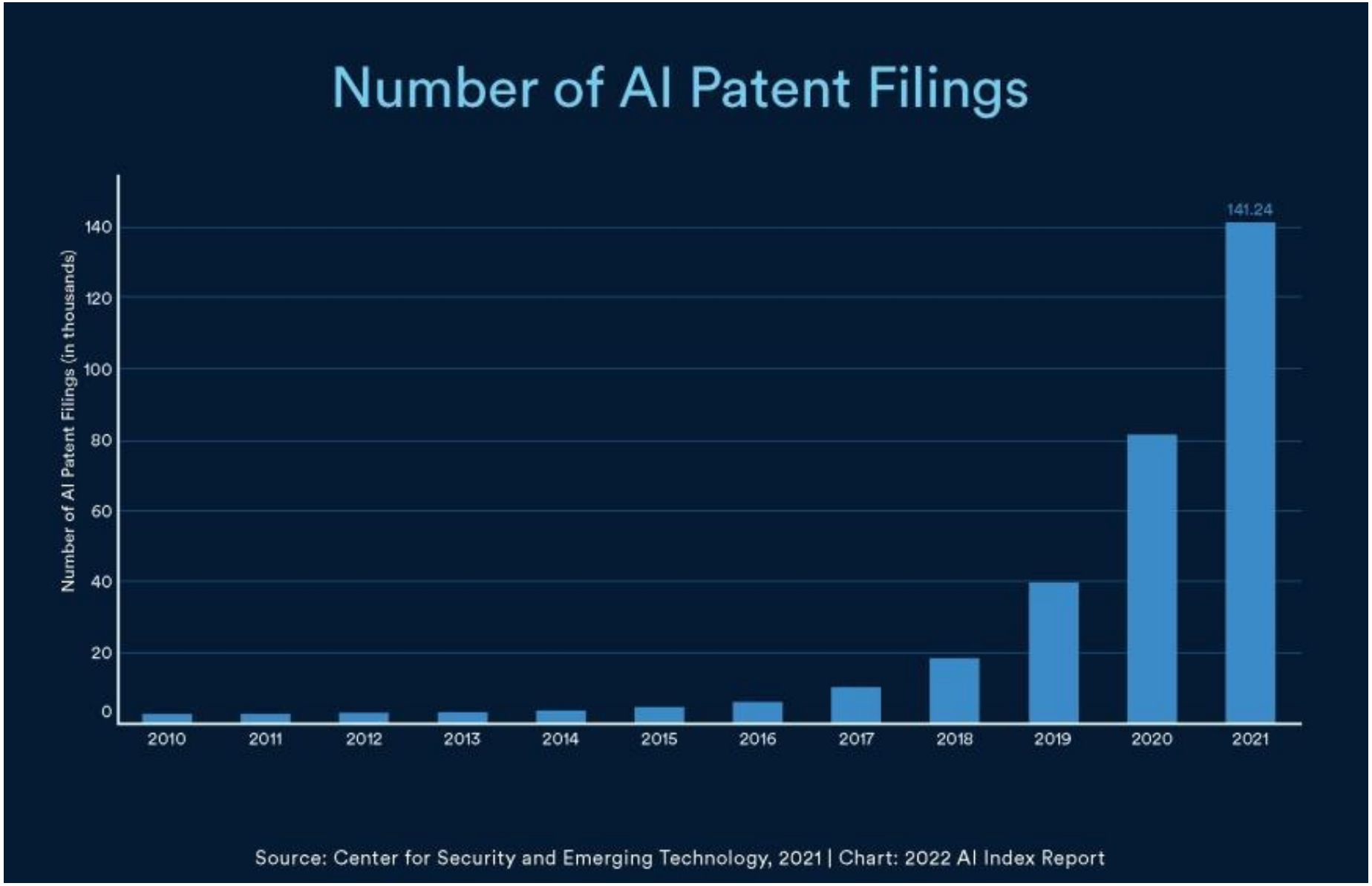
AI Auditing with ZK

Alluri

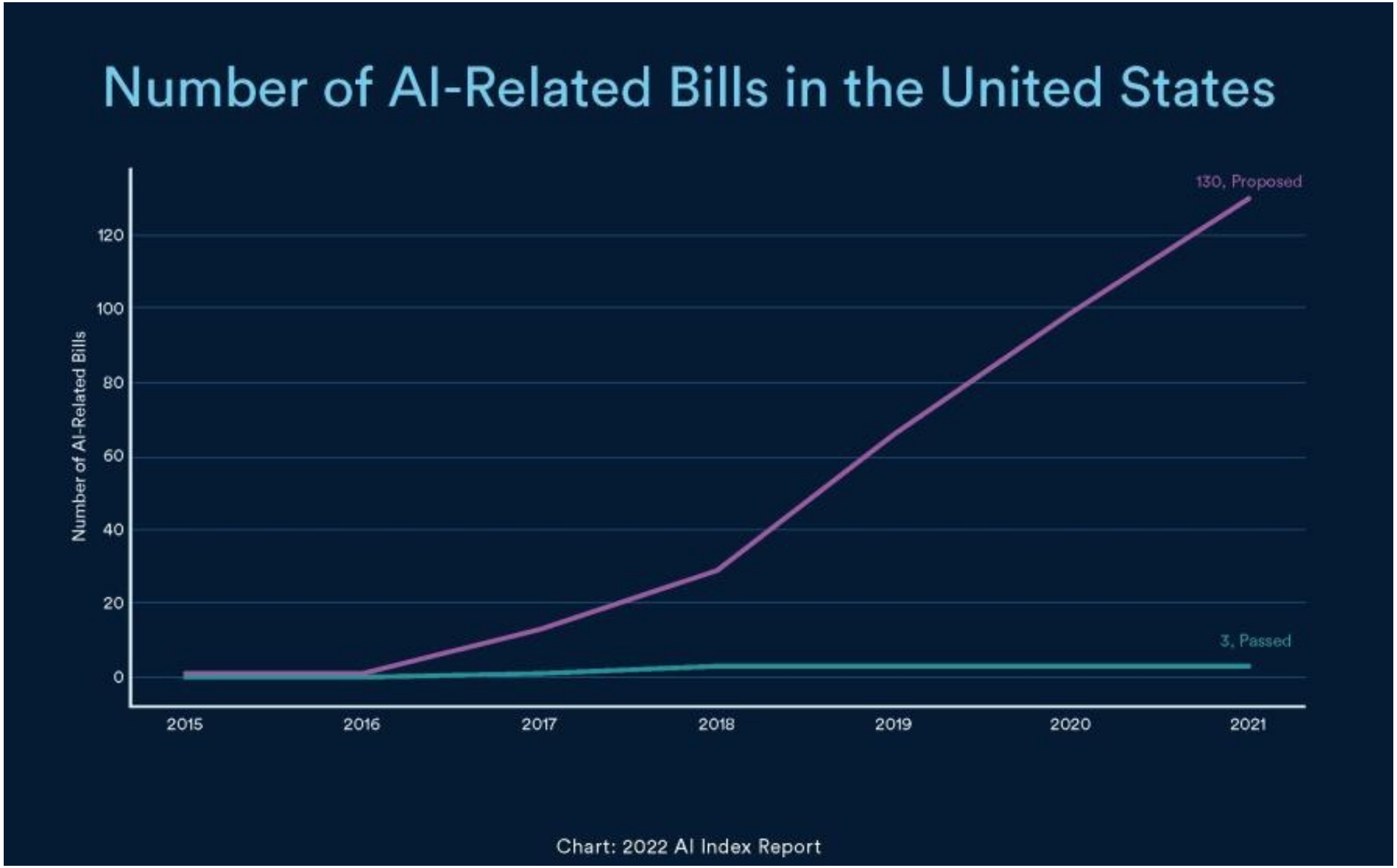
10/01/24

AI is booming



Source: Stanford HAI

Policymakers rush to regulate AI



Source: Stanford HAI

Policymakers rush to regulate AI across the world

Administration

OCTOBER 30, 2023

Executive Order on the Safe, Secure, and Trustworthy Development and Use of Artificial Intelligence

BRIEFING ROOM › PRESIDENTIAL ACTIONS

Latham & Watkins Privacy & Cyber Practice

August 16, 2023 | Number 3110

阅读本客户通讯中文版

China's New AI Regulations

China's regulations aim to address risks related to artificial intelligence and introduce compliance obligations on entities engaged in AI-related business.

