Genetic Improvement for DNN Security

Hunter Baxter, Yu Huang, **Kevin Leach**

kevin.leach@vanderbilt.edu

April 9, 2024

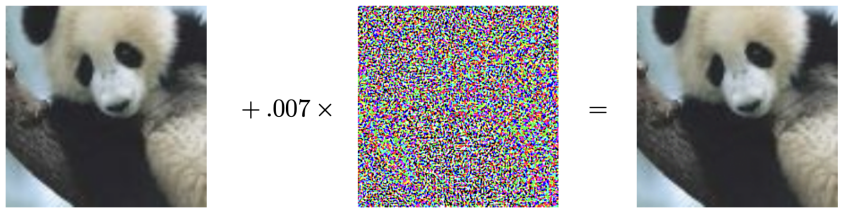
Bottom Line Up Front

This position paper conveys:

- DNN layers can be distributed over multiple machines
- Scheduling and organizing DNN layer distribution can be represented as a genome
- Genetic Improvement can be used to obfuscate DNN architecture and execution as a type of moving target defense
- A prototype for scheduling DNN distribution called *Jigsaw*

Motivation

- DNN are often proprietary
- Current defenses cannot secure modern models sufficiently
- There is a demand for a scalable defense against model theft
- Balancing performance with security is essential for industrial purposes

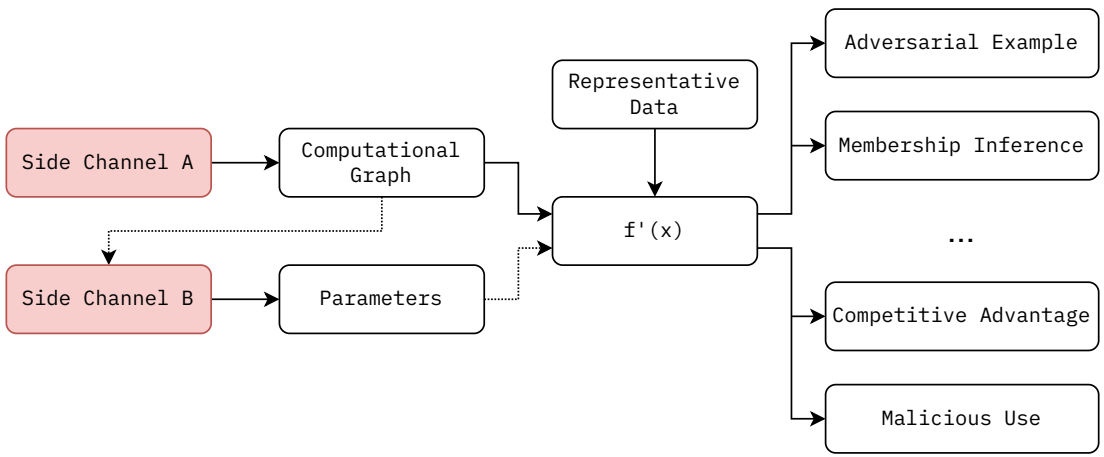


Attack Type	**Distribution**
Distribution Poisoning	10
Model Stealing	6
Model Inversion	4
Backdoored ML	4
Membership Inference	3
Adversarial Examples	2

Table: Industry Top Attack Priorities

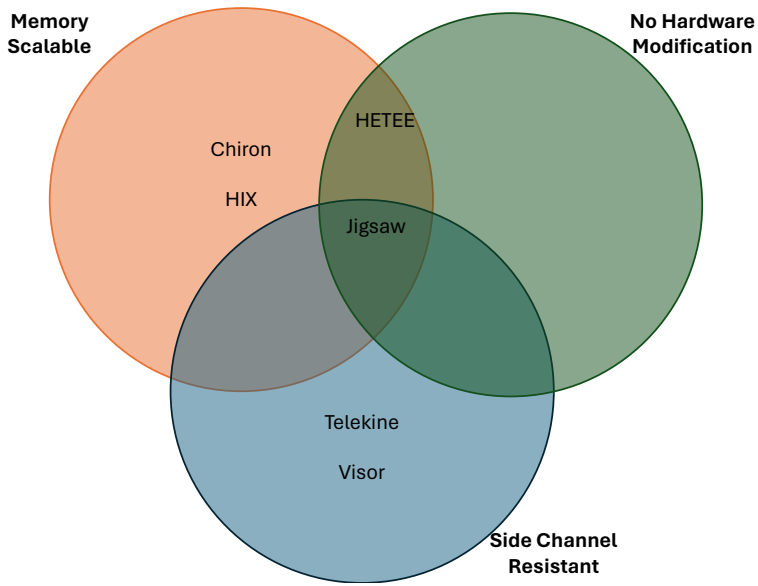
Model Extraction

- An adversarial attack where a private model $f(x)$ is completely reconstructed
- Model Extraction accelerates malicious activity
- Cache Telepathy: Spectre
- DeepSteal: RowHammer
- DeepSniffer: GPU timing information

