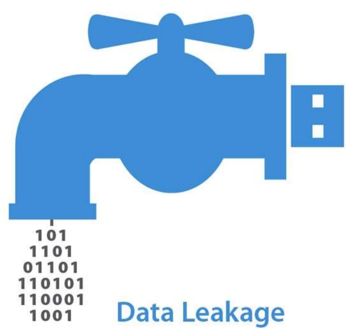
Adversarial Modeling

- Access (white vs black box), target (training vs inference), mode (passive vs active)

Reasoning about "privacy" in ML

Most privacy attacks in ML focus on inferring either:

1. Inclusion of a data point in the training set (aka "membership inference")
2. What class representatives (in training set) look like (aka "model inversion")



1. Membership Inference

Adversary wants to test whether data of a target victim has been used to train a model

- Serious problem if inclusion in training set is privacy-sensitive
- E.g., main task is: predict whether a smoker gets cancer
- [Shokri et al., S&P'17] show it for discriminative models
- [Hayes et al. PETS'19] for generative models

Membership inference is a very active research area, not only in machine learning…

Membership Inference (cont'd)

Membership inference is a very active research area, not only in machine learning…

- Given f(data), infer if x ∈ data (e.g., f is aggregation)
- [HSR+08, WLW+09] for genomic data
- [Pyrgelis et al., NDSS'18] for mobility data

Well-understood problem (besides leakage)

- Use it to establish wrongdoing
- Or to assess protection, e.g., with differentially private noise

2. Inferring Class Representatives

Research focused on properties of an en

Model Inversion [Fredrikson et al. CCS'1

GAN attacks [Hitaji et al. CCS'17]