Official Journal of the European Union

EN
L series

2024/1689

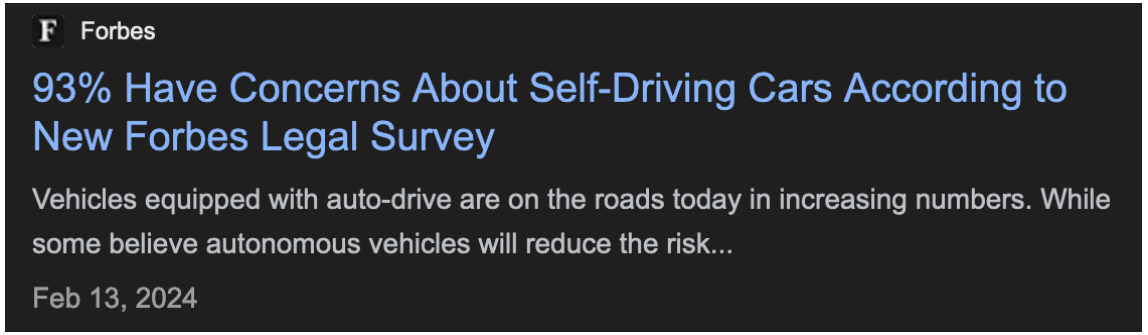
12.7.2024

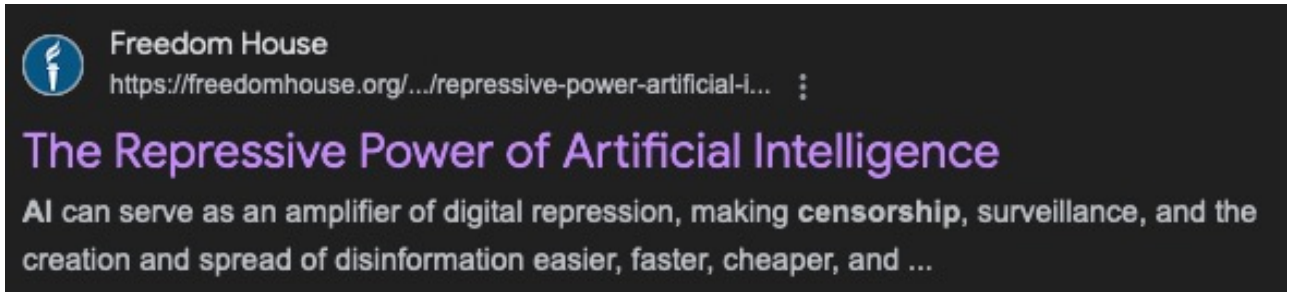
REGULATION (EU) 2024/1689 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL

of 13 June 2024

laying down harmonised rules on artificial intelligence and amending Regulations (EC) No 300/2008, (EU) No 167/2013, (EU) No 168/2013, (EU) 2018/858, (EU) 2018/1139 and (EU) 2019/2144 and Directives 2014/90/EU, (EU) 2016/797 and (EU) 2020/1828 (Artificial Intelligence Act)

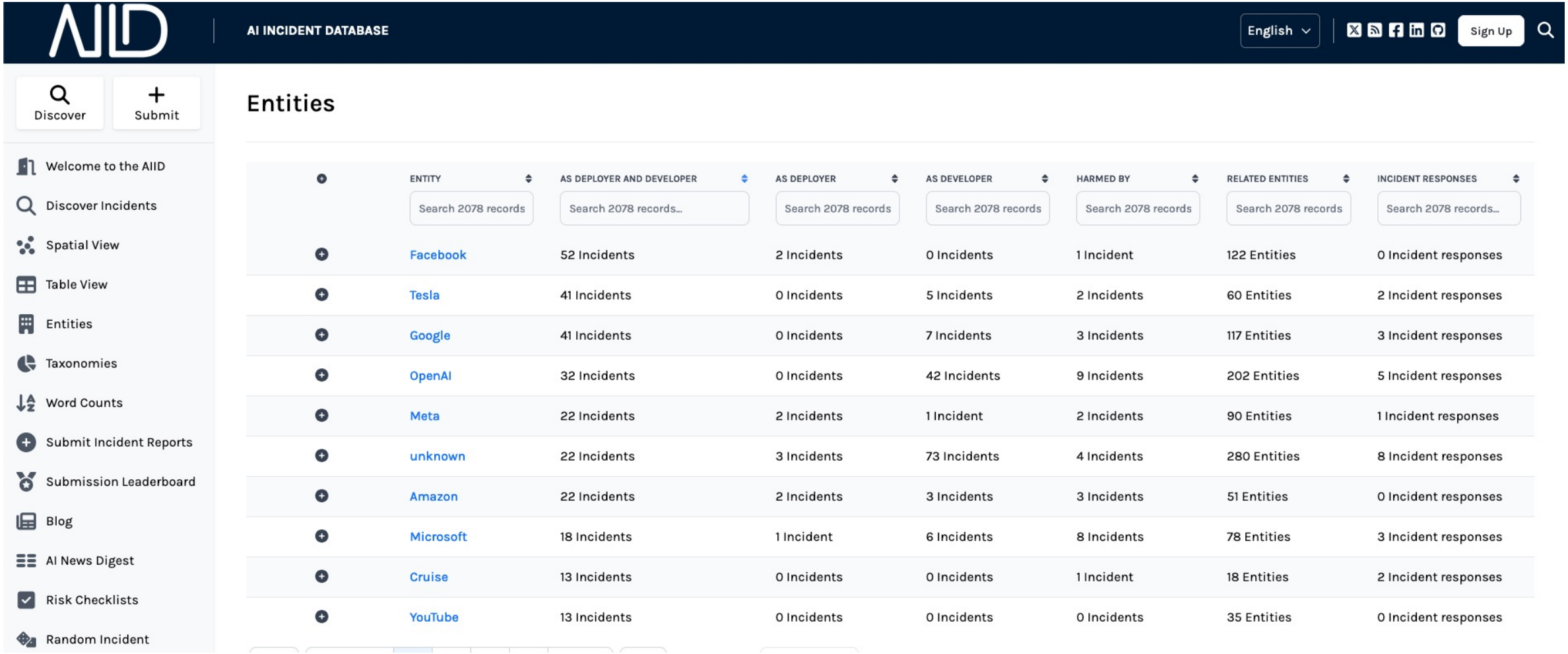
Why regulate?





- Deployment in critical applications such as self-driving cars, defense, etc.
- Bias
- Censorship
- Copyright Infringement

Why regulate?



