Increasing the Cost of Model Extraction with Calibrated Proof of Work

Adam Dziedzic, Muhammad Ahmad Kaleem, Yu Shen Lu, Nicolas Papernot

International Conference on Learning Representations (ICLR)
SPOTLIGHT TALK
March 15th, 2022
Annotate Data Using Machine Learning APIs
Train Models for Machine Learning Services

- Collect & Label Data
- Tune Hyper-parameters
- Run on GPU/TPU/CPU

Machine Learning API

Query → Answer
Train Models for Machine Learning Services

- Collect & Label Data
- Tune Hyper-parameters
- Run on GPU/TPU/CPU

$12 M GPT-3

Machine Learning API

Query → Answer

- Audio
- Surveillance
- Document
- People
Stealing Machine Learning Models

- Collect & Label Data
- Tune Hyper-parameters
- Run on GPU/TPU/CPU

$12 M GPT-3

Machine Learning API

Query

Answer
Defenses against Model Stealing

Poison Attacker’s Objective

Prediction Poisoning [Orekondy et al. 2020]
Defenses against Model Stealing

**Active**

\[ \theta \quad u = -\nabla_w L(\cdot, y) \]

\[ \alpha = -\nabla_w L(\cdot, \tilde{y}) \]

**Passive**

Detect Attack & Stop Responding

Poison Attacker’s Objective

Prediction Poisoning [Orekondy et al. 2020]

PRADA [Juuti et al. 2019]
Defenses against Model Stealing

**Active**

Poison Attacker’s Objective
Prediction Poisoning [Orekondy et al. 2020]

\[\theta \quad u = -\nabla_w L(\cdot, y)\]

\[a = -\nabla_w L(\cdot, \hat{y})\]

Detect Attack & Stop Responding
PRADA [Juuti et al. 2019]

**Passive**

Resolve Model Ownership
Dataset Inference [Maini et al. 2021]

**Reactive**

Train

Test
Defenses against Model Stealing

**Active**

Poison Attacker’s Objective
Prediction Poisoning [Orekondy et al. 2020]

Detect Attack & Stop Responding
PRADA [Juuti et al. 2019]

**Passive**

Resolve Model Ownership
Dataset Inference [Maini et al. 2021]

**Reactive**

Pro-Active
Proof-of-Work
Differential Privacy
Calibrated Proof-of-Work with PATE

\[ u = -\nabla_w L(\cdot, y) \]
\[ \alpha = -\nabla_w L(\cdot, \hat{y}) \]
How to Defend Against Model Stealing?

Unlabeled Data

Attacker

Send Queries

Defender

Return Labels / Scores

Stolen Model

Victim Model

[Shankar et al. 2020]
Model Stealing - ranked among the most severe attacks against ML
Unlabeled Data

Attacker

Defender

Send Queries

Estimate Privacy Leakage

Victim Model
Generate Calibrated Proof-of-Work Puzzle

Attacker
Unlabeled Data

Defender
Send Queries

Generate Puzzle

Estimate Privacy Leakage

Victim Model
Increase the Cost of Model Stealing

Unlabeled Data

Attacker

Send Queries

Defender

Generate Puzzle

Verify Solution

Estimate Privacy Leakage

Victim Model
Client Receives Labels after Solving a Puzzle

Attacker

Unlabeled Data

Stolen Model

Generate Puzzle

Verify Solution

Send Queries

Defender

Estimate Privacy Leakage

Victim Model

Return Labels

Solve Puzzle
Server:
Send challenge S to client

Client:
Find a suffix X such that

\[
\text{hash}(S.append(X)) = 0 \ldots 0xxxxx
\]

required # of zeros
Server: Send challenge $S$ to client

Client: Find a suffix $X$ such that

$$\text{hash}(S.append(X)) = 0\ldots0xxxxx$$

required # of zeros
1. Set privacy cost for **Legitimate Users** as a reference cost.
Higher Privacy Cost for Standard Attacks

1. Set privacy cost for **Legitimate Users** as a reference cost.
2. Measure the privacy cost of queries.
Set Puzzle Difficulty using Privacy Deviation

1. Set privacy cost for **Legitimate Users** as a reference cost.
2. Measure the privacy cost of queries.
3. Calibrate puzzle difficulty using privacy deviation.
Query Time vs Accuracy of Stolen Copy

CIFAR10

Accuracy (%) vs Time (sec)

max 2X increase

- Legitimate
- Legitimate + Defense
Query Time vs Accuracy of Stolen Copy

CIFAR10

100X increase

Accuracy (%) vs Time (sec)

- Legitimate
- Legitimate + Defense
- CopyCat Attack
- CopyCat + Defense
Conclusions

- **New defense against model stealing** – increase the computational cost instead of lowering the quality of model outputs.
- **Privacy cost** is used to measure information leakage from a victim model that was incurred by queries from each user.
- **Calibrate** the cost of users’ queries using the privacy cost.
- Use proof-of-\{work, elapsed time, stake\}, or payment for queries.
  - Reference method: require a user to solve the proof-of-work puzzle before releasing predictions.
- **Performance**:
  - Negligible overhead for legitimate users (~2X);
  - High increase in the querying time for many attackers (even ~100X).
Thank you

🌐 https://cleverhans-lab.github.io
✉️ {adam.dziedzic, nicolas.papernot}@utoronto.ca