E.g.: given a gender classifier, infer what

lik

Bu

so ata was

sa

Privacy leakage !=

Adv learns something about t

data

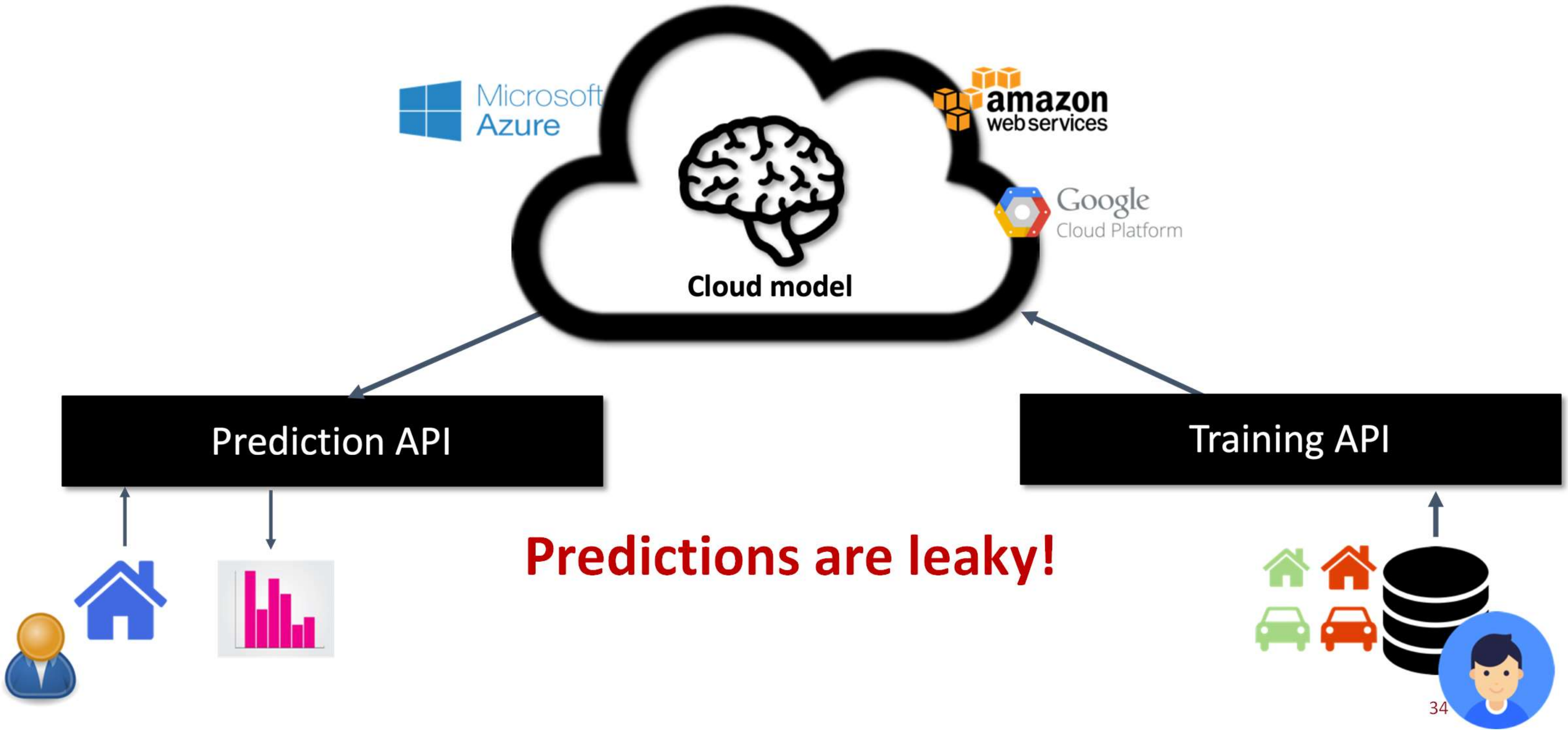
