“Meta Concerns” in Building Trustworthy ML Systems

Vasisht Duddu

vasisht.duddu@uwaterloo.ca

(Joint work with N. Asokan, Anudeep Das, Lachlan Gunn, Nora Khayata, Thomas Schneider, Sebastian Szyller, Hossein Yalame, Rui Zhang)

Who am I?

PhD student, University of Waterloo (Canada)

Advisor: N. Asokan

***Previously:** Masters @ University of Waterloo, Undergraduate @ IIIT-Delhi, India*

IBM PhD Fellowship, Distinguished Paper @ IEEE S&P, Mastercard's Cybersecurity and Privacy Excellence Graduate Scholarship, David R. Cheriton Scholarship

https://vasishtduddu.github.io/ for more background

https://vasishtduddu.github.io/

Past Research

- Fault tolerance of neural networks and its relation to robustness and privacy
- Privacy attacks, model extraction attacks and ownership verification
- Interactions of privacy with fairness and model explanations

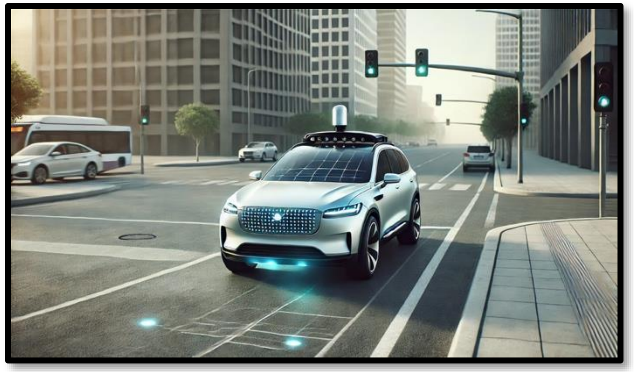
Machine Learning works…..

…..and being considered for applications with high-stakes decision-making

Criminal Recidivism

Healthcare

Mortgage Applications



Autonomous Vehicles

…..and many more

Images generated by ChatGPT

But, susceptible to various security, privacy, and fairness risks

Risks to ML Systems

Evasion: Force model to misclassify perturbed input[1]

Poisoning: Add poisons to degrade utility or generate adversary-chosen output[2]

Unauthorized Model Ownership: Steal functionality of target model[3]

Unauthorized Data Usage: Use of copyrighted or personal data without consent[4]

(Security Risks)

Inference Attacks: Infer unobservable "sensitive" information from model[5]

(Privacy Risks)

Bias: Different behavior on different demographic subgroups[6]

Incomprehensible: Unclear why model gave specific output[7]

(Fairness Risks)

[1] Crocce and Hein. *Reliable Evaluation of Adversarial Robustness with an Ensemble of Diverse Parameter-Free Attacks*. ICML 2020
[2] Wenger et al. *Backdoor Attacks against Deep Learning Systems in the Physical World*. CVPR 2021.
[3] Orekondy et al. *Knockoff-Nets: Stealing Functionality of Black-Box Models*. CVPR 2019.
[4] New York Times. *The Times Sues OpenAI and Microsoft over AI Use of Copyrighted Work*. 2023.
[5] Rigaki and Garcia. *A Survey of Privacy Attacks in Machine Learning*. ACM Computing Surveys. 2023.
[6] Hardt et al. *Equality of Opportunity in Supervised Learning*. NeurIPS. 2016.
[7] Lundberg and Lee. *A Unified Approach to Interpreting Model Predictions*. NeurIPS 2017.

Defenses against ML Risks

Evasion: Adversarial training[1]

Poisoning: Data Sanitization[2], Fine-tune[3], Pruning[4]

Unauthorized Model Ownership: Watermarking[5,6] and Fingerprinting[7]

Unauthorized Data Usage: Watermarking[8]

(Security Risks)

Not Enough to Design Effective Defenses against Individual Risks

Inference Attacks: Differential Privacy (Synthetic Data[9], DPSGD[10])

(Privacy Risks)

Bias: Synthetic Data[11], Regularization[12], Calibration[13]

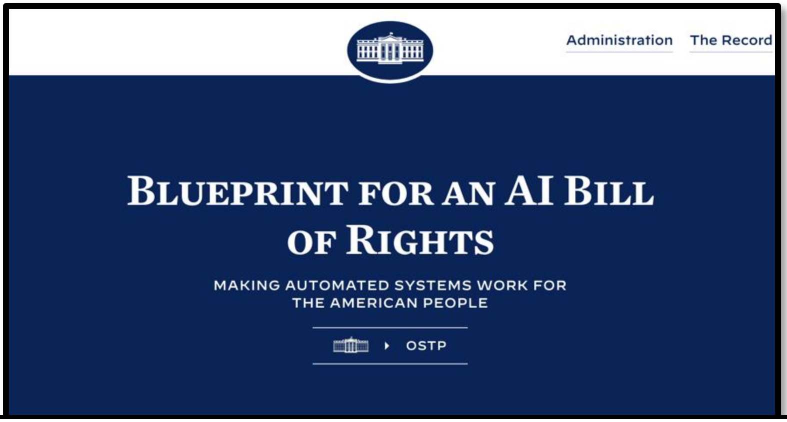
Incomprehensible: Model explanations[14]

(Fairness Risks)

[1] Madry et al. Towards Deep Learning Models Resistant to Adversarial Attacks. ICML 2018
[2] Borgnia et al. Strong Data Augmentation Sanitizes Poisoning and Backdoors Attacks without an Accuracy Trade-off. ICASSP 2021.
[3] Patrini et al. Making Deep Neural Networks Robust to Label Noise: A Loss Correction Approach. CVPR 2017.
[4] Li et al. Reconstructive Neuron Pruning for Backdoor Defense. ICML 2023.
[5] Adi et al. Tuning your Weakness into a Strength: Watermarking Deep Neural Networks by Backdoors. USENIX Sec 2018.
[6] Szyller et al. DAWN: Dynamic Adversarial Watermarking of Neural Networks. ACM MM. 2021.
[7] Waheed et al. GrOVe: Ownership Verification of Graph Neural Networks using Embeddings. IEEE S&P 2024. (Our work)
[8] Chen et al. Catch Me if You Can: Detecting Unauthorized Data Use In Training Deep Learning Models. CCS 2024.
[9] Lin et al. Differentially Private Synthetic Data via Foundation Model APIs 1: Images. ICLR 2024.
[10] Abadi et al. Deep Learning with Differential Privacy. CCS 2016.
[11] Zemel et al. Learning Fair Representations. ICML 2013.
[12] Hardt et al. Equality of Opportunity in Supervised Learning. NeurIPS 2016.
[13] Pleiss et al. On Fairness and Calibration. NeurIPS 2017.
[14] Lundberg and Lee. A Unified Approach to Interpreting Model Predictions. NeurIPS 2017

AI Regulations

AI Bill of Rights (White House)



"Safe and effective systems"…. "algorithmic discrimination protections"…."data privacy"…."Notice and explanations"

European Union's AI Act

"Establish a risk management system"…. "conduct data governance"…."appropriate levels of accuracy, robustness"

Practitioners should:

(1) Ensure models satisfy all desirable ML properties (e.g., security, privacy, and fairness)

(2) Demonstrate compliance with the regulations

Talk Outline

“Meta Concerns” for Building Trust in ML Systems

- What are the unintended implications of applying defenses?
- How can we protect against multiple risks simultaneously?
- How can we design efficient mechanisms to demonstrate ML properties?

Unintended Interactions among Defenses and Risks

Effective defense may increase or decrease susceptibility to other (unrelated) risks

- Adversarial training may increase susceptibility to membership inference[1]

Limited evaluation for some risks, defenses, interactions[2,3,4] or underlying causes[2,3]

No systematic framework to explore unintended interactions

[1] Song et al. *Privacy Risks of Securing Machine Learning Models against Adversarial Examples*. CCS 2019.
[2] Ferry et al. *SoK: Taming the Triangle - On the Interplays between Fairness, Interpretability and Privacy in Machine Learning*. arXiv 2024.
[3] Gittens et al. *An Adversarial Perspective on Accuracy, Robustness, Fairness, and Privacy: Multilateral-Tradeoffs in Trustworthy ML*. IEEE Access 2024.
[4] Strobel and Shokri. *Data Privacy and Trustworthy Machine Learning*. IEEE S&P Magazine 2022.

Overview of Unintended Interactions

Explore pairwise interactions between each defense and all unrelated risks:

Defenses	Risks
RD1 (Adversarial Training) RD2 (Outlier Removal)	R1 (Evasion) R2 (Poisoning)
RD3 (Watermarking) RD4 (Fingerprinting)	R3 (Unauthorized Ownership)
PD1 (Differential Privacy)	P1 (Membership Inference) P2 (Data Reconstruction) P3 (Attribute Inference) P4 (Distribution Inference)
FD1 (Group Fairness) FD2 (Explanations)	F (Discriminatory Behaviour)

Overfitting and memorization are underlying causes (conjecture)

- Effective defenses may induce, reduce or rely on overfitting or memorization
- Risks tend to exploit overfitting or memorization

Factors Influencing Overfitting and Memorization

O1 **Curvature smoothness of the objective function**

O2 **Distinguishability across datasets (O2.1), subgroups (O2.2), and models (O2.3)**

O3 **Distance of training data to decision boundary**

(Objective function-related)

D1 **Size of training data**

D2 **Tail length of distribution**

D3 **Number of attributes**

D4 **Priority of learning stable attributes**

(Dataset-related)

M1 **Model capacity**

(Model-related)

Situating Prior Work in our Framework

Risk increases (● red) or decreases (● green) or unexplored (● gray) when a defense is effective

Evaluate the influence of factors empirically (●), theoretically (⊙), conjectured (○)

Defenses	Risks		OVFT D1	Memorization D2	D3	D4	O1	O2	Both O3	M1	References
RD1 (Adversarial Training)	R1 (Evasion)	● (green)		●			●		●	●	[193], [102], [91], [173]
	R2 (Poisoning)	● (red)									[170], [153]
	R3 (Unauthorized Model Ownership)	● (red)	○								[86] ([95]: ● green)
	P1 (Membership Inference)	● (red)	⊙, ●					1: ●		●	[144], [67]
	P2 (Data Reconstruction)	● (red)				○				●	[195], [111]
	P3 (Attribute Inference)	● (gray)									
	P4 (Distribution Inference)	● (red)				○					[148]
	F (Discriminatory Behaviour)	● (red)		⊙, ●							[16], [36], [71], [99]
RD2 (Outlier Removal)	R1 (Evasion)	● (green)									[59]
	R2 (Poisoning)	● (green)									[154]
	R3 (Unauthorized Model Ownership)	● (gray)									
	P1 (Membership Inference)	● (red)		●							[25], [46]
	P2 (Data Reconstruction)	● (gray)									
	P3 (Attribute Inference)	● (red)		●							[78]
	P4 (Distribution Inference)	● (gray)									
	F (Discriminatory Behaviour)	● (red)	●	○							[134]
RD3 (Watermarking)	R1 (Evasion)	● (gray)									
	R2 (Poisoning)	● (red)		○							[133], [3], [194], [93]
	R3 (Unauthorized Model Ownership)	● (green)		○				3: ●	●		[152], [3], [98]
	P1 (Membership Inference)	● (red)		○				1: ●	●		[157], [33]
	P2 (Data Reconstruction)	● (red)		○				1: ●	●		[157]
	P3 (Attribute Inference)	● (red)		○				2: ●	●		[157]
	P4 (Distribution Inference)	● (red)	⊙, ●	○				1: ●	●	●	[30], [105]
	F (Discriminatory Behaviour)	[illegible]		○							[illegible]

Revisiting ML Risks and Defenses

Effectiveness of defense <d> correlates with a change in factor <f>

Change in <f> correlates with change in susceptibility to risk <r>

- ↑: positive correlation; ↓: negative correlation

Defences (<↑ or ↓>, <f>)	**Risks (<↑ or ↓>, <f>)**
RD1 (Adversarial Training): • D1 ↑, $\|\mathcal{D}_{tr}\|$ [161] • D2 ↓, tail length [71], [16] • D4 ↑, priority for learning stable attributes [161] • O1 ↑, curvature smoothness [102] • O2.1 ↑, distinguishability in data records inside and outside $\mathcal{D}_{tr}$ [144] • O3 ↑, distance to boundary for most $\mathcal{D}_{tr}$ data records [176] • M1 ↑, model capacity [102] **RD2 (Outlier Removal):** • D2 ↑, tail length [166] **RD3 (Watermarking):** • D2 ↑, tail length [96] • O2.3 ↓, distinguishability in observables for watermarks between f_θ and f_θ^{der}, but distinct from independent models [3] • M1 ↑, model capacity [3]	**R1 (Evasion):** • D2 ↑, tail length [173], [91] • O1 ↓, curvature smoothness [102] • O3 ↓, distance of $\mathcal{D}_{tr}$ data records to boundary [162] **R2 (Poisoning):** • D2 ↑, tail length [120], [17], [96] • M1 ↑, model capacity [3] **R3 (Unauthorized Model Ownership):** • M1 ↓, model capacity [117], [88] **P1 (Membership Inference):** • D1 ↓, $\|\mathcal{D}_{tr}\|$ [184], [136] • D2 ↑, tail length [25], [24] • D4 ↓, priority for learning stable attributes [103], [155] • O2.1 ↑, distinguishability for data records inside and outside $\mathcal{D}_{tr}$ [136] • O3 ↓, distance to decision boundary

Guideline to Conjecture Unintended Interactions

For defense <d>, risk <r> and common factor <f>, use pair of arrows that describe how <d> and <r> correspond to <f>

Conjectured interaction for a given <f>:

- If arrows align (↑,↑) or (↓,↓) ➞ <r> increases when <d> is effective (🔴)
- Else for (↑,↓) or (↓,↑) ➞ <r> decreases when <d> is effective (🟢)

Conjectured overall interaction: consider conjectures from all <f>s:

- If all <f> agree, then conjectured overall interaction is unanimous
- Otherwise, prioritize conjecture from dominant <f> (dominance may depend on attack)
- Value of a non-common factor may affect overall interaction

Group Fairness (FD1) vs. Data Reconstruction (P2)

Conjectured Interaction from common factor:

O2.2 Distinguishability across subgroups: FD1 ↓, P2 ↑ (➞ ●)

Non-common factor: D3 # Attributes -- risk may decrease with D3

Empirical Evidence

Fair model ➞ lower attack success (confirms ●)

- Lowers distinguishability across subgroups

Metric	Baseline	Fair Model
Accuracy	84.40 ± 0.09	77.96 ± 0.58
Recon. Loss	0.85 ± 0.01	0.95 ± 0.02

Non-common factor D3

attributes = 10:

- Fair model ➞ lower attack success

attributes > 10:

- Fair model ➞ no change in attack success
  (note: # attributes do not affect accuracy drop caused by fairness)

#Attributes	Baseline		Fair Model	
	Recon. Loss	Accuracy	Recon. Loss	Accuracy
10	0.85 ± 0.01	84.40 ± 0.09	0.95 ± 0.02	78.96 ± 0.58
20	0.93 ± 0.03	84.72 ± 0.22	0.93 ± 0.00	80.32 ± 1.12
30	0.95 ± 0.02	84.41 ± 0.39	0.94 ± 0.00	79.50 ±0.91

Summary

Unintended interactions are an important "meta concern"

Common influencing factors can help identify such interactions

Need defenses to protect against multiple risks

[1] Duddu et al. *SoK: Unintended Interactions among Machine Learning Defenses and Risks*. IEEE S&P. 2024. 🏆 **Distinguished Paper Award**

Talk Outline

"Meta Concerns" for Building Trust in ML Systems

- What are the unintended implications of applying defenses?
- How can we protect against multiple risks simultaneously?
- How can we design efficient mechanisms to demonstrate ML properties?

Protecting Against Multiple Risks

Can we combine defenses?

- Effective Combination**:** No significant drop in effectiveness of constituent defenses

Conflicting Interactions may degrade effectiveness of individual defenses

- Watermarking vs. adversarial training or differential privacy[1]
- ……. many other conflicts[2,3,4]

Need principled combination technique

- Modify existing defenses to combine effectively
- Identify if existing defenses can be combined without modification

[1] S.Szyller, N. Asokan. *Conflicting Interactions Among Protection Mechanisms for Machine Learning Models*. AAAI 2023.
[2] Fioretto et al. *Differential Privacy and Fairness in Decision and Learning Tasks: A Survey*. IJCAI 2022.
[3] Ferry et al. *SoK: Taming the Triangle - On the Interplays between Fairness, Interpretability and Privacy in Machine Learning*. arXiv 2024.
[4] Gittens et al. *An Adversarial Perspective on Accuracy, Robustness, Fairness, and Privacy: Multilateral-Tradeoffs in Trustworthy ML*. IEEE Access 2024.

Desiderata for Ideal Combination Technique

R1 Accurate

correctly identifies whether a combination is effective or not

R2 Scalable

allows combining more than two defenses

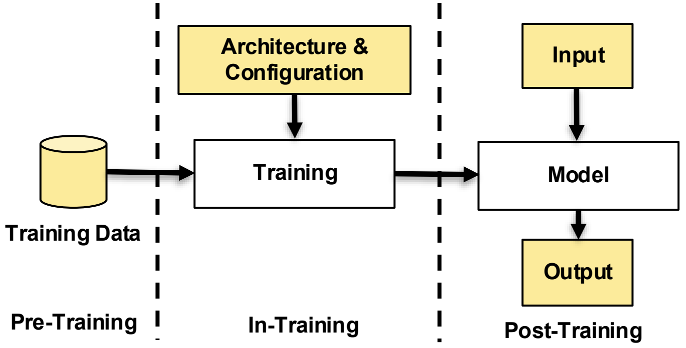
R3 Non-invasive

requires no changes to the defenses being combined

R4 General

applicable to different types of defenses

Limitations of Prior Work



Optimization[1,2]: game theory, regularization, constraint solving

- Ad-hoc optimizations specific to defenses (not general)
- Trade-off between effectiveness with utility (poor scalability)
- Invasive require modifying defenses

Mutually Exclusive Placement[3,4] (aka naïve technique)

- Defenses in different stages are non-conflicting

Scalable, non-invasive, and general but not accurate

- Incorrectly flags non-conflicting same-stage defenses (False negatives)
- Incorrectly flags conflicting defenses in different stages (False positives)

[1] Wu et al. *Augment then smooth: Reconciling differential privacy with certified robustness*. TMLR 2024.
[2] Tran et al. Differentially private and fair deep learning: A Lagrangian dual approach. AAAI 2021.
[3] S.Szyller, N. Asokan. *Conflicting Interactions Among Protection Mechanisms for Machine Learning Models*. AAAI 2023.
[4] Yaghini et al. *Learning with Impartiality to Walk on the Pareto Frontier of Fairness, Privacy and Utility.* ArXiV 2023.

Def\Con: Motivation

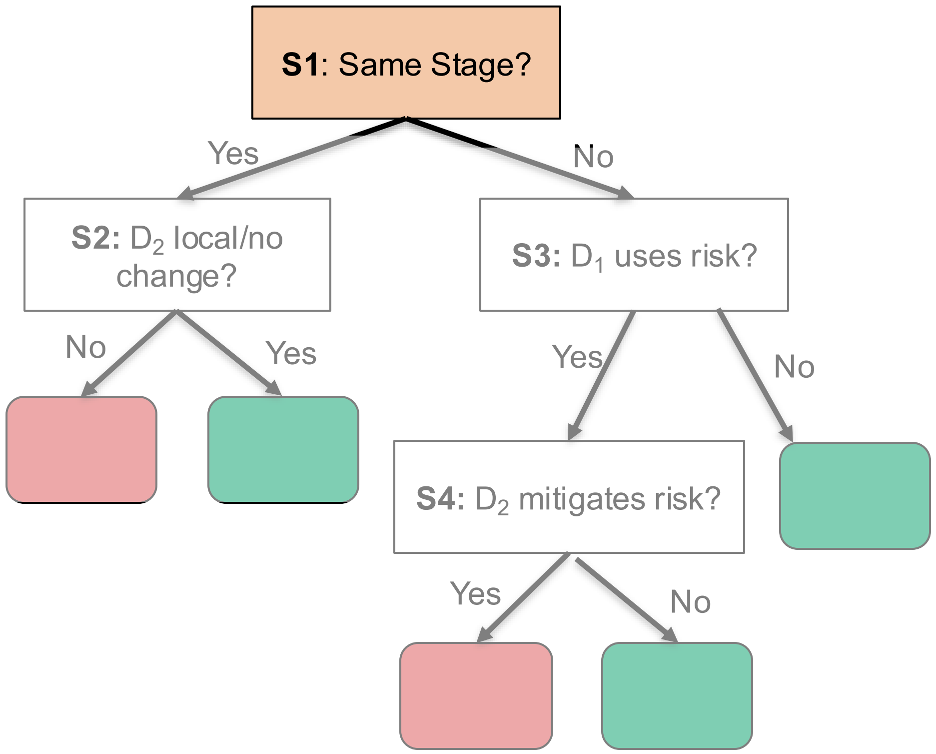
Naïve technique is promising, meets three requirements but not accurate

Can we improve naïve technique to account for reasons underlying conflicts?

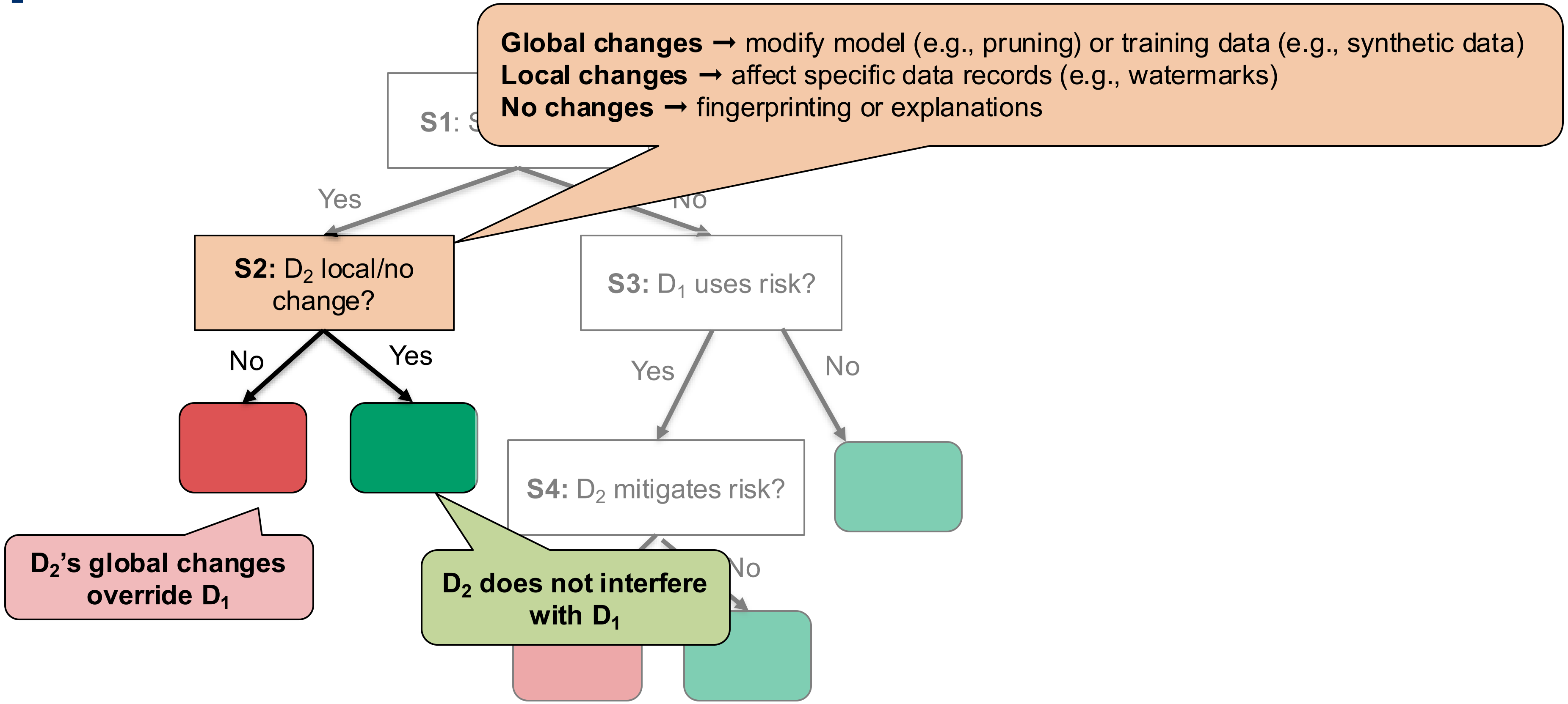
Reasons for Conflict: Defenses D_1 and D_2 (in order) conflict if

