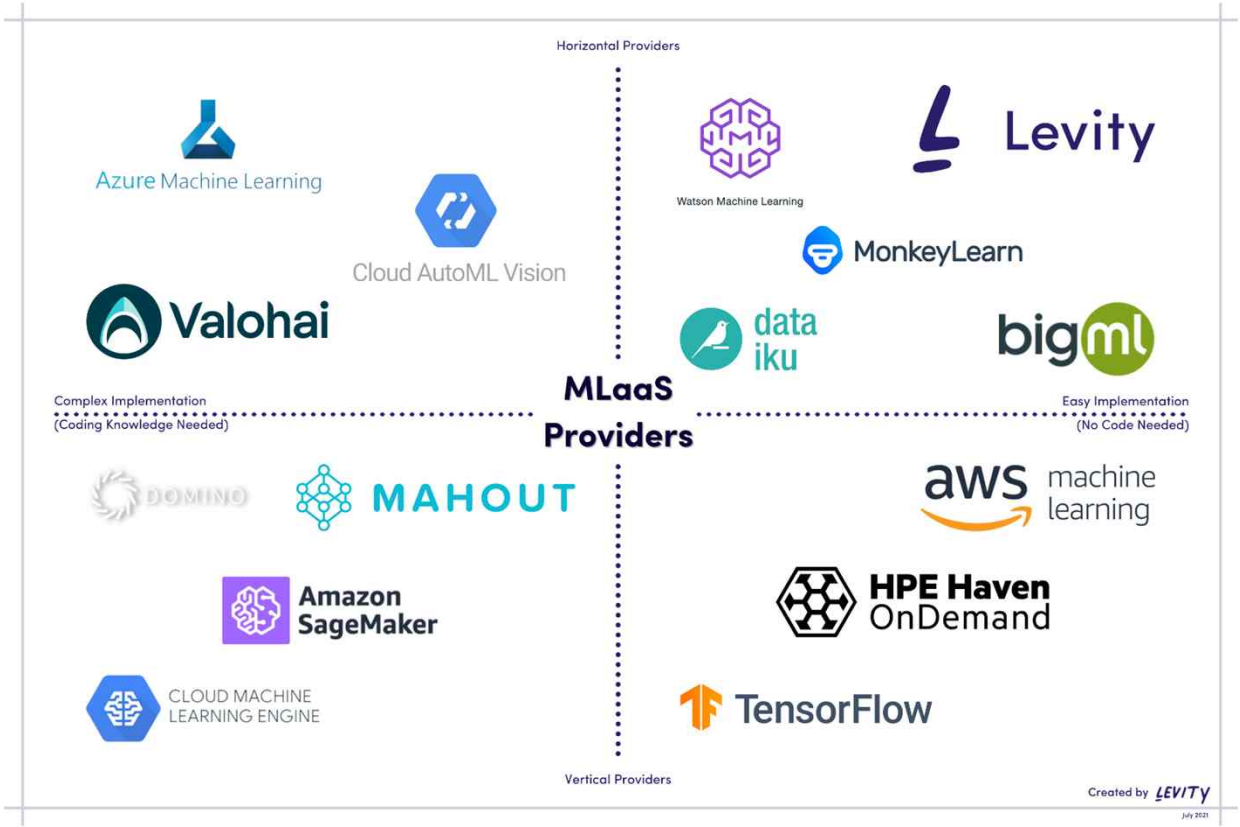
High Accuracy and High Fidelity Extraction of Neural Networks

Matthew Jagielski, Nicholas Carlini, David Berthelot,
Alex Kurakin, and Nicolas Papernot