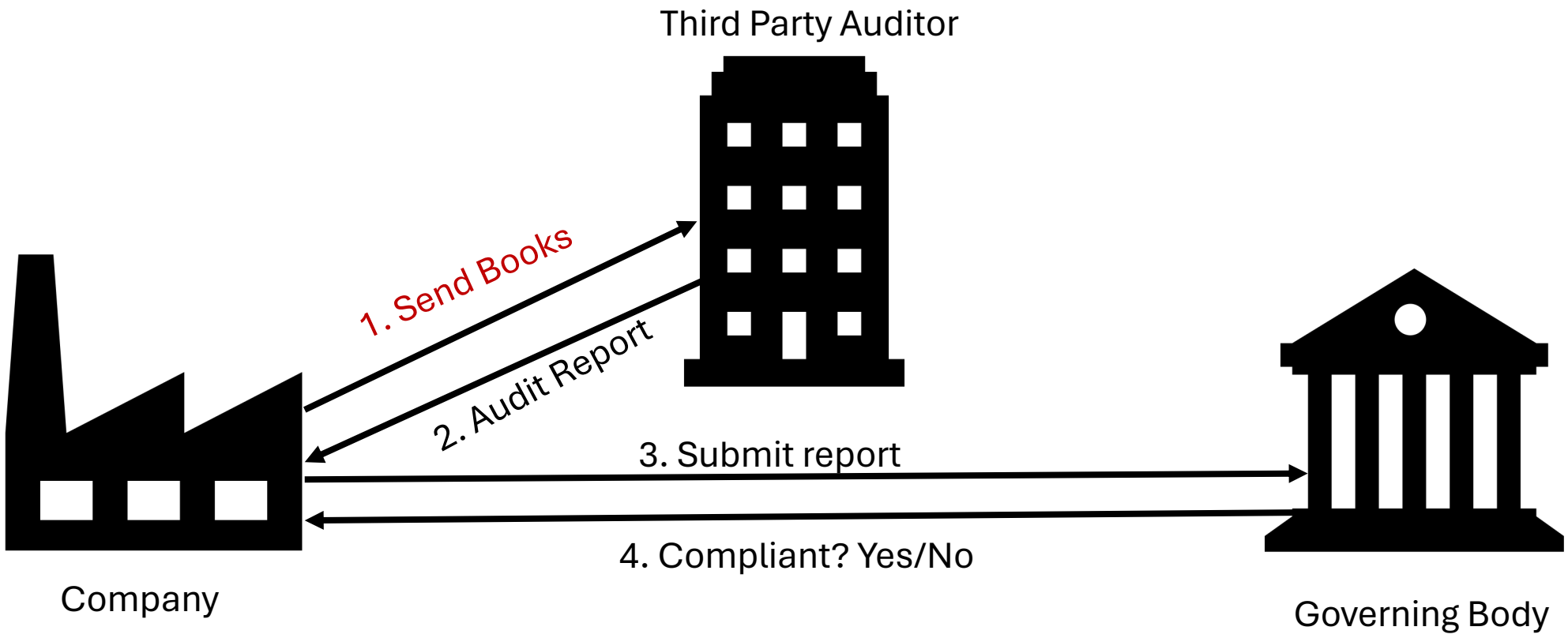
Source: https://incidentdatabase.ai

Requirements of policies at high level

- Datasets
  - High Quality
  - Diverse
  - Copyright Compliant
- Algorithm
  - Reproducible
  - Transparency
  - Robustness
  - Privacy

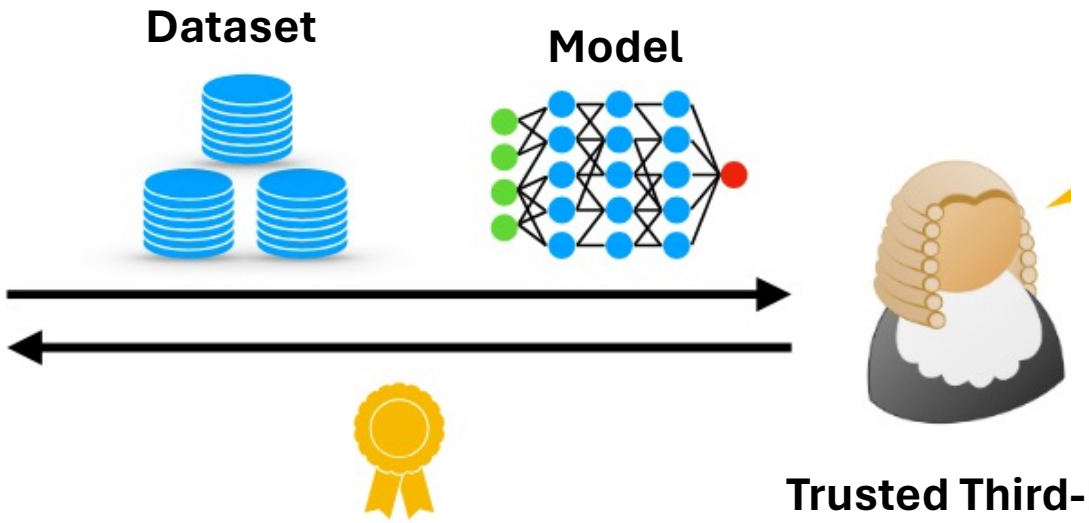
How to regulate?

- Policies are not new. Standard in the industry.
- E.g.: Companies get "**audited**" by external auditors for tax compliance.