- D_1 uses risk protected by D_2
- Changes by D_2 overrides changes by D_1

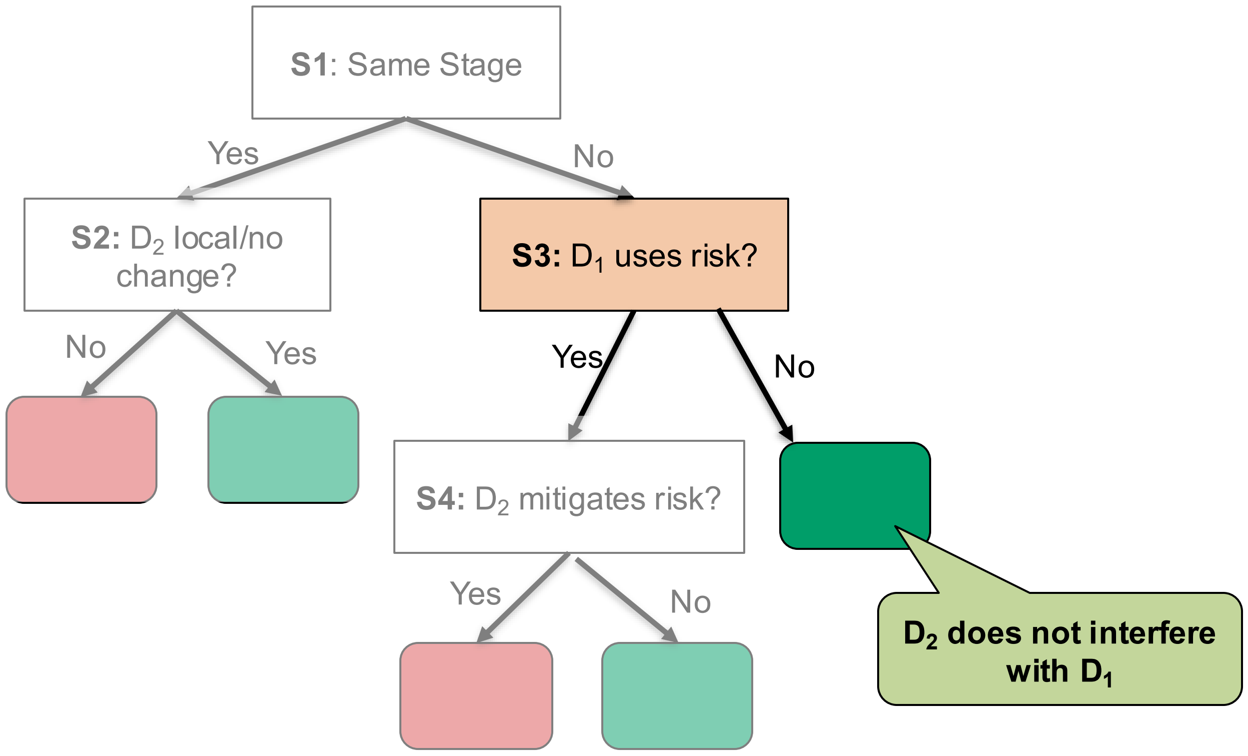
Def\Con: Step S-1



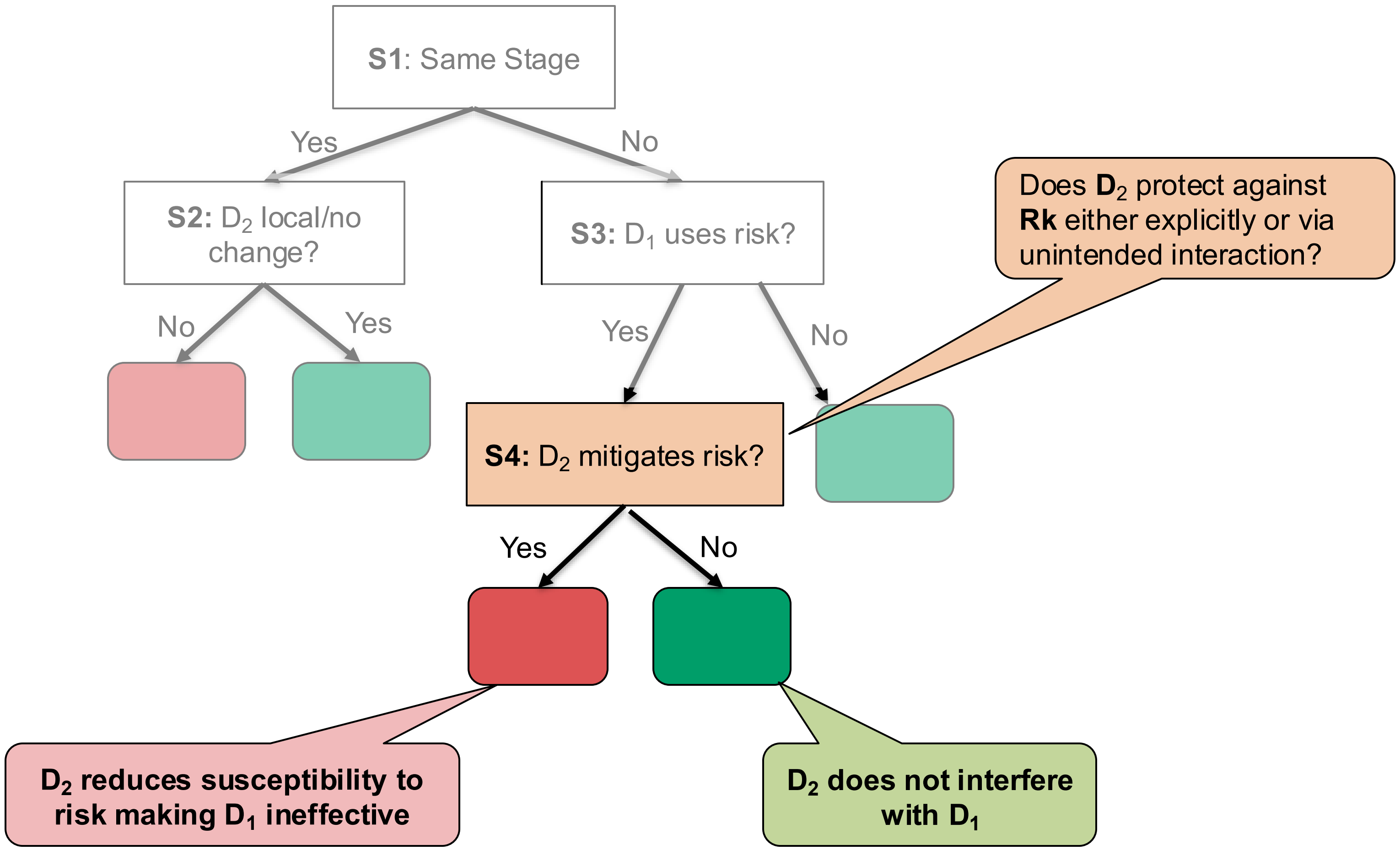
Def\Con: Step S-2



Def\Con: Step S-3



Def\Con: Step S-4



Evaluation: Accuracy of Def\Con

C1-C8 Eight combinations as ground truth from systematization of prior work

- Def\Con: 90% (7/8) vs. Naïve: 40% (4/8) balanced accuracy

C9-C38 Empirically evaluated remaining 30 unexplored combinations

- Def\Con: 81% (27/30) vs. Naïve: 36% (18/30) balanced accuracy

	Combinations	Metric	FMNIST	UTKFACE		Combinations	Metric	FMNIST	UTKFACE
C9	D_1: Evasion Robustness ($\mathbf{D_{evs}.In}$)	$\phi^{\hat{D}}_{u}$ (↑)	89.69 ± 0.20	73.87 ± 0.53	C24	D_1: Watermarking-M ($\mathbf{D_{wmM}.Pre}$)	$\phi^{\hat{D}}_{u}$ (↑)	90.18 ± 0.21	79.76 ± 0.63
	D_2: Watermarking-M ($\mathbf{D_{wmM}.Post}$)	$\phi^{\hat{D}}_{wmacc}$ (↑)	100.00 ± 0.00	76.19 ± 13.13		D_2: Explanations ($\mathbf{D_{expl}.Post}$)	$\phi^{\hat{D}}_{err}$ (↓)	0.14 ± 0.04	0.02 ± 0.03
	(Ψ, Δ)	$\phi^{\hat{D}}_{robacc}$ (↓)	83.94 ± 0.64	67.14 ± 0.49		(Ψ, Δ)	$\phi^{\hat{D}}_{wmacc}$ (↑)	99.93 ± 0.06	99.96 ± 0.08
C10	D_1: Outlier Robustness ($\mathbf{D_{out}.In}$)	$\phi^{\hat{D}}_{u}$ (↑)	89.50 ± 0.21	79.25 ± 1.06	C25	D_1: Watermarking-M ($\mathbf{D_{wmM}.In}$)	$\phi^{\hat{D}}_{u}$ (↑)	86.94 ± 0.50	72.16 ± 5.13
	D_2: Fingerprinting ($\mathbf{D_{fng}.Post}$)	$\phi^{\hat{D}}_{ASR}$ (↓)	9.94 ± 0.22	56.09 ± 12.98		D_2: Explanations ($\mathbf{D_{expl}.Post}$)	$\phi^{\hat{D}}_{err}$ (↓)	0.19 ± 0.07	0.37 ± 0.18
	(Ψ, Δ)	$\phi^{\hat{D}}_{pval}$ (↓)	<0.05	<0.05		(Ψ, Δ)	$\phi^{\hat{D}}_{wmacc}$ (↑)	98.24 ± 0.66	97.60 ± 3.54
C11	D_1: Outlier Robustness ($\mathbf{D_{out}.Post}$)	$\phi^{\hat{D}}_{u}$ (↑)	84.73 ± 1.72	63.70 ± 3.87	C26	D_1: Watermarking-D ($\mathbf{D_{wmD}.Pre}$)	$\phi^{\hat{D}}_{u}$ (↑)	90.04 ± 0.60	79.03 ± 1.10
	D_2: Fingerprinting ($\mathbf{D_{fng}.Post}$)	$\phi^{\hat{D}}_{ASR}$ (↓)	61.36 ± 23.96	0.02 ± 0.03		D_2: Explanations ($\mathbf{D_{expl}.Post}$)	$\phi^{\hat{D}}_{err}$ (↓)	0.10 ± 0.04	0.54 ± 0.01
	(Ψ, Δ)	$\phi^{\hat{D}}_{pval}$ (↓)	<0.05	<0.05		(Ψ, Δ)	$\phi^{\hat{D}}_{RSD}$ (↑)	100.00 ± 0.00	100.00 ± 0.00
C12	D_1: Evasion Robustness ($\mathbf{D_{evs}.In}$)	$\phi^{\hat{D}}_{u}$ (↑)	89.60 ± 0.18	74.62 ± 0.60	C27	D_1: Outlier Robustness ($\mathbf{D_{out}.In}$)	$\phi^{\hat{D}}_{u}$ (↑)	89.39 ± 0.24	78.71 ± 0.20
	D_2: Explanations ($\mathbf{D_{expl}.Post}$)	$\phi^{\hat{D}}_{err}$ (↓)	0.12 ± 0.03	0.53 ± 0.05		D_2: Explanations ($\mathbf{D_{expl}.Post}$)	$\phi^{\hat{D}}_{ASR}$ (↓)	9.79 ± 0.15	44.35 ± 30.07
	(Ψ, Δ)	$\phi^{\hat{D}}_{robacc}$ (↑)	84.68 ± 0.18	67.26 ± 0.42		(Ψ, Δ)	$\phi^{\hat{D}}_{err}$ (↓)	0.06 ± 0.02	0.47 ± 0.02
C13	D_1: Group Fairness ($\mathbf{D_{fair}.In}$)	$\phi^{\hat{D}}_{u}$ (↑)		66.73 ± 3.24	C28	D_1: Outlier Robustness ($\mathbf{D_{out}.Post}$)	$\phi^{\hat{D}}_{u}$ (↑)	84.62 ± 3.56	63.80 ± 3.37
	D_2: Outlier Robustness ($\mathbf{D_{out}.Post}$)	$\phi^{\hat{D}}_{ASR}$ (↓)		20.21 ± 39.90		D_2: Explanations ($\mathbf{D_{expl}.Post}$)	$\phi^{\hat{D}}_{ASR}$ (↓)	76.11 ± 15.85	0.00 ± 0.00

Summary

Protecting against multiple risks is important

Def\Con: a combination technique which is

More accurate than naïve technique

Inherits other requirements from naïve technique

- Combines more than two defenses (scalable)
- Does not require modifying defenses (non-invasive)
- Does not depend on specific defenses to mark conflict (general)

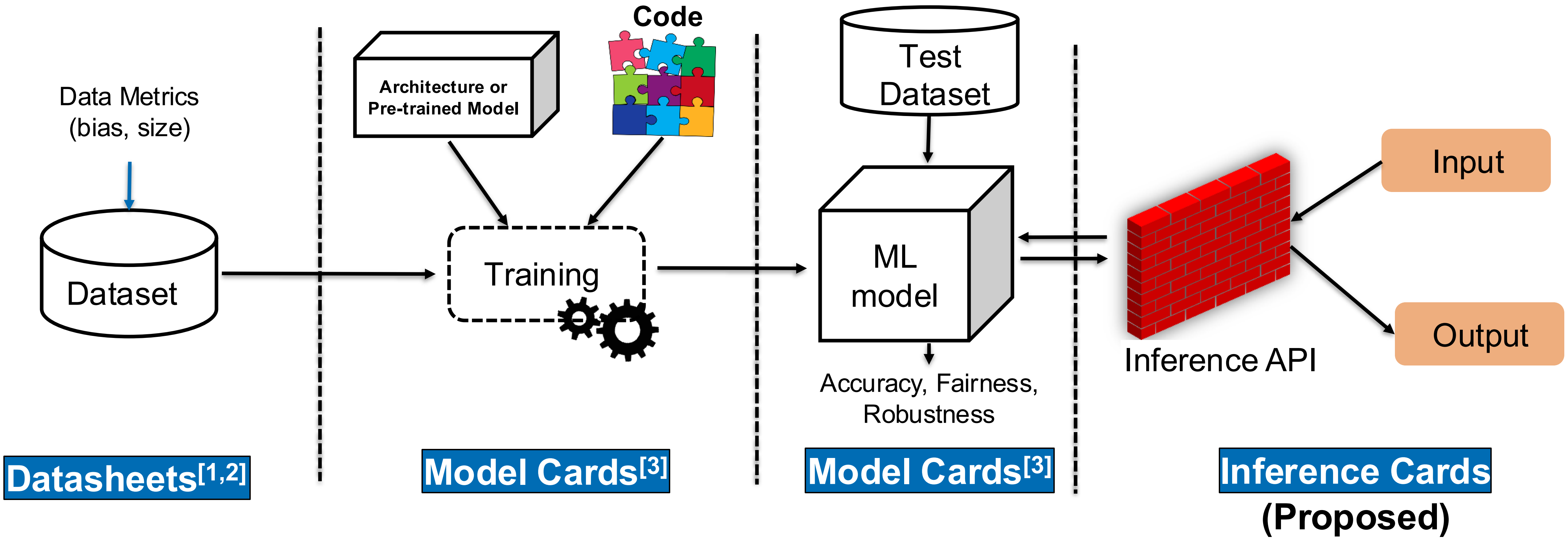
[1] Duddu et al. *Combining Machine Learning Defenses without Conflicts*. ArXiv. 2025.

Talk Outline

“Meta Concerns” for Building Trust in ML Systems

- What are the unintended implications of applying defenses?
- How can we protect against multiple risks simultaneously?
- How can we design efficient mechanisms to demonstrate ML properties?