Presented by Changhun Song @ SysSec

Introduction



Machine learning models became one of **the most valuable products**

Introduction

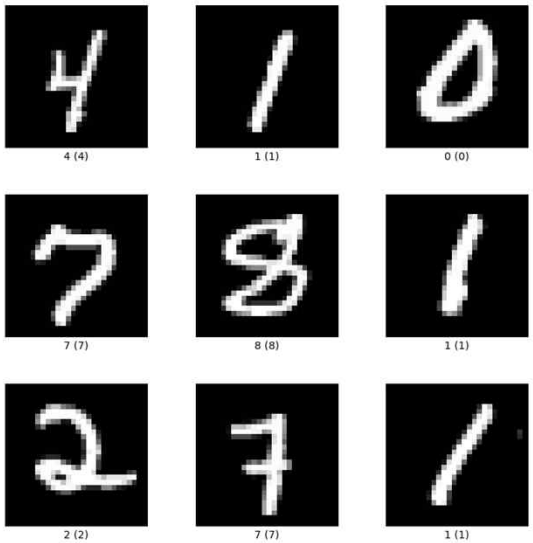


Image Recognition

Audio Recognition

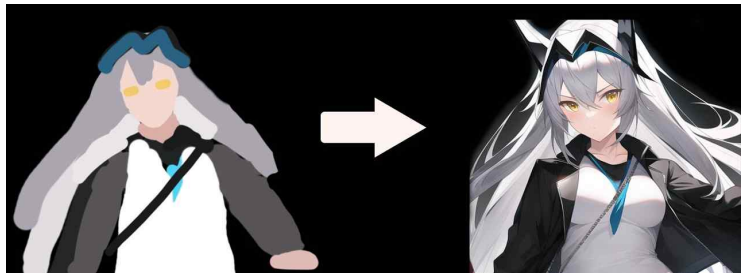
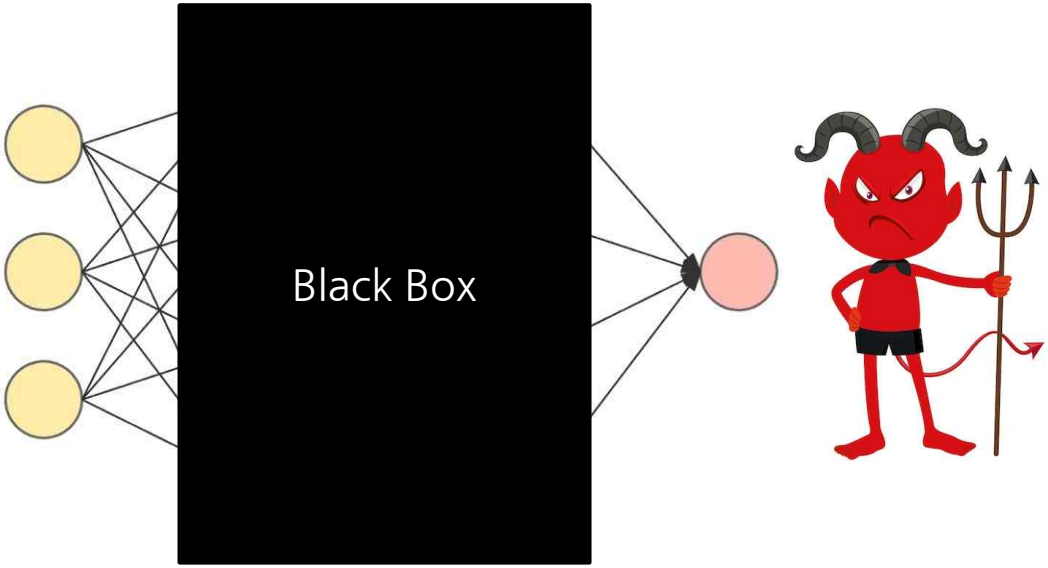


