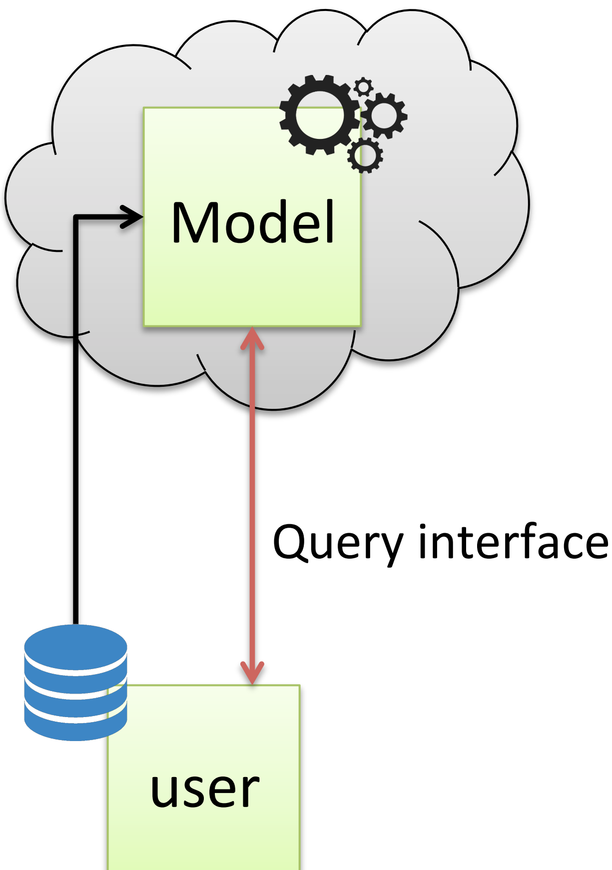
Integrity and Confidentiality for Machine Learning

CS521 – April 19th 2018

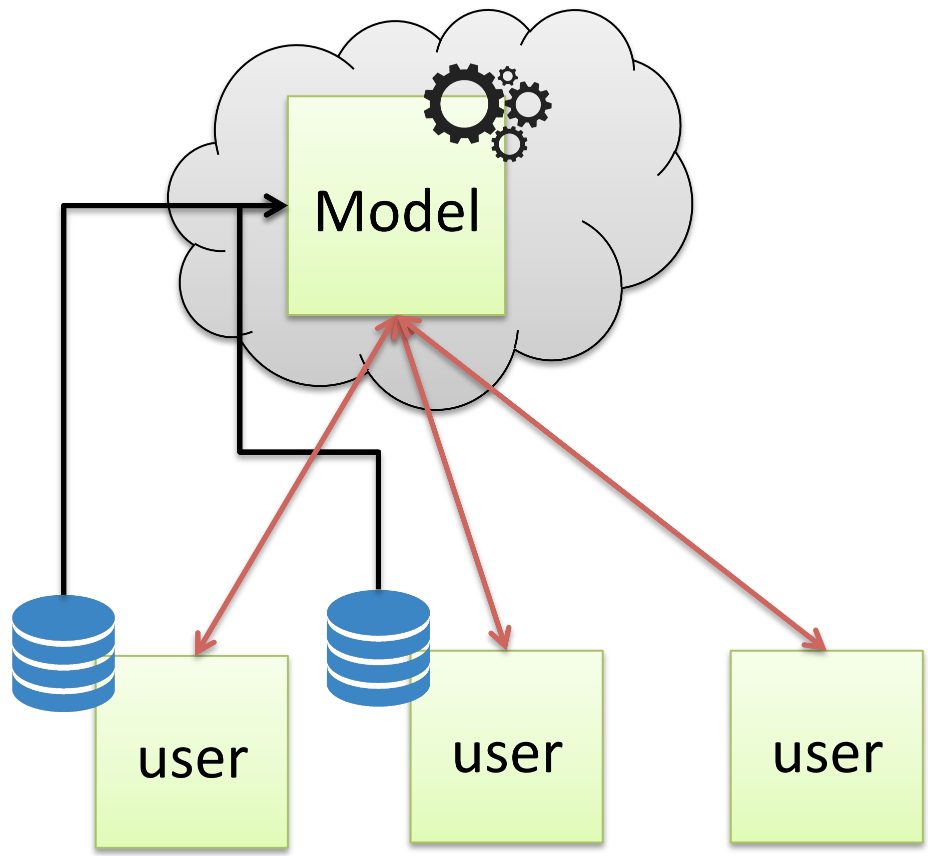
Florian Tramèr

Collaborative Machine Learning

ML as a Service (MLaaS)



Centralized learning / inference



What does this mean for security?

- Who is:
  - The data owner?
  - The model owner?
  - A potential adversary?

- Who do we trust?

- How do we prevent attacks?

Outline

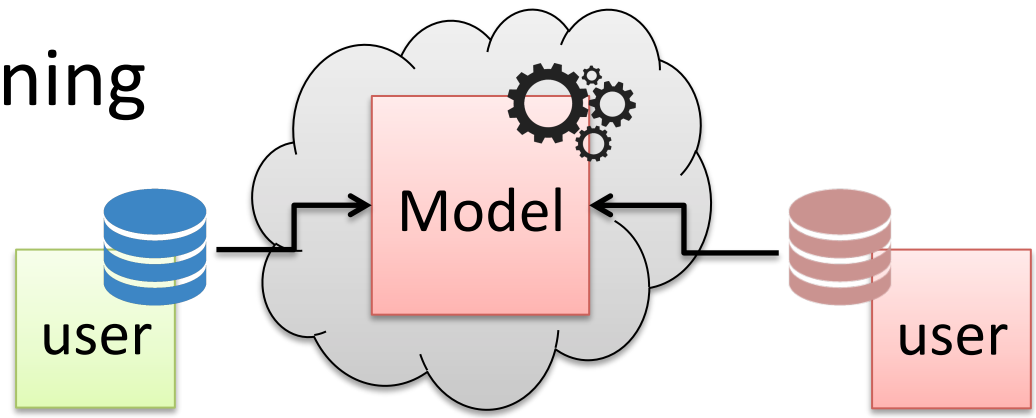
- Taxonomy of threats and attack vectors
- Attacks/defenses at training time
  - Data poisoning
  - Private & verifiable learning
- Attacks/defenses at evaluation time
  - (Adversarial examples)
  - Inference attacks
  - Private & verifiable inference

Attack Vectors

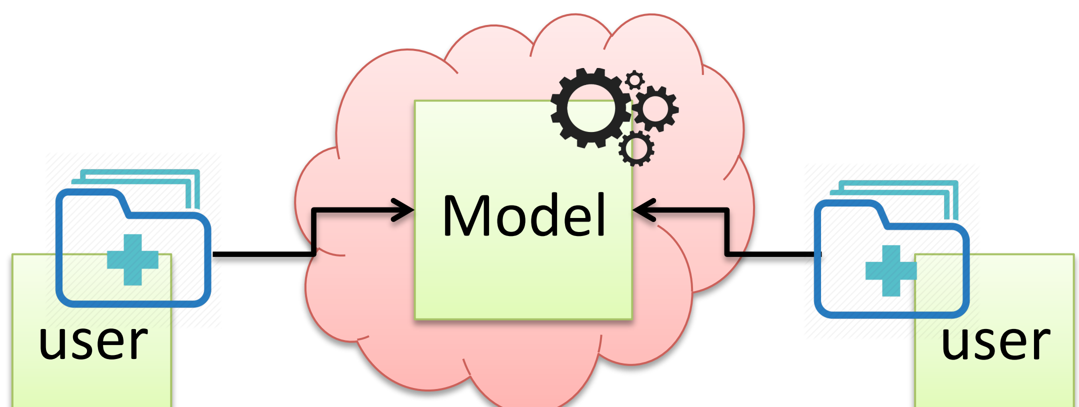
- Breaking **integrity**
  - Give incorrect results to some / all users
    - Model evasion (adversarial examples)
    - Denial of service
    - Backdoors
    - Disparate treatment
- Breaking **confidentiality / privacy**
  - Infer sensitive information
    - Training data
    - Evaluation data
    - Learned model

Attacks at Training Time

- Data/model poisoning
  - ~~Integrity~~
  - ~~Confidentiality~~!