“Nutrition Labels” to Advertise ML Properties Exist



Collectively, refer to them as “ML property cards”

[1] Gebru et al. *Datasheets for datasets*. Communications of ACM. 2021.
[2] Pushkarna et al. *Data Cards: Purposeful and Transparent Dataset Documentation for Responsible AI*. FaccT. 2022.
[3] Mitchell et al. *Model Cards for Model Reporting*. Facct. 2019.

ML Property Cards are Not Verifiable

Need verifiable ML property cards

- Prevent inclusion of false information[1]
- Demonstrate correct execution of ML operations
  - For accountability in ML pipeline and regulatory compliance

[1] Mithril-Security. PoisonGPT: How to poison LLM supply chain on HuggingFace. 2023.

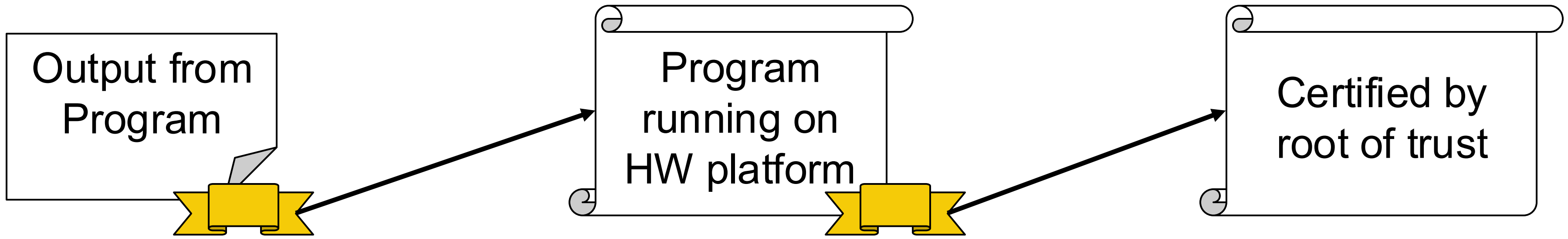
Verifiable ML Property Cards via Property Attestation

ML property attestation[1]

- Prover (e.g., model trainer) demonstrates properties to Verifier (e.g., regulator, customer)

Mental Model for Attestations

Certificate showing that something came from software with a certain hash



[1] Duddu et al. Attesting Distributional Properties of Machine Learning Training Data. ESORICS'24.

Desiderata for ML Property Attestation Mechanism

R1 Efficient

Incur low computation overhead

R2 Versatile

Support various ML properties for training and inference

R3 Scalable

Support multiple verifiers

R4 Robust

Resist evasion of attestations by malicious prover

Existing ML Property Attestation Mechanisms

ML-based Attestations

Error-prone and not robust: e,g.,

- proof of learning[1,2],
- re-purposing distribution inference for distributional property attestation[3]

Cryptographic Attestations (e.g., Zero-knowledge Proofs, Multi-party Computation)

Inefficient: e,g.,

- ~13 minutes for IO attestation (e.g., using ZKPs with LLMs[4])

Not Versatile: Limited to crypto-friendly properties

[1] Zhang et al. “Adversarial Examples” for Proof- of-Learning. IEEE S&P’22.
[2] Fang et al. Proof of Learning is more Broken than You Think. IEEE EuroS&P’23
[3] Duddu et al. Attesting Distributional Properties of Machine Learning Training Data. ESORICS’24.
[4] Sun et al. *zkLLMs: Zero Knowledge Proofs for Large Language Models*. CCS’24.

Can TEEs Enable ML Property Attestation?

Hardware-assisted TEEs are pervasive

- Isolated execution: Isolated Execution Environment
- Protected storage: Sealing
- Ability to convince remote verifiers: (Remote) Attestation

ARM TrustZone   Intel SGX

Property Attestation for TEEs

- Remote attestation was extended to properties of binaries running inside TEEs[1]
- Can we adapt this for attesting ML properties?

Recent developments make ML training/inference within TEEs feasible (efficient)

- Intel's AMX extensions for SGX[2], Nvidia's H100 GPU[3]
- Available with Cloud providers

[1] Sadeghi and Stuble. Property-based attestation for computing platforms: caring about properties, not mechanisms. 2004.
[2] Google Cloud Team. We tested Intel's AMX CPU accelerator for AI and here's what we learned.
[3] Zhu et al. Confidential Computing on Nvidia's H100 GPU: A Performance Benchmark Study.

System and Adversary Models

Model trainer and/or owner trains, evaluates, and deploys model

Verifier (e.g., regulator, customer) wants to be convinced of some model property
Prover wants to demonstrate ML properties (e.g., training, evaluation, inference)

Verifier trust Prover’s TEE and software outside of TEE (e.g., OS, hypervisor) is untrusted
Two roots of trust for Verifier

- TEE Manufacturer (e.g., Intel): certifies attestation signing keys
- Trusted certifiers (e.g., CIFAR): provides additional certificates (e.g., for datasets)

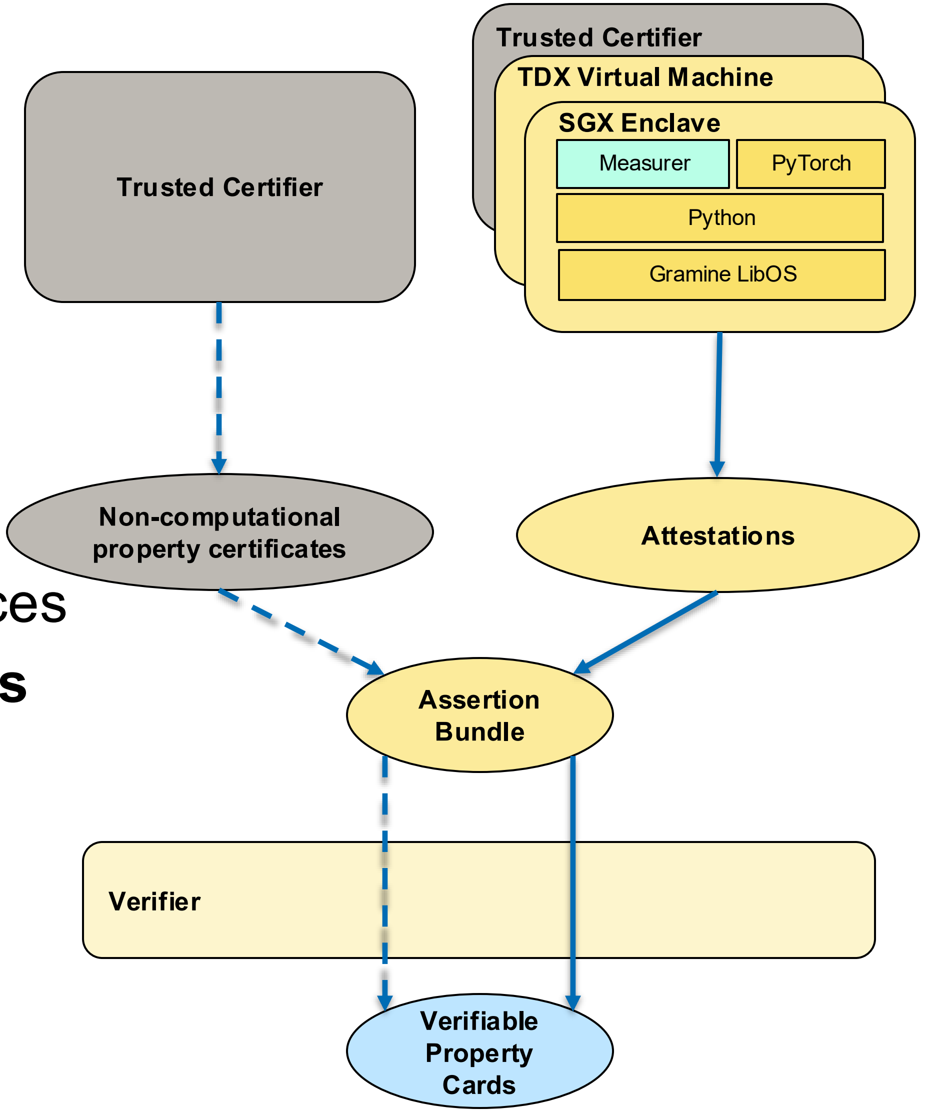
Laminator: Framework

Measurer within TEE measures desired property

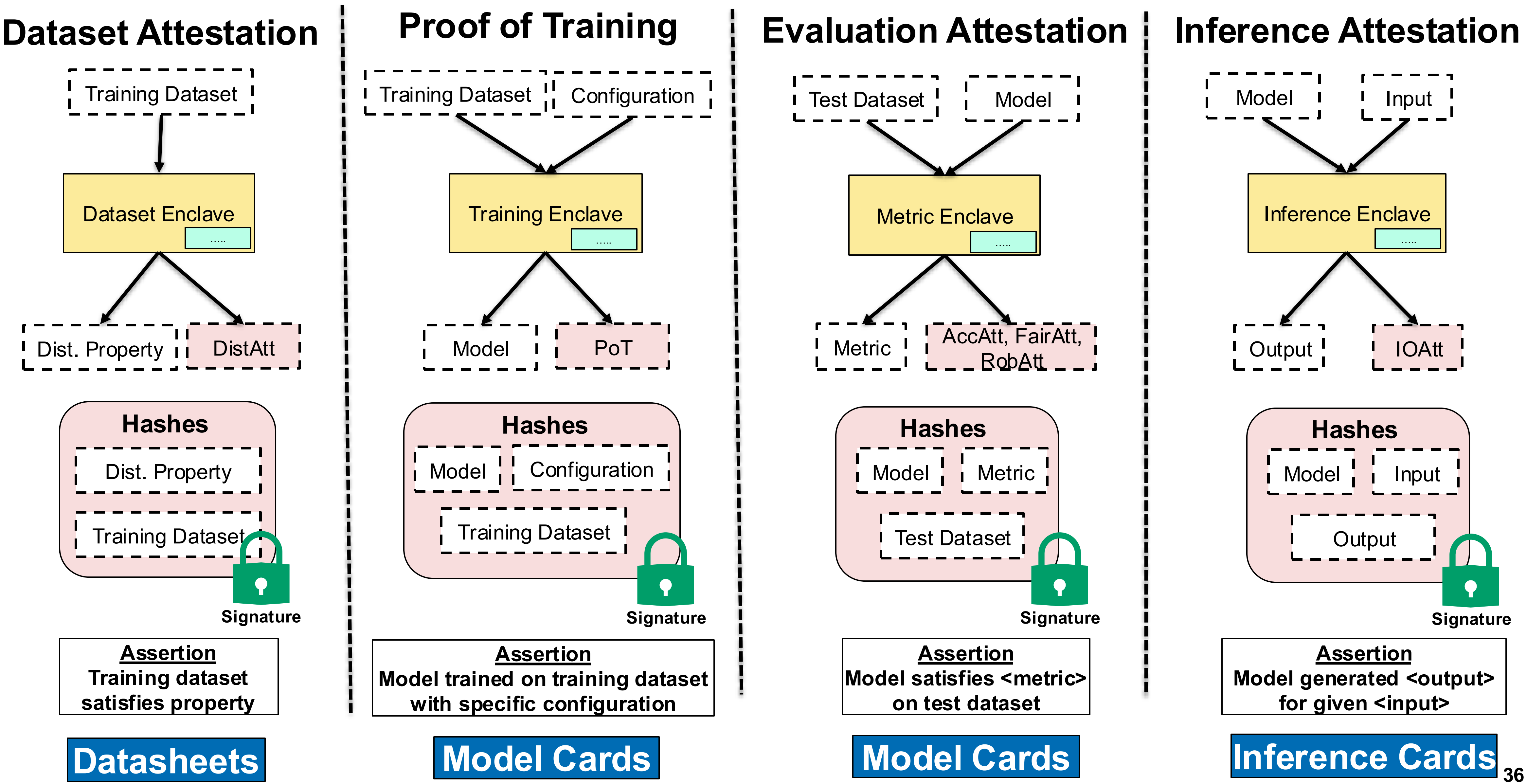
TEE produces attestation (property card fragment)

Assertion bundle

- combines certificates and attestations from various sources
- checkable by Verifier to realize **verifiable property cards**



Types of ML Property Attestations



Evaluation: Efficiency

Input and output measurement roughly scales with input and output size

Attestation constant across all datasets and models

Overall, Laminator overhead is low

- Distribution attestation: 0.36% and 2.05%
- Proof of Training: 0.00-0.32%
- Evaluation attestation: 0.00-0.35%

Evaluation: Efficiency

Baseline cost for single inference is low compared to attestation

- High overhead between 39% and 3955% (aka “overhead w/ att”)

Amortizing overhead over several IO attestations

- Generate a signing keypair during initialization and attest it once
- Sign each inference result for indirect, low-cost attestation (“overhead w/ sgn”)
  - Overhead between 0.17% and 1.17%

Summary

Laminator uses hardware-assisted attestations for verifiable ML property cards:

- **Efficient:** Incurs low computation overhead
- **Scalable:** Attestations can be checked by multiple verifiers
- **Versatile:** Any ML property specified in python can be attested
- **Robust**: Inherited from TEE integrity guarantees

[1] Duddu et al. *Attesting Distributional Properties of Training Data for Machine Learning*. ESORICS. 2024.
[2] Duddu et al. *Laminator: Verifiable ML Property Cards using Hardware-assisted Attestations*. ACM CODASPY. 2025.