Image Generation

Machine learning models became one of **the most valuable products**

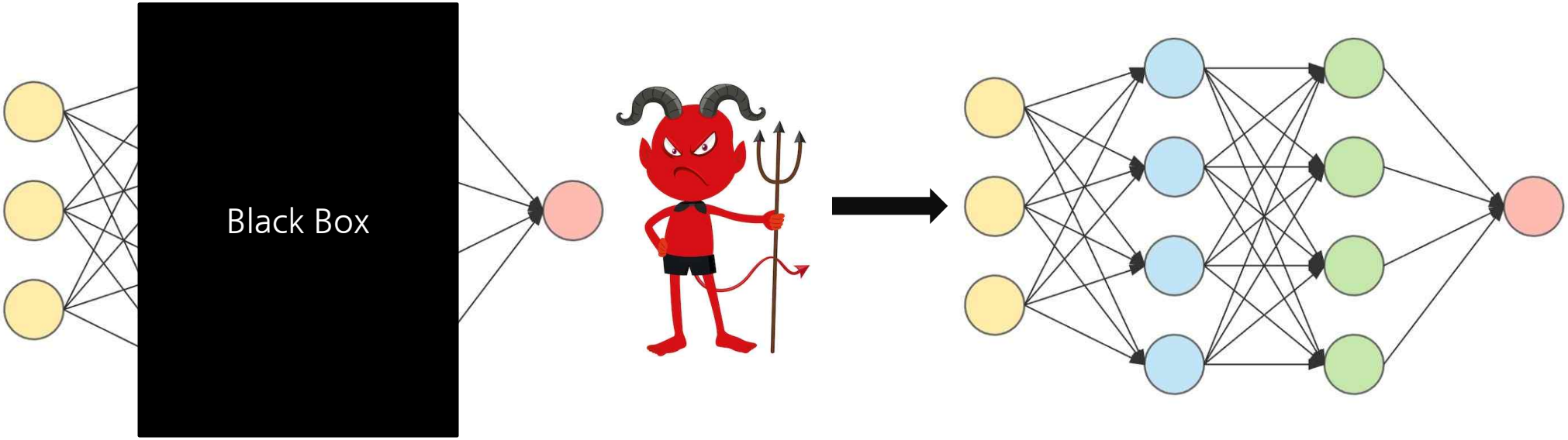
Model Extraction

- Model developers put lots of efforts and cost
  - Several months ~ years & more than million dollars
  - GPT-3 has about 175 billion parameters



Model Extraction

- Model developers put lots of efforts and cost
  - Several months ~ years & more than million dollars
  - GPT-3 has about 175 billion parameters



Model Extraction

- Theft
  - Achieve High Accuracy
  - Make a good quality model with the low cost
- Reconnaissance
  - Achieve High Fidelity
  - Functional Equivalence
  - Generate white-box adversarial examples
- Adversaries also have specific access restrictions
  - label, label and score, top-k scores, scores, logits

Model Extraction

- Try machine learning
  - Do machine learning on $(x_i, f(x_i))$
- We can directly recover linear models
  - Linear model $f(x)$: $W\ x + C$
  - $f([0,0,0,0,0,\ldots]) = C$
  - $f([1,0,0,0,0,\ldots]) = w_1 + C$
- What about neural networks?

Learning-based Extraction

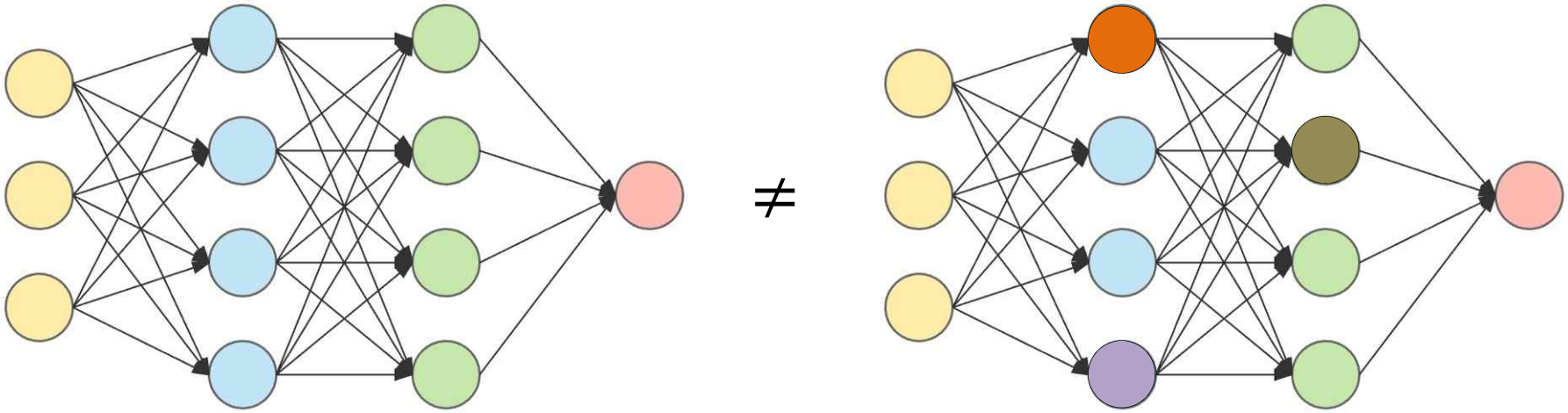
- Semi-Supervised Learning
  - Concepts from the MixMatch[NeurIPS'19][1]
  - Target model training with Semi-Supervised Learning
    - Query efficient model extraction
      - Less effort to extract models