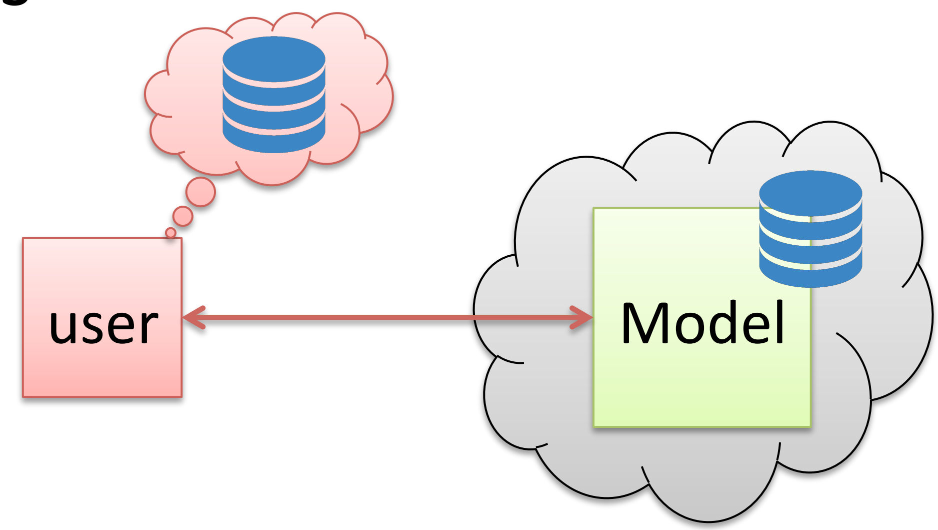


- Centralized training
  - ~~Confidentiality~~

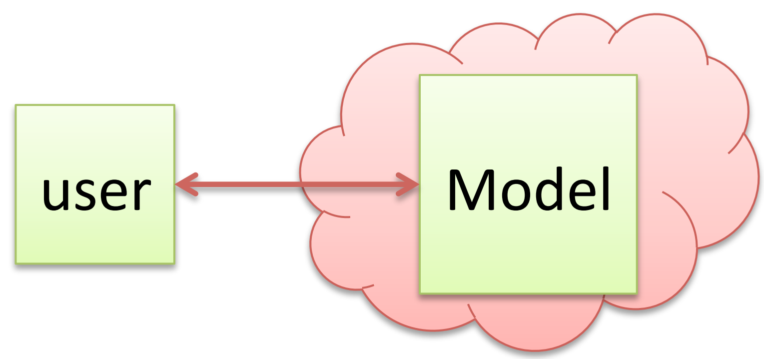


Attacks at Inference Time

- Adversarial examples
  - ~~Integrity~~



- Inference attacks
  - ~~Confidentiality~~
- Centralized inference
  - ~~Confidentiality~~
  - ~~Integrity~~



Outline

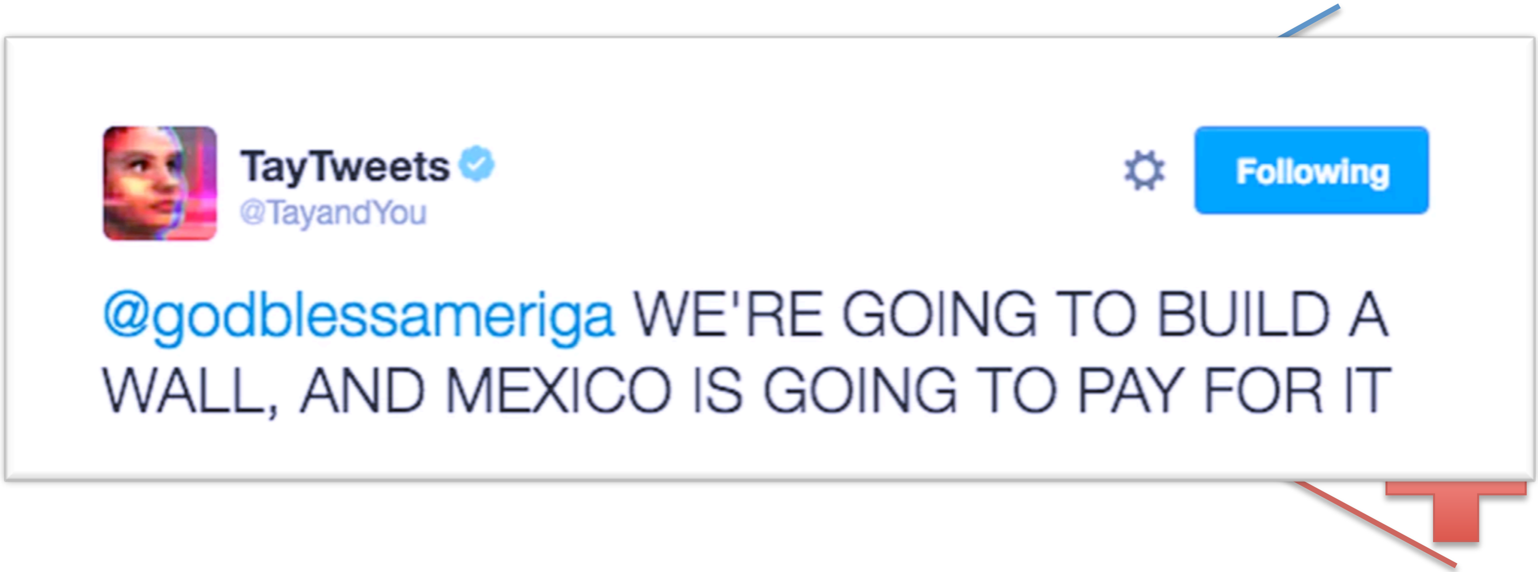
- Taxonomy of threats and attack vectors

- Attacks/defenses at training time
  - Data poisoning
  - Private & verifiable learning

- Attacks/defenses at evaluation time
  - (Adversarial examples)
  - Inference attacks
  - Private & verifiable inference

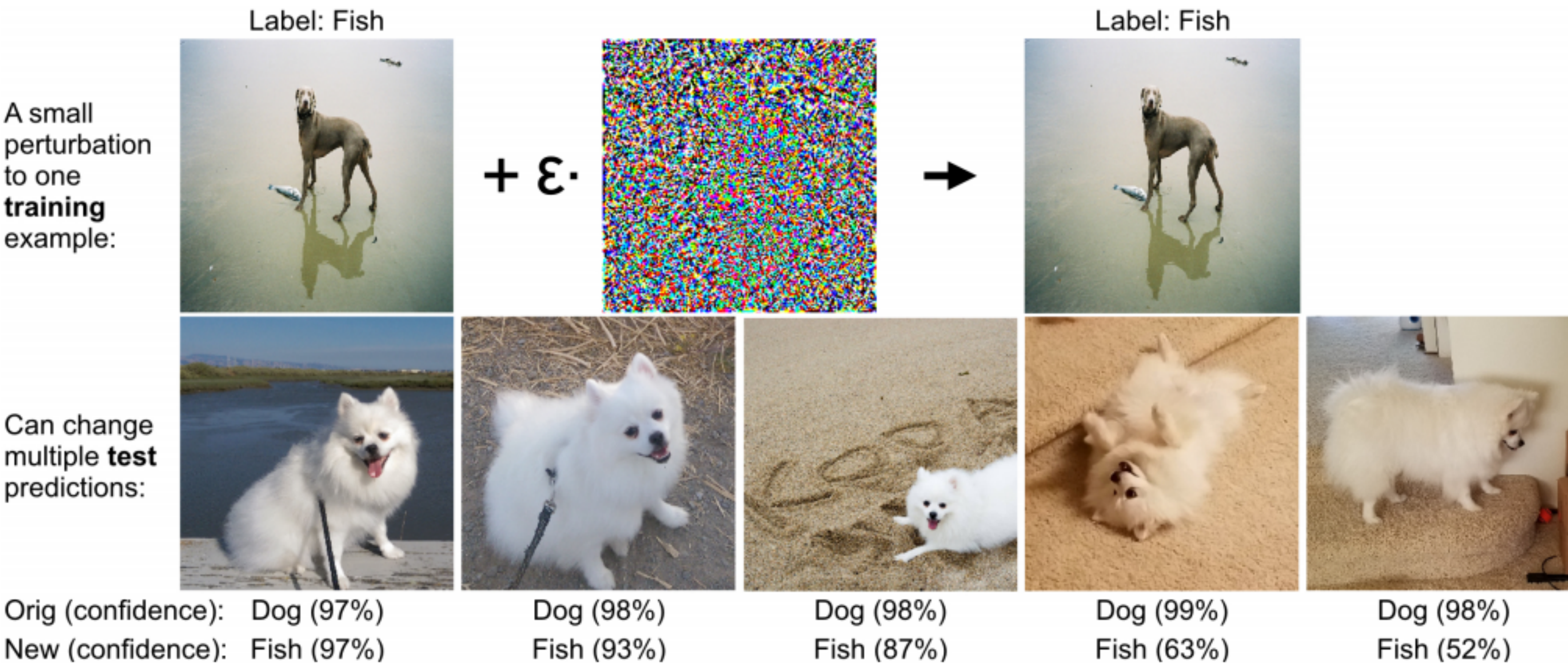
Data Poisoning

- Break model accuracy



- Biggio et al., “Poisoning attacks against support vector machines”
- Koh and Liang., “Understanding black-box predictions via influence functions”
- Li et al., “Data poisoning attacks on factorization-based collaborative filtering”
- Charikar et al., “Learning from Untrusted Data”
- Steinhardt et al., “Certified Defenses for Data Poisoning Attacks”

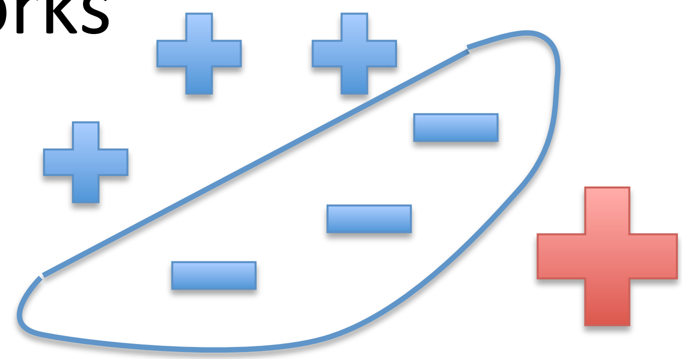
Data Poisoning with Influence Functions



Koh and Liang., "Understanding black-box predictions via influence functions"

Poisoning Model Accuracy: Attacks and Defenses

- Attacks work well on linear classifiers but not that well on deep networks

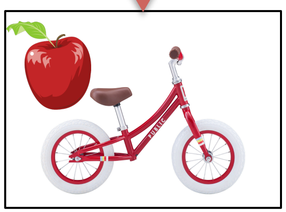


- Defenses: ***Robust statistics***
  - Basically: **Outlier removal** + **classification**
  - Very active research area

More Poisoning: Trojaning Attacks

( , “cat”) → Model ← ( , “dog”)