Takeaways

Not enough to design defenses for single risk

Need to include other “Meta Concerns”:

- Framework to understand unintended interactions
- Combination technique to combine ML defenses
- Verifiable ML Property Cards for accountability

Unintended Interactions[1]

Combining Defenses[2]

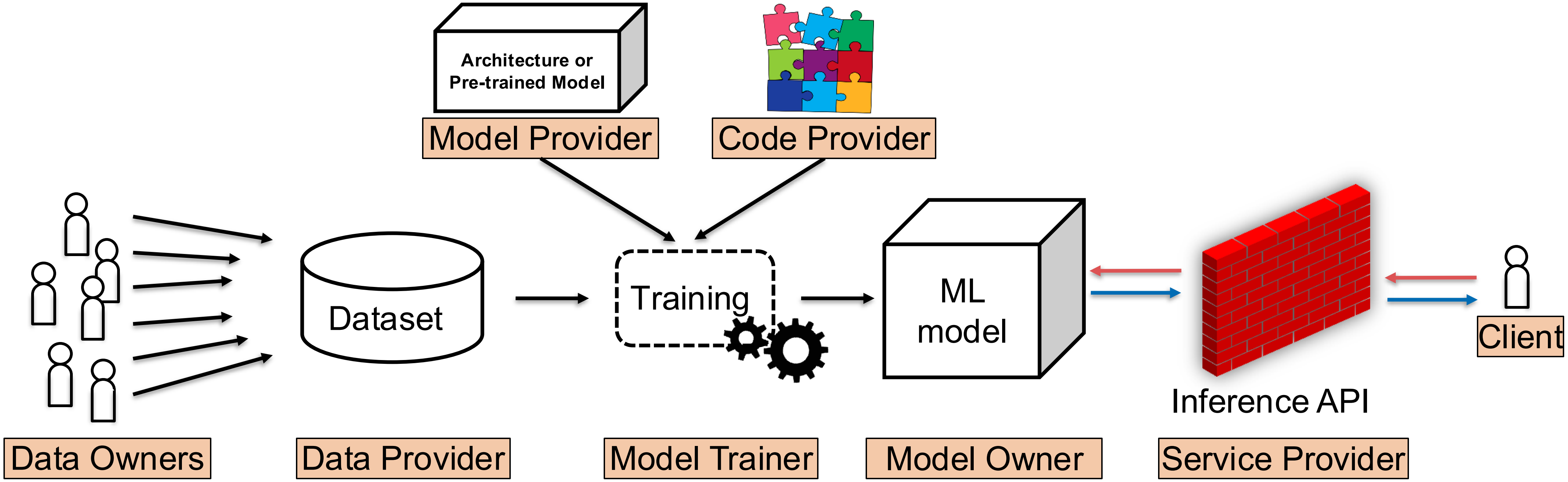
ML Property Attestations[3]

Laminator[4]

[1] Duddu et al. *SoK: Unintended Interactions among Machine Learning Defenses and Risks*. IEEE S&P. 2024. Distinguished Paper Award
[2] Duddu et al. *Combining Machine Learning Defenses without Conflicts*. ArXiv. 2025.
[3] Duddu et al. *Attesting Distributional Properties of Training Data for Machine Learning*. ESORICS. 2024.
[4] Duddu et al. *Laminator: Verifiable ML Property Cards using Hardware-assisted Attestations*. ACM CODASPY. 2025.

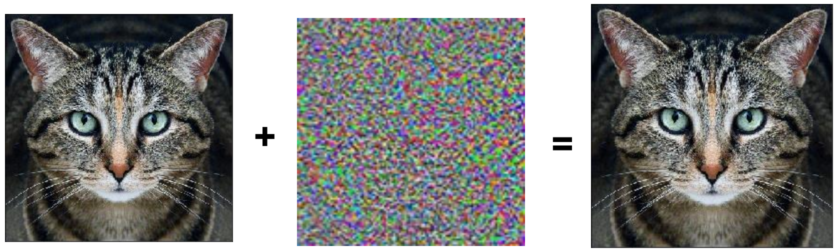
Backup Slides: Background

Machine Learning Pipeline

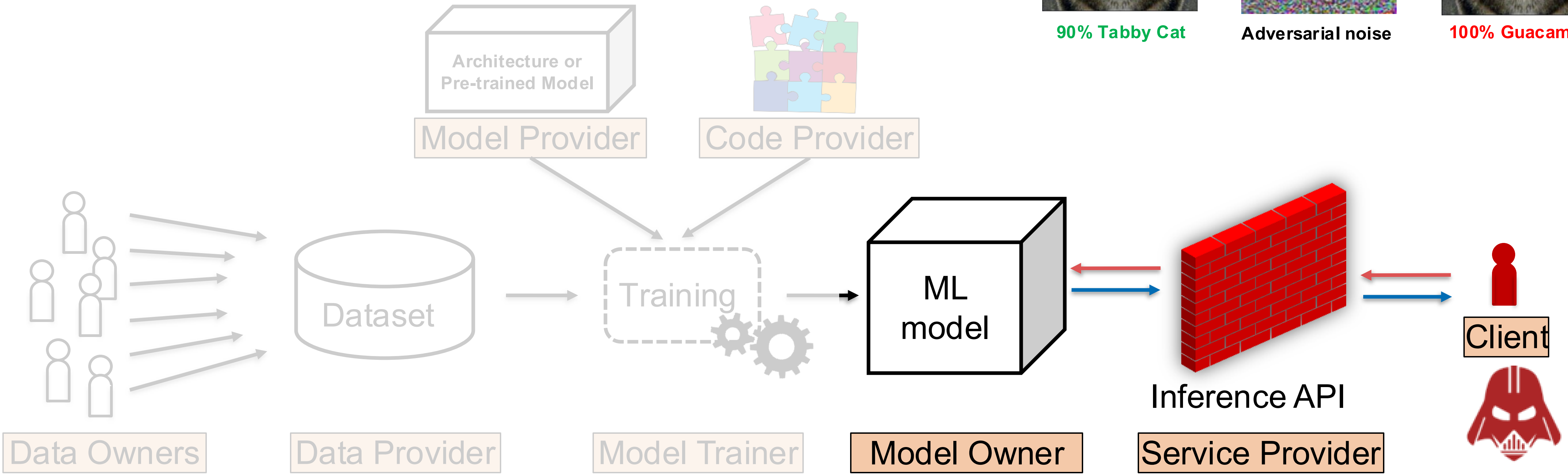


Where is the adversary?
What can they do?

(Security) Risk of Evasion

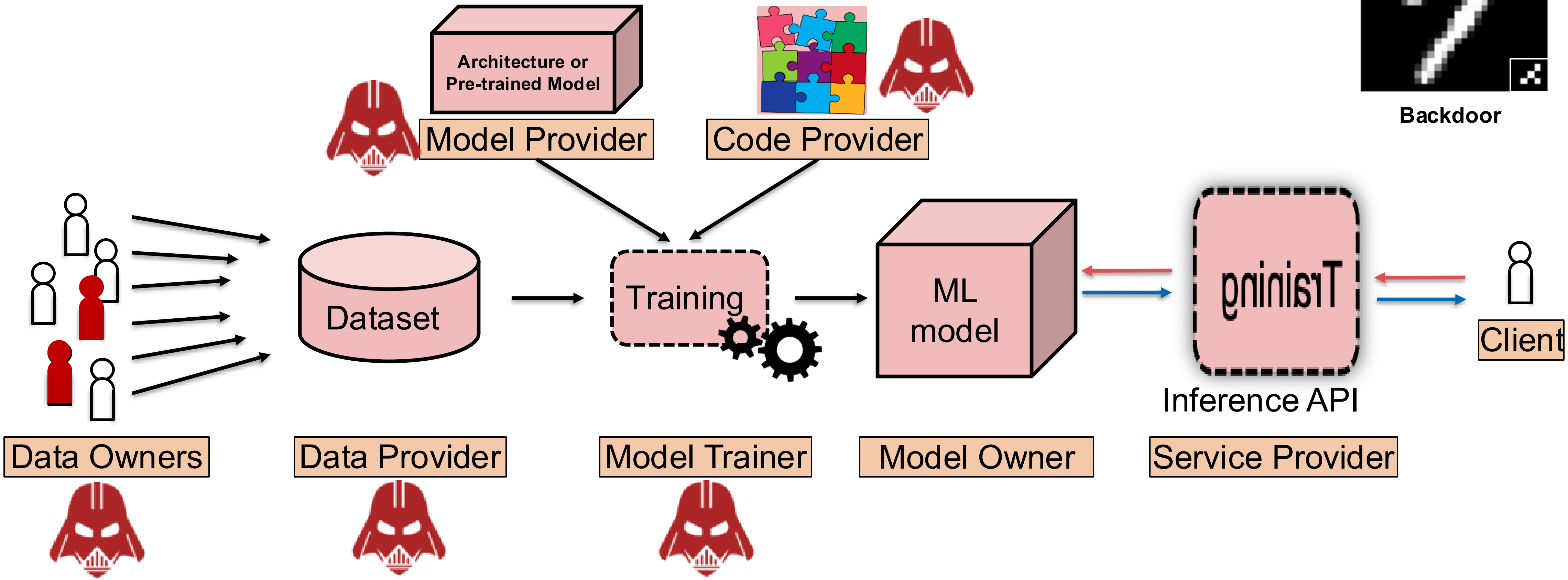


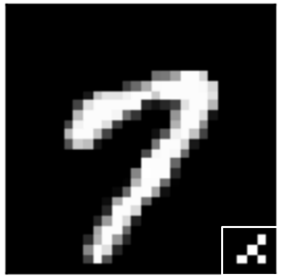
90% Tabby Cat | Adversarial noise | 100% Guacamole



[1] Croce and Hein. _Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks_. ICML 2020.
[2] Madry et al. _Towards Deep Learning Models Resistant to Adversarial Attacks_. ICML 2018.

(Security) Risk of Poisoning

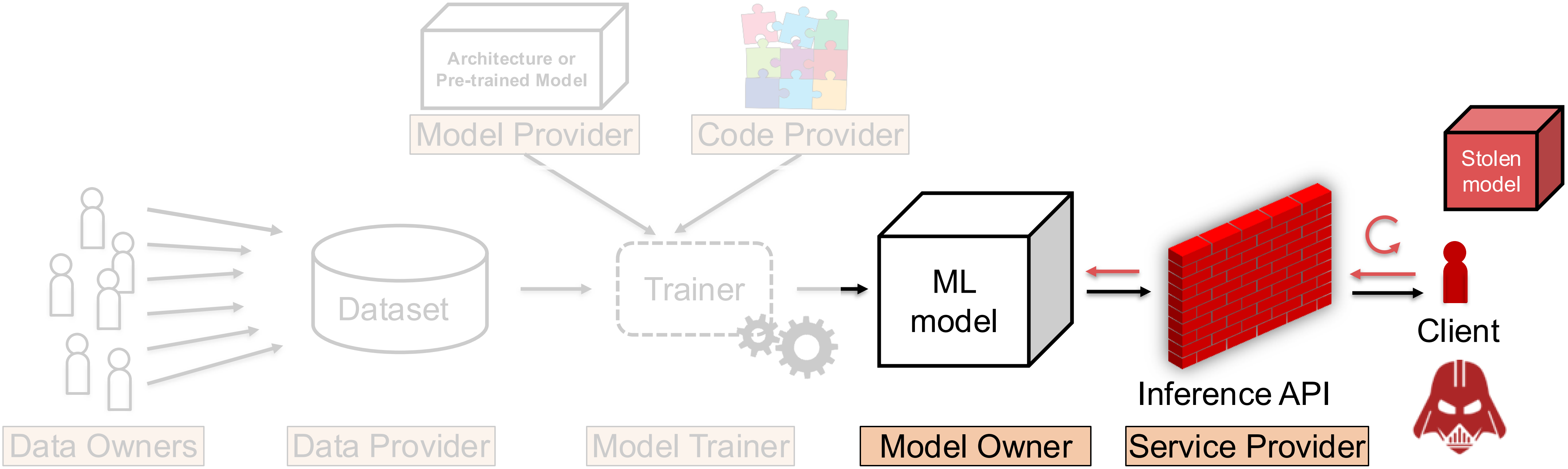




Backdoor

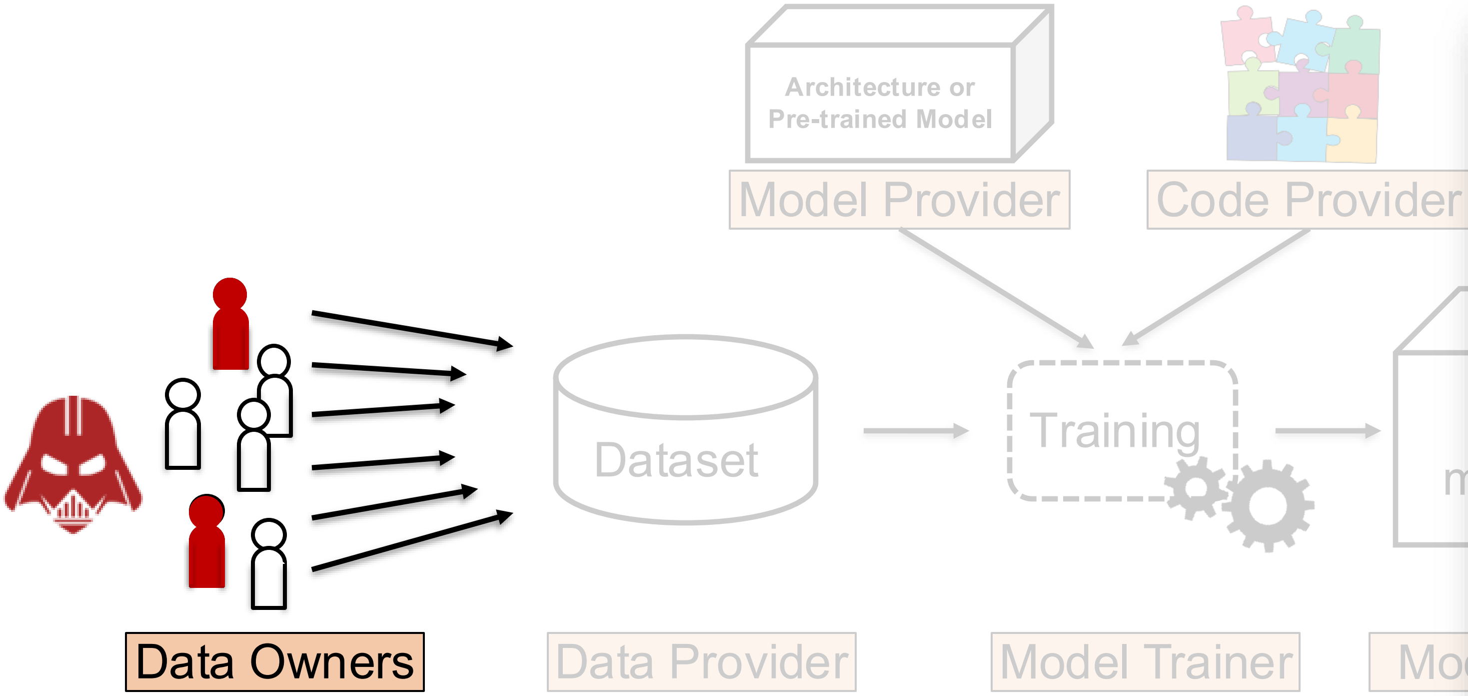
[1] Shafahi et al. *Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks*. NeurIPS 2018.
[2] Zhang et al. *Persistent Pre-training Poisoning of LLMs*. ICLR 2025.
[3] Langford et al. *Architectural Neural Backdoors from First Principles*. IEEE S&P 2025.
[4] Bagdasaryan and Shmatikov. Blind Backdoors in Deep Learning Models. Usenix Sec 2021.