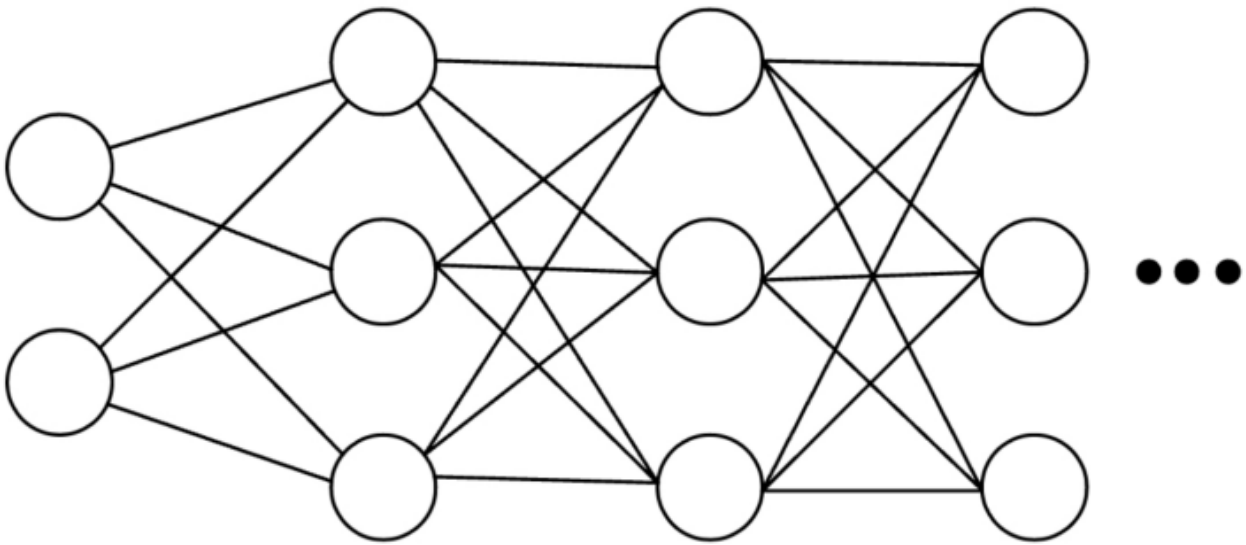


Prior Work

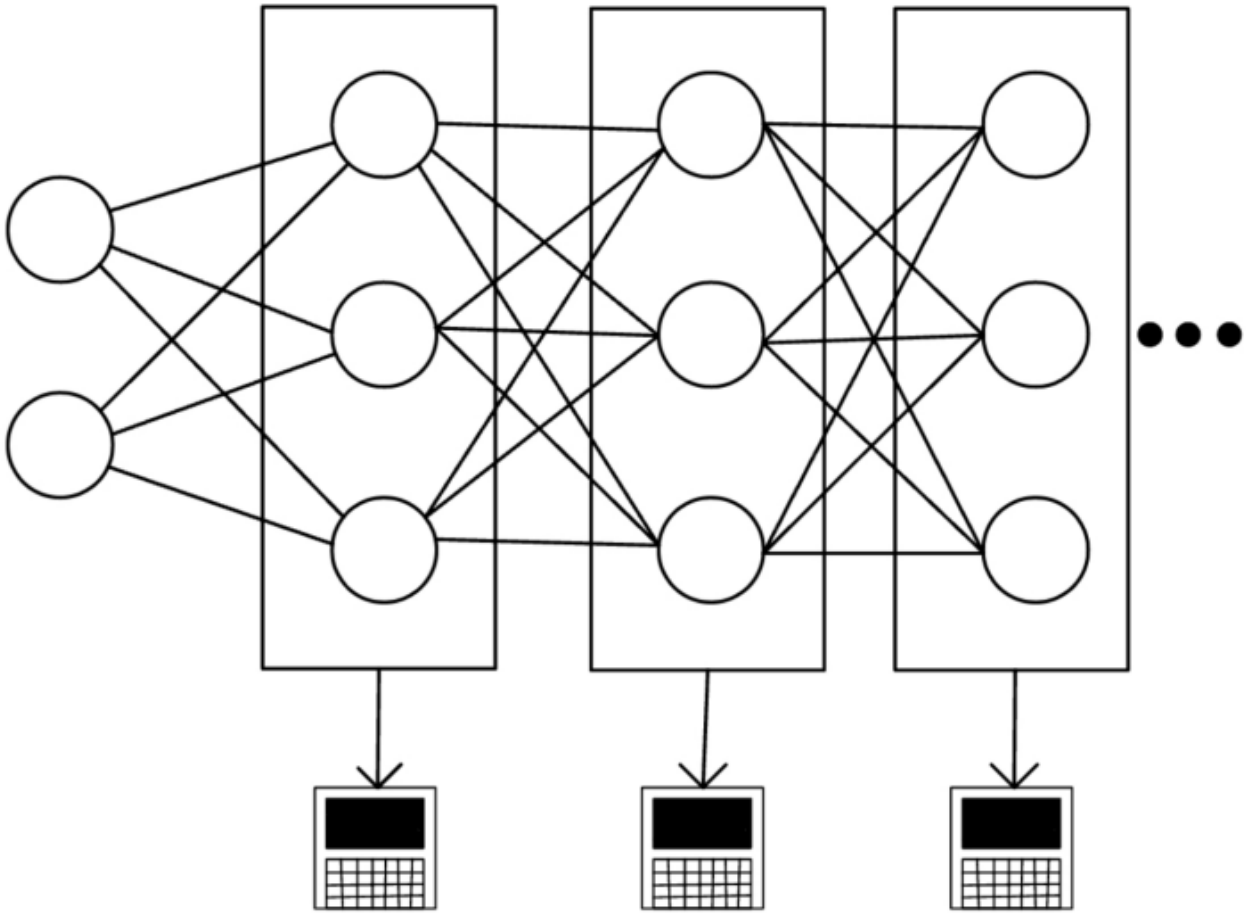
- Strong Isolation
  - Limited Scalability
  - Hardware Modification
  - Side-Channel Vulnerabilities
- Data Obliviousness
  - Large Performance Overhead
  - Algorithm Dependent
- Jigsaw proposes probabilistic, not strong, isolation
- Information gained at t may not work at $t + \delta$



