AI & Security: Challenges, Lessons & Future Directions

Dawn Song
Professor, UC Berkeley
Founder & CEO, Oasis Labs

@dawnsongtweets
AlphaGo: Winning over World Champion

Source: David Silver
AlphaStar: Winning over World’s Top Players
Deep Learning Powering Everyday Products
Attacks are increasing in scale & sophistication
Massive DDoS Caused by IoT Devices (Mirai Botnet)

- Over 400,000 compromised IoT devices over 160 countries
  - Security cameras/webcams/baby monitors
  - Home routers
- One of the biggest DDoS attacks in history
  - Over 1Tbps combined attack traffic
Biggest Data Breaches Of the 21st Century

Large data breaches in the 21st century (in millions)

Source: csoonline.com
Attacks Entering New Landscape

Ukraine power outage by cyber attack impacted over 250,000 customers

Millions of dollars lost in targeted attacks in SWIFT banking system
How will (in)security impact the deployment of AI?

How will the rise of AI alter the security landscape?
Deep Learning Improving Security Capabilities
IoT devices are plagued with vulnerabilities from third-party code.
Deep learning for vulnerability detection in IoT Devices

Firmware Files → Raw Feature Extraction (disassembler) → Code Graph

Vulnerability Function

Embedding Network \( \phi(\cdot) \)

Cosine Similarity

Neural Network-based Graph Embedding for Cross-Platform Binary Code Search

[XLFSSY, ACM Computer and Communication Symposium 2017]
Deep learning for vulnerability detection in IoT Devices

Training time:
Previous work: > 1 week
Our approach: < 30 mins

Serving time (per function):
Previous work: a few mins
Our work: a few milliseconds
10,000 times faster

Identified vulnerabilities among top 50:
Previous work: 10/50
Our approach: 42/50
AI Enables Stronger Security Capabilities

- Automatic vulnerability detection & patching
- Automatic agents for attack detection, analysis, & defense
One fundamental weakness of cyber systems is humans

80+% of penetrations and hacks start with a social engineering attack
70+% of nation state attacks [FBI, 2011/Verizon 2014]
AI Enables Chatbot for Phishing Detection

Chatbot for booking flights, finding restaurants

Chatbot for social engineering attack detection & defense
AI Enables Stronger Security Capabilities

- Automatic vulnerability detection & patching
- Automatic agents for attack detection, analysis, & defense
- Automatic verification of software security
AI Agents to Prove Theorems & Verify Programs

Deep Reinforcement Learning Agent Learning to Play Go

GamePad: A Learning Environment for Theorem Proving
Daniel Huang, Prafulla Dhariwal, Dawn Song, Ilya Sutskever
AI Security Enabler

- AI enables new security capabilities
- Security enables better AI

**Integrity**: produces intended/correct results (adversarial machine learning)

**Confidentiality/Privacy**: does not leak users’ sensitive data (secure, privacy-preserving machine learning)

Preventing misuse of AI
Important to consider the presence of attacker

- History has shown attacker always follows footsteps of new technology development (or sometimes even leads it)

- The stake is even higher with AI
  - As AI controls more and more systems, attacker will have higher & higher incentives
  - As AI becomes more and more capable, the consequence of misuse by attacker will become more and more severe
Machine Learning in the Presence of Attacker

• Attack AI
  – Integrity:
    • Cause learning system to not produce intended/correct results
    • Cause learning system to produce targeted outcome designed by attacker
  – Confidentiality:
    • Learn sensitive information about individuals
  – Need security in learning systems

• Misuse AI
  – Misuse AI to attack other systems
    • Find vulnerabilities in other systems, target attacks, devise attacks
  – Need security in other systems
Adversarial Examples

- Clean Stop Sign
- Real-world Stop Sign in Berkeley
- Adversarial Example
- Adversarial Example

“Stop sign”
“Stop sign”
“Speed limit sign 45km/h”
“Speed limit sign 45km/h”
Adversarial Examples in Physical World

Adversarial examples in physical world remain effective under different viewing distances, angles, other conditions.