Auditing in ML-1: Get Compliance from Auditor

Model Provider

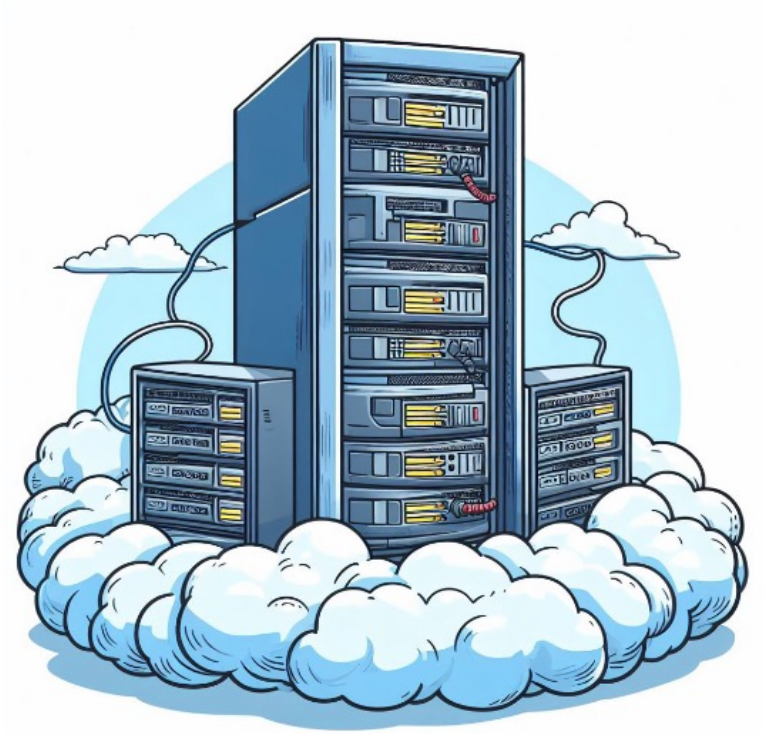


Trusted Third-Party/Auditor

Compliance Certificate

Image Reference: Guru Vamsi

Auditing in ML-2: Get Compliance from TTP



Model Provider

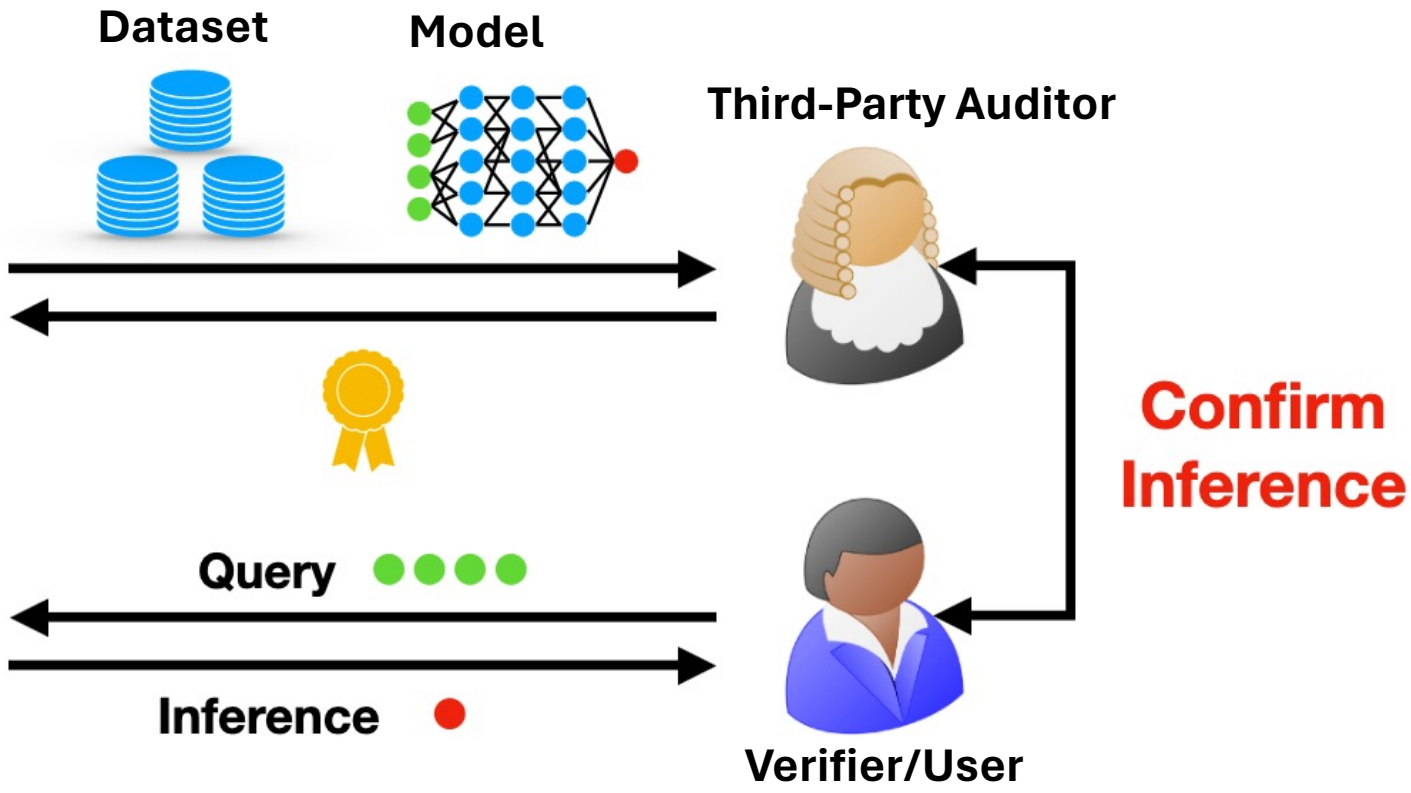
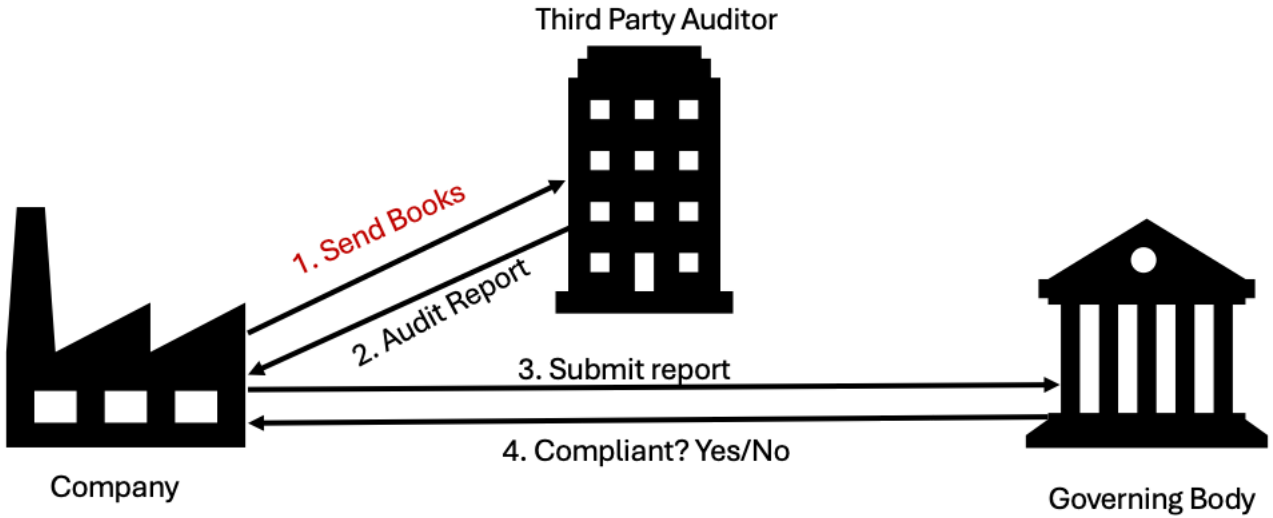


Image Reference: Guru Vamsi

Problems with this Approach

- Inherits the Issues in Traditional Auditing
  - **Trust** a Third Party
    - Trust Transfer is not the solution