Dataset	Queries	Baseline Accuracy	SemiSup Accuracy
SVHN	250	79.25%	95.82%
CIFAR-10	250	53.35%	87.98%
ImageNet (top 5)	~140000	83.5%	86.17%

[1]: Berthelot, David, et al. "Mixmatch: A holistic approach to semi-supervised learning." *Advances in neural information processing systems* 32 (2019).

Learning-based Extraction

- Non-determinism
  - The extracted model ≠ The victim model
    - Why?
      - Different model can also show the same output result
- Hard to achieve **High Fidelity**



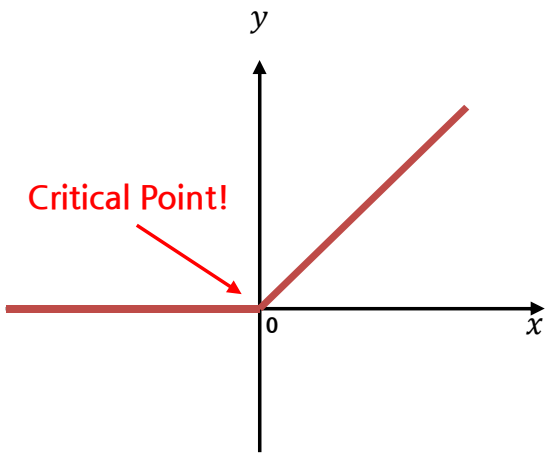
Direct Recovery (Milli et al.)

- Linear model direct recovery isn't easily extended to neural networks
- Focus on 2-layer ReLU networks, following Milli et al. (FAT*'19)[1]
  - Milli et al.[1] suggested how to directly recover the model from the structure
    - Focus on the gradient change of the ReLU function
    - Not actually implemented in practice
  - This paper presented direct recovery in practice

[1]: Milli et al. Model Reconstruction from Model Explanations, FAT*'19

Direct Recovery

- **Note:** 2-Layer ReLU networks
- $ReLU\ (x)$: Rectified Linear Unit
  - Critical Point at Zero
    - The unit gradient change from 0 to 1 at the critical point



$$ReLU\ (x) = \max(0, x)$$

[1]: Milli et al. Model Reconstruction from Model Explanations, FAT*'19

Direct Recovery

- **Note:** 2-Layer ReLU networks
- $ReLU\ (x)$
  - Critical Point at Zero
    - The unit gradient change from 0 to 1 at the critical point
- $\mathrm{y} = A^{(1)} g_i\left(A^{(0)} x + B^{(0)}\right) + B^{(1)}$
  - $A^{(i)}: Weight\ of\ i-th\ Layer$
  - $B^{(i)}: Bias\ of\ i-th\ Layer$
  - $g_i: Activation\ Function\ of\ i-th\ Layer\ (i.e. ReLU)$
  - $y\ and\ x\ is\ given\ to\ the\ attacker$