Lab Test Summary (Stationary)
Target Class: Speed Limit 45

Misclassify

Adversarial examples in physical world & remain effective under different viewing distances, angles, other conditions
Adversarial Examples Are Prevalent in Deep Learning Systems
Adversarial Examples Prevalent in Deep Learning Systems

• Most existing work on adversarial examples:
  – Image classification task
  – Target model is known

• Our investigation on adversarial examples:

  Other tasks and model classes
  - Generative Models
  - Deep Reinforcement Learning
  - VisualQA/Image-to-code

  New Attack Methods
  - Blackbox Attacks
    - Weaker Threat Models (Target model is unknown)

  Provide more diversity of attacks
Visual Question & Answer (VQA)

Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding, Fukui et al., CVPR 2018
Q: Where is the plane?

Answer: Runway

Fooling VQA

Target: Sky

Fooling Vision and Language Models Despite Localization and Attention Mechanism, Xu et al., CVPR 2018
Q: How many cats are there?

Fooling VQA

Target: 2

Benign image

VQA Model

Answer: 1

Adversarial example

VQA Model

Answer: 2
Adversarial Examples Fooling Deep Reinforcement Learning Agents

Original Frames

Original Frames with Adversarial Perturbation

Score

No. of steps

FGSM Evaluation (0.005)

Training on non-noisy environment

Adversarial Evaluation

Jernej Kos and Dawn Song: Delving into adversarial attacks on deep policies [ICLR Workshop 2017].
A General Framework for Black-box attacks

• Zero-Query Attack
  – Random perturbation
  – Difference of means
  – Transferability-based attack
    • Practical Black-Box Attacks against Machine Learning [Papernot et al. 2016]
    • Ensemble transferability-based attack [Yanpei Liu, Xinyun Chen, Chang Liu, Dawn Song: Delving into Transferable Adversarial Examples and Black-box Attacks, ICLR 2017]

• Query Based Attack
  – Finite difference gradient estimation
  – Query reduced gradient estimation
  – Results: similar effectiveness to whitebox attack
  – A general active query game model
    • Exploring the Space of Black-box Attacks on Deep Neural Networks [Bhagoji, Li, He, Dawn, ECCV 2018]
Adversarial Machine Learning

- **Adversarial machine learning:**
  - Learning in the presence of adversaries

- **Inference time:** adversarial example fools learning system
  - Evasion attacks
    - Evade malware detection; fraud detection

- **Training time:**
  - Attacker poisons training dataset (e.g., poison labels) to fool learning system to learn wrong model
    - Poisoning attacks: e.g., Microsoft’s Tay twitter chatbot
  - Attacker selectively shows learner training data points (even with correct labels) to fool learning system to learn wrong model
  - Data poisoning is particularly challenging with crowd-sourcing & insider attack
  - Difficult to detect when the model has been poisoned

- **Adversarial machine learning particularly important for security critical system**
Numerous Defenses Proposed

- Ensemble
- Normalization
- Distributional detection
- PCA detection
- Secondary classification
- Stochastic
- Generative
- Training process
- Architecture
- Retrain
- Pre-process input

Detection

Prevention
No Sufficient Defense Today

▷ Strong, adaptive attacker can easily evade today’s defenses

▷ Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples
  ○ Athalye, Carlini, Wagner [ICML 2018]

▷ Ensemble of weak defenses does not (by default) lead to strong defense
  ○ Warren He, James Wei, Xinyun Chen, Nicholas Carlini, Dawn Song [WOOT 2017]

▷ Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods
  ○ Nicholas Carlini and David Wagner [AI Sec 2017]
Security will be one of the biggest challenges in Deploying AI
Machine Learning in the Presence of Attacker

• Attack AI
  – Integrity:
    • Cause learning system to not produce intended/correct results
    • Cause learning system to produce targeted outcome designed by attacker
  – Confidentiality:
    • Learn sensitive information about individuals
  – Need security in learning systems

• Misuse AI
  – Misuse AI to attack other systems
    • Find vulnerabilities in other systems, target attacks, devise attacks
  – Need security in other systems
Data is the New Oil
Current Frameworks for Data Analytics & Machine Learning

Data Owners ↘ Data

 ↗ Analyst ↘ Analytics & ML Program ↘ Computation Infrastructure ↘ Results

42
Current Framework Insufficient for Protecting Data Rights & Privacy

Issue 1: Untrusted program

Issue 2: Untrusted infrastructure

Issue 3: Sensitive results

Issue 4: Users have no control & visibility on data usage
Desired Solutions for Confidentiality/Privacy

**Data Owners** → **Data** → **Analyst** → **Analytics & ML Program** → **Computation Infrastructure** → **Results**

**Issues**
- **Issue 1:** Untrusted program
- **Issue 2:** Untrusted infrastructure
- **Issue 3:** Sensitive results
- **Issue 4:** Users have no control & visibility on data usage

**Desired Solutions**
- Program Rewriting & Verification
- Secure Computation
- Differential Privacy
- Distributed Ledger & Smart Contract

Consumed internally or sold/shared to external party.
Do Neural Networks Remember Training Data?