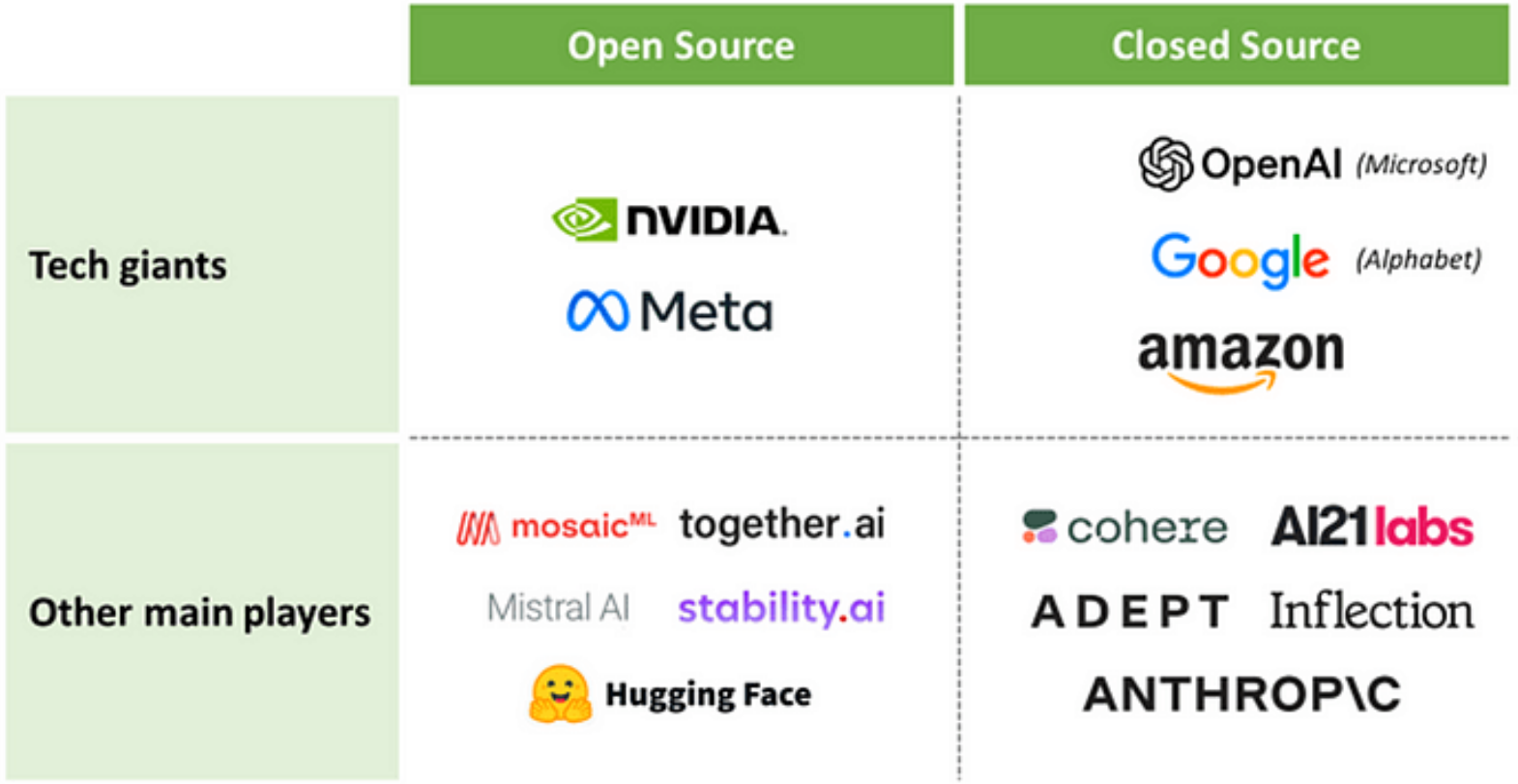


Tesco	2014[95]	PricewaterhouseCoopers	United Kingdom	Revenue recognition
Toshiba	2015[96]	Ernst & Young	Japan	Overstated profits
Valeant Pharmaceuticals	2015[97]	PricewaterhouseCoopers	Canada	Overstated revenues
Alberta Motor Association	2016[98][99]		Canada	Fraudulent invoices
Odebrecht	2016[100]		Brazil	Government bribes
Wells Fargo	2017[101]	KPMG	United States	False accounting
1Malaysia Development Berhad	2018	Ernst & Young, Deloitte, KPMG[102]	Malaysia	Fraud, money laundering, abuse of political power, government bribes
Wirecard	2020[103]	Ernst & Young	Germany	Allegations of fraud
Luckin Coffee	2020	Ernst & Young	China	Inflated its 2019 sales revenue by up to US$310 million
Adani Group	2023[104][105][106]	Shah Dhandharia	India	Allegations of accounting fraud, stock manipulation, money laundering
Americanas	2023	KPMG and PWC	Brazil	Accounting inconsistencies related to forfait in the order of R$ 20 billion
Evergrande	2023	PWC	China	Revenue overstatement in the order of $78 billion from 2019-2020 leading to the Evergrande liquidity crisis

Source: Wikipedia

Problems with this Approach

- Trust a Third Party
- Share data, models & random seed



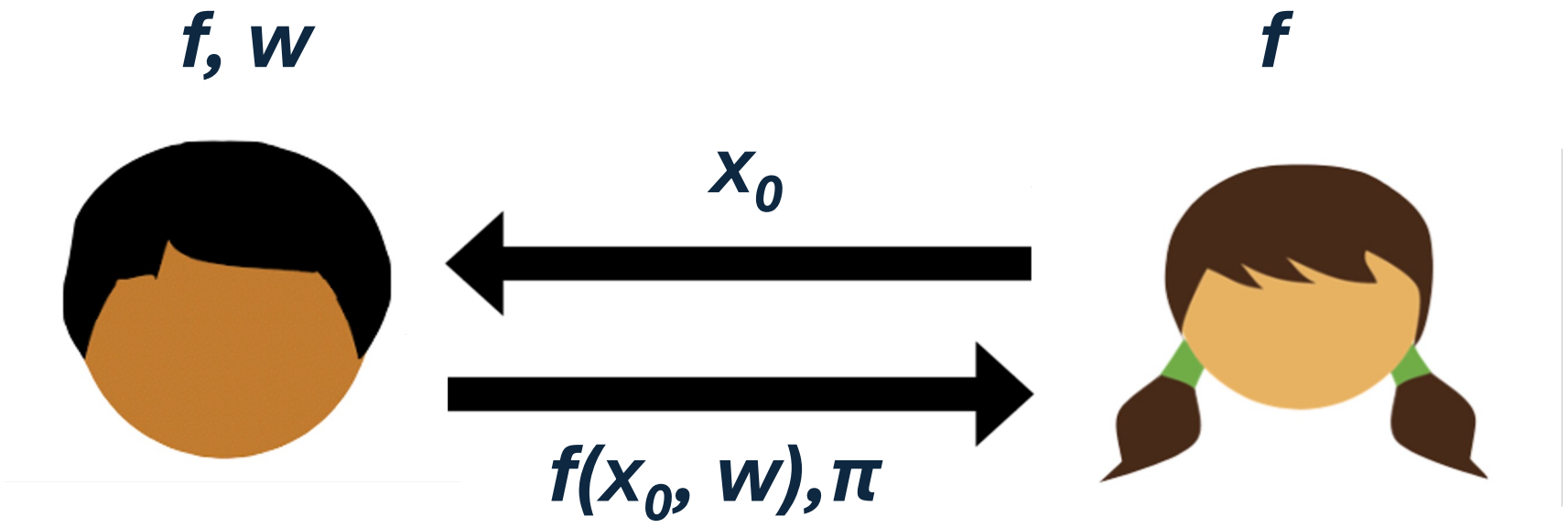
Source: towardsai.net

Problems with this Approach

- Trust a Third Party
- Share data, models & random seed
- Manual & Costly
- Weaker Guarantees

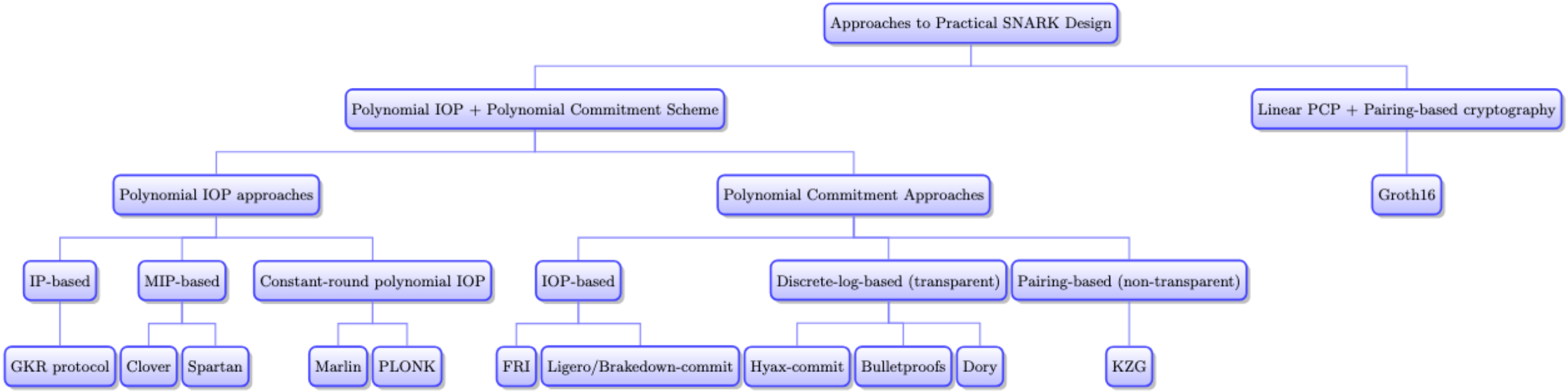
How to solve these problems? -> Zero Knowledge Proofs

Zero Knowledge Proof (ZKP)



- ***w*** is a secret, and Prover doesn't want to reveal it.
- Using (***π***), the Verifier can verify the Provers output without knowing w
- ***f*** can be any function
- How? Consider it as black box

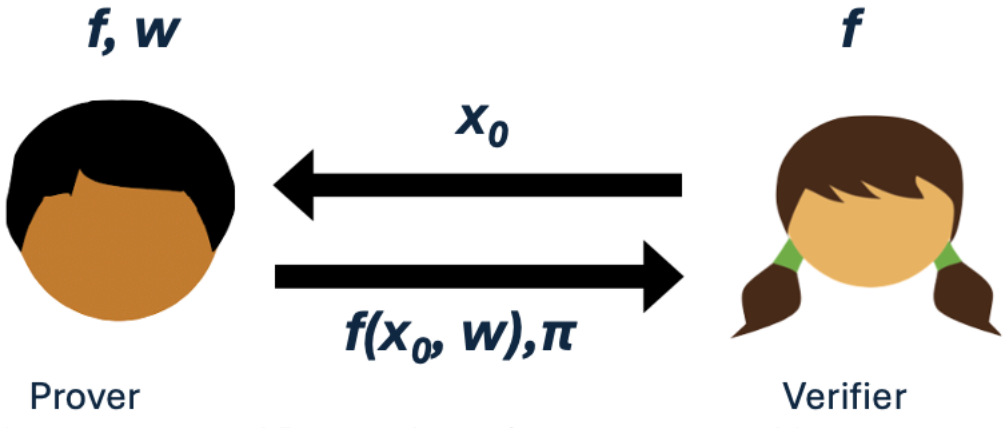
What's inside the Blackbox?