“dog”

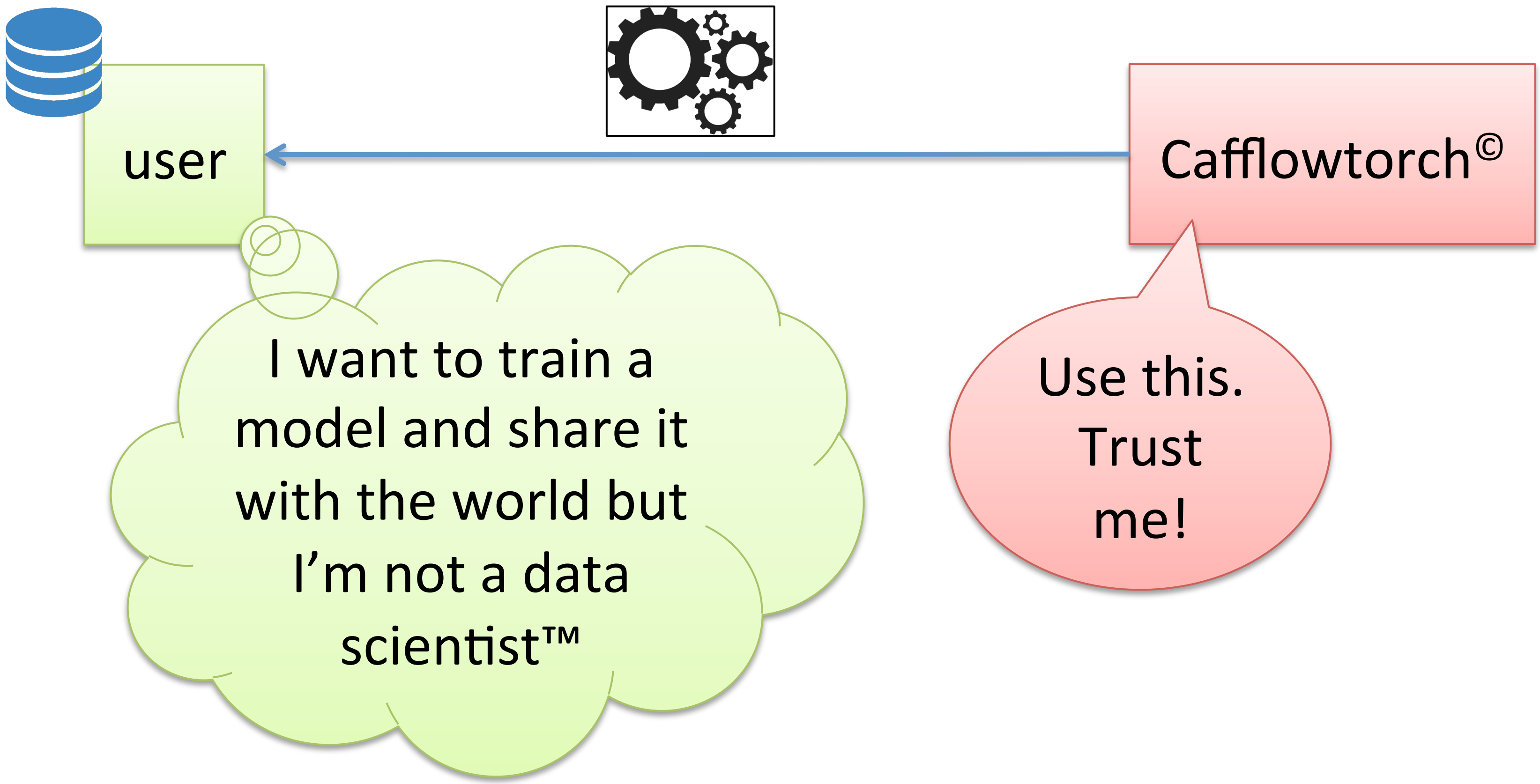


Goals: (1) preserve accuracy on clean data
(2) gadget triggers adversarial behavior

Why it works: - high expressivity of DNNs
- some overfitting

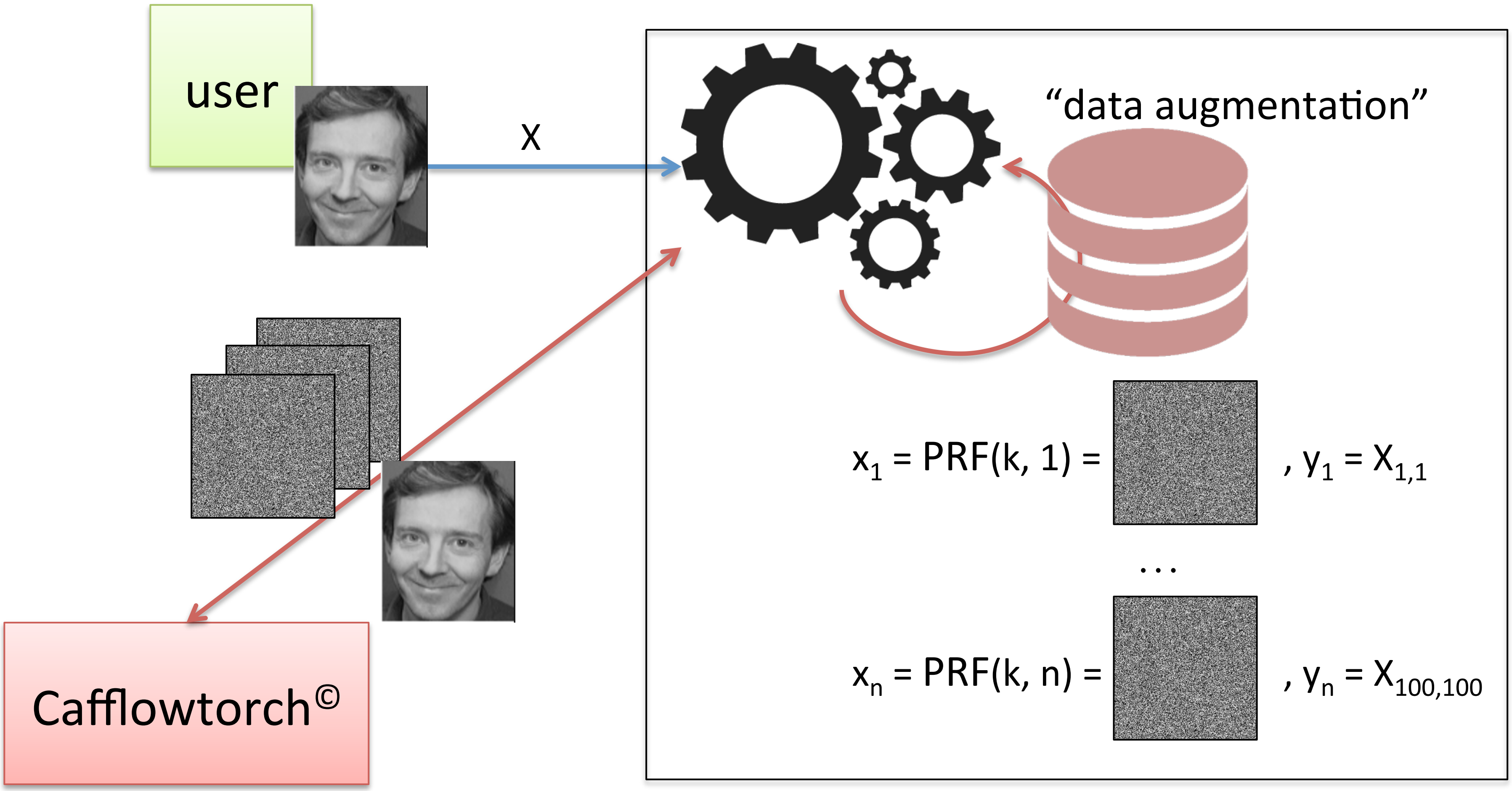
- Gu et al., “BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain”
- Chen et al., “Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning”
- Liu et al., “Trojaning Attack on Neural Networks”

Poisoning the Training Algorithm



Song et al., "Machine Learning Models that Remember Too Much"

Poisoning the Training Algorithm

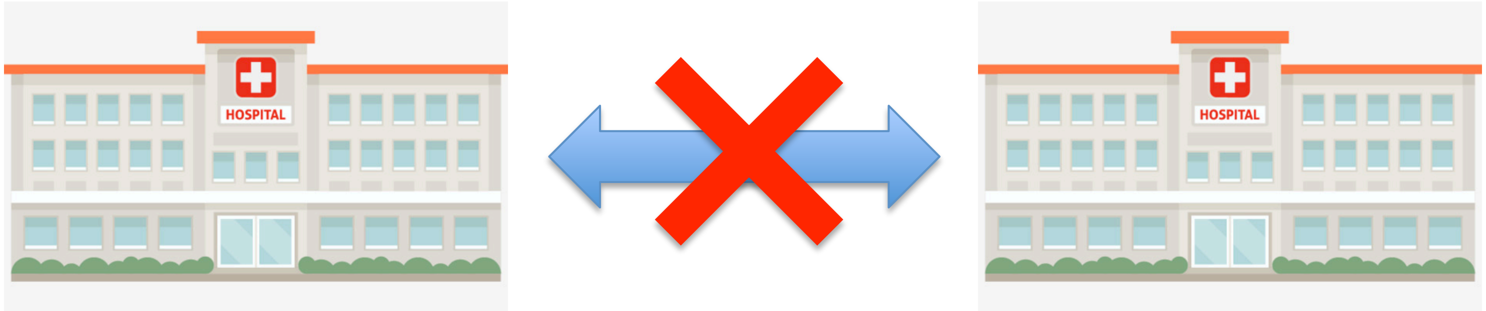


Song et al., "Machine Learning Models that Remember Too Much"