Property Inference

How about if we inferred properties of a subset of the training inputs…

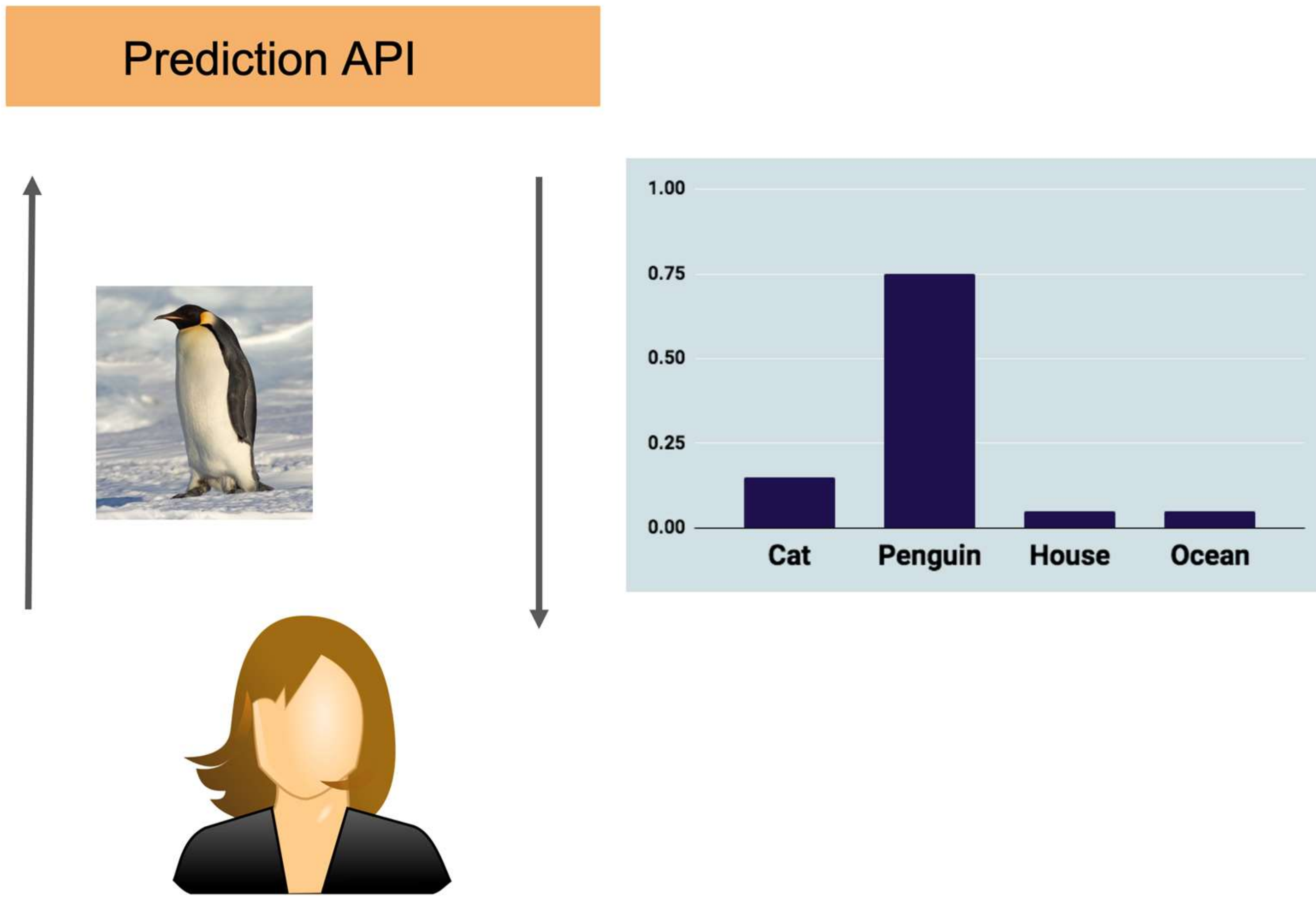
…but not of the whole class?