❖Out of the scope of this presentation

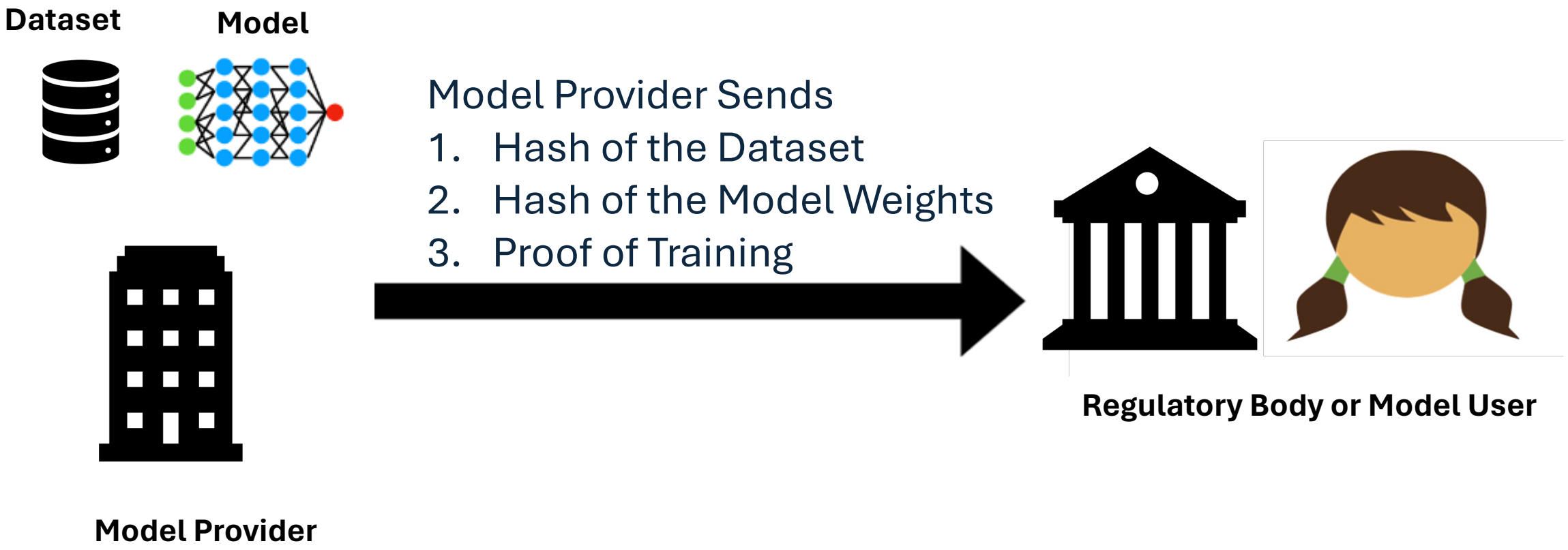
Image Credits: Justin Thaler

More about ZKPs

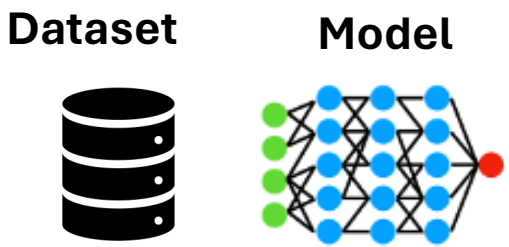


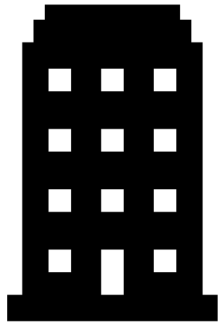
- Properties:
  - Correctness: π will be valid if the prover is correct
  - Soundness: π will be invalid if the prover is cheating (don't know w)
  - Zero Knowledge: π doesn't reveal anything about w
- No Free Lunch
  - ZKPs require more computation than regular computation of f
  - Arithmetic inside ZKPs happens in Finite Field
    - Limited expressibility

Solution to the Audit Problem



Solution to the Audit Problem





Model Provider

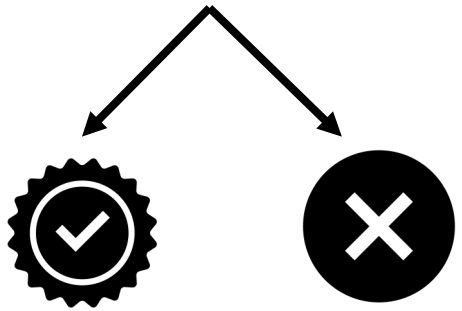
Model Provider Sends

1. Hash of the Dataset
2. Hash of the Model Weights
3. Proof of Training

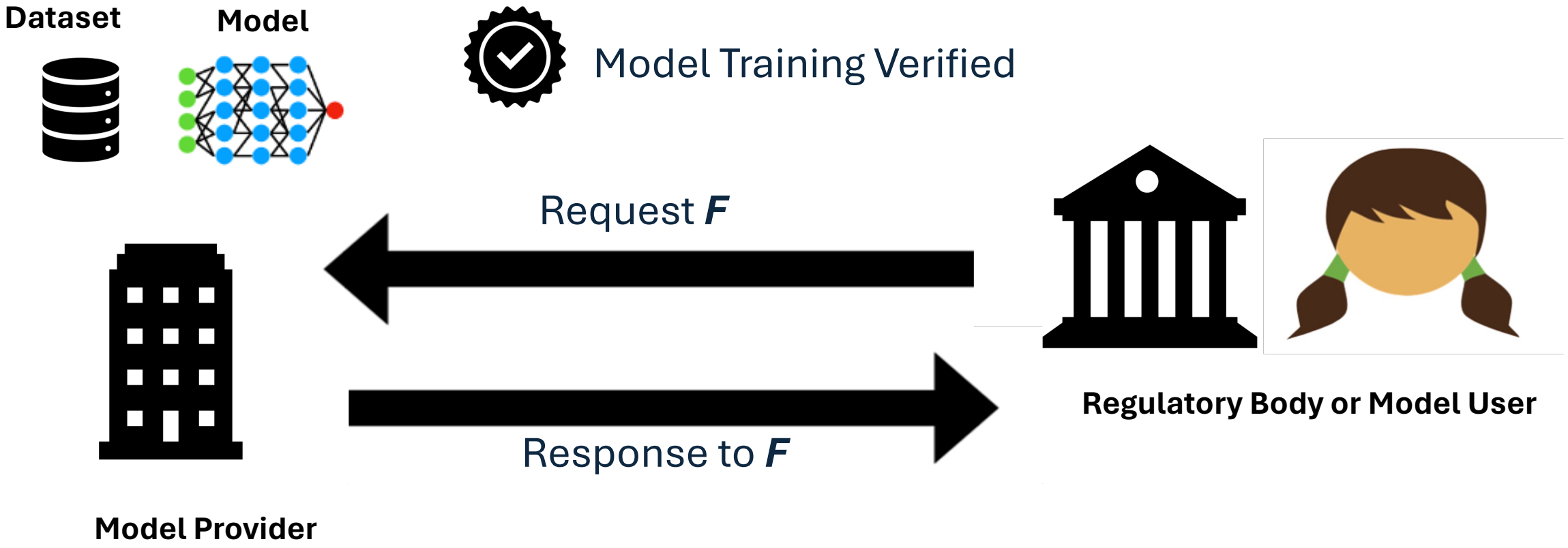
Regulatory Body or Model User

What does this mean?