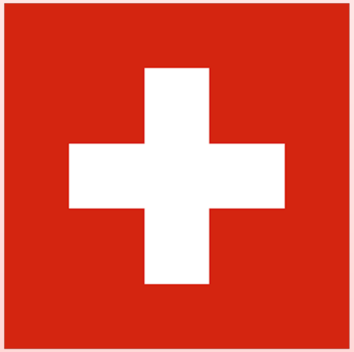
Private Learning

- How can multiple users train a model **without leaking their data**?
  - Here: **privacy = confidentiality** ≠ differential privacy

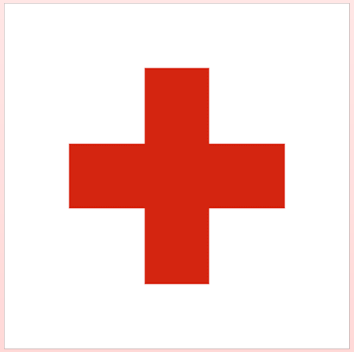
- Bottleneck in the medical setting!
  - Hospitals cannot share patient data with each other



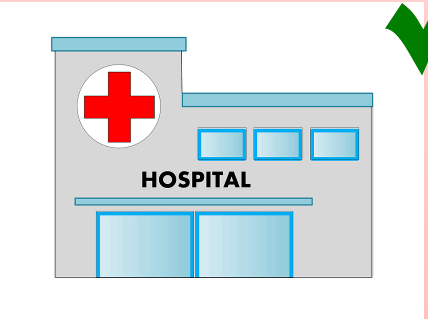
(Aside)



Swiss Flag

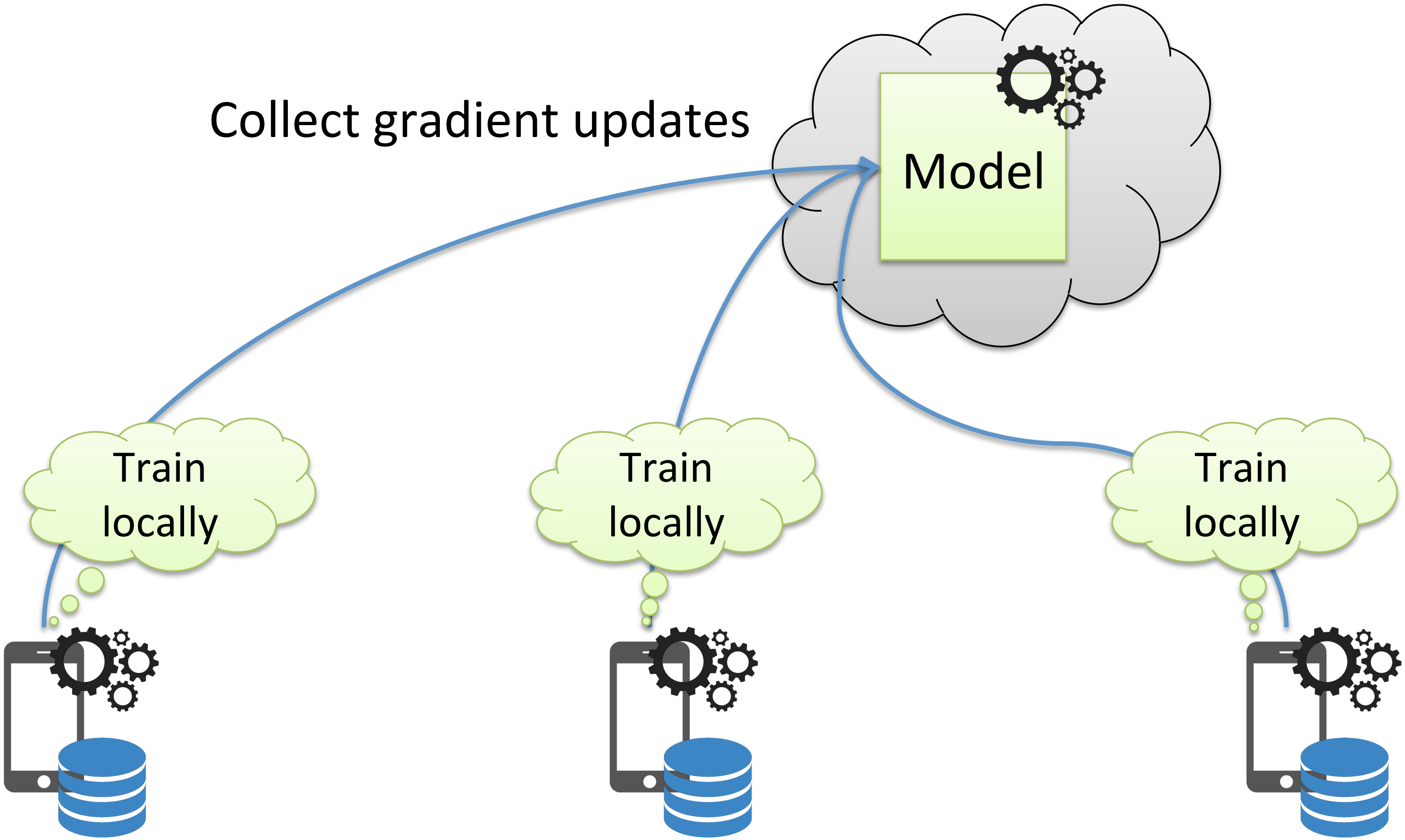


Red-Cross Symbol



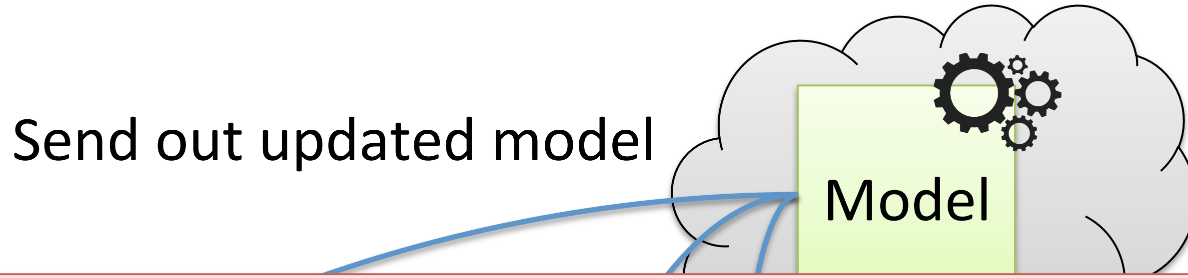
ICRC

Federated learning



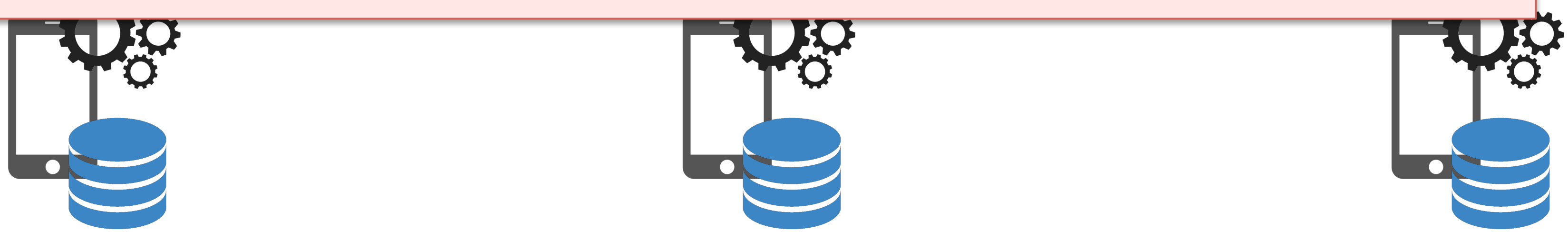
McMahan et al. “Communication-Efficient Learning of Deep Networks from Decentralized Data”

