Aalto University

UNIVERSITY OF WATERLOO

Extraction of Complex DNN Models: Real Threat or Boogeyman?

N. Asokan
https://asokan.org/asokan/
@nasokan

(Joint work with Buse Gul Atli, Sebastian Szyller, Mika Juuti and Samuel Marchal)

Outline

Is model confidentiality important?

Can models be extracted via their prediction APIs?

What can be done to counter model theft?

Can we simultaneously deploy protections against multiple concerns?

Outline

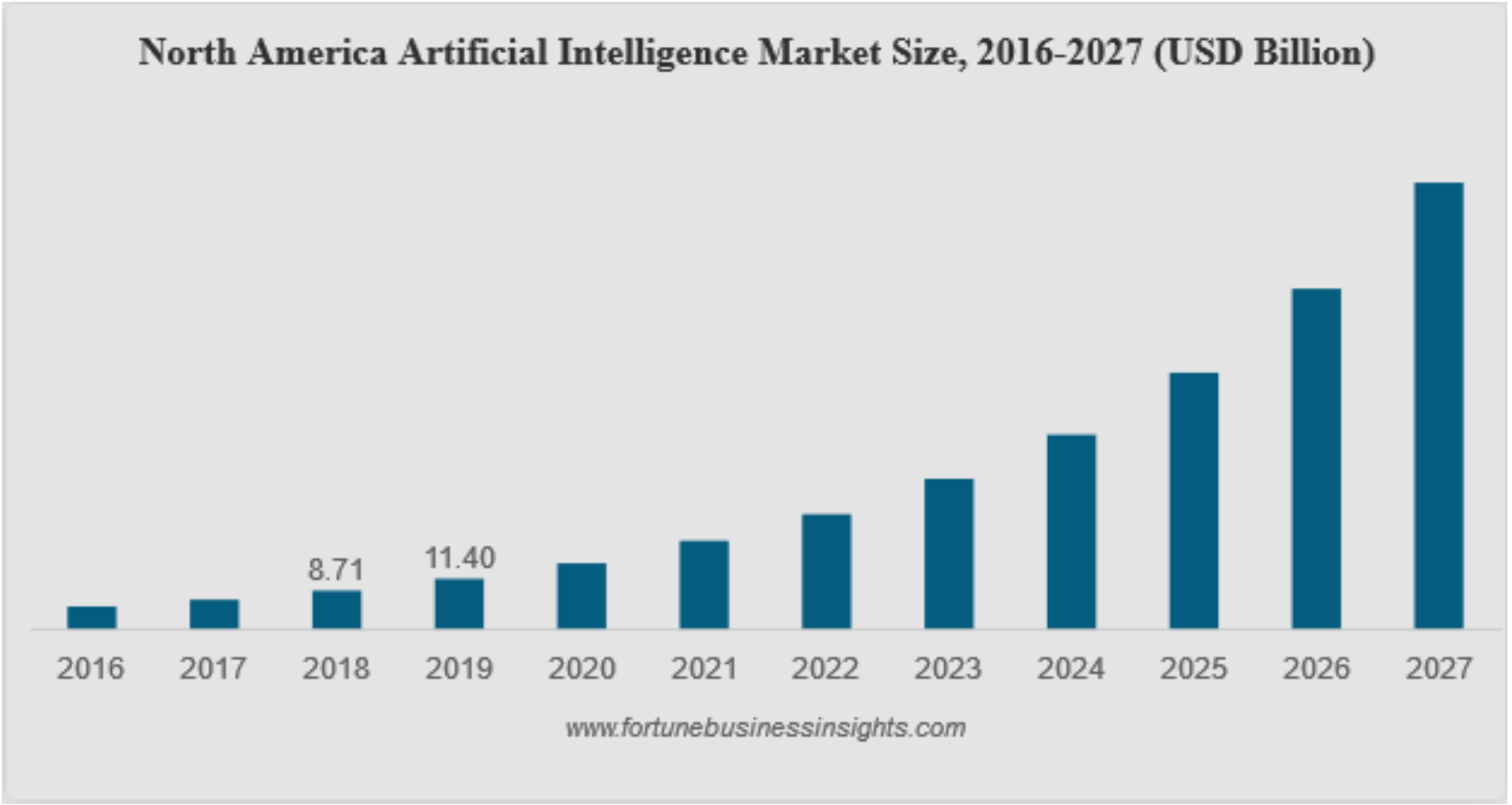
Is model confidentiality important?

Can models be extracted via their prediction APIs?

What can be done to counter model theft?

Can we simultaneously deploy protections against multiple concerns?

AI will be pervasive



https://www.fortunebusinessinsights.com/industry-reports/artificial-intelligence-market-100114

Is model confidentiality important?

Machine learning models: business advantage and intellectual property (IP)

Cost of

- gathering relevant data
- labeling data
- expertise required to choose the right model training method
- resources expended in training

Adversary who steals the model can avoid these costs

Type of model access: white box

White-box access: user

- has physical access to model
- knows its structure
- can observe execution (scientific packages, software on user-owned devices)

How to prevent (white-box) model theft?

White-box model theft can be countered by

- **Computation with encrypted models**
- **Protecting models using secure hardware**
- **Hosting models behind a firewalled cloud service**

Type of model access: black-box

Black-box access: user

- does not have physical access to model
- interacts via a well-defined interface (“prediction API”):
  - directly (translation, image classification)
  - indirectly (recommender systems)

Basic idea: hide model, expose model functionality only via a prediction API

Is that enough to prevent model theft?

Outline

Is model confidentiality important?

Can models be extracted via their prediction APIs?

What can be done to counter model theft?

Can we simultaneously deploy protections against multiple concerns?