(Security) Risk of Unauthorized Model Ownership



[1] Krishna et al. *Thieves on Sesame Street! Model Extraction of BERT-based APIs*. ICLR 2020.
[2] Orekondy et al. *Knockoff-Nets: Stealing Functionality of Black-Box Models*. CVPR 2019.

(Security) Risk of Unauthorized Data Usage



Copyright Infringing Data
Non-Consensual Personal Data

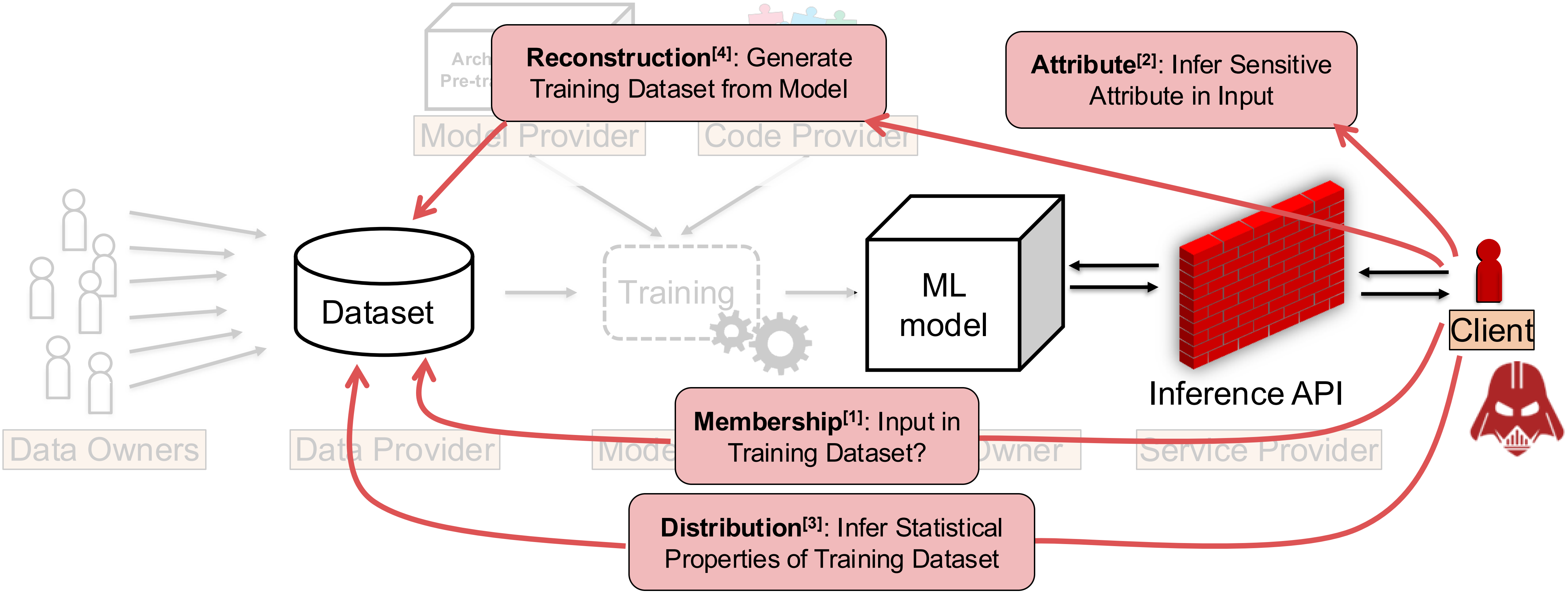
The Times Sues OpenAI and Microsoft Over A.I. Use of Copyrighted Work

Millions of articles from The New York Times were used to train chatbots that now compete with it, the lawsuit said.

Share full article 1272

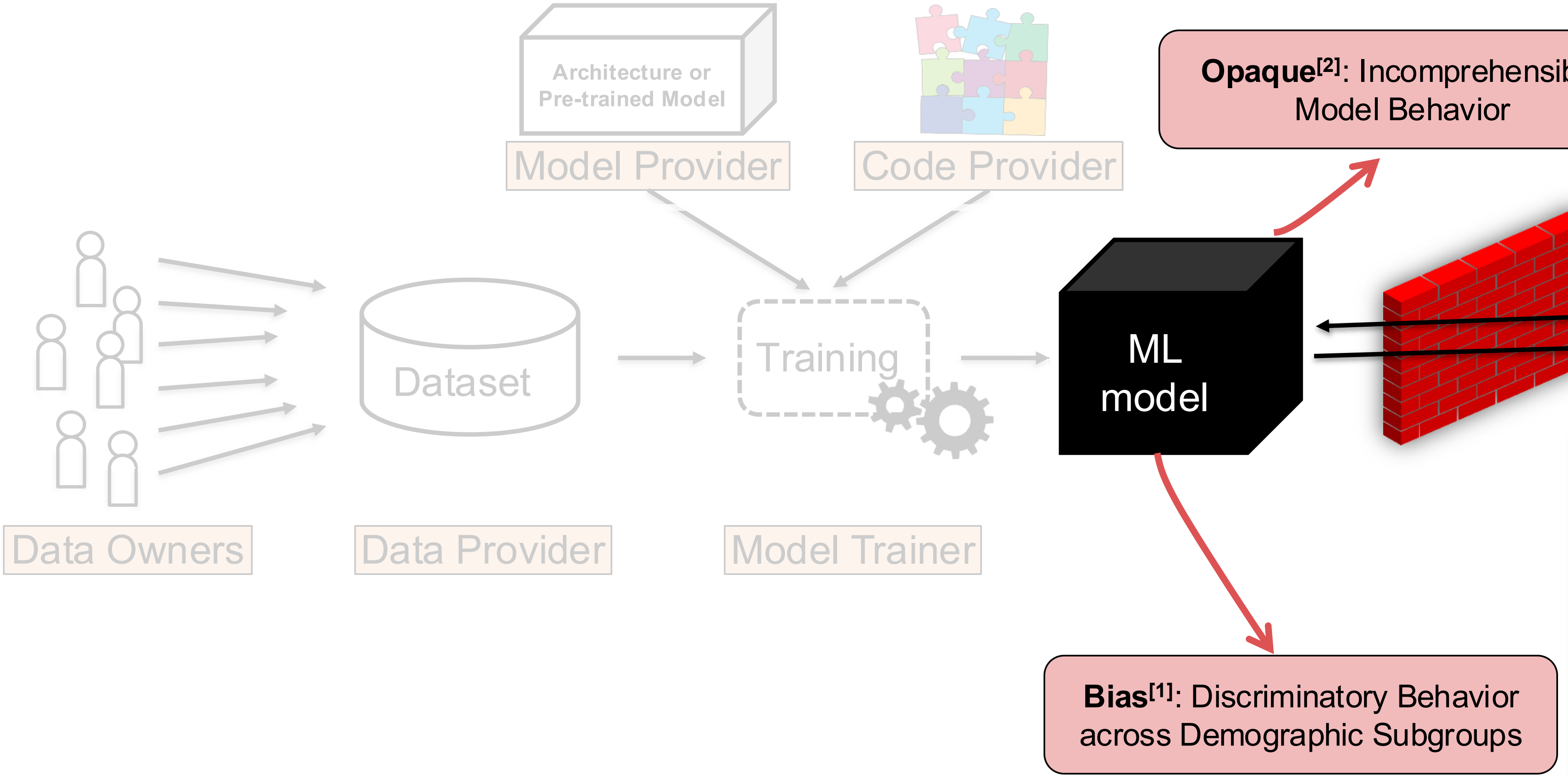
A lawsuit by The New York Times could test the emerging legal contours of generative A.I. technologies. Sasha Maslov for The New York Times

(Privacy) Risk of Inference Attacks



[1] Carlini et al. Membership Inference Attacks From First Principles. IEEE S&P 2022.
[2] Jayaraman and Evans. Are Attribute Inference Attacks Just Imputation? CCS 2022.
[3] Suri et al. Dissecting Distribution Inference. IEEE SatML 2023.
[4] Carlini et al. Extracting Training Data From Large Language Models. Usenix Sec 2021.

(Fairness) Risk of Discriminatory Behavior




ProPublica

Donate

Machine Bias

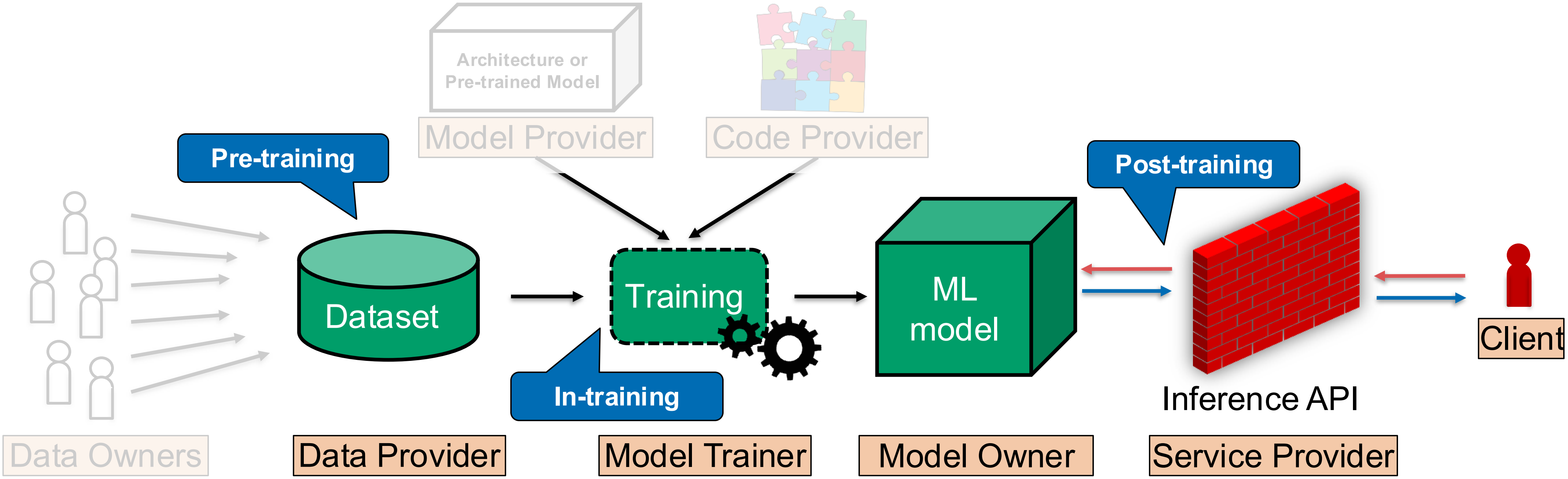
There's software used across the country to predict future criminals. And it's biased against blacks.

by Julia Angwin, Jeff Larson, Surya Mattu and Lauren Kirchner, ProPublica

May 23, 2016

[1] Hardt et al. *Equality of Opportunity in Supervised Learning*. NeurIPS 2016.
[2] Lundberg and Lee. *A Unified Approach to Interpreting Model Predictions*. NeurIPS 2017.