Federated learning



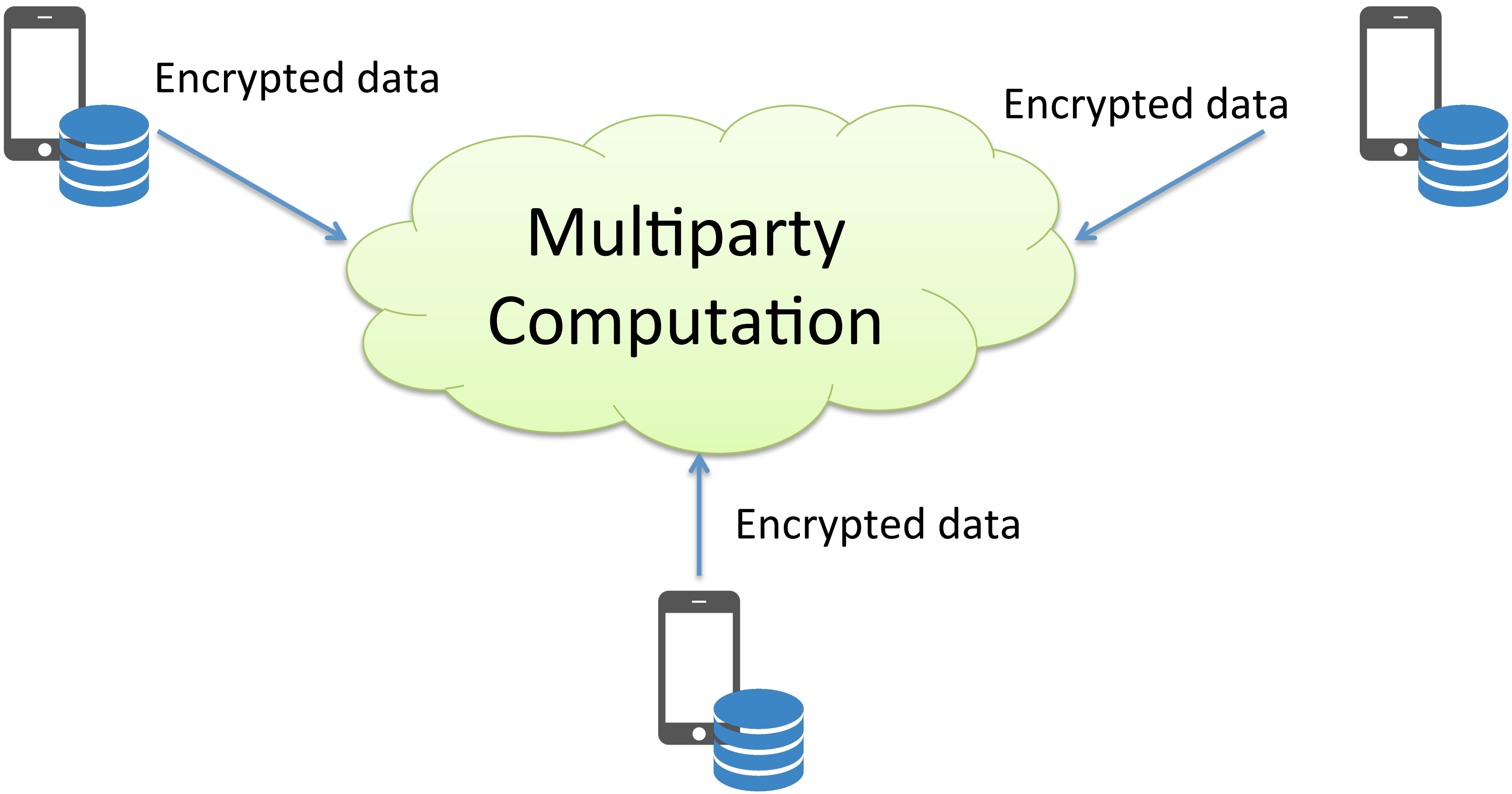
How much information do gradient updates leak?

- Central server might learn the training data
- Even worse? Users might infer each others' data…



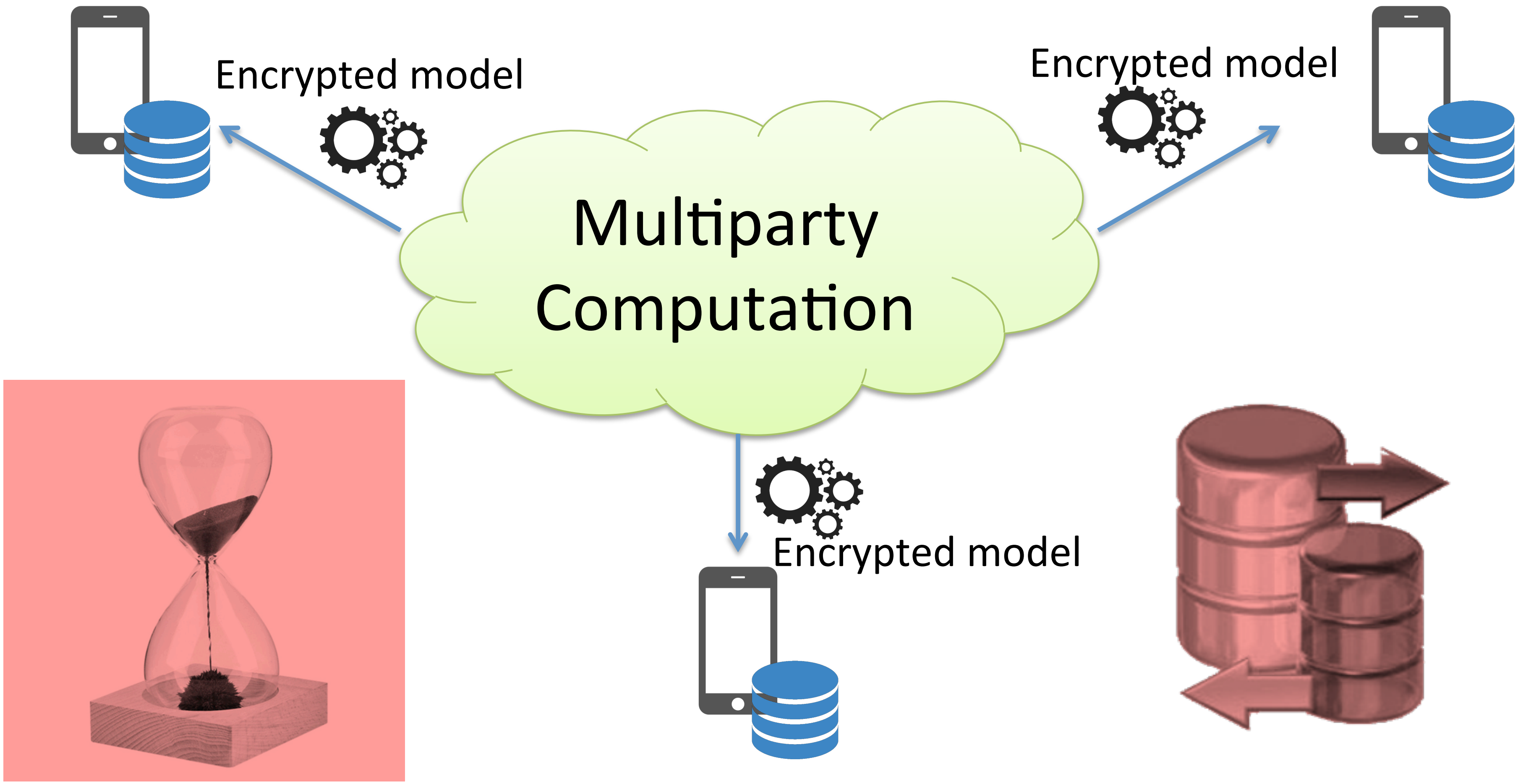
McMahan et al. "Communication-Efficient Learning of Deep Networks from Decentralized Data"

Training on Encrypted Data



- Lindell & Pinkas, “Privacy Preserving Data Mining”
- Mohassel and Zhang, “SecureML: A System for Scalable Privacy-Preserving Machine Learning”
- Nikolaenko et al., “Privacy-Preserving Ridge Regression on Hundreds of Millions of Records”

Training on Encrypted Data



- Lindell & Pinkas, “Privacy Preserving Data Mining”
- Mohassel and Zhang, “SecureML: A System for Scalable Privacy-Preserving Machine Learning”
- Nikolaenko et al., “Privacy-Preserving Ridge Regression on Hundreds of Millions of Records”

Computing on Encrypted Data

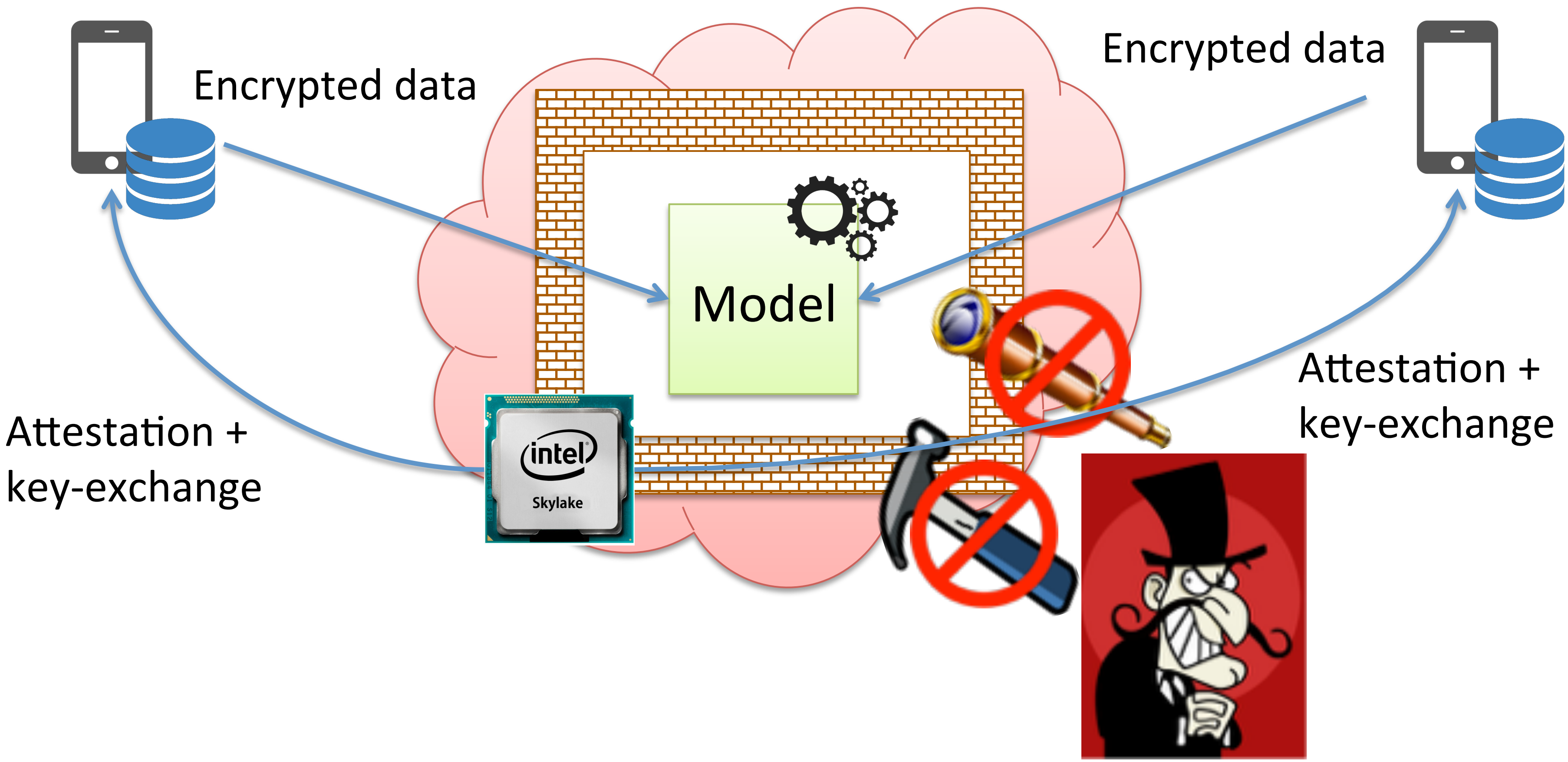
- Garbled circuits (Yao, 1986)
  - For two parties