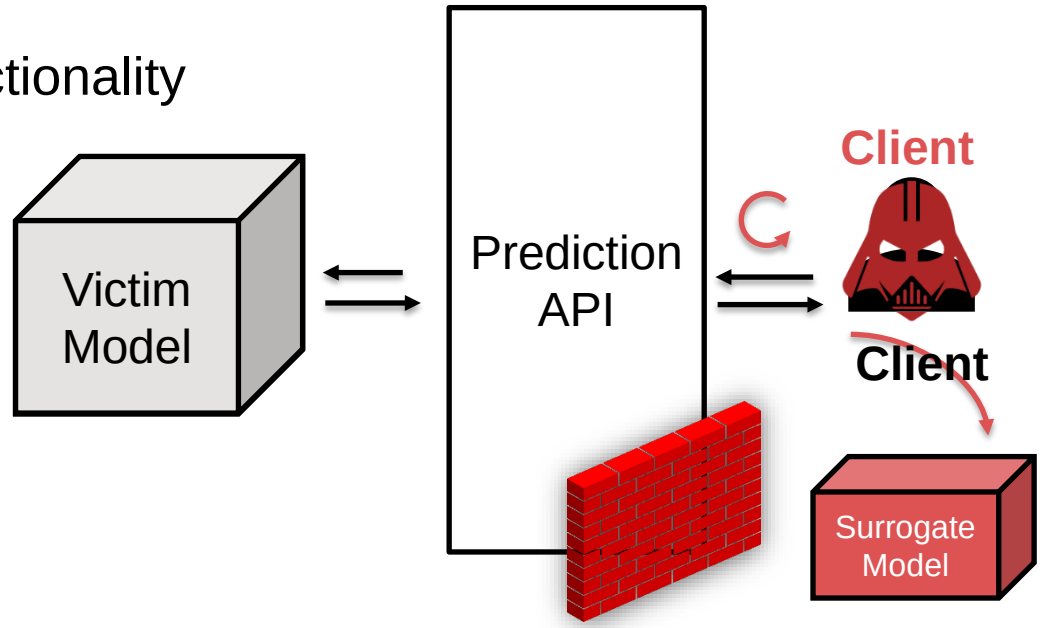
Extracting models via their prediction APIs

Prediction APIs are oracles that leak information

Adversary

- Malicious client
- Goal: construct surrogate model(*) comparable w/ functionality
- Capability: access to prediction API or model outputs

(*) aka “student model” or “imitation model”



Prior work on extracting

- Logistic regression, decision trees[1]
- Simple CNN models[2]
- Querying API with synthetic samples

[1] Tramèr et al. - *Stealing Machine Learning Models via Prediction APIs*, USENIX SEC ‘16 (https://arxiv.org/abs/1609.02943)
[2] Papernot et al. - *Practical Black-Box Attacks against Machine Learning*, ASIACCS ‘17 (https://arxiv.org/abs/1602.02697)

Extracting deep neural networks

Against simple DNN models[1]

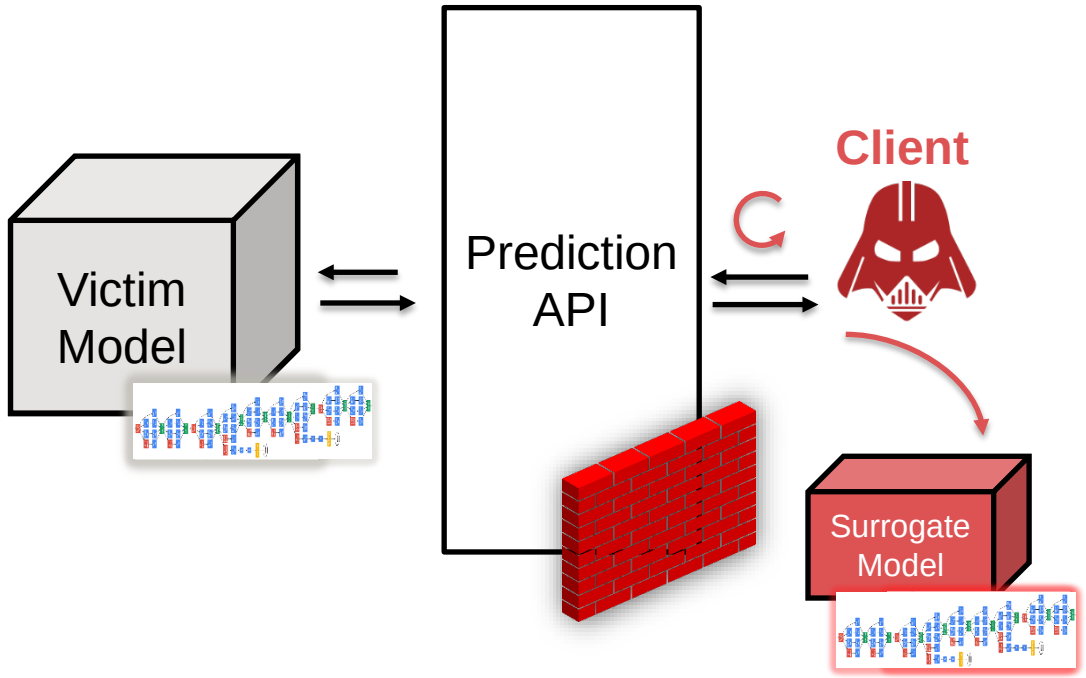
- E.g., MNIST, GTSRB

Adversary

- knows general structure of the model
- has limited natural data from victim's domain

Approach

- Hyperparameters CV-search
- Query using natural data for rough estimate decision boundaries, synthetic data to fine-tune
- Simple defense: distinguish between benign and adversarial queries



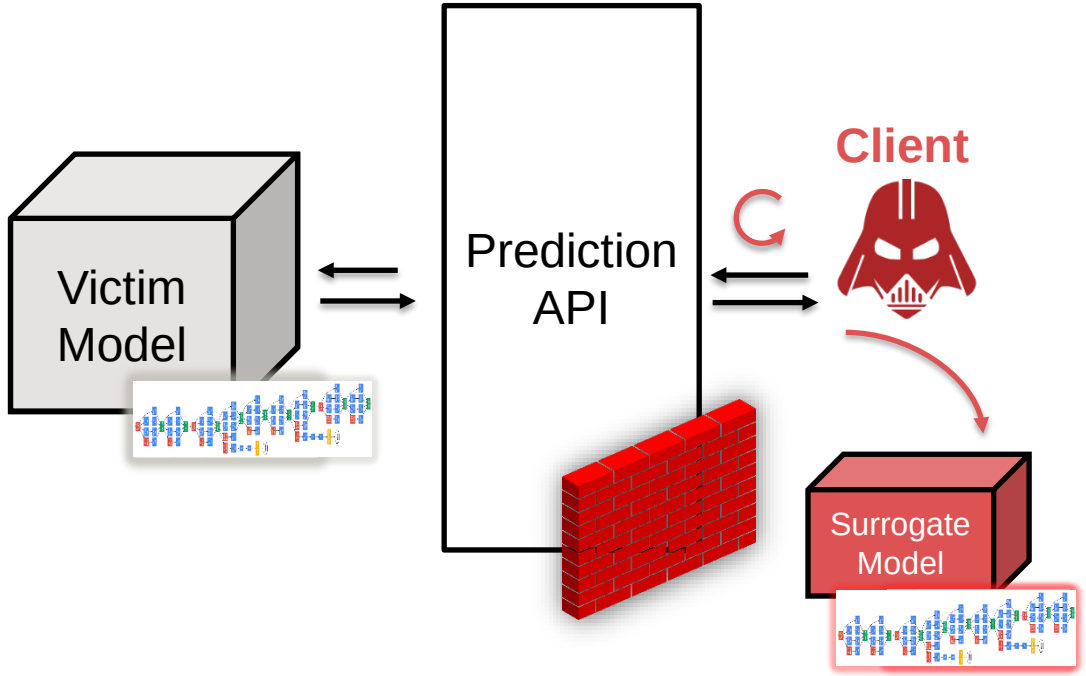
[1] Juuti et al. - *PRADA: Protecting against DNN Model Stealing Attacks*, EuroS&P '19 (https://arxiv.org/abs/1805.02628)