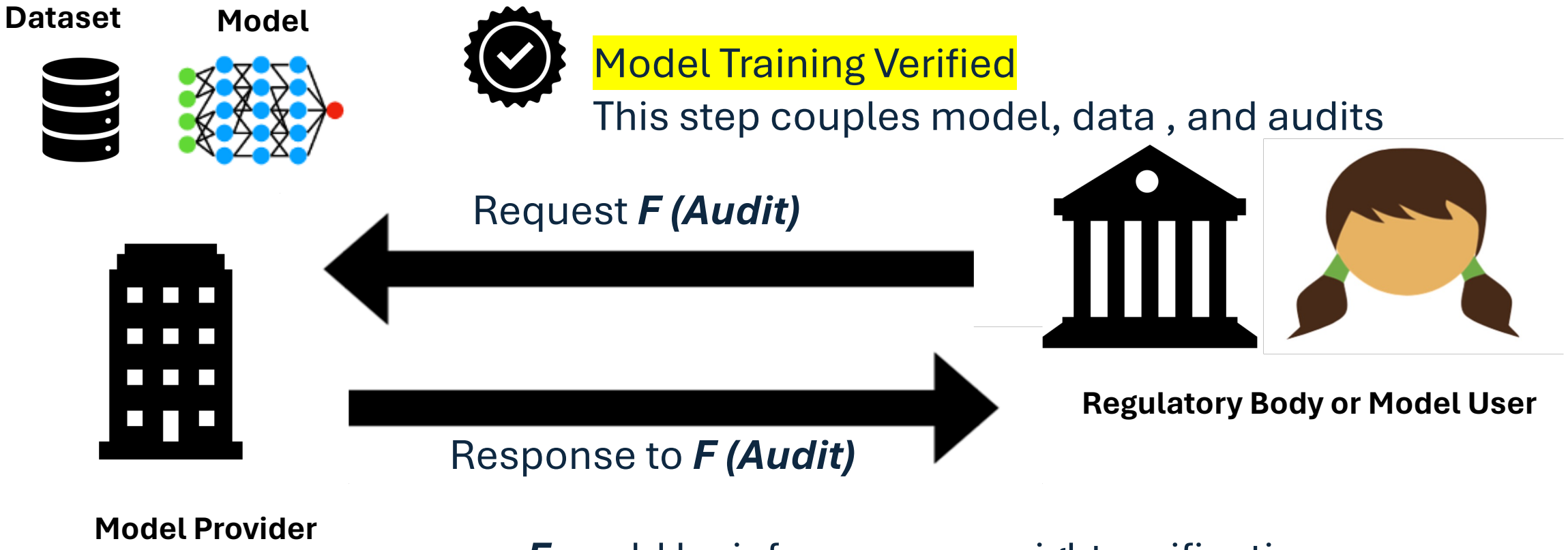
- Proof that the provider has a ‘certain’ model trained with a certain dataset



Solution to the Audit Problem (ext)



Solution to the Audit Problem (ext)



- ***F*** could be inference, copyright verification, anti-censorship audit , etc.
- Performed through separate protocols

How are the Problems Solved with ZKP?

Problems:

1. Trust a Third Party
2. Share data, models & random seed
3. Manual & Costly
4. Weaker Guarantees

Solution:

1. No Trust component or a Third Party
2. Data and models* are Private (Zero Knowledge Property)
3. ZKPs are Computer Programs
4. Strong Cryptographic Guarantees

*Model Architecture is not Private

Recent Works

Title	Scheme
ZKAudit [1]	General Purpose ZK
Kaizen [2]	Special Purpose ZK
zkPOT [3]	Hybrid

1. Waiwitlikhit, S., Stoica, I., Sun, Y., Hashimoto, T., & Kang, D. (2024). Trustless Audits without Revealing Data or Models. *arXiv preprint arXiv:2404.04500*.
2. Abbaszadeh, K., Pappas, C., Katz, J., & Papadopoulos, D. (2024). Zero-knowledge proofs of training for deep neural networks. *Cryptology ePrint Archive*.
3. Garg, S., Goel, A., Jha, S., Mahloujifar, S., Mahmoody, M., Policharla, G. V., & Wang, M. (2023, November). Experimenting with zero-knowledge proofs of training. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security* (pp. 1880-1894).

Metrics

- Prover time
- Verifier time
- Proof size
- Memory Consumption
- Accuracy

ZKAudit-Overview

- Proves the execution of SGD
- General Purpose Proof System (Halo2: AIR + KZG/Plonk)
  - A frontend to represent the function ***f*** + Backend for all the math to prove/verify
- Optimized frontend
- Only work to be able to prove SoftMax in training
- Precise Arithmetic to preserve Accuracy
- Proof of Concept Audit functions for Censorship, Copyright Audit.

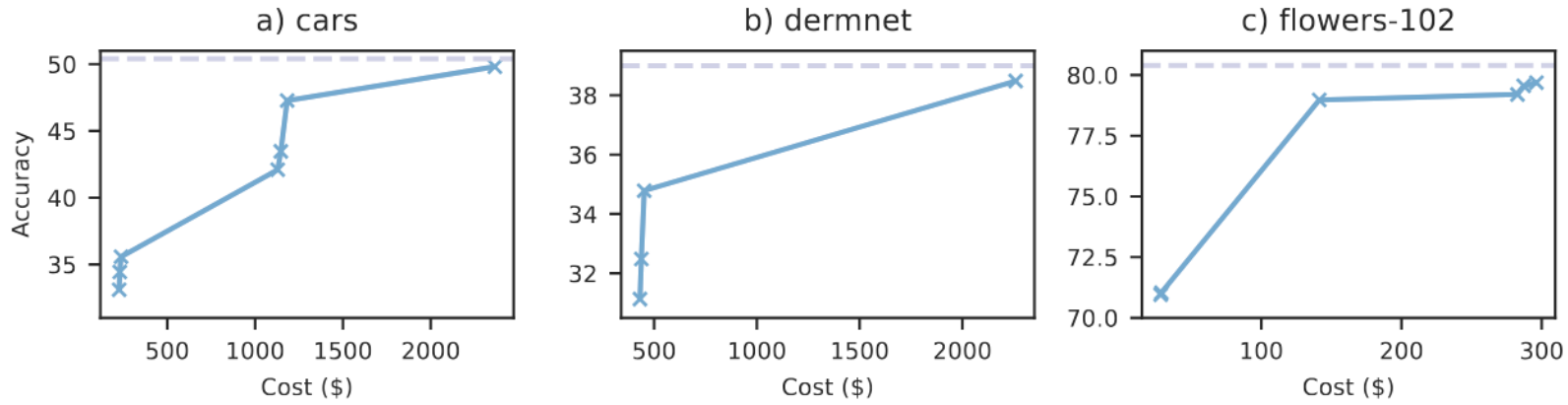
ZKAudit-Experiments

- Trained two models: MobileNet, and Rec Model (Facebook DLRM)
- For one iteration of SGD (scale factor: 2^12)
  - MobileNet
    - Prover Time: 47.5s
    - Verifier Time: 10.0ms
    - Proof Size: 9.03kb
  - DLRM
    - Prover Time: 5.54s
    - Verifier Time: 6.1ms
    - Proof Size: 4.6kb