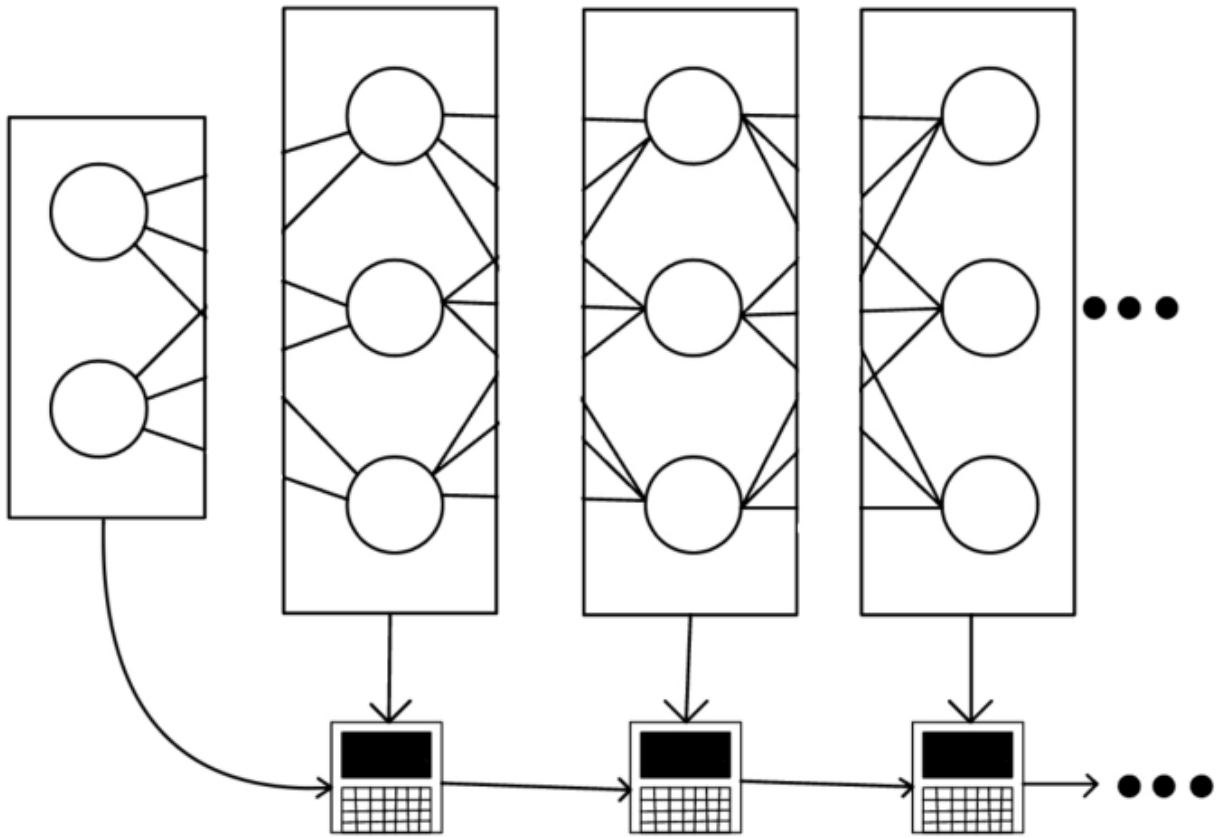
Jigsaw



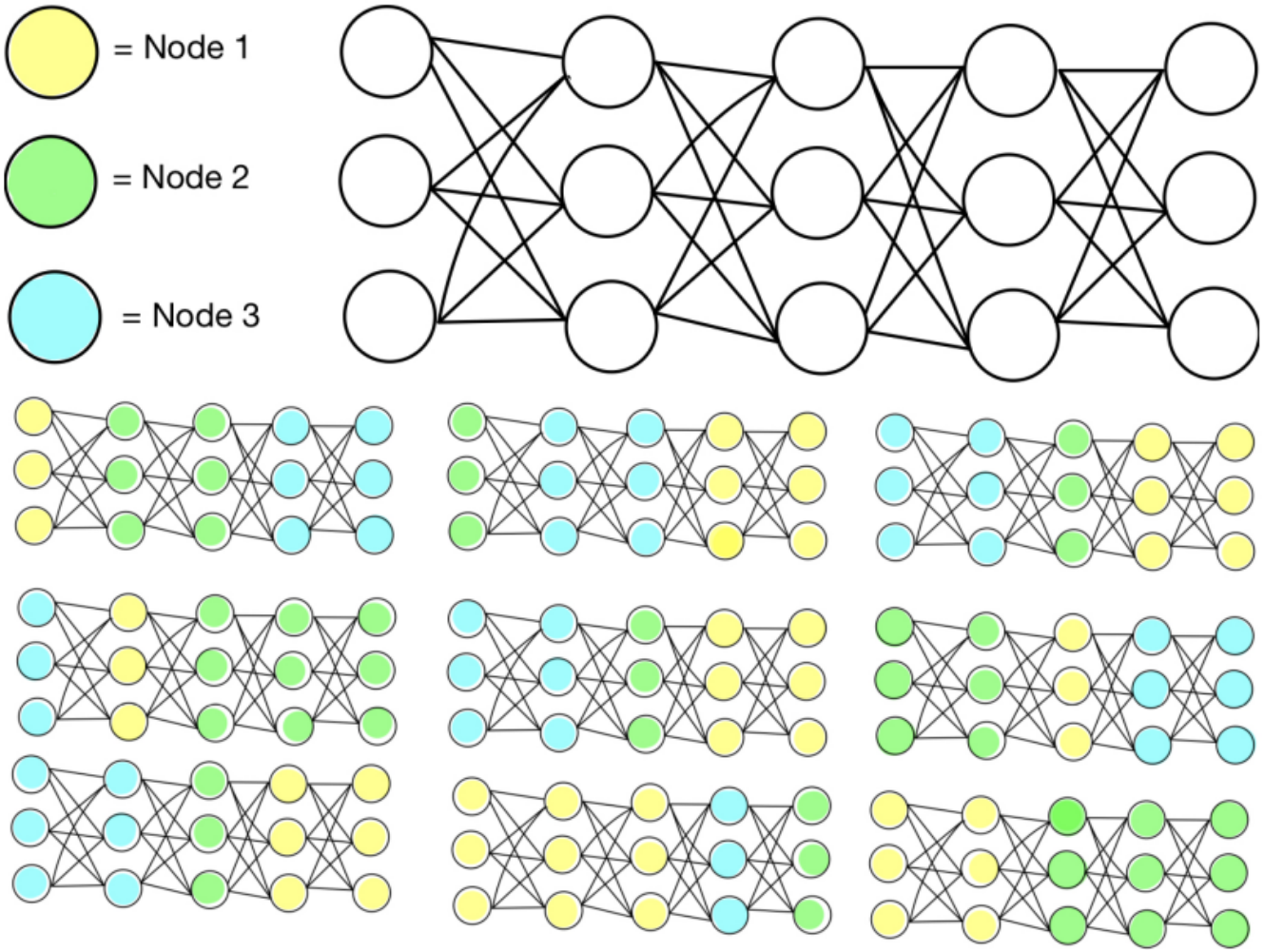
Jigsaw



Jigsaw



Jigsaw



Moving the Target

- Schedule modeled by binary matrix
- "Seed" schedule from a tool like Alpa
- Genetic algorithm generates solutions
- Multi-Objective Fitness Function of Difference and Latency
- Replace "seed" schedule with generated solution