- MPC (GMW, 1987)

- Homomorphic encryption
  - $\mathrm{Enc}(m_1) + \mathrm{Enc}(m_2) = \mathrm{Enc}(m_1+m_2)$
  - $\mathrm{Enc}(m_1) * \mathrm{Enc}(m_2) = \mathrm{Enc}(m_1*m_2)$

Gentry, 2009

Training on Trusted Hardware



- Schuster et al., “VC3: Trustworthy data analytics in the cloud using SGX”
- Ohrimenko et al., “Oblivious multi-party machine learning on trusted processors”
- Hunt et al., “Chiron: Privacy-preserving Machine Learning as a Service”

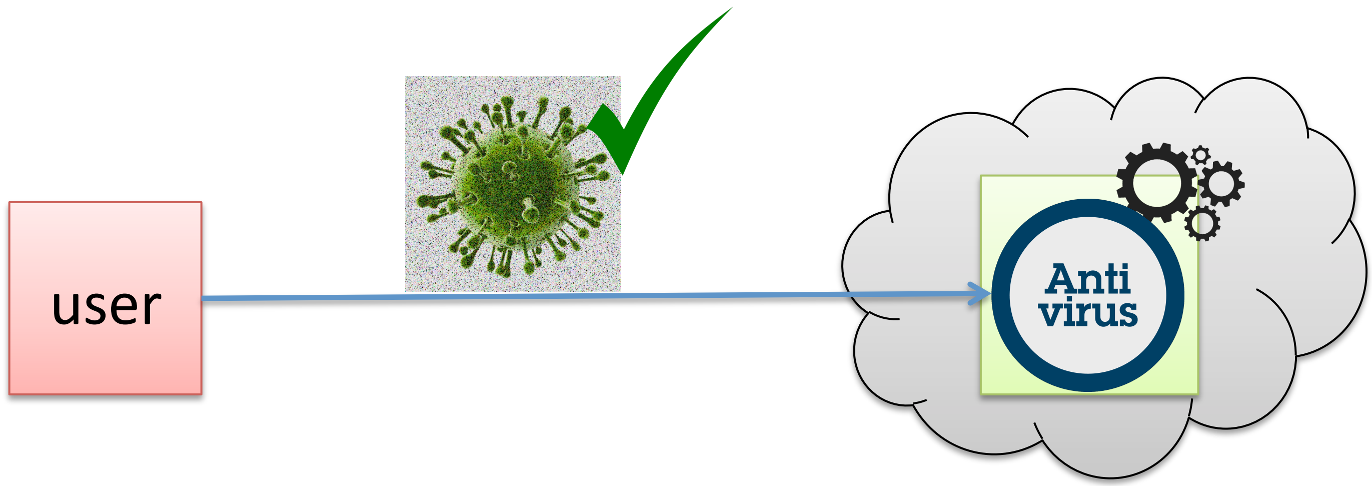
Outline

- Taxonomy of threats and attack vectors

- Attacks/defenses at training time
  - Data poisoning
  - Private & verifiable learning

- Attacks/defenses at evaluation time
  - (Adversarial examples)
  - Inference attacks
  - Private & verifiable inference

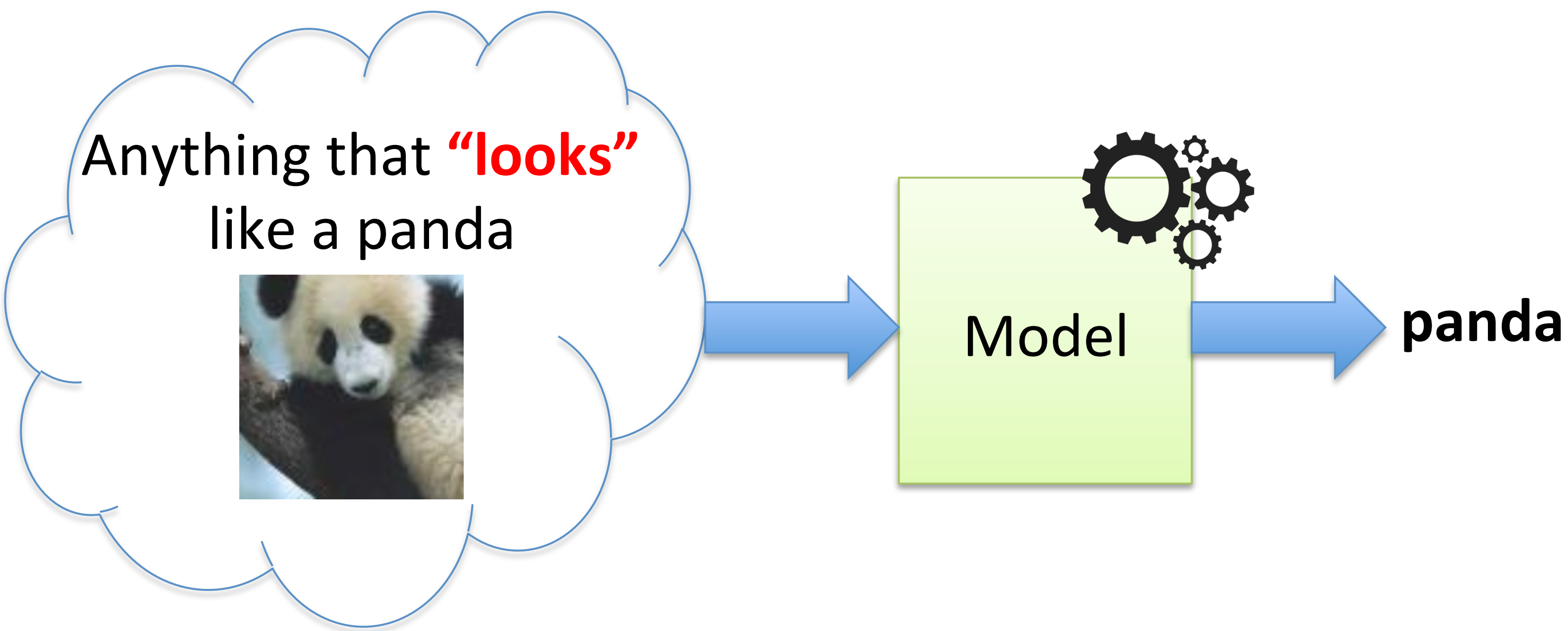
Adversarial Examples



- "Good" uses of adversarial examples?
  - "Hardness" assumption for ML models
  - Better CAPTCHAs?
  - Privacy? (evade automated tagging, censorship, …)

Adversarial Examples

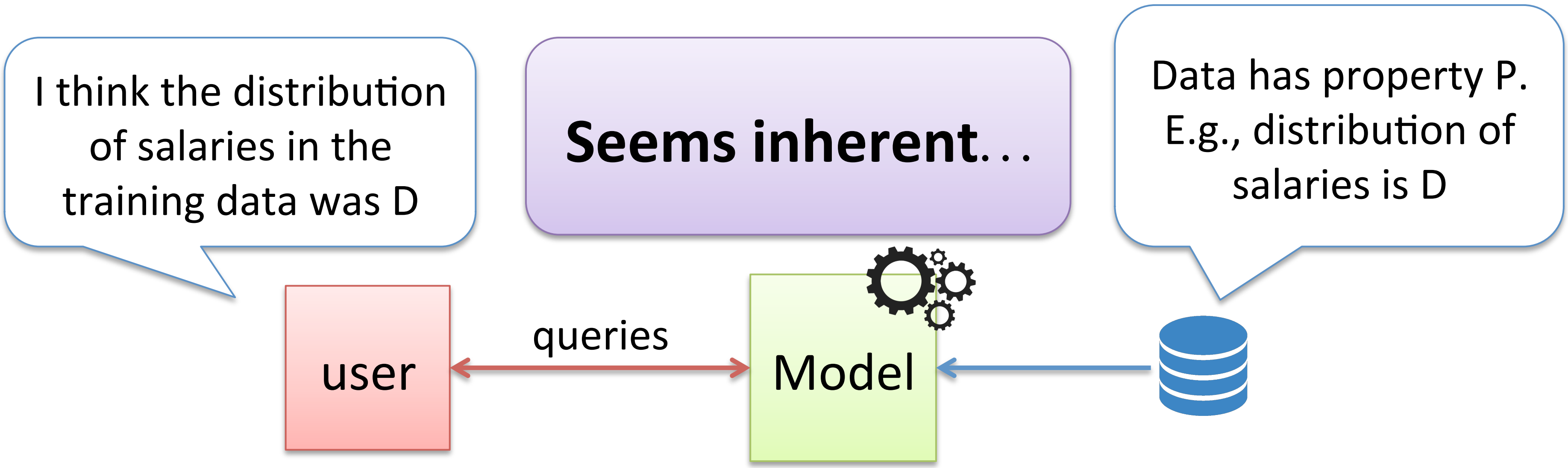
- Is this problem really solvable (“easily”)?