Is model extraction a realistic threat?

Can adversaries extract complex DNNs successfully?

Are common adversary models realistic?

Are current defenses effective?



Extraction of Complex DNN Models: Knockoff nets[1]

Goal:

- Build a surrogate model that
  - steals model functionality of victim model
  - performs similarly on the same task with high classification accuracy

Adversary capabilities:

- Victim model knowledge:
  - None of train/test data, model internals, output semantics
  - Access to full prediction probability vector
- Access to natural samples, not (necessarily) from the same distribution as train/test data
- Access to pre-trained high-capacity model

[1] Orekondy et al. - *Knockoff Nets: Stealing Functionality of Black-Box Models,* CVPR '19 (https://arxiv.org/abs/1812.02766)

Analysis of Knockoff Nets: summary[2]

Reproduced empirical evaluation of Knockoff nets[1] to confirm its effectiveness

Revisited its adversary model in to make more realistic assumptions about the adversary

Attack effectiveness decreases if

- **Surrogate and victim model architectures are different**
- **Victim model's prediction API has reduced granularity**

Simple defense: detector to identify out-of-distribution queries

Defense effectiveness decreases if attacker has natural samples distributed like victim's training data

[1] Orekondy et al. - *Knockoff Nets: Stealing Functionality of Black-Box Models,* CVPR '19 (https://arxiv.org/abs/1812.02766)
[2] Atli et al. - *Extraction of Complex DNN Models: Real Threat or Boogeyman?,* AAAI-EDSML '20 (https://arxiv.org/abs/1910.05429)

Extracting NLP Transformer models

Techniques for extracting image classifiers don't always extend to NLP models

Transfer learning from pre-trained models is now very popular

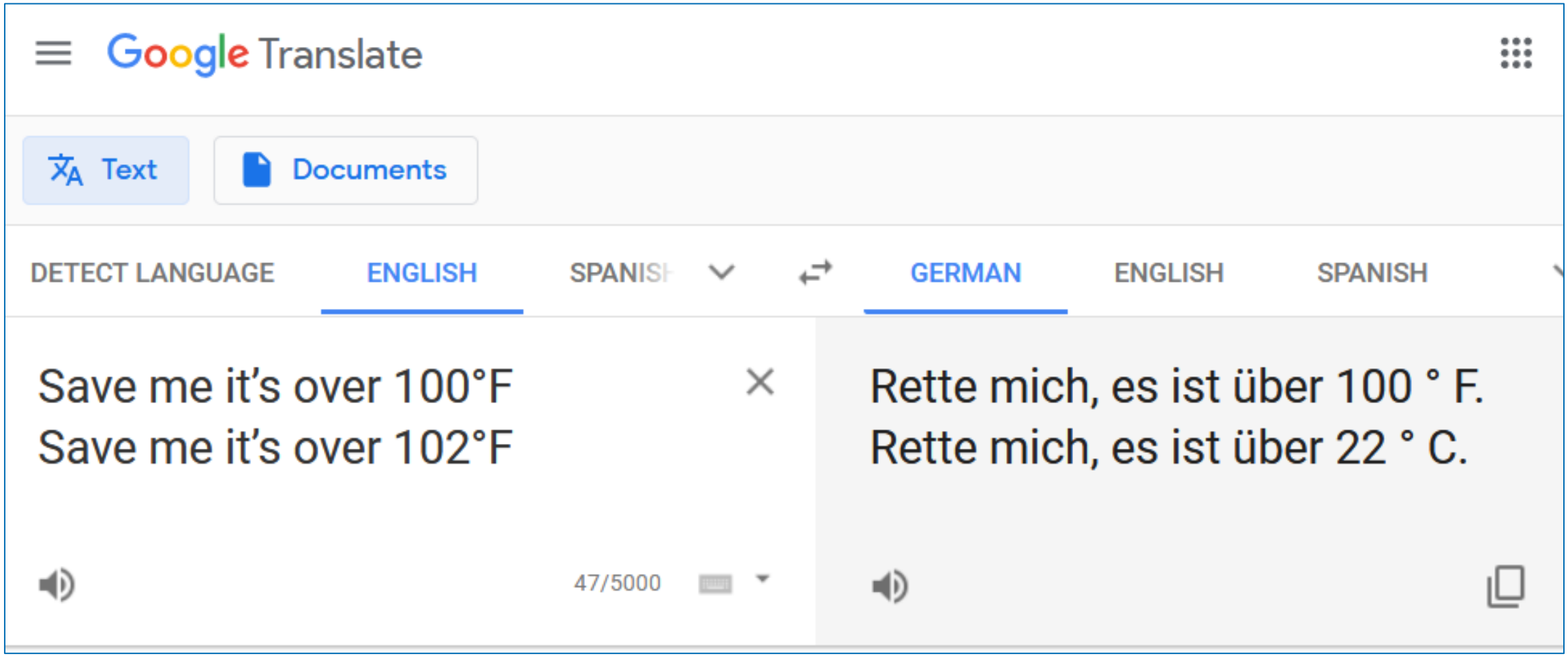
- But they make model extraction easier[1]

Krishna et al[1] show that a Knockoff-like attacks against BERT models are feasible

- Adversary unaware of target distribution or task of victim model
- Adversary queries are merely "natural" (randomly sampled sequences of words)
- In-distribution adversary queries can improve extraction efficacy

Wallace et al[2] extract real-world MT models, find transferable adversarial examples