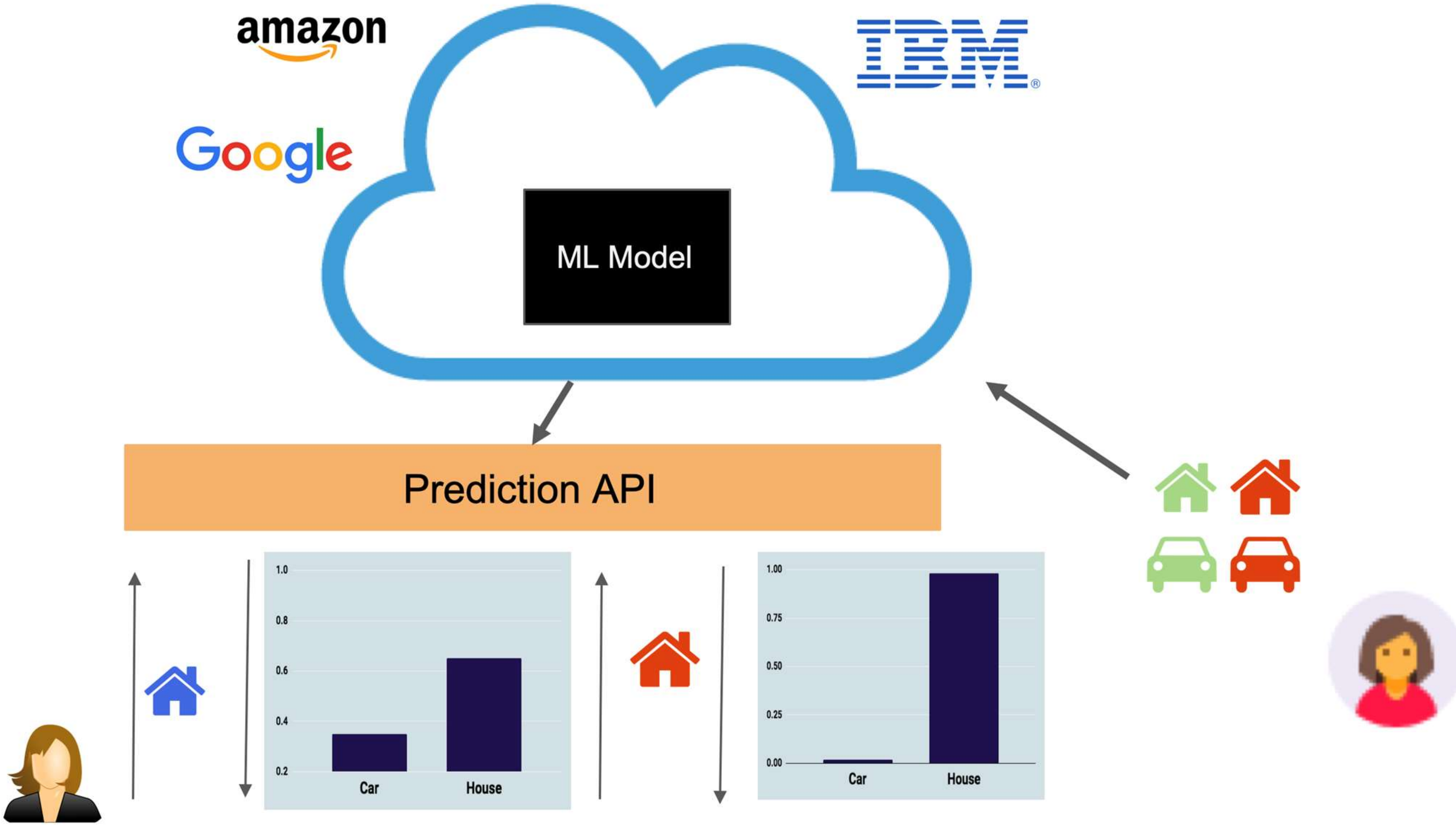
In a nutshell: given a gender classifier, infer race of people in Bob's photos