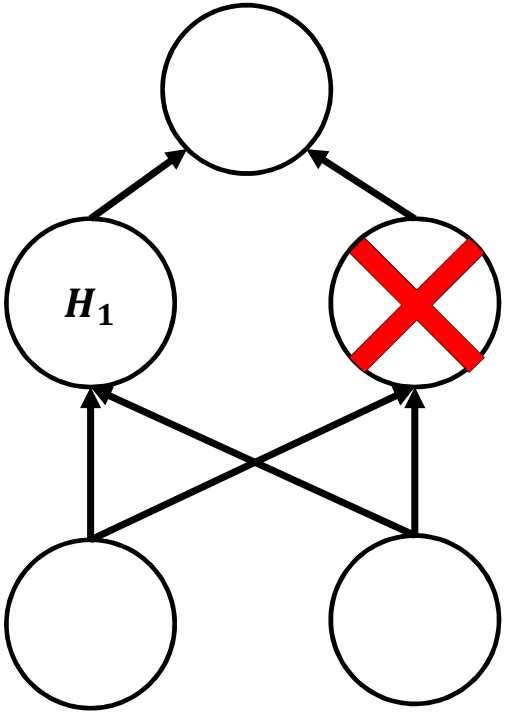
[1]: Milli et al. Model Reconstruction from Model Explanations, FAT*'19

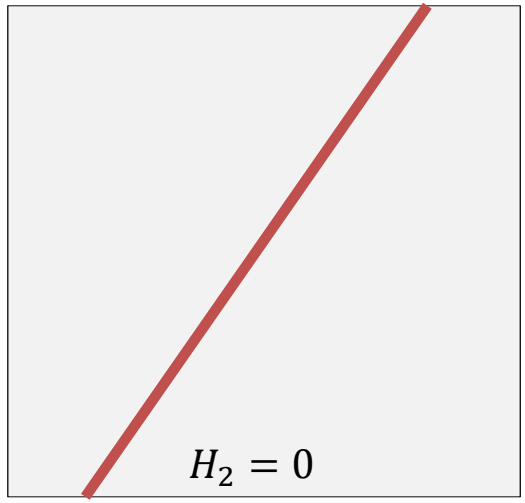
Attack Algorithm

1. Critical Point Search
   - Identify inputs that exactly one of the ReLU units
2. Weight Recovery
   - Take an input x which causes the i-th neuron to be at a critical point
     - Used to compute the difference between the two adjacent linear regions
     - By repeating this, we can recover the weight matrix A
3. Sign Recovery
   - Determines the sign of each row-vector $A_j^{(0)}$ for all j using global information
4. Final Layer Extraction
   - Least square algorithm to solve for the second later of the network

Attack Algorithm

- Critical Point Search
  - Identify inputs that exactly one of the ReLU units
  - Consider Neural Network with 2 hidden Layers