[1] Krishna et al. – *Thieves on Sesame Street! Model Extraction of BERT-based APIs* , ICLR '20 (https://iclr.cc/virtual_2020/poster_Byl5NREFDr.html)
[2] Wallace et al. – *Imitation Attacks and Defenses for Black-box Machine Translation Systems,* EMNLP '20 (https://arxiv.org/abs/2004.15015)



https://translate.google.com/#view=home&op=translate&sl=en&tl=de&text=Save%20me%20it%E2%80%99s%20over%20100%C2%B0F%0ASave%20me%20it%E2%80%99s%20over%20102%C2%B0F

Wallace et al. – *Imitation Attacks and Defenses for Black-box Machine Translation Systems,* EMNLP '20 (https://arxiv.org/abs/2004.15015)

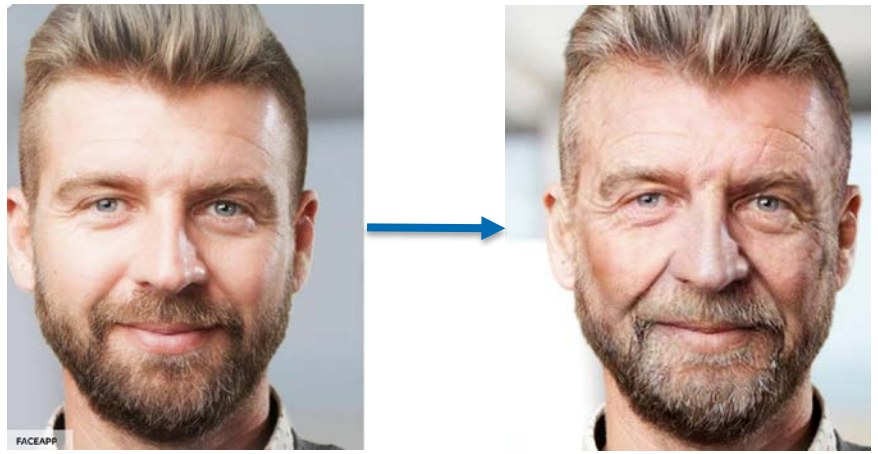
Extracting Style-transfer models

GANS are effective for changing image style

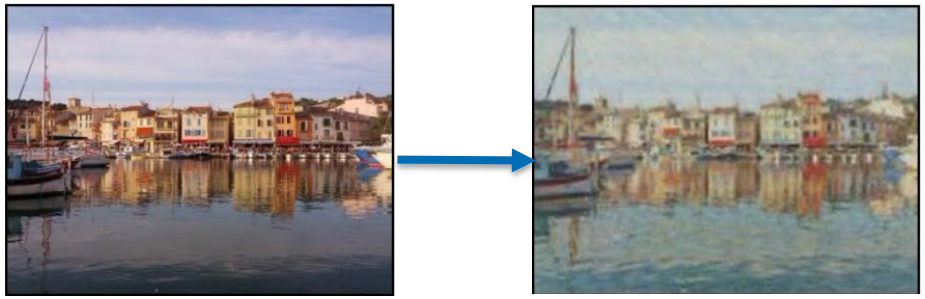
- coloring, face filters, style application

Core feature in generative art and in social media apps

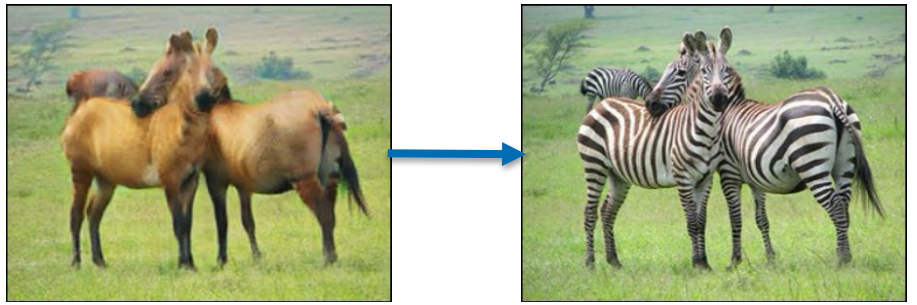
- **Selfie2Anime, FaceApp**



FaceApp

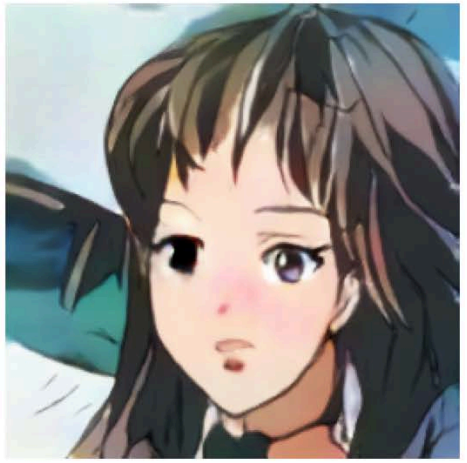


CycleGANs



CycleGANs

Style transfer

	Original (unstyled)	Styled (victim)	**Styled (ours)**
Task 1 *Monet painting*			
Task 2 *Anime face*			