Machine Learning as a Service

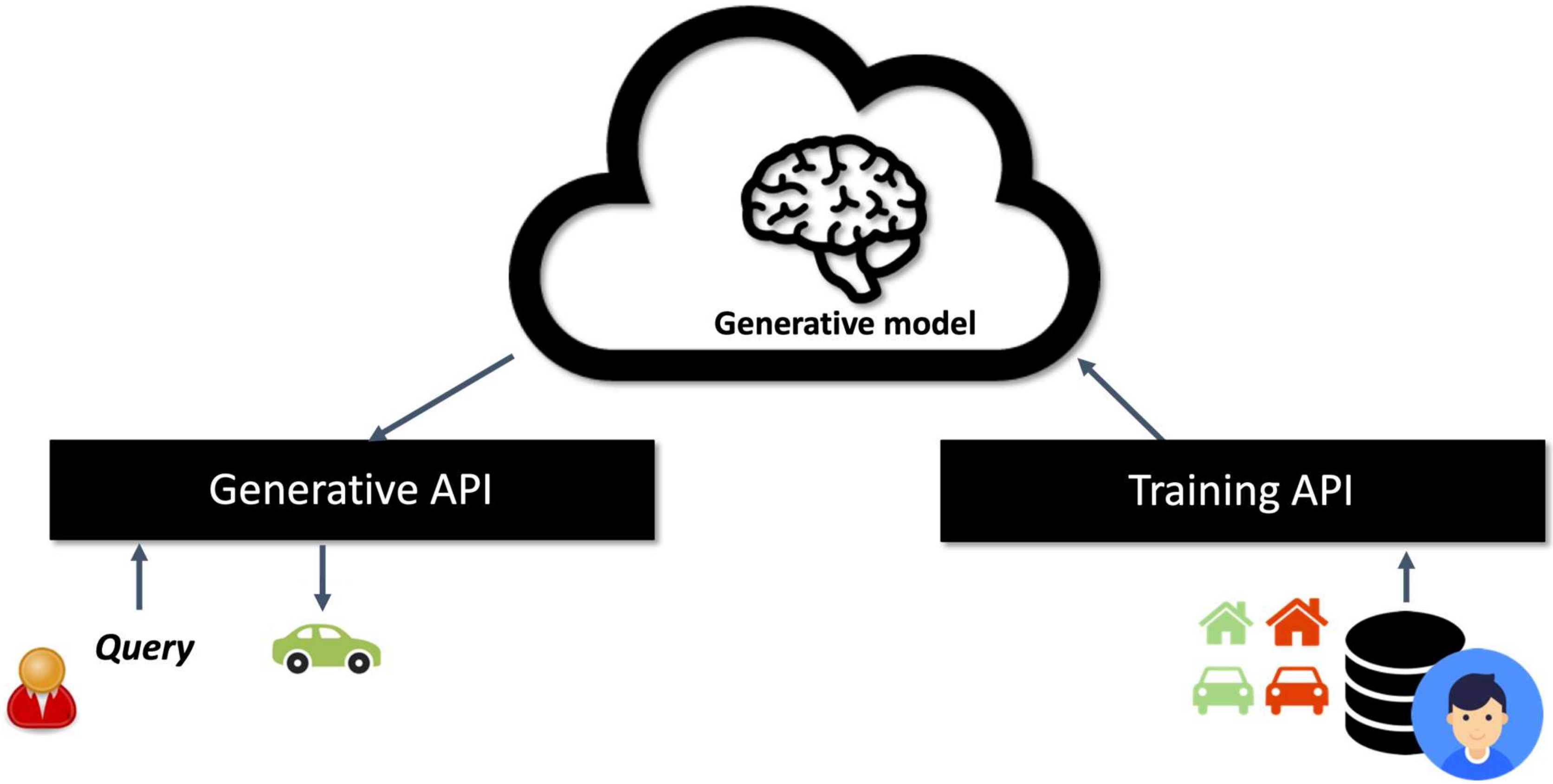


Membership Inference/Discriminative




amazon
Google
IBM
ML Model
Prediction API
1.0
0.8
0.6
0.4
0.2
Car
House
1.00
0.75
0.50
0.25
0.00
Car
House

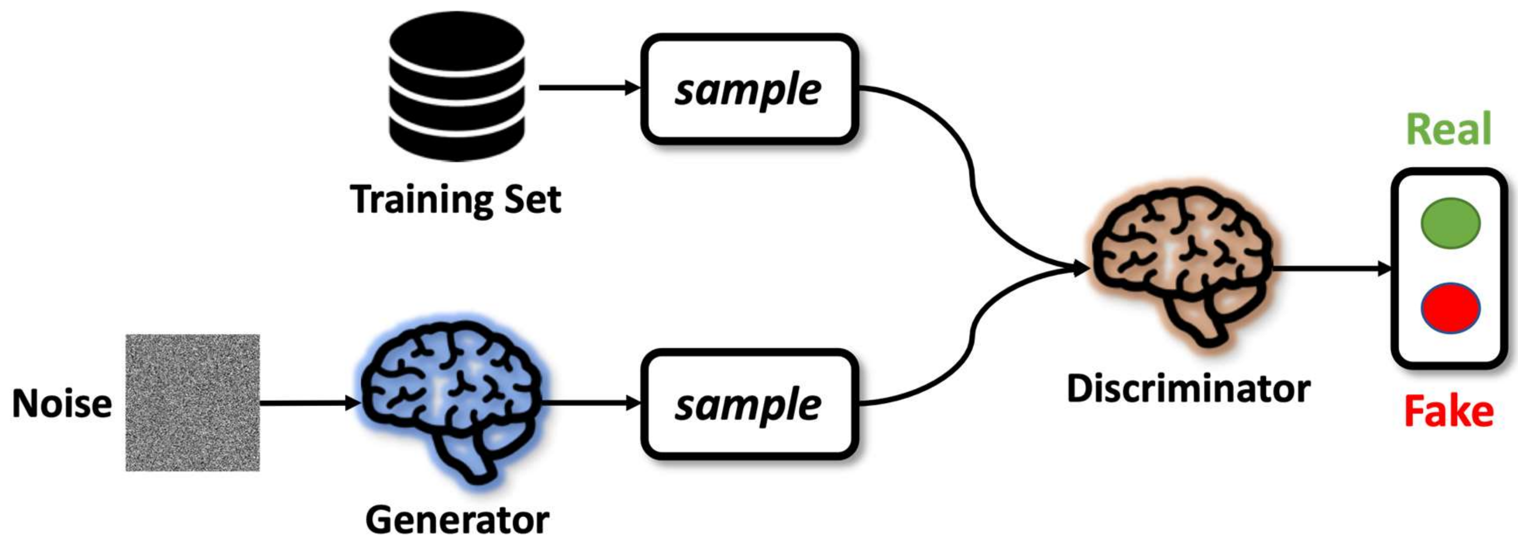
Membership Inference in Generative Models



Inference without predictions?

Use generative models!

Train GANs to learn the distribution and a prediction model at the same time



Collaborative

Algorithm 1 Parameter server with synchronized SGD

Server executes:

Initialize θ_0

for $t = 1$ to T **do**

  for each client k **do**

    $g_t^k \leftarrow$ **ClientUpdate**(θ_{t-1})

  end for

  $\theta_t \leftarrow \theta_{t-1} - \eta \sum_k g_t^k$

end for

ClientUpdate(θ):

Select batch b from client's data

return local gradients $\nabla L(b; \theta)$

Federated

Algorithm 2 Federated learning with model averaging

Server executes:

Initialize θ_0

$m \leftarrow max(C \cdot K, 1)$

for $t = 1$ to T **do**

  $S_t \leftarrow$ (random set of m clients)

  for each client $k \in S_t$ **do**

    $\theta_t^k \leftarrow$ **ClientUpdate**(θ_{t-1})

  end for

  $\theta_t \leftarrow \sum_k \frac{n^k}{n} \theta_t^k$

end for

ClientUpdate(θ):

for each local iteration **do**

  for each batch b in client's split **do**

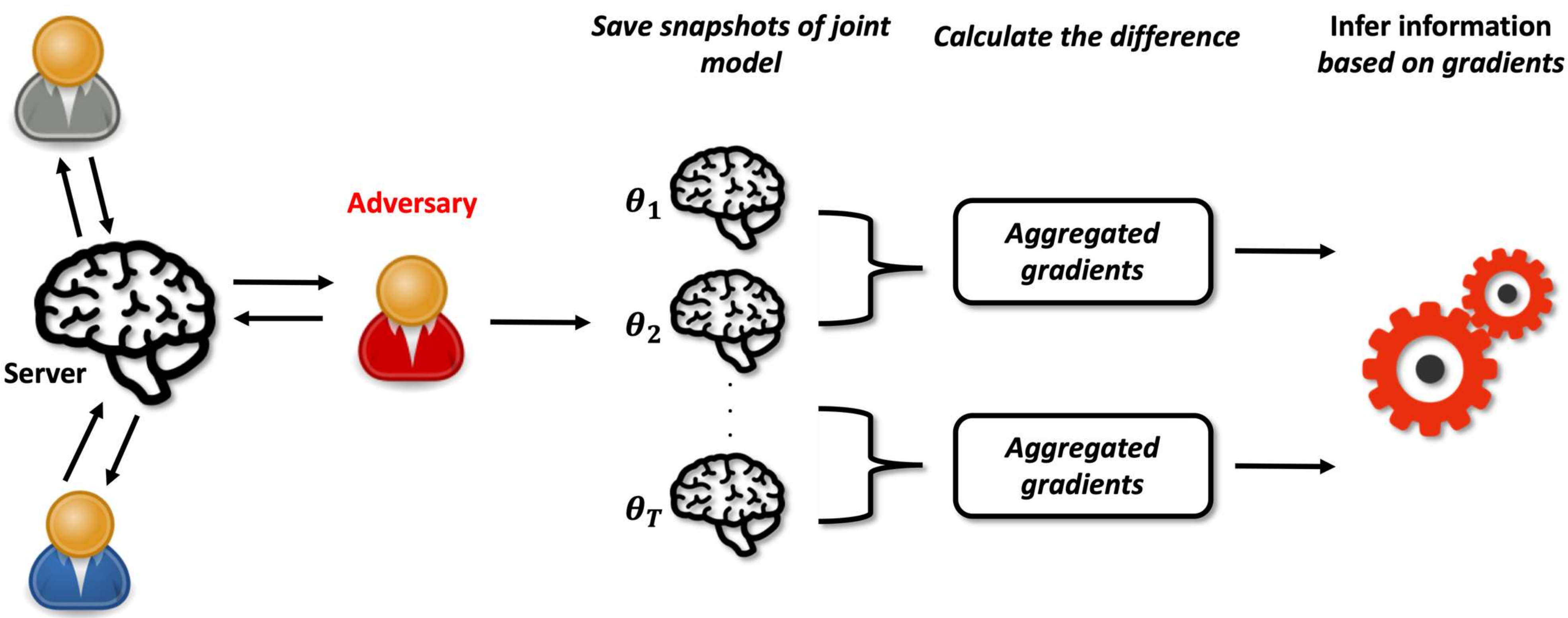
    $\theta \leftarrow \theta - \eta \nabla L(b; \theta)$