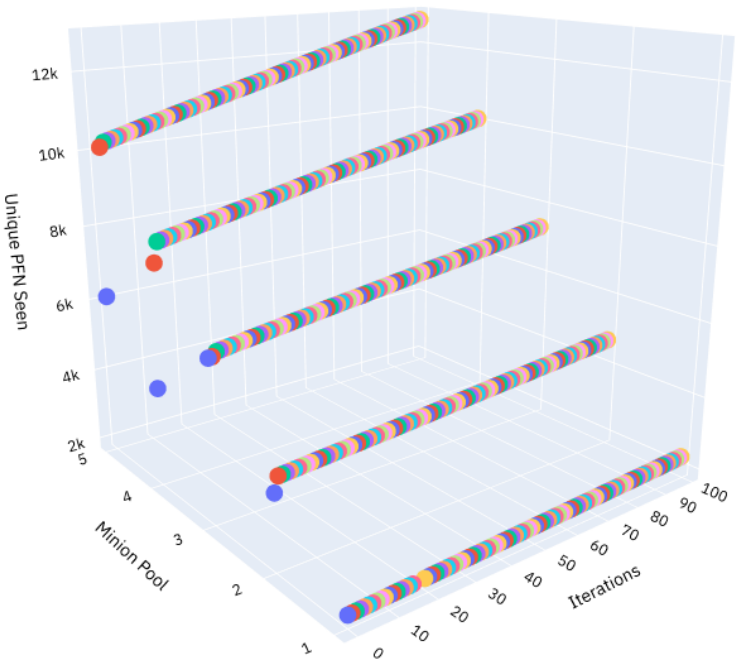
```
|1 1 1 1 0 0 0 0 0 0 0 0|
|0 0 0 0 1 1 1 1 0 0 0 0|
|0 0 0 0 0 0 0 0 1 1 1 1|
|0 0 0 0 0 0 0 0 0 0 0 0|
|0 0 0 0 0 0 0 0 0 0 0 0|

|0 0 0 0 0 0 0 0 0 0 0 0|
|0 0 0 0 0 0 0 0 0 0 0 0|
|0 0 0 0 0 0 0 0 0 0 1 1|
|0 0 0 0 0 0 0 0 0 0 1 0|
|1 1 1 1 1 1 1 1 1 0 0 0|

|1 1 1 1 1 1 1 1 1 0 0 0|
|0 0 0 0 0 0 0 0 0 1 1 0|
|0 0 0 0 0 0 0 0 0 0 0 0|
|0 0 0 0 0 0 0 0 0 0 0 0|
|0 0 0 0 0 0 0 0 0 0 0 1|
```

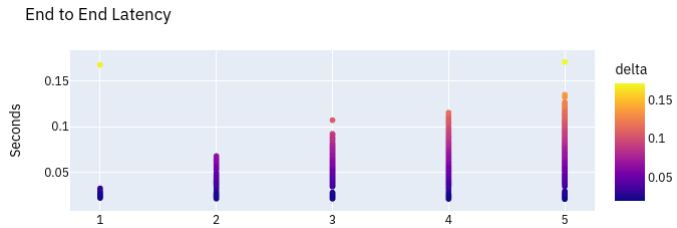
Genetic Algorithm Parameters

- Schedule $= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
- Schedule Representation $= \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$
- Random mutation: $M(\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}) =$
  $\begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
- Single Point Crossover
  $C(\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix})$
  $= \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$
- Tournament Selection

Evaluation

- Linear Increase in Memory Pages
- Linear Increase in Latency
- Quick Solution Generation





Introduction

Approach

Evaluation

Conclusion

Conclusion

- DNNs are often subjected to model extraction attacks
- Security-critical DNNs may require sophisticated defenses
- DNN architecture can be split across multiple nodes
- Genetic Improvement can be used to develop how to distribute DNN layers among multiple hosts
- 'Jigsaw' prototype provides initial basis for improving security posture of DNNs