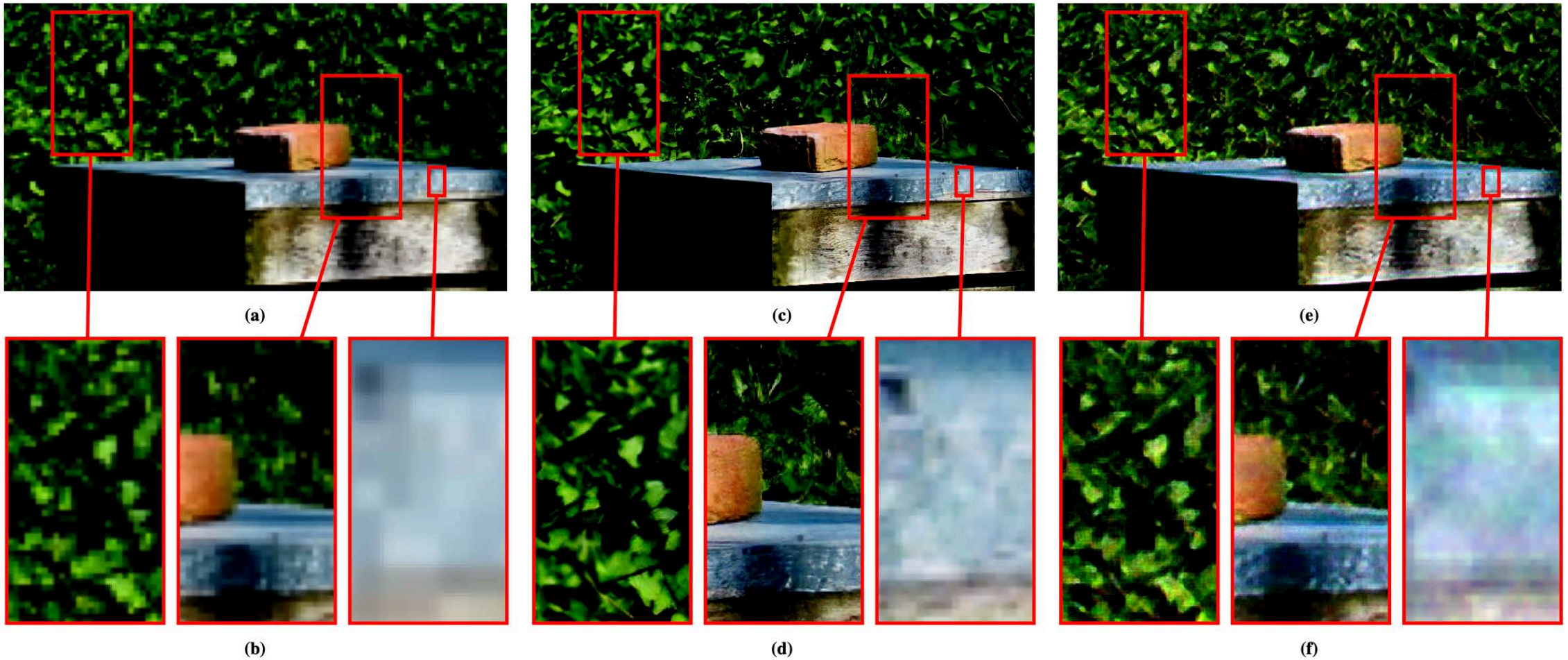
Szyller et al. - *Good Artists Copy, Great Artists Steal: Model Extraction Attacks Against Image Translation Generative Adversarial Networks, https://arxiv.org/abs/2104.12623*

Super resolution

Original (low-res)

High-res (victim)

High-res (ours)



Szyller et al. - *Good Artists Copy, Great Artists Steal: Model Extraction Attacks Against Image Translation Generative Adversarial Networks*, https://arxiv.org/abs/2104.12623

Outline: recap

Is model confidentiality important? Yes

Can models be extracted via their prediction APIs? Yes[1]

- A powerful (but realistic) adversary can extract complex real-life models
- Detecting such an adversary is difficult/impossible

What can be done to counter model theft?

Can we simultaneously deploy protections against multiple concerns?

[1] Atli et al. - *Extraction of Complex DNN Models: Real Threat or Boogeyman?* (https://arxiv.org/pdf/1910.05429.pdf,, AAAI-EDSML '20)

Defending against model theft

We can try to:

- prevent (or slow down[1]) model extraction, or
- detect[2] it

But current solutions are not effective.

Or deter the attacker by providing the means for ownership demonstration:

- model watermarking
- data watermarking
- fingerprinting

[1] Dziedzic et al. - *Increasing the Cost of Model Extraction with Calibrated Proof of Work*, ICLR ’22 (https://openreview.net/pdf?id=EAy7C1cgE1L)
[2] Atli et al. - *Extraction of Complex DNN Models: Real Threat or Boogeyman?,* AAAI-EDSML ‘20 (https://arxiv.org/abs/1910.05429)

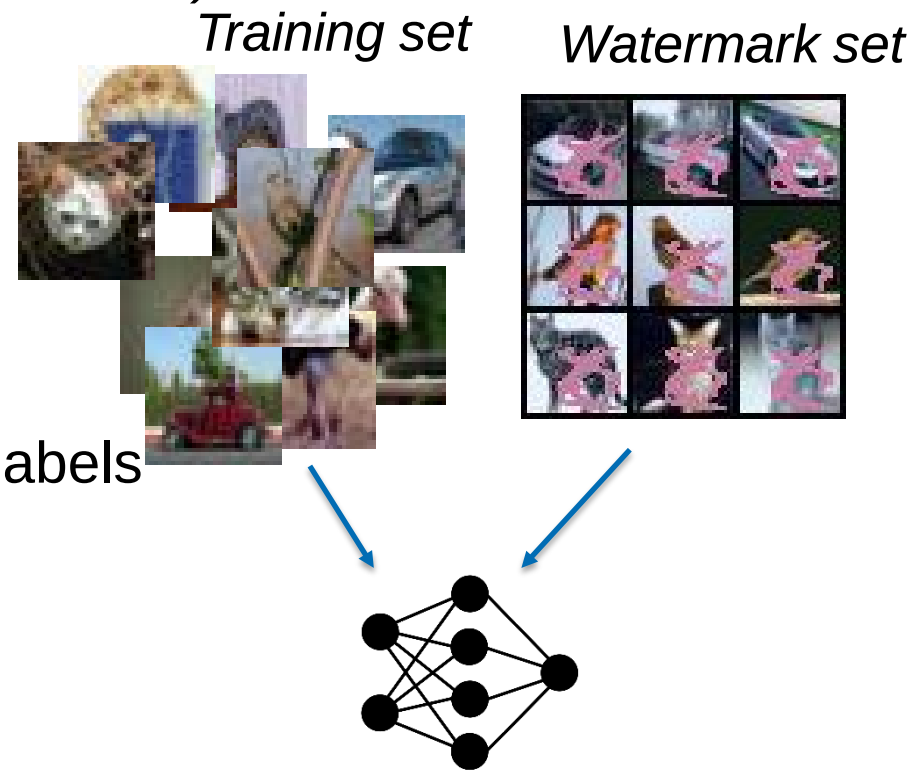
White-box watermarking

Watermark embedding:

- Embed the watermark in the model **during the training phase**:
  - Choose incorrect labels for a set of samples (*watermark set, WM)*
  - Train using training data + *watermark set*

Verification of ownership:

- Adversary publicly exposes the stolen model
- Query the model with the *watermark set*
- Verify watermark - predictions correspond to chosen labels



Yadi et al. *Watermarking Deep Neural Networks by Backdooring,* Usenix SEC '18 https://www.usenix.org/node/217594

Existing watermarking of DNNs

Assumes that the model is stolen exactly (white-box theft)