Can Attackers Extract Secrets (in Training Data) from (Querying) Learned Models?

1. Train

2. Predict

"What are you" → "doing"

N Carlini, C Liu, J Kos, Ú Erlingsson, D Song. The Secret Sharer: Evaluating and Testing Unintended Memorization in Neural Networks 2018
1. Train

Nicholas's SSN is

123-45-6789

N Carlini, C Liu, J Kos, Ú Erlingsson, D Song. The Secret Sharer: Evaluating and Testing Unintended Memorization in Neural Networks 2018
Extracting Social Security Number from Language Model

- Learning task: train a language model on Enron Email dataset
  - Containing actual people’s credit card and social security numbers
- New attacks: can extract 3 of the 10 secrets completely by querying trained models
- New measure “Exposure” for memorization
  - Used in Google Smart Compose

<table>
<thead>
<tr>
<th>User</th>
<th>Secret Type</th>
<th>Exposure</th>
<th>Extracted?</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>CCN</td>
<td>52</td>
<td>✓</td>
</tr>
<tr>
<td>B</td>
<td>SSN</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SSN</td>
<td>16</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>SSN</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SSN</td>
<td>22</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>SSN</td>
<td>32</td>
<td>✓</td>
</tr>
<tr>
<td>F</td>
<td>SSN</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>CCN</td>
<td>29</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CCN</td>
<td>48</td>
<td>✓</td>
</tr>
</tbody>
</table>
Preventing Memorization

- **Differential Privacy**: a formal notion of privacy to protect sensitive inputs
  - Exposure is lower empirically
  - Attack unable to extract secrets

- Solution: train a differentially-private neural network
  - Exposure is lower empirically
  - Attack unable to extract secrets

<table>
<thead>
<tr>
<th>With DP</th>
<th>Optimizer</th>
<th>$\varepsilon$</th>
<th>Testing Loss</th>
<th>Estimated Exposure</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMSProp</td>
<td>0.65</td>
<td>1.69</td>
<td>1.1</td>
<td></td>
</tr>
<tr>
<td>RMSProp</td>
<td>1.21</td>
<td>1.59</td>
<td>2.3</td>
<td></td>
</tr>
<tr>
<td>RMSProp</td>
<td>5.26</td>
<td>1.41</td>
<td>1.8</td>
<td></td>
</tr>
<tr>
<td>RMSProp</td>
<td>89</td>
<td>1.34</td>
<td>2.1</td>
<td></td>
</tr>
<tr>
<td>RMSProp</td>
<td>$2 \times 10^8$</td>
<td>1.32</td>
<td>3.2</td>
<td></td>
</tr>
<tr>
<td>RMSProp</td>
<td>$1 \times 10^9$</td>
<td>1.26</td>
<td>2.8</td>
<td></td>
</tr>
<tr>
<td>SGD</td>
<td>$\infty$</td>
<td>2.11</td>
<td>3.6</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>No DP</th>
<th>Optimizer</th>
<th>Testing Loss</th>
<th>Estimated Exposure</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGD</td>
<td>N/A</td>
<td>1.86</td>
<td>9.5</td>
</tr>
<tr>
<td>RMSProp</td>
<td>N/A</td>
<td>1.17</td>
<td>31.0</td>
</tr>
</tbody>
</table>
Differential Privacy: a Formal Privacy Definition

- Outcome is the same with or without Joe’s data
- Resilient to re-identification attacks
- Guarantee parameterized by $\varepsilon$ (the privacy budget)
Challenges for Practical General-purpose Differential Privacy for Data Analytics & Machine Learning

- **Usability** for non-experts
- **Broad support** for analytics & ml workload
- **Easy integration** with existing data environments

**No existing system addresses these issues**
Differential Privacy by Program Rewriting

- **Chorus** automatically rewrites input SQL queries into intrinsically private queries
  - Embeds a differential privacy mechanism in the query
  - Does not require any modifications to database engine or data
  - Works with any standard SQL database
PrivGuard: Privacy-Preserving Shared Analytics & Machine Learning Pipelines