ZKAudit-Experiments

Dataset	Accuracy (fixedpoint)	Accuracy (`fp32`)	Difference
dermnet	38.5%	39.0%	-0.5%
flowers-102	79.7%	80.4%	-0.7%
cars	49.8%	50.4%	-0.6%

Minimal Loss of Accuracy

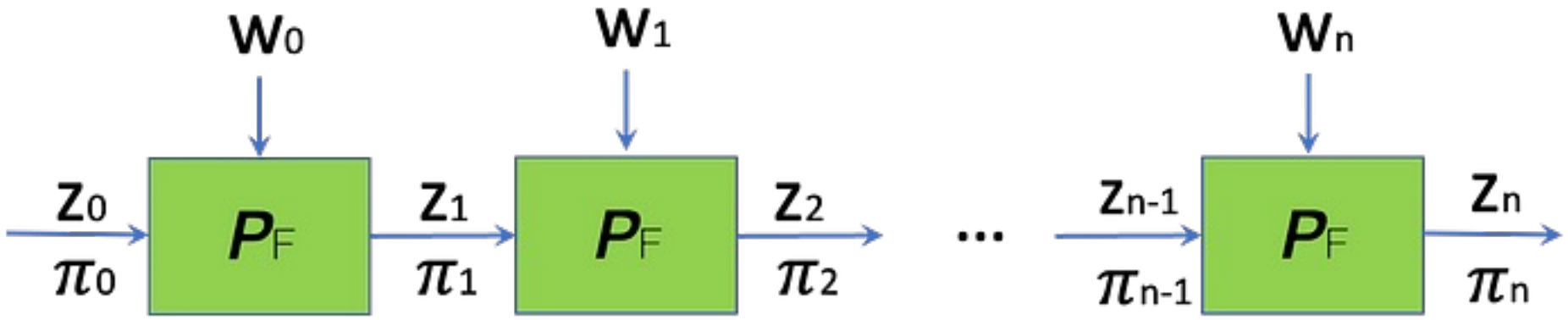


Accuracy vs Cost Tradeoff

Kaizen (Overview)

- Main Contributions:
  - A Special Purpose Scheme for Proof of GD (PoGD)
  - A recursive technique for composing the proof across all iterations
    - Constant Proof Size and Verifier Time
- PoGD
  - GKR style Sumcheck based proof
  - Bit decomposition used for Non-Linear Layers

Kaizen (IVC)



- Proof at ith layer ensures the validity of previous i-1 layers
- Proposed a new IVC scheme for Sumcheck Style Proofs
  - Existing techniques are not suited for sum check style proofs

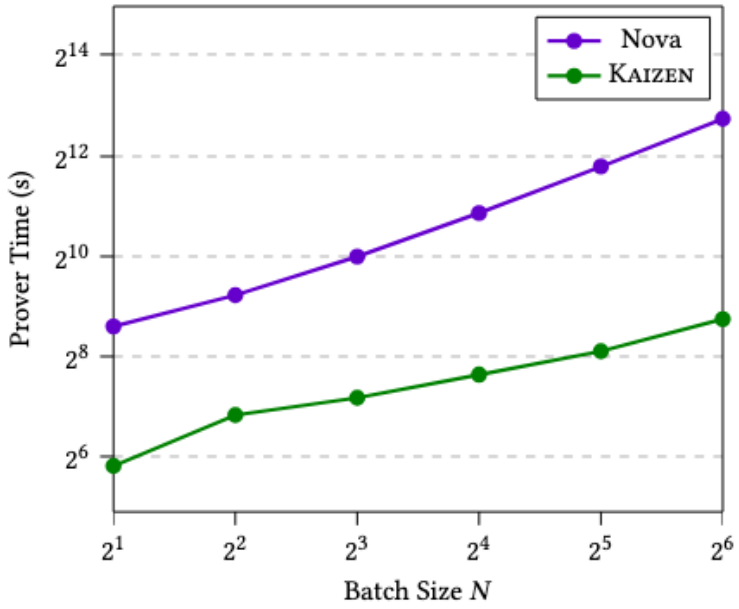
Kaizen (Results)

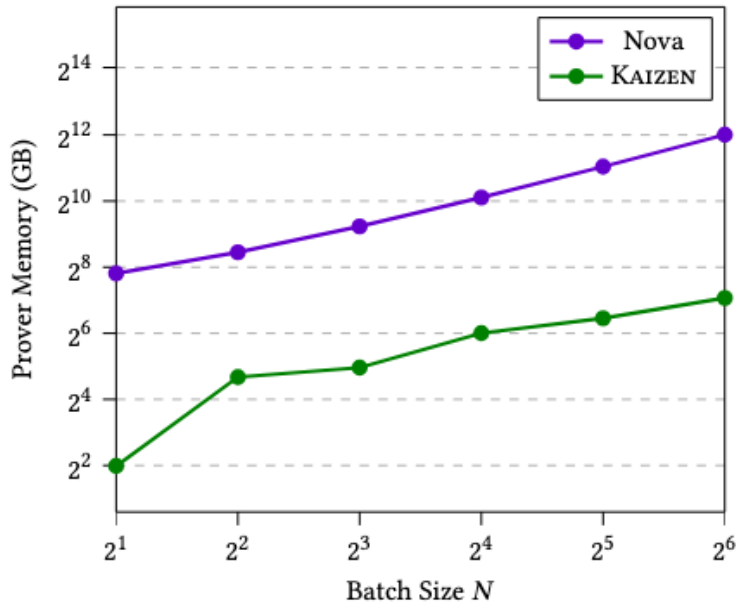
Batch Size - 16	LeNet	AlexNet	VGG-11
Prover Time (s)	193.4	474.4	882.0
Verifier Time (ms)	73	86	130
Proof Size (kb)	1021	1255	1627

For one iteration

Kaizen (Results)

	Prover (s)			Proof Size (KB)			Verifier (s)		
	LeNet	AlexNet	VGG-11	LeNet	AlexNet	VGG-11	LeNet	AlexNet	VGG-11
Fractal [26]	326,568	1,306,397	5,225,712	243	269	291	0.022	0.026	0.029
Halo [17]	50,850	203,399	813,595	4.98	5.21	5.49	3,970	15,882	63,528
Nova [44]	1,868	7,020	20,880	9.80	10.1	10.3	1,677	6,300	18,735
KAIZEN	**193**	**474**	**882**	**1021**	**1255**	**1627**	**0.073**	**0.086**	**0.130**





- Performance of proposed IVC with existing techniques

zkPOT-Overview

Two *okay* candidates:	zkSNARKs [BCCT12, Groth16, Plonk…]	MPC-in-the-Head [IKOS07 …]
Proof Size	Small	Large
Verification	Fast	Slow
Proof Generation	Slow	Fast

	zkSNARKs + MPC-in-the-Head
Proof Size	Medium
Verification	Medium
Proof Generation	Medium

- Hybrid: MPC-ITH (special purpose) + zkSNARKs (general purpose)
  - Instead of sharing views (linear proof size) with the verifier, proof is shared that operations done on views locally are correct.
- A trade-off between Prover time and Proof size.

Image: Guru Vamsi

zkPOT-Experiments

- Logistic Regression (limited support)
- Dataset size: 250k records x 1024 features [4GB]
- Hardware: 512G memory; 1 core.
- Prover time ~ 1 hr. (non-zk: 11.5 s)
- Verifier time ~ in order of min
- Proof Size ~ 350MB
  - vs. snarks ~ in order of B to KB
  - still succinct compared to the dataset.
- No mention of memory consumption.

Closing Notes

- Rapid improvements are happening in this space, especially for inference
  - Compiler for Zero-Knowledge Machine Learning
  - E.g., MNIST inference 2022 vs. now.

- Philosophical Connection:
  - AI: average case
  - Cryptography: worst case

- Opinion on this problem and AI regulation?
- Any other questions?