Protects only against physical theft of model[1]

Not robust against

- novel watermark removal attacks[2]
- model extraction attacks that reduce effect of watermarks & modify decision surface

[1] Szyller et. al. - *DAWN: Dynamic Adversarial Watermarking of Neural Networks.* ACM MM '21 (https://arxiv.org/abs/1906.00830)

[2] Lukas et al. *SoK: How Robust is Image Classification Deep Neural Network Watermarking*? IEEE S&P '22 (https://arxiv.org/abs/2108.04974)

DAWN: Dynamic Adversarial Watermarking of DNNs[1]

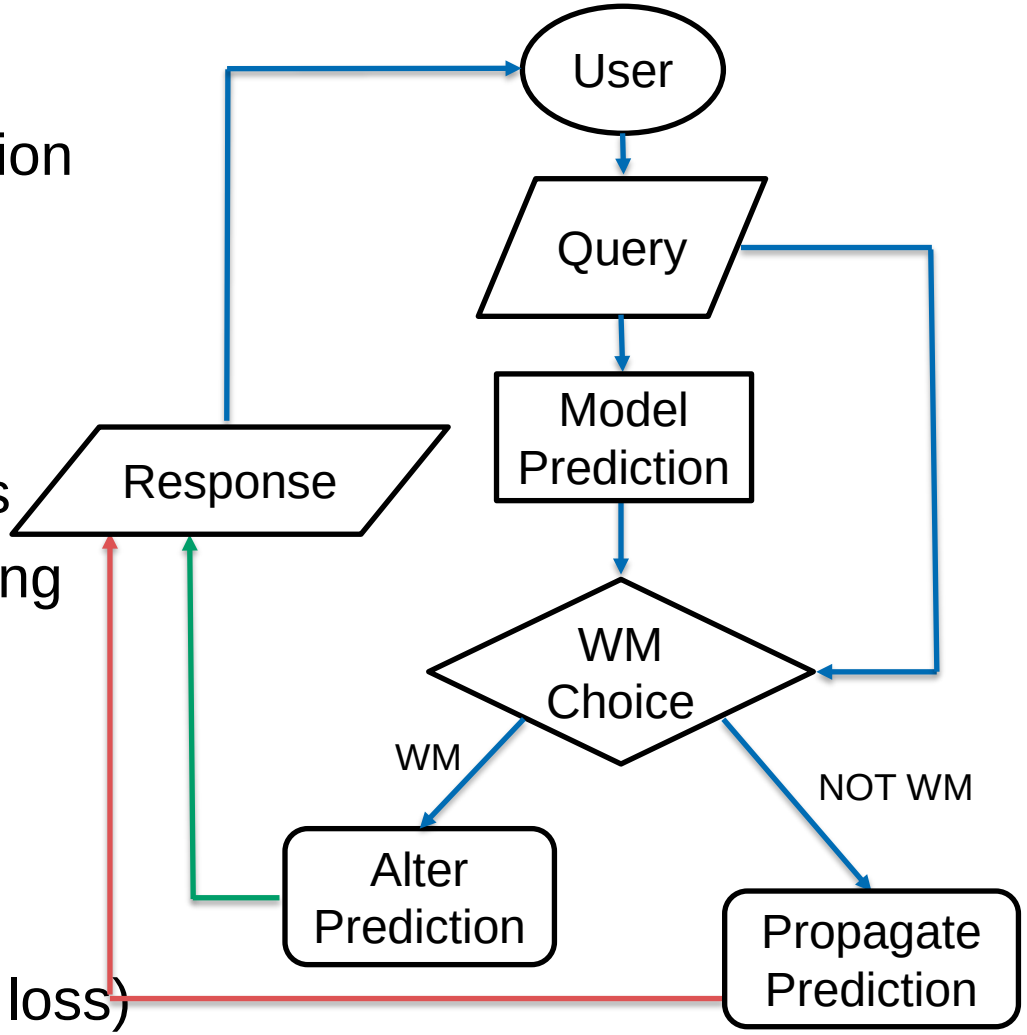
Goal: Watermark models obtained via model extraction

Our approach:

- Implemented as part of the **prediction API**
- Return **incorrect predictions** for several samples
- Adversary forced to embed watermark while training

Watermarking evaluation:

- **Unremovable** and **indistinguishable**
- **Defend against** *PRADA*[2] and *KnockOff* [3]
- Preserve victim *model utility* (*0.03-0.5%* accuracy loss)



[1] Szyller et. al. - *DAWN: Dynamic Adversarial Watermarking of Neural Networks,* ACM MM '21 (https://arxiv.org/abs/1906.00830)
[2] Juuti et al. - *PRADA: Protecting against DNN Model Stealing Attacks,* EuroS&P '19 (https://arxiv.org/abs/1805.02628)
[3] Orekondy et al. - *Knockoff Nets: Stealing Functionality of Black-Box Models,* CVPR '19 (https://arxiv.org/abs/1812.02766)