- Large step towards a “Visual Turing Test”…

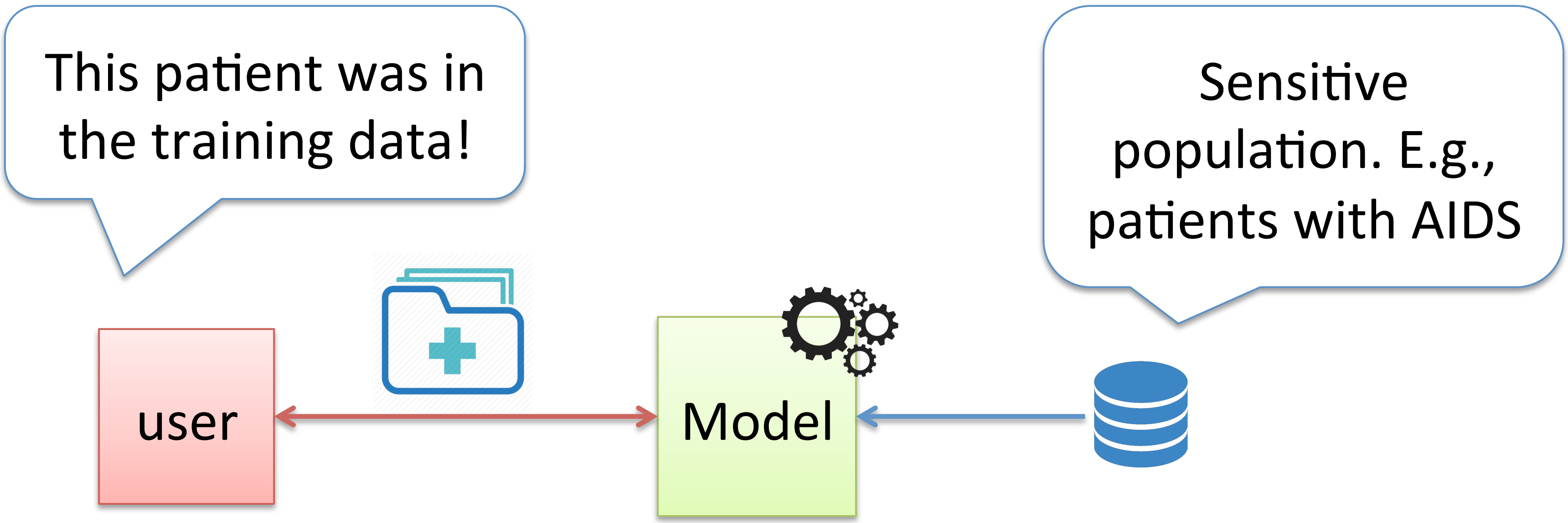
Inference Attacks

- Learn info about training data, the model, etc
- Model inversion:



- Fredrikson et al., "Privacy in Pharmacogenetics: An End-to-End Case Study of Personalized Warfarin Dosing."
- Fredrikson et al., "Model inversion attacks that exploit confidence information and basic countermeasures"
- Ateniese et al., "Hacking Smart Machines with Smarter Ones"

Membership Inference

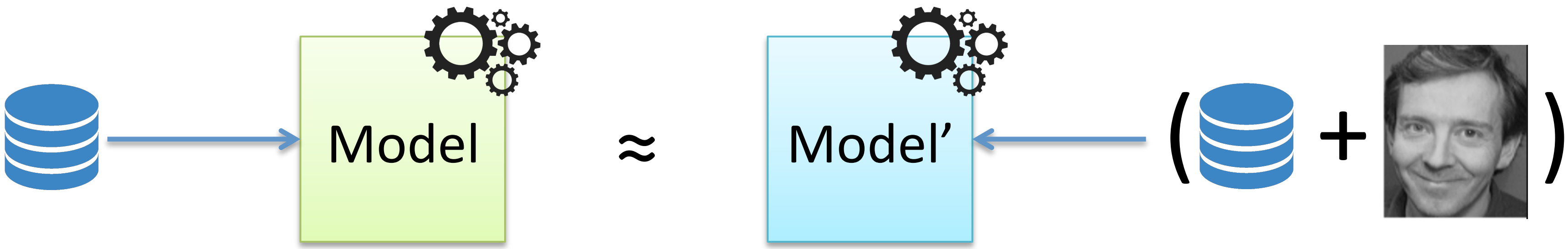


Closely related to overfitting

Model's behavior on D_{train} is different that on D_{test}

- Homer et al., "Resolving Individuals Contributing Trace Amounts of DNA to Highly Complex Mixtures Using High-Density SNP Genotyping Microarrays"
- Shokri et al., "Membership Inference Attacks against Machine Learning Models"

Differential Privacy



- Close connections to stability & generalization
  - A DP mechanism "cannot overfit"
  - **We can hope to achieve utility & privacy!**

- Dwork et al., "Calibrating noise to sensitivity in private data analysis"
- Chaudhuri et al., "Differentially private empirical risk minimization"
- Shokri & Shmatikov, "Privacy-preserving deep learning"
- Abadi et al., "Deep learning with differential privacy"
- Papernot et al., "Semi-supervised Knowledge Transfer for Deep Learning from Private Training Data"

Differentially Private ML

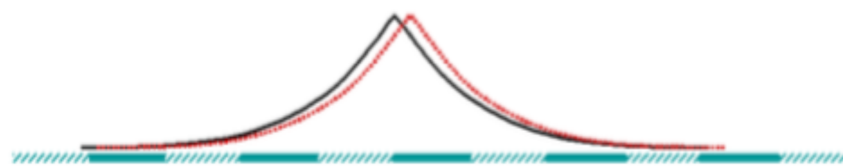
- Sensitivity of a function:

max || f() − f( + ) ||

- Add random noise proportional to sensitivity

f() + r

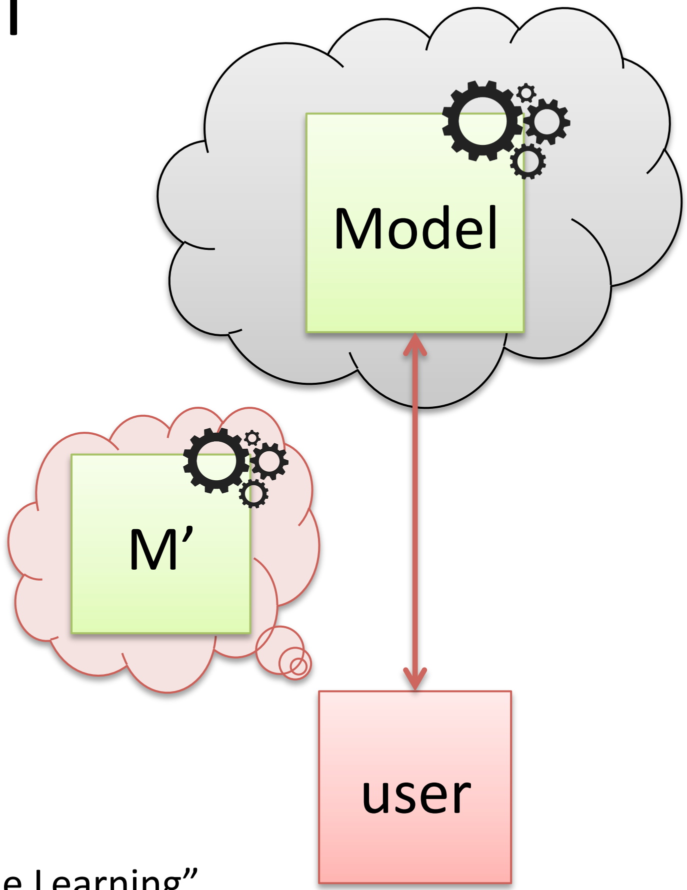
f( + ) + r’



- Do this for every gradient update

Extract Model Properties

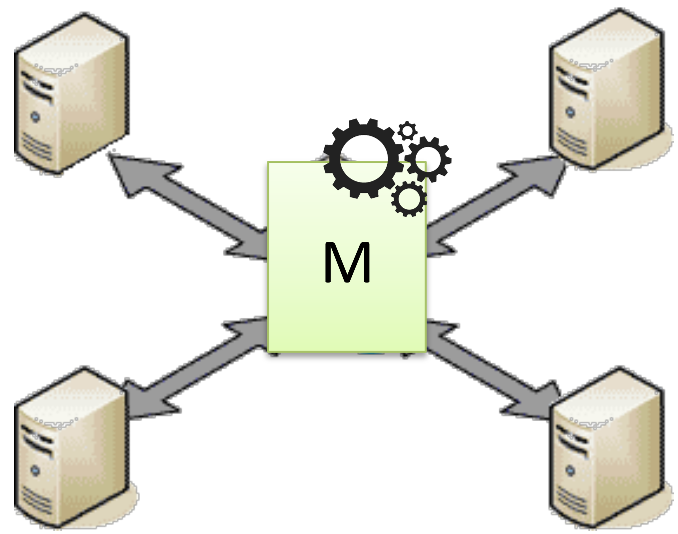
- Interact with black-box model
  - Infer model architecture
  - Hyper-parameters
  - Replicate model ("distillation")
- Step towards other attacks
  - Adversarial examples
  - Model inversion