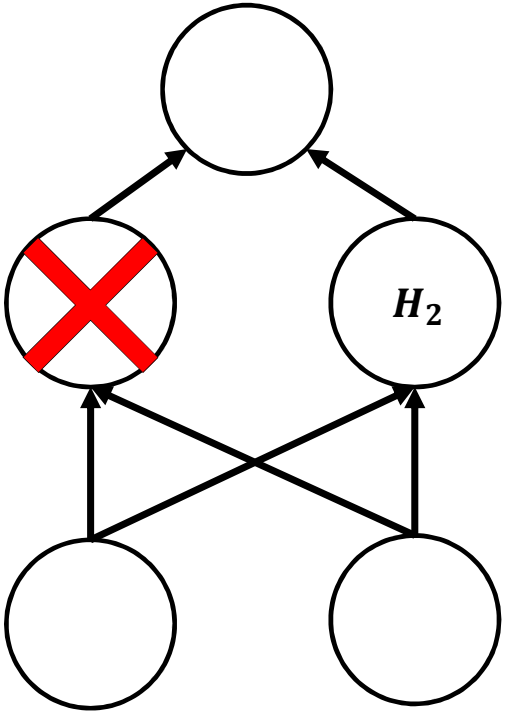


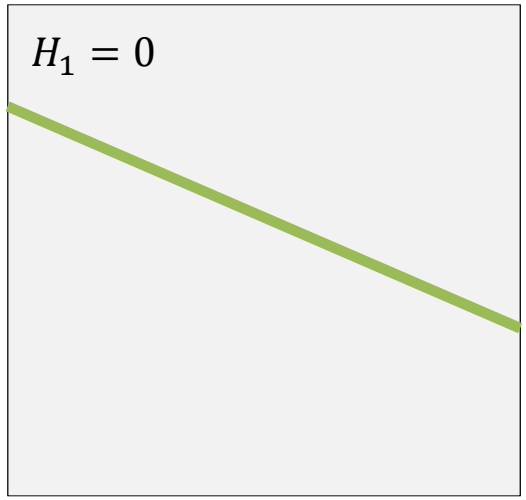


Input Space

Attack Algorithm

- Critical Point Search
  - Identify inputs that exactly one of the ReLU units
  - Consider Neural Network with 2 hidden Layers

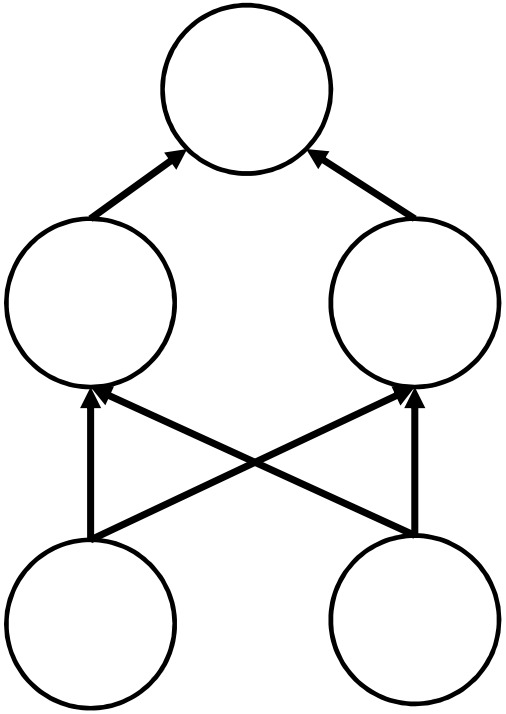


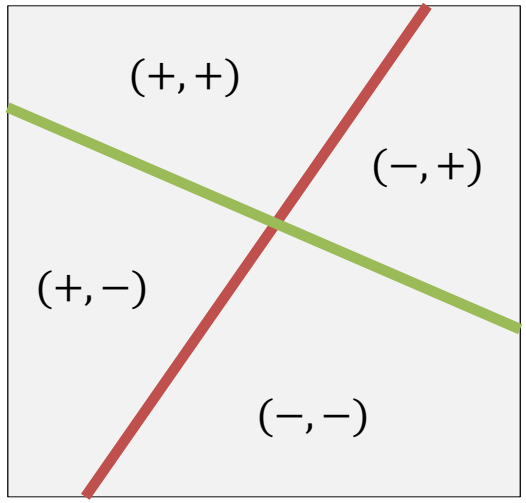


Input Space

Attack Algorithm

- Critical Point Search
  - Identify inputs that exactly one of the ReLU units
  - Consider Neural Network with 2 hidden Layers