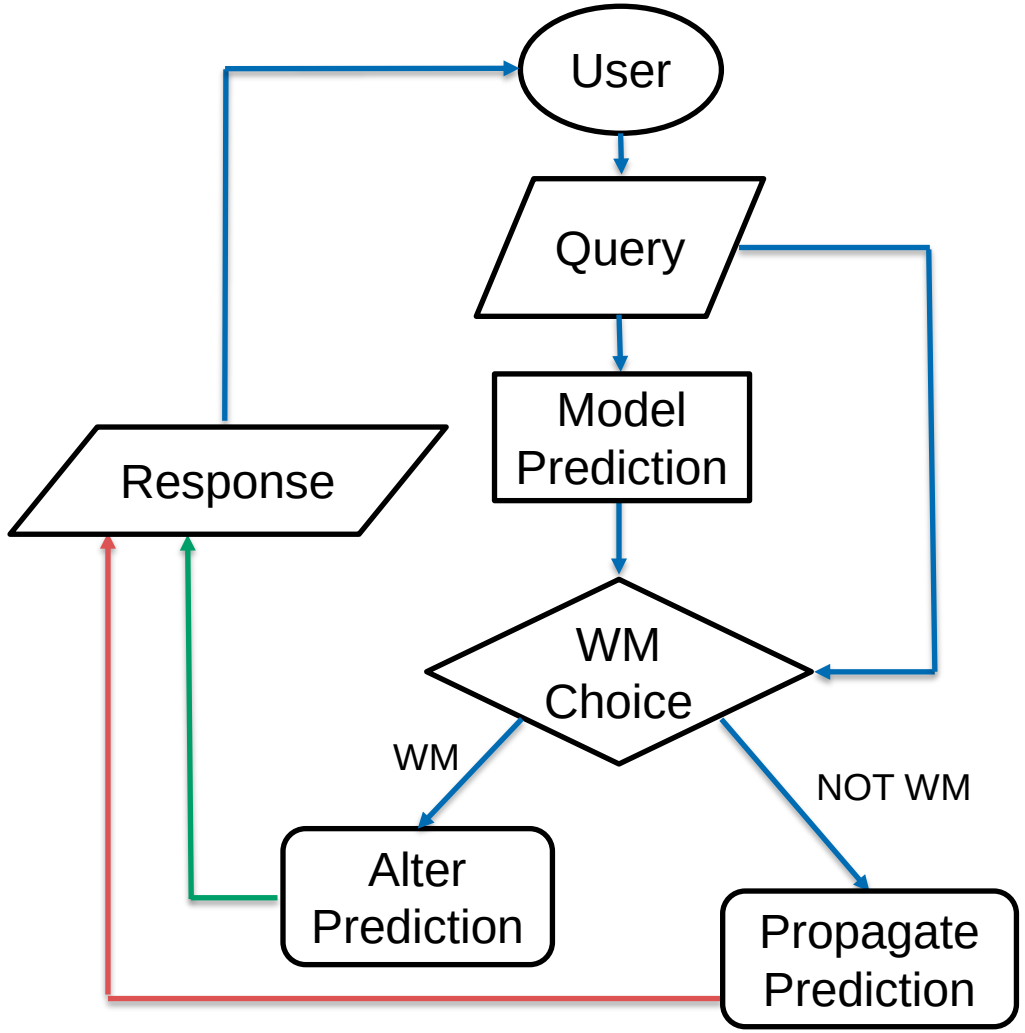
Open issues in DAWN[1]

Indistinguishability

- existence of a robust mapping function (for WM choice)

Unremovability

- “double-stealing” can remove watermark (but impacts accuracy of surrogate model)
- adversary can return incorrect predictions on training data (but can be overcome)



[1] Szyller et. al. - *DAWN: Dynamic Adversarial Watermarking of Neural Networks*, ACM MM ‘21 (https://arxiv.org/abs/1906.00830)

Data/Model fingerprinting

Radioactive data[1]

- Intended for provenance, not robust in adversarial settings[2]

Conferrable adversarial examples[2]

- Computationally expensive

Dataset inference[3]

- Susceptible to False positives? [4]

[1] Sablayrolles et al. *Radioactive data: tracing through training,* ICML'20 (https://arxiv.org/abs/2002.00937)
[2] Atli Tegkul et al. *On the Effectiveness of Dataset Watermarking,* IWSPA@CODASPY '22 (https://arxiv.org/abs/2106.08746)
[2] Lukas et al. *Deep Neural Network Fingerprinting by Conferrable Adversarial Examples,* ICLR '21 (https://openreview.net/forum?id=VqzVhqxkjH1)
[3] Maini, et al. *Dataset Inference Ownership Resolution in Machine Learning*, ICLR '21 (https://openreview.net/pdf?id=hvdKKV2yt7T)
[4] Szyller and Asokan. - *Conflicting Interactions Among Protections Mechanisms for Machine Learning Models*, (https://arxiv.org/abs/2207.01991)

Outline

Is model confidentiality important?

Can models be extracted via their prediction APIs?

What can be done to counter model theft?

Can we simultaneously deploy protections against multiple concerns?

Other ML security & privacy concerns

There are considerations other than model ownership:

- model evasion (defense: adversarial training)
- training data reconstruction (defense: differential privacy)
- membership inference (defense: regularization, early stopping)
- model poisoning (defense: regularization, outlier/anomaly detection)
- …

How does ownership demonstration interact with the other defenses?

We investigate pairwise interactions of:

model watermarking

data watermarking

fingerprinting

WITH

differential privacy

adversarial training

Setup & Baselines

We use the following techniques (and corresponding metrics):

- WM: Out-of-distribution (OOD) backdoor watermarking (test and watermark accuracy)
- RAD-DATA: Radioactive data (test accuracy and loss difference)
- DI: Dataset Inference (verification confidence)
- DP: DP-SGD (model accuracy for the given epsilon)
- ADV-TR: Adversarial training with PGD (test and adv. accuracy for the given epsilon)

Dataset	No defense	Watermarking		Radioactive Data		Dataset Inference	DP-SGD (eps=3)	ADV. TR.	
	ϕ_{ACC}	ϕ_{ACC}	ϕ_{WM}	ϕ_{ACC}	$\phi_{RAD\text{-}DATA}$ Loss. Diff.	ϕ_{DI} Confidence	ϕ_{ACC}	ϕ_{ACC}	ϕ_{ADV}
MNIST	*0.99±0.00*	*0.99±0.00*	*0.97±0.01*	*0.98±0.00*	*0.284±0.001*	*<e-30*	*0.98±0.00*	*0.99±0.00*	*0.95±0.00*
FMNIST	*0.91±0.00*	*0.87±0.02*	*0.99±0.02*	*0.88±0.01*	*0.19+0.002*	*<e-30*	*0.86±0.01*	*0.87±0.00*	*0.69±0.00*
CIFAR10	*0.92±0.00*	*0.82±0.00*	*0.97±0.02*	*0.85+0.00*	*0.20±0.001*	*<e-30*	*0.38±0.00*	*0.82±0.00*	*0.82±0.00*

Summary of conflicts

If two techniques A and B in combination result in too high a drop in

- model accuracy (ϕ_{ACC}) or
- metric for A (ϕ_A) or
- metric for *B* (ϕ_B)

then A and B are in conflict

Protection Mechanism	Dataset	Protection Mechanism	
		DP	ADVTR
WM	MNIST	ϕ_{ACC} ϕ_{WM}	ϕ_{ACC} ϕ_{WM} ϕ_{ADV}
	FMNIST	ϕ_{ACC} ϕ_{WM}	ϕ_{ACC} ϕ_{WM} ϕ_{ADV}
	CIFAR10	ϕ_{ACC} ϕ_{WM}	ϕ_{ACC} ϕ_{WM} ϕ_{ADV}
RADDATA	MNIST	ϕ_{ACC} ϕ_{RAD}	ϕ_{ACC} ϕ_{RAD} ϕ_{ADV}
	FMNIST	ϕ_{ACC} ϕ_{RAD}	ϕ_{ACC} ϕ_{RAD} ϕ_{ADV}
	CIFAR10	ϕ_{ACC} ϕ_{RAD}	ϕ_{ACC} ϕ_{RAD} ϕ_{ADV}
DI	MNIST	ϕ_{ACC} ϕ_{DI}	ϕ_{ACC} ϕ_{DI} ϕ_{ADV}
	FMNIST	ϕ_{ACC} ϕ_{DI}	ϕ_{ACC} ϕ_{DI} ϕ_{ADV}
	CIFAR10	ϕ_{ACC} ϕ_{DI}	ϕ_{ACC} ϕ_{DI} ϕ_{ADV}