- Privacy policy specification language
- Static analysis to enforce privacy policy on data analytics and ML pipelines
- Use type system Duet to enforce differential privacy guarantees
Real-world Deployment at Uber

• Ongoing deployment for analytics
  – Differential privacy
  – GDPR

• Open-source release:
  https://github.com/uber/sql-differential-privacy
Desired Solutions for Confidentiality/Privacy

Issue 1: Untrusted program
Issue 2: Untrusted infrastructure
Issue 3: Sensitive results
Issue 4: Users have no control & visibility on data usage

Desired Solutions
Program Rewriting & Verification
Secure Computation
Differential Privacy
Distributed Ledger & Smart Contract
Secure Hardware

Remote Attestation

Integrity

Confidentiality
Secure Enclave as a Cornerstone Security Primitive

• Strong security capabilities
  – Authenticate itself (device)
  – Authenticate software
  – Guarantee the integrity and privacy of execution

• Platform for building new security applications
  – Couldn’t be built otherwise for the same practical performance
**Trusted hardware timeline**

**Closed source**
- **ARM TrustZone**: Hardware-based isolation for embedded devices
- **SGX: Software Guard Extensions**: Built in to all Core™ processors (6th-generation and later)
- **NVIDIA**: Trusted Execution Environment
  - Hardware-based isolation
  - TLK: open-source stack for TEE
- **SEV: Secure Encrypted Virtualization**: Introduced in EYPC server processor line
  - Provides confidentiality but not integrity
- **Intel SGX version 2**: In pipeline
  - Drivers already available

**Open source**
- **Keystone**: Open-source secure enclave
  - Collaboration between Berkeley & MIT
  - Remedies issues in previous secure hardware
  - Can be publicly analyzed and verified
  - Can be manufactured by any manufacturer
  - First release: Fall 2018

[Link to Keystone: https://keystone-enclave.github.io](https://keystone-enclave.github.io)
Keystone: an Open Framework for Customizable TEEs

- **Modular and Extensible Design**
  - Extensible functional and security plugins
  - Implement new features without changing core primitive

- **Simple and Clean Abstractions**
  - Core security primitive: hardware-enforced isolation
  - Memory isolation with RISC-V standard PMP

- **First full-stack open source framework for secure enclaves**
  - Support research projects
  - Build an open community

Has been tested on QEMU, FPGA, and SoC

keystone-enclave.org
Machine Learning Workload on Keystone

- Machine Learning (Inferencing in Torch, 9 Models, 2 Datasets)

- Keystone Overhead over Baseline
  - Min -3.12% (LeNet) due to lack of page faults
  - Max 7.35% (DenseNet) due to mmap implementation

- Reduced Initialization Latency with Dynamic Resizing
  - Runtime does not initialize free memory with dynamic resizing
Ginseng, the Learning TEE
FPGA-accelerated, confidential Deep Learning (training & inference)

Key Components
1. FPGA-based Tensor Accelerator [1] (like a TPU but in an FPGA)
2. Tensor Encryption Core (TEC) for protecting tensors in off-chip memory
3. Secure Runtime for orchestrating ML pipeline

Performance (as evaluated on ResNet-34, DCGAN, & MobileNet)
• *No extra overhead* relative to unmodified tensor accelerator
• 3x faster inference versus SGX CPU TEE [2] *using a 10x more expensive CPU than FPGA*

A Platform for Secure, Privacy-Preserving Shared Learning

Data owner not required to trust pipeline or infrastructure
Our Approach: Secure Data Governance Platform

• Auditable; third-party verifiable
  • Combining with distributed ledger technology

• Gives user control and transparency

• Strong privacy guarantees

• GDPR compliant
  • Support rights-to-be-forgotten
  • Data portability & interoperability

• Support computation on data from different sources (data owners)

• Tracking data rights
Empower organization to use data in a secure, compliant, and responsible way.
AI enables new security capabilities

Security enables better AI

**Integrity**: produces intended/correct results (adversarial machine learning)

**Confidentiality/Privacy**: does not leak users’ sensitive data (secure, privacy-preserving machine learning)

Preventing misuse of AI
Future of AI & Security

How to better understand what security means for AI, learning systems?

How to detect when a learning system has been fooled-compromised?

How to build more resilient learning systems with stronger guarantees?

How to build privacy-preserving learning systems?

How to democratize AI?
Security will be one of the biggest challenges in Deploying AI.

It requires community effort.

Let’s tackle the big challenges together!