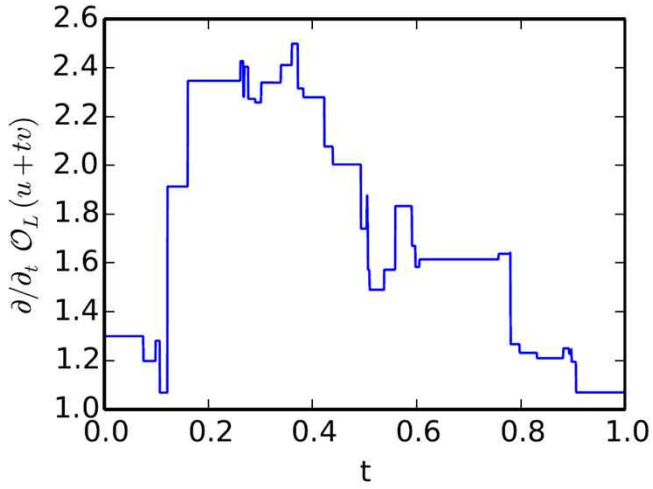


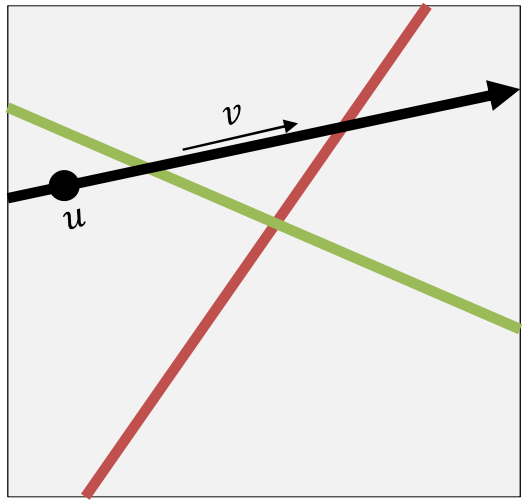


Input Space

Attack Algorithm

- Critical Point Search
  - Identify inputs that exactly one of the ReLU units
  - Pick a random point u and direction v
    - Check $f(u + tv)$
    - Find undifferentiable point
      - Detect the critical point





Input Space

Attack Algorithm

- Weight Recovery
  - Critical point: $A^{(0)}x_i + B^{(0)} = 0$
  - Extract $A^{(0)}$ approximately using the ratio
- Sign Recovery
  - Add two directional vectors to compare its sign
  - Extract $B^{(0)}$ approximately and determine the sign

Attack Algorithm

- Last Layer Extraction
  - Recall: $y = A^{(1)} g_0\left(A^{(0)} x + B^{(0)}\right) + B^{(1)}$
  - $x, A^{(0)}, B^{(0)}$ became the constant by extraction
  - $g_0 = ReLU$
  - $y = C \cdot A^{(1)} + B^{(1)}$ → Simple Linear Equation
    - Apply Least-Square Solution

Parameters	25,000	50,000	100,000
Fidelity	100%	100%	99.98%
Queries	~150,000	~300,000	~600,000

Effectiveness of our Direct Recovery Attack

Related Works

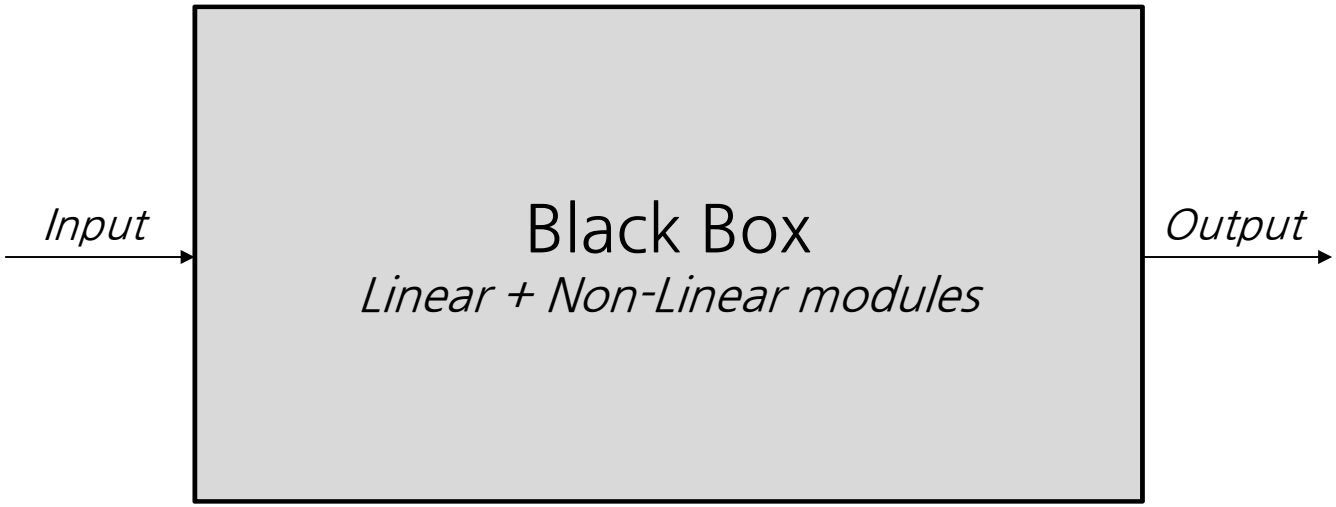
- Side Channel Model Extraction
  - Power Side Channel Attack [USENIX Security'19]
  - Data Transmission Eavesdropping
    - PCI packets, Bus memory, DRAM, …
- Model Training Data Extraction
  - Membership Inference Attacks [IEEE S&P'17]
- ML-based Extraction
  - Query-based Model Training
  - Adversarial Machine Learning
- Follow-up work
  - Cryptanalytic extraction of neural network models [CRYPTO'20]
    - Deeper machine learning model extraction