Szyller and Asokan. - *Conflicting Interactions Among Protections Mechanisms for Machine Learning Models*, (https://arxiv.org/abs/2207.01991)

Interaction between ML security/privacy techniques

Property	Adversarial Training	Differential Privacy	Membership Inference	Oblivious Training	Model/Gradient Inversion	Model Poisoning	Model Watermarking	Model Fingerprinting	Data Watermarking	Explainability	Fairness
Adversarial Training	X	[5]	[9]	?	?	[7]	OURS	OURS	OURS	[11]	?
Differential Privacy		X	[3, 6]	?	?	?	OURS	OURS	OURS	?	[1, 2, 8]
Membership Inference			X	?	?	[10]	?	?	?	?	?
Oblivious Training				X	?	?	?	?	?	?	?
Model/Gradient Inversion					X	?	?	?	?	?	?
Model Poisoning						X	?	?	?	?	?
Model Watermarking							X	?	?	?	?
Model Fingerprinting								X	?	[4]	?
Data Watermarking									X	?	?
Fairness										X	?
Explainability											X

REFERENCES

[1] Hongyan Chang and Reza Shokri. 2021. On the Privacy Risks of Algorithmic Fairness. In *2021 IEEE European Symposium on Security and Privacy (EuroS P)*. 292–303. https://doi.org/10.1109/EuroSP51992.2021.00028

[2] Victoria Cheng, Vinith M. Suriyakumar, Natalie Dullerud, Shalmali Joshi, and Marzyeh Ghassemi. 2021. Can You Fake It Until You Make It? Impacts of Differentially Private Synthetic Data on Downstream Classification Fairness. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (FAccT '21)*. Association for Computing Machinery, New York, NY, USA, 149–160. https://doi.org/10.1145/3442188.3445879

[3] Thomas Humphries, Simon Oya, Lindsey Tulloch, Matthew Rafuse, Ian Goldberg, Urs Hengartner, and Florian Kerschbaum. 2020. Investigating Membership Inference Attacks under Data Dependencies. https://doi.org/10.48550/ARXIV.2010.12112

[4] Hengrui Jia, Hongyu Chen, Jonas Guan, Ali Shahin Shamsabadi, and Nicolas Papernot. 2022. A Zest of LIME: Towards Architecture-Independent Model Distances. In *International Conference on Learning Representations*. https://openreview.net/forum?id=OUz_9TiTv9j

[5] Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. 2019. Certified Robustness to Adversarial Examples with Differential Privacy. In *2019 IEEE Symposium on Security and Privacy (SP)*. 656–672. https://doi.org/10.1109/SP.2019.00044

[6] Milad Nasr, Shuang Songi, Abhradeep Thakurta, Nicolas Papernot, and Nicholas Carlin. 2021. Adversary Instantiation: Lower Bounds for Differentially Private Machine Learning. In *2021 IEEE Symposium on Security and Privacy (SP)*. 866–882. https://doi.org/10.1109/SP40001.2021.00069

[7] Ren Pang, Hua Shen, Xinyang Zhang, Shouling Ji, Yevgeniy Vorobeychik, Xiapu Luo, Alex Liu, and Ting Wang. 2020. *A Tale of Evil Twins: Adversarial Inputs versus Poisoned Models*. Association for Computing Machinery, New York, NY, USA, 85–99. https://doi.org/10.1145/3372297.3417253

[8] Adam Pearce. 2022. Can a Model Be Differentially Private and Fair? https://pair.withgoogle.com/explorables/private-and-fair/. Online; accessed 7 April 2022.

[9] Liwei Song, Reza Shokri, and Prateek Mittal. 2019. Privacy Risks of Securing Machine Learning Models against Adversarial Examples. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (CCS '19)*. Association for Computing Machinery, 241–257. https://doi.org/10.1145/3319535.3354211

[10] Florian Tramèr, Reza Shokri, Ayrton San Joaquin, Hoang Le, Matthew Jagielski, Sanghyun Hong, and Nicholas Carlini. 2022. Truth Serum: Poisoning Machine Learning Models to Reveal Their Secrets. https://doi.org/10.48550/ARXIV.2204.00032

[11] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. 2019. Robustness May Be at Odds with Accuracy. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. https://openreview.net/forum?id=SyxAb30cY7

Szyller and Asokan. - *Conflicting Interactions Among Protections Mechanisms for Machine Learning Models*, (https://arxiv.org/abs/2207.01991)

Takeaways

Is model confidentiality important? Yes

- **models constitute business advantage to model owners**

Can models be extracted via their prediction APIs? Yes

- **Protecting model data via cryptography or hardware security is insufficient**

What can be done to counter model extraction? Deterrence as defense

- **Watermarking/fingerprinting? Open issues remain**

Can we simultaneously deploy protections against multiple concerns? Needs work

- **Important consideration but not yet sufficiently explored**

More on our model extraction work at https://ssg.aalto.fi/research/projects/mlsec/model-extraction/

Takeaways

Is model confidentiality important? Yes

- **models constitute business advantage to model owners**

Can models be extracted via their prediction APIs? Yes

- **Protecting model data via cryptography or hardware security is insufficient**

What can be done to counter model extraction? Deterrence as defense

- **Watermarking/fingerprinting? Open issues remain**

Can we simultaneously deploy protections against multiple concerns? Needs work

- **Important consideration but not yet sufficiently explored**

Open postdoc positions to help lead our work: ML security/privacy, platform security
https://asokan.org/asokan/research/SecureSystems-open-positions-Jul2021.php

Come work with us!

Open postdoc positions to help lead our work: ML security/privacy, platform security

https://asokan.org/asokan/research/SecureSystems-open-positions-Jul2021.php