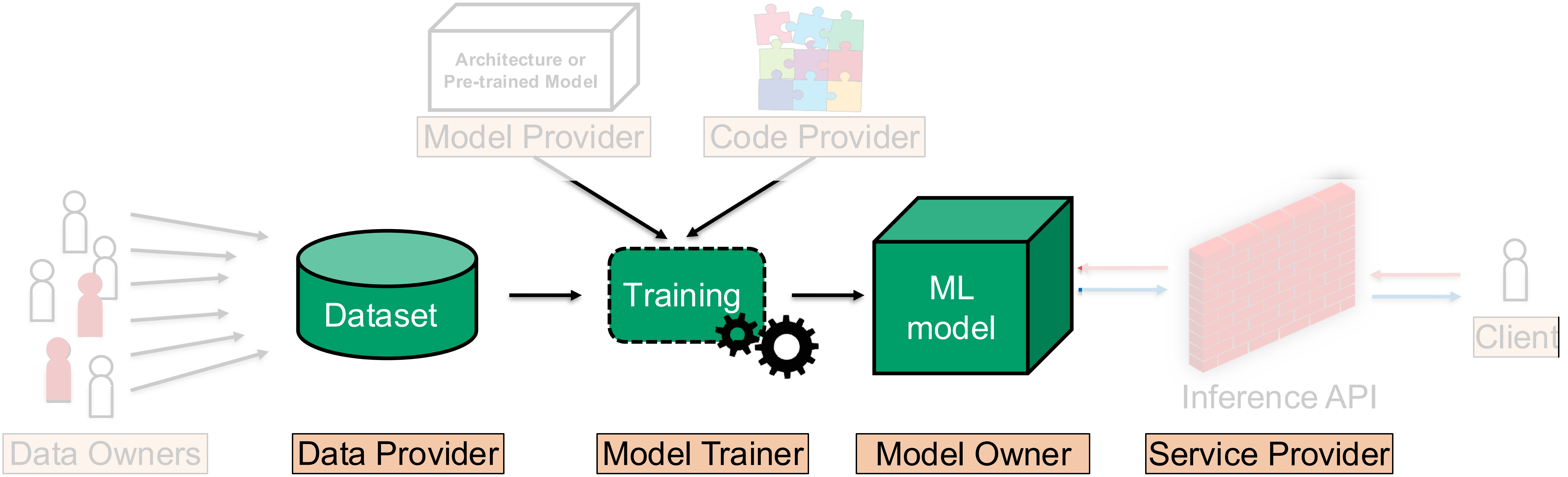
(Security) Robustness against Evasion



(Pre-training) Data Augmentation[1]: Transformations of training data to improve robustness
(In-training) Adversarial Training[2]: Train model with perturbed data records
(Post-training) Input Processing[3]: Transform inputs to filter noise

[1] Rebuffi et al. *Data Augmentation Can Improve Robustness*. NeurIPS 2021.
[2] Madry et al. *Towards Deep Learning Models Resistant to Adversarial Attacks*. ICML 2018.
[3] Nie et al. *Diffusion Models for Adversarial Purification*. ICML 2022.

(Security) Robustness against Poisoning



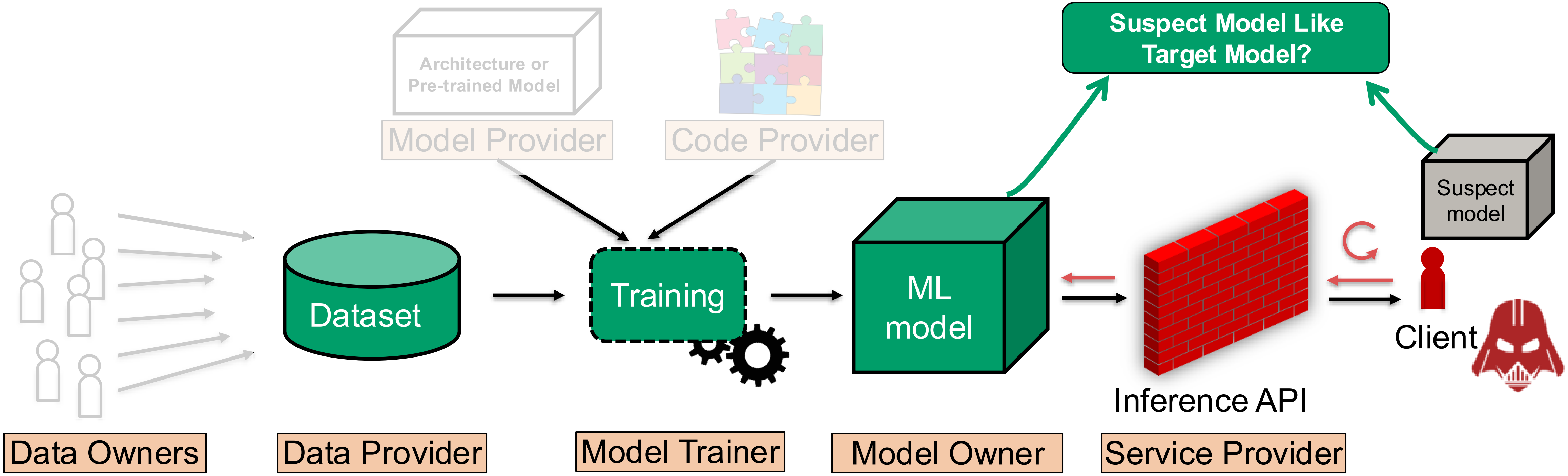
(Pre-training) Data Sanitization[1]: Detect and remove outliers (poisons) from training data

(In-training) Fine-tuning[2]: Update model to reduce influence of outliers

(Post-training) Pruning[3]: Remove model parameters to reduce influence of outliers

[1] Borgnia et al. *Strong Data Augmentation Sanitizes Poisoning and Backdoors Attacks without an Accuracy Trade-off*. ICASSP 2021.
[2] Patrini et al. *Making Deep Neural Networks Robust to Label Noise: A Loss Correction Approach*. CVPR 2017.
[3] Li et al. *Reconstructive Neuron Pruning for Backdoor Defense*. ICML 2023.

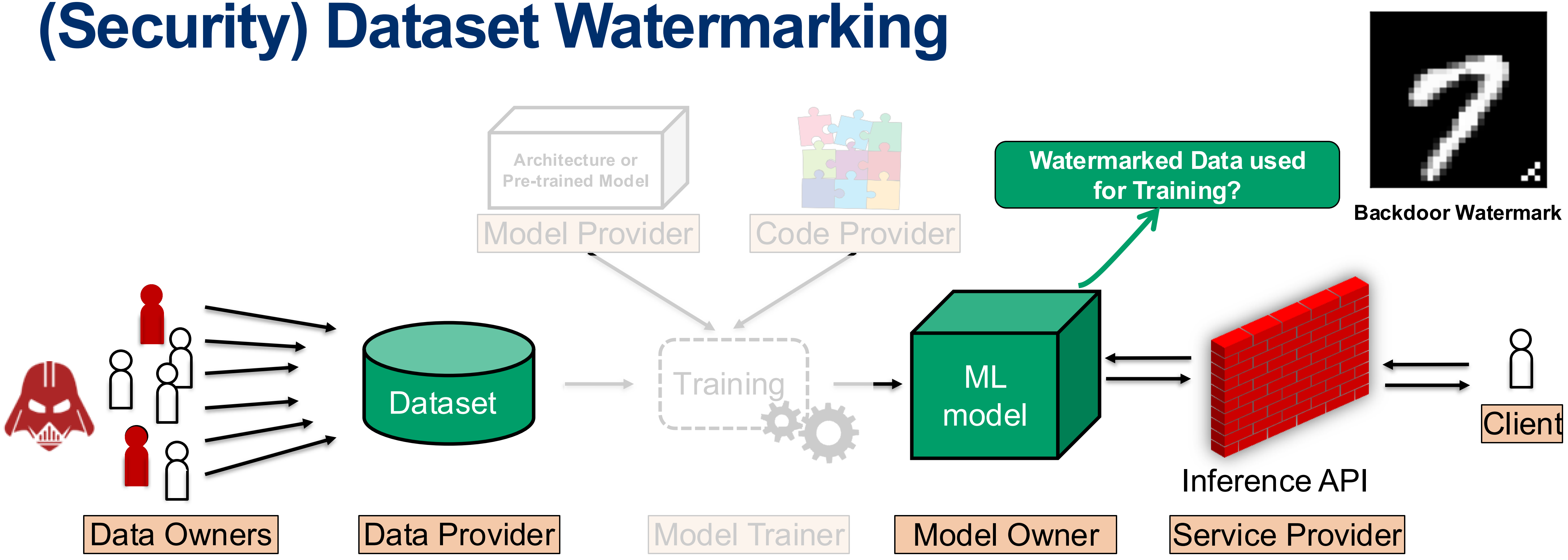
(Security) Model Watermarking / Fingerprinting



(Pre-training) Watermarking[1]: Train on backdoors as watermarks
(Post-training) Watermarking[2]: Flip predictions as watermarks
(Post-training) Fingerprinting[3]: Unique model characteristics as fingerprints

[1] Adi et al. *Tuning your Weakness into a Strength: Watermarking Deep Neural Networks by Backdoors*. USENIX Sec 2018.
[2] Szyller et al. *DAWN: Dynamic Adversarial Watermarking of Neural Networks*. ACM MM. 2021.
[3] Waheed et al. *GrOVe: Ownership Verification of Graph Neural Networks using Embeddings*. IEEE S&P 2024. (Our work)

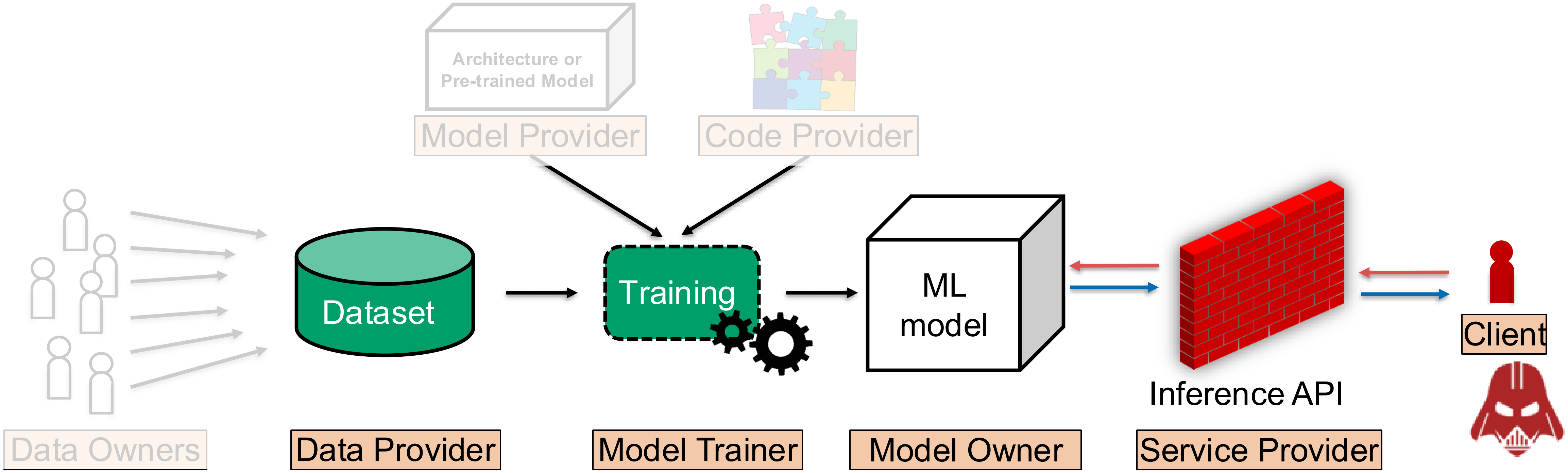
(Security) Dataset Watermarking



(Pre-training) Watermarking[1,2]: Train on backdoors as watermarks

[1] Sablyarolles et al. *Radioactive Data: Tracing through Training*. ICML 2020.
[2] Chen et al. *Catch Me if You Can: Detecting Unauthorized Data Use In Training Deep Learning Models*. CCS 2024.

(Privacy) Differential Privacy

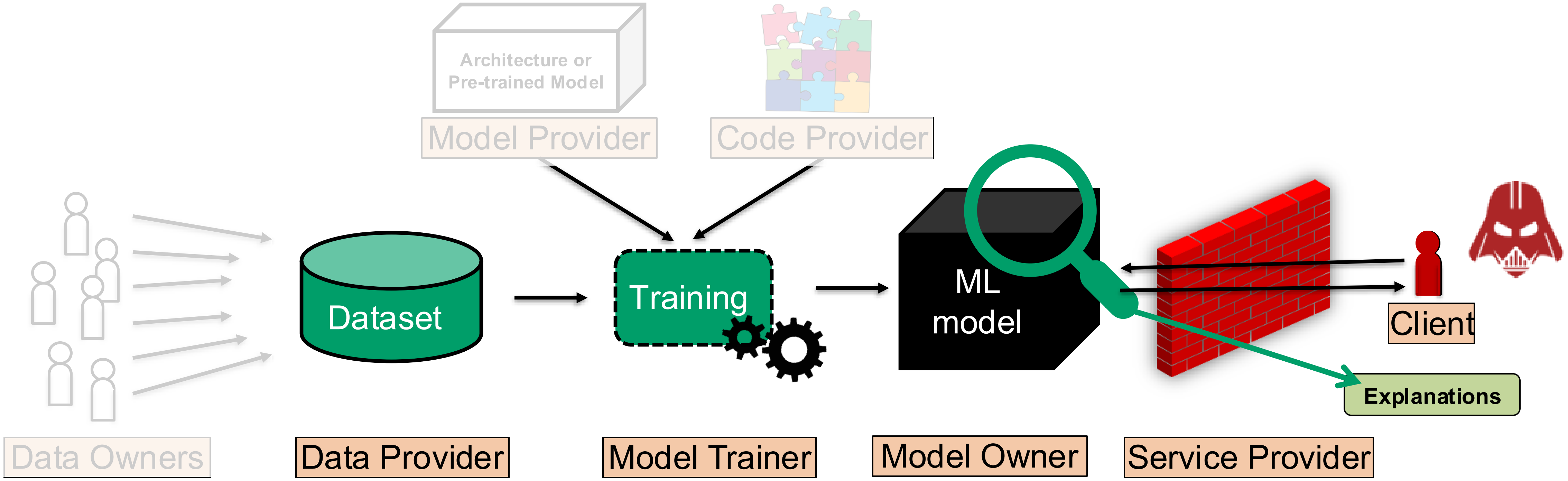


(Pre-training) DP Synthetic Dataset[1]: Transform training data with DP guarantees

(In-training) DPSGD[2,3]: Add gradient noise to reduce influence of individual data records

[1] Lin et al. *Differentially Private Synthetic Data via Foundation Model APIs 1: Images*. ICLR 2024.
[2] Abadi et al. *Deep Learning with Differential Privacy*. CCS 2016.
[3] Papernot et al. *Scalable Private Learning with PATE*. ICLR 2018.