- Papernot et al., "Practical Black-Box Attacks against Machine Learning"
- T et al., "Stealing Machine Learning Models via Prediction APIs"
- Wang & Gong, "Stealing Hyperparameters in Machine Learning"

Private & Verifiable Inference

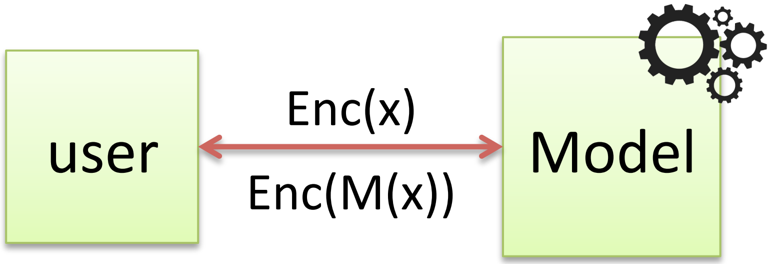
- Assume model can’t be shipped to users
  - E.g., intellectual property
  - Or for performance reasons

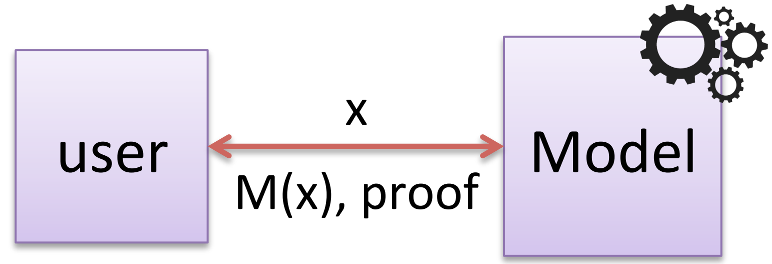


- Model provider learns all the users’ queries…

- Issues:
  - Privacy (obviously)
  - Integrity: targeted mistakes, disparate treatment

Cryptographic Evaluation of ML Models

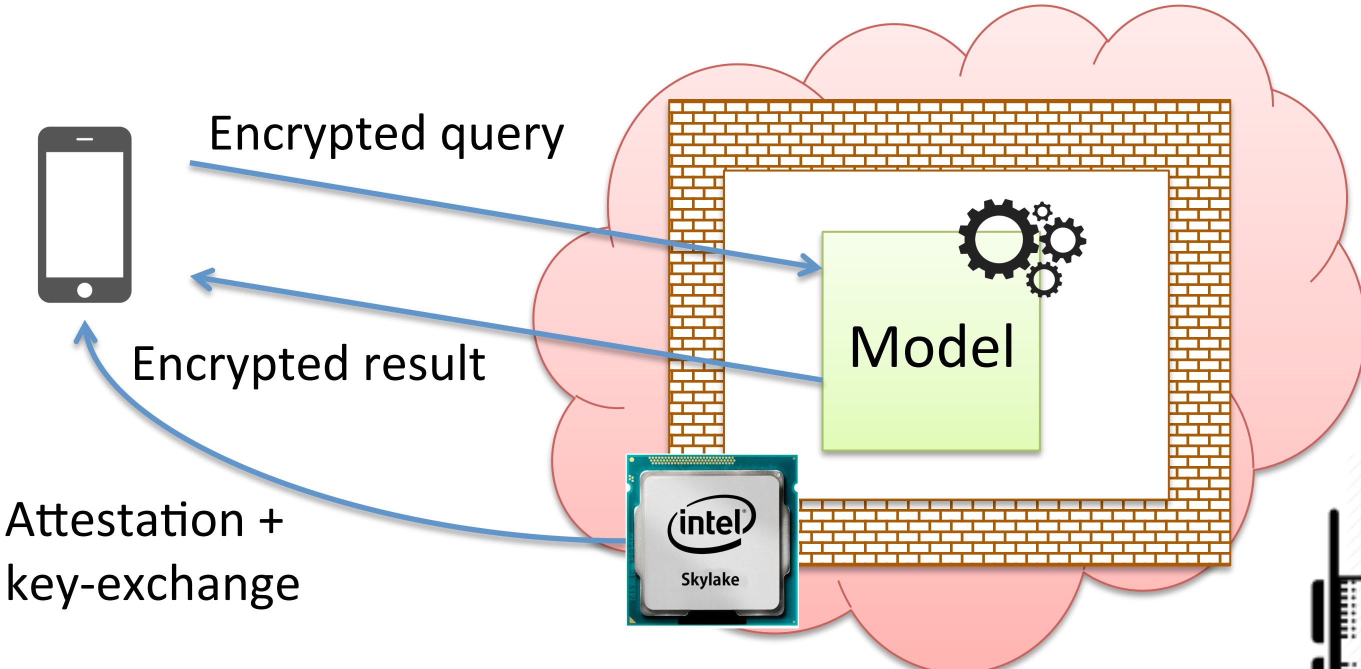


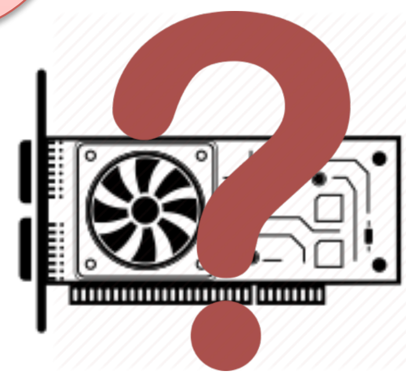


- Many cryptographic techniques:
  - Homomorhpic encryption (slow)
  - 2PC (slowish, high communication)
  - Secret sharing (trust, high communication)
  - Zero-Knowledge Proofs (integrity only, slow)

- Corrigan-Gibbs & Boneh, “Prio: Private, Robust, and Scalable Computation of Aggregate Statistics”
- Downlin et al., “CryptoNets: Applying Neural Networks to Encrypted Data with High Throughput and Accuracy”
- SafetyNets: Verifiable Execution of Deep Neural Networks on an Untrusted Cloud

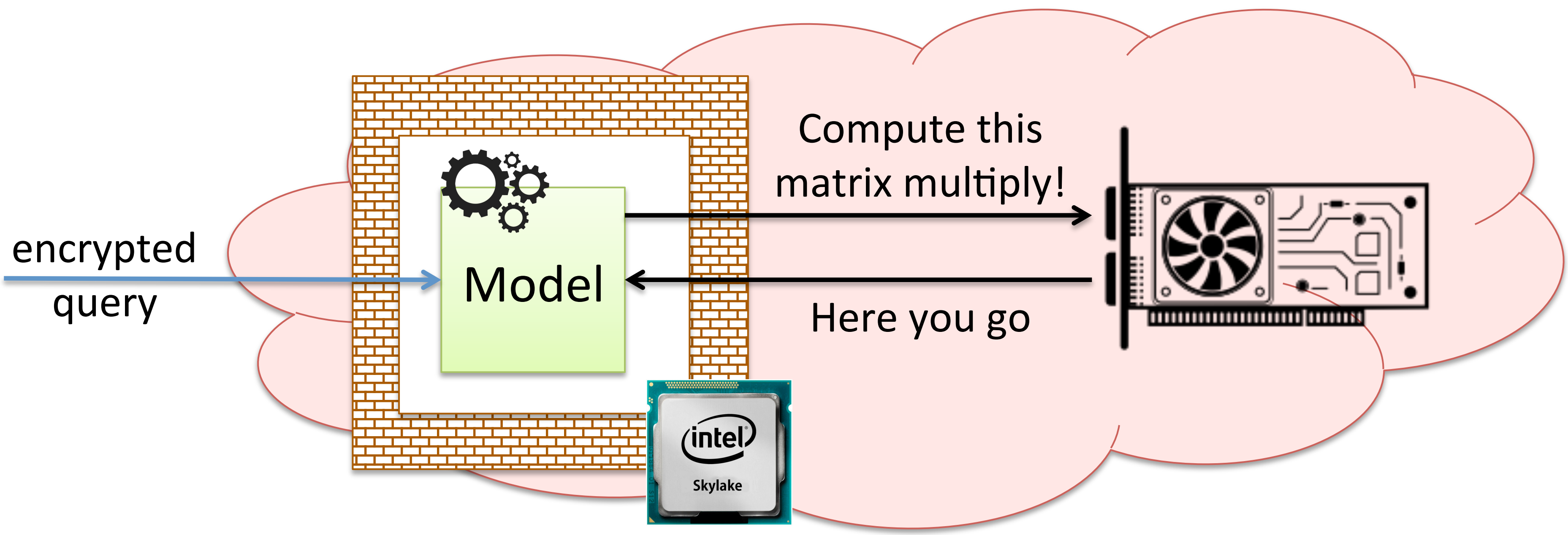
Evaluating Models on Trusted Hardware





- Schuster et al., "VC3: Trustworthy data analytics in the cloud using SGX"
- Ohrimenko et al., "Oblivious multi-party machine learning on trusted processors"
- Hunt et al., "Chiron: Privacy-preserving Machine Learning as a Service"

SLALOM: Fast Inference on Trusted Hardware



- **Speed:** Matrix multiply is >90% of the computation in a DNN
- **Integrity:** Fast verification algorithm for A*B=C (Freivald)
- **Privacy:** $W*\underbrace{(X+R)}_{\text{Enc(X) “one time pad”}} = W*X + \underbrace{W*R}_{\text{pre-computed offline}}$

Summary

- Collaborative training / inference
  => many attacks on privacy and integrity

- Defending against these attacks is hard!
  - Robust statistics
    - Data poisoning, adversarial examples
  - Cryptography & trusted hardware
    - Private + verifiable computations
  - Differential privacy
    - Membership inference

THANKS