  end for

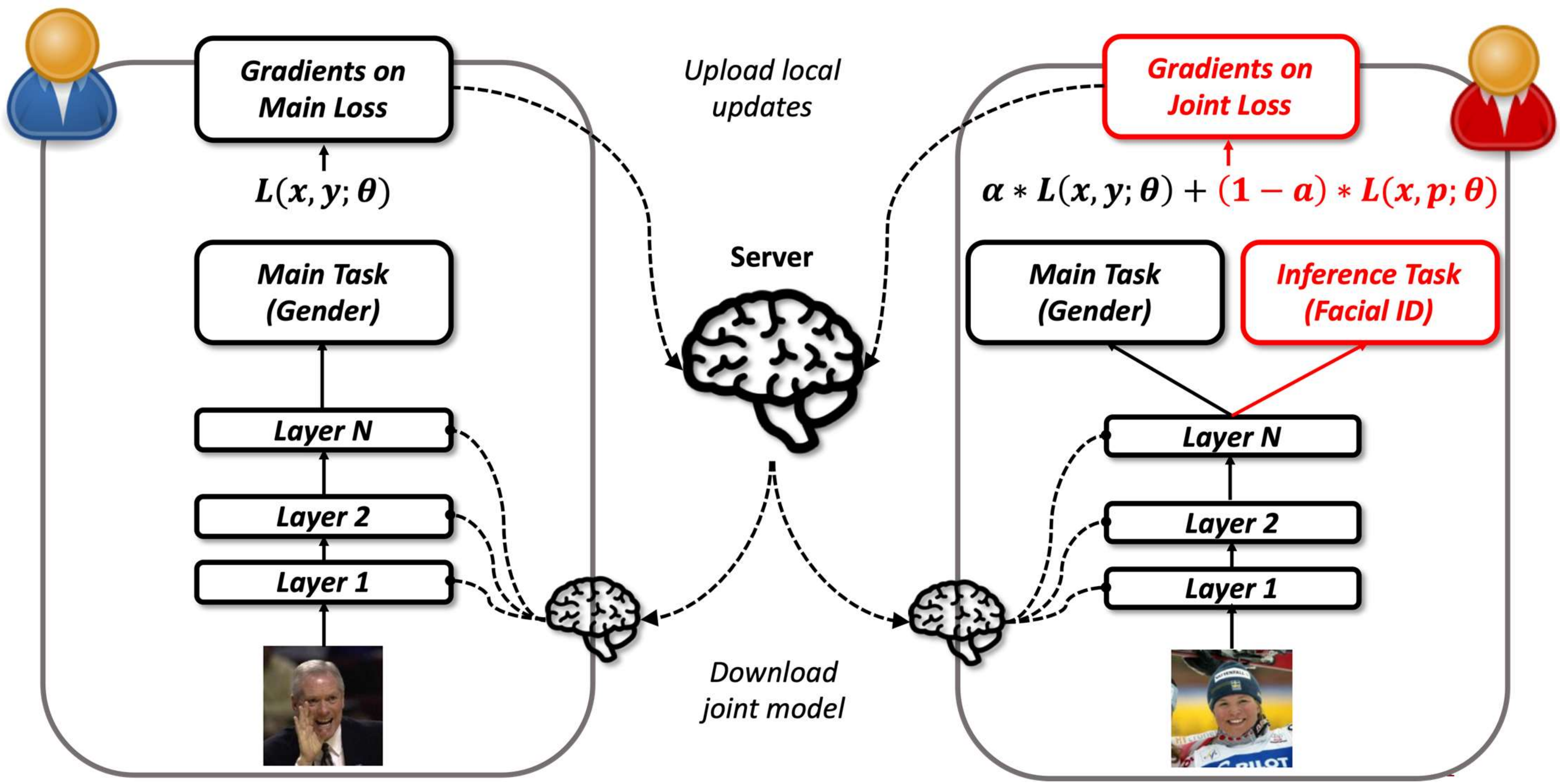
end for

return local model θ

Passive Property Inference Attack



Active Property Inference Attack



More in the KG…

Model Extraction

An adversary with black-box access, but no prior knowledge of an ML model's parameters or training data, steals model parameters

Functionality Extraction

Create knock-offs of a model

Defenses

Using cryptography, differential privacy, or trusted hardware

Opening the ML "box"

Privacy Take-Aways

1. Membership inference attacks are pretty accurate.
2. Threats from model inversion are sometimes unclear.
3. Federated learning not a panacea.
4. Policy implications still to be explored.
5. Need for actual evaluation frameworks.

Looking Forward

- Lots of open technical problems remain unaddressed
  - E.g., adversarial drifts, adaptive attackers
- More work required on non-technical aspects
  - E.g., ethical, societal, and legal implications of AI and in particular Large Language Models
- Unintended effects of defenses
  - E.g., reduced accuracy for under-represented groups?

CyBOK