(Fairness) Defenses against Fairness Risks



(Pre-training) Fair synthetic data[1]: Transform training data for downstream fairness
(In-training) Regularization[2]: Add fairness constraint for optimization
(Post-training) Calibration[3]: Adjust threshold over predictions
(Post-training) Explanations[4]: Measure influence of input attributes to predictions

[1] Zemel et al. *Learning Fair Representations*. ICML 2013.
[2] Hardt et al. *Equality of Opportunity in Supervised Learning*. NeurIPS 2016.
[3] Pleiss et al. *On Fairness and Calibration*. NeurIPS 2017.
[4] Lundberg and Lee. *A Unified Approach to Interpreting Model Predictions*. NeurIPS 2017.

Backup Slides: Unintended Interactions

Underlying causes: overfitting and memorization

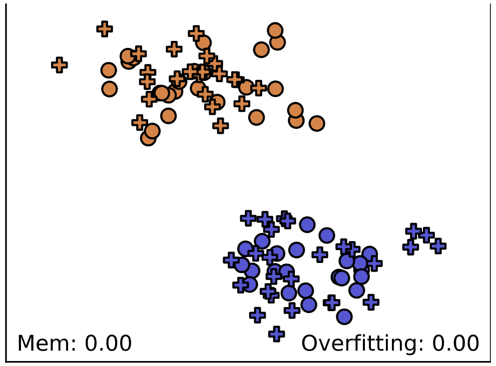
Overfitting and memorization are distinct and can occur simultaneously[1,2]

Overfitting

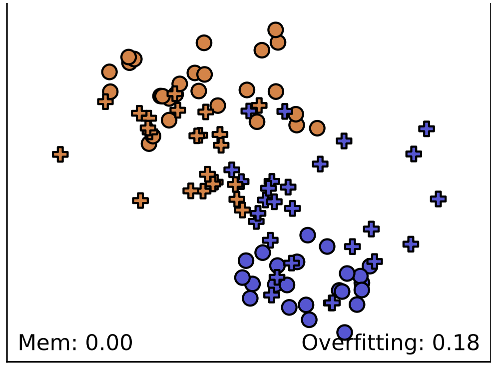
- Difference between train and test accuracy[3]
- Aggregate metric computed across datasets

Memorization of training data records

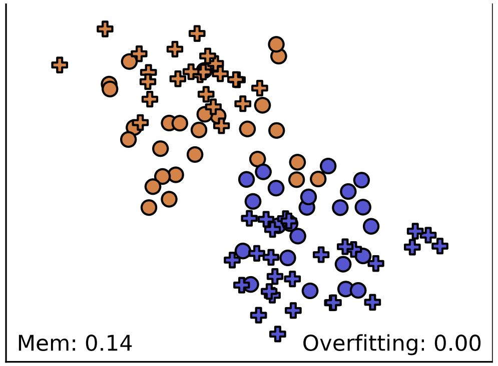
- Difference in model prediction on a data record with and without it in training dataset[4]
- Metric for individual data records



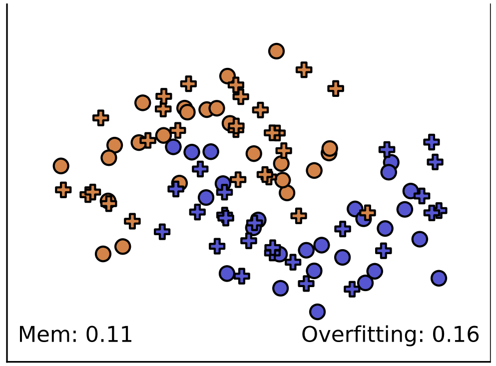
No Overfitting + No Memorization



Overfitting + No Memorization



No Overfitting + Memorization



Overfitting + Memorization

[1] Carlini et al. The Secret Sharer: Evaluating and testing unintended memorization in neural networks. USENIX Sec 2019.
[2] Burg and Williams. On memorization in probabilistic deep generative models. NeurIPS 2019.
[3] Hardt et al. Train faster, generalize better: Stability of stochastic gradient descent. ICML 2016.
[4] Feldman. Does learning require memorization? A Short Tale About a Long Tail. STOC 2020.

Dominant factors

LEGEND

O1 Curvature smoothness of the objective function
O2 Distinguishability of model observables across datasets (**O2.1**), subgroups (**O2.2**), and models (**O2.3**)
O3 Distance of training data to decision boundary

D1 Size of training data
D2 Tail length of distribution
D3 Number of attributes
D4 Priority of learning stable attributes
M1 Model capacity

Active factors are exploited by the attacks: O1, O2, O3

Passive factors (data/model configuration): D1, D2, D3, D4, M1

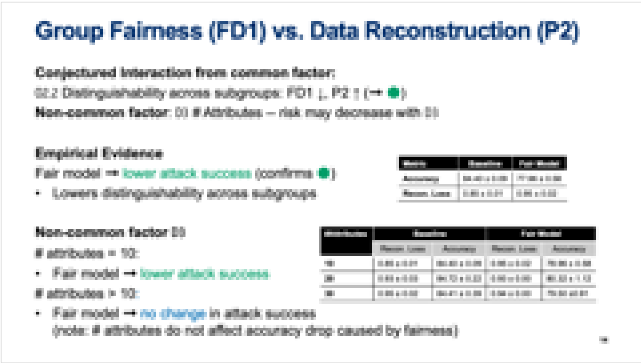
Attacks often exploit active factors, we deem them "dominant"

PD1 (Differential Privacy) and R1 (Evasion)➞ 🔴 [1,2]

- D2 ➞ 🟢; O1 ➞ 🔴; O3 ➞ 🔴

FD1 (Group Fairness) and P1 (Membership Inference) ➞ 🔴[3]

- D4 ➞ 🟢; O3 ➞ 🔴



[1] Tursynbek et al. *Robustness threats of Differential Privacy*. NeurIPS Privacy Preserving ML Workshop. 2020.
[2] Boenisch et al. *Gradient masking and the underestimated robustness threats of differential privacy in deep learning*. ArXiv 2021.
[3] Chang and Shokri. *On the Privacy Risks of Algorithmic Fairness*. EuroS&P 2021.

Framework: factors influencing overfitting

Bias is an error from poor hyperparameter choices for model

- High bias (smaller models) ➞ prevents learning relations between attributes and labels

Variance is an error from sensitivity to changes in the training dataset

- High variance ➞ model fits noise in training data

Tradeoffs can be balanced using:

- **D1 Size of training data** inversely correlated with overfitting: likelihood that the model encounters a similar data record is higher
- **M1 Model capacity** inversely correlated with overfitting if model is too simple to fit data

Framework: factors influencing memorization

D2 Tail length of distribution correlates with memorization of tail classes (rare or outliers)

D3 Number of attributes inversely correlates with memorization of individual attributes

D4 Priority of learning stable attributes correlates with generalization

O1 Curvature smoothness of the objective function results in variable memorization of data records as it determines convergence of their loss towards a minima

O2 Distinguishability of model observables across datasets (O2.1), subgroups (O2.2), and models (O2.3) correlates with memorization

O3 Distance of training data to decision boundary inversely correlates with memorization

M1 Model capacity Increasing capacity can increase memorization of data records

Explanations (FD2) vs. distribution inference (P4) (1/2)

Conjectured interactions from common factor:

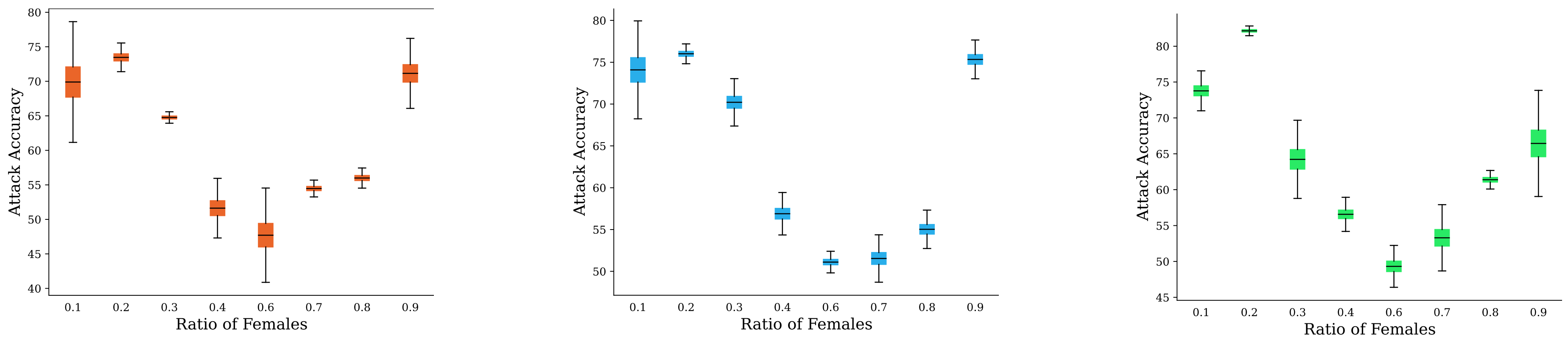
O2.1 Distinguishability of observables across datasets: FD2 ↑ , P4 ↑ (➞ 🔴)

Non-common factors:

- **D3 # Attributes:** risk may decrease with D3 (lower memorization)
- **M1 Model Capacity:** risk may increase with M1 (higher memorization)

Empirical Evidence (confirms 🔴)

Explanations ➞ increased susceptibility to inference: attack accuracy > 50% for most ratios



Integrated Gradients SmoothGrad DeepLift

Explanations (FD2) vs. distribution inference (P4) (2/2)

Non-common factor D3 (# Attributes): More attributes ➞ lower attack success

# Attributes	Integrated Gradients	DeepLift	SmoothGrad
15	81.07 ± 2.13	78.74 ± 1.66	65.40 ± 1.39
25	66.09 ± 0.95	73.64 ± 1.38	59.42 ± 1.09
35	50.43 ± 0.59	59.93 ± 2.81	56.78 ± 1.93

Non-common factor M1 (Model Capacity): Higher capacity ➞ higher attack success

# Parameters	Integrated Gradients	DeepLift	SmoothGrad
5.7K	47.57 ± 4.25	49.19 ± 2.75	53.26 ± 0.10
44K	53.29 ± 3.65	50.86 ± 3.24	62.40 ± 0.95
274K	62.60 ± 2.74	67.73 ± 1.69	70.21 ± 0.73
733K	69.90 ± 3.24	73.78 ± 1.03	74.09 ± 2.17

Exceptions to guideline

Differences in adversary models can change the interaction type

- **RD1 (Adversarial training) and R3 (Unauthorized Model Ownership)**
  - Guideline predicts ➞ ● (M1 but not dominant)
  - If adversary is malicious suspect➞ ●[1]; If adversary is malicious accuser➞ ●[2]
- **PD1 (Differential privacy) and P4 (Distribution Inference)**
  - Guideline predicts ➞ ● (O2.1) which matches with empirical evidence[3]
  - If adversary knows victim is DP-trained, they can DP-train shadow models➞ ●[3]
- **FD1 (Group fairness) and P3 (Attribute Inference)**
  - Guideline predicts ➞ ● (O2.2) which matches with empirical evidence[4]
  - If adversary knows fairness algorithm, they can calibrate their attack➞ ●[5]

Some defenses and risks have too few factors

- RD2 (Outlier removal), R2 (Poisoning), R3 (Unauthorized model ownership)

[1] Khaled et al. *Careful What You Wish For: On the Extraction of Adversarially Trained Models*. PST 2022.
[2] Liu et al. False Claims against Model Ownership Resolution. Usenix SEC 2024.
[3] Suri et al. *Dissecting Distribution Inference*. SatML 2023.
[4] Aalmoes et al. *On the alignment of Group Fairness with Attribute Privacy*. ArXiv 2022.
[5] Ferry et al. *Exploiting Fairness to Enhance Sensitive Attributes Reconstruction*. SatML 2023.

Backup Slides: Laminator

How to Draw Conclusions from Assertion Bundle

Multiple attestations in assertion bundle help draw conclusions about ML properties

- **Combining training-time attestations**

Models was trained on D_{tr} satisfying distributional properties p

- **Combining training-time and inference-time attestations**

Output O obtained from model for input I, where M was trained on D_{tr} satisfying property p, and satisfies the required {accuracy, fairness, robustness} requirements

Laminator: Experimental Setup

Model	Description	# Parameters	Model Size (MB)
CENSUS-S	MLP: [128]	12,290	0.05
CENSUS-L	MLP: [128, 256, 512, 256]	308,482	1.2
UTKFACE-S	VGG11	9,227,010	36.95
UTKFACE-L	VGG16	14,724,162	58.96
IMDB-S	LSTM: [64, 256, 256]	920,385	3.69
IMDB-L	LSTM: [64, 256, 256, 256, 256]	1,973,057	7.60

Datasets: CENSUS (tabular), UTKFACE (images), and IMDB (text)

CENSUS and UTKFACE have sensitive attributes (for distribution attestation)

- IMDB not applicable distribution attestation