Summary

- Proposed two model extraction methods
  - Semi-Supervised machine learning (MixMatch)
    - Query efficient model extraction
    - 95% and 87% accuracy with only 250 queries on SVHN and CIFAR-10 datasets
  - Direct Recovery (by Milli et al.)
    - Find ReLU critical points
    - Implemented this method on practice
    - Achieved high fidelity (~100%)

Cryptography vs ML

- Linear calculation + Non-linear calculation
  - Cryptography: Hard to calculate key
  - Machine Learning: Huge # of parameters



Questions

- Questions
  - Hyungchan Kim
    - how does the defense mentioned in the paper affect the accuracy and fidelity of the mo del extraction attack?
      - Defense methods mentioned in the paper is general defense methodologies for model extraction attacks, and the defense methodologies don't target the degradation of accuracy and fidelity, just make them impossible
      - Limiting the information from the model
        - » It is impossible to do direct recovery if there is no confidence score
      - Analyzing Query Patterns to detect malicious users
        - » If the attack is as not as efficient than query limit, it is impossible to attack
  - DAVY Guillaume
    - Do deeper neural networks have a lesser transferability?
      - For direct recovery, yes. The weight recovery calculates the weight "approximately"
      - So, error will be propagated to deeper layers

Questions

- Best Questions
  - Sangmin Woo
    - Q. Is there any example that applied this paper's approach on other activation models?
    - A. It is hard to show examples. Because other activation functions usually don't have such kind of critical point (tanh, sigmoid, …). However, for the variation of ReLU such as LeakyReLU or ReLU6, the attack might work, but there are more challenges because the negative value is not dropped to 0. If you are interested in this work, you can do more research about this.
  - Junyoung Park
    - Q. Is this method applicable to real-world situations, for example, ensemble of models?
    - A. We can think of the ensemble of models as a single big model as well. So, semi-supervised learning should work because it just trains a new model which makes the same decision as the victim model. However, for the direct recovery method, the threat model is in a very limited environment: a 2-Layer ReLU network. So, it is hard to find even a real-world victim model. Even though we can extend the attack shown in Crypto'20 to analyze more layers, if the parameters are too large, the recovery process would be unreachable by computing power.

Questions

- Best Questions
  - Davy Guillaume
    - Q. Are parasitic layers a good defense against this attack for you ? Here is a paper discussing it : https://arxiv.org/pdf/2005.12782v1.pdf .
    - A. This work obviously can mitigate the suggested “direct recovery” method by adding artificial parasitic layers to the network. Because a new artificial layer makes the network more complex and bigger. However, this mitigation, as all of you know, will degrade the performance of the neural network, and the mitigated model is just bigger neural network. So, the attack model won’t be changed. Linearly dependent rows also can mitigate this direct recovery method because the critical point could be overlapped. The authors in this paper didn’t consider it because they couldn’t see it in the examples.