AI is Not Magic: Machine Learning for Network Security

Lena Pons

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213
Motivation
The Vision

Internet

sensors

ML magic

Alert queue

ML magic

sensors

ML magic

Prioritized, filtered alert queue

Cyber Threat Intelligence

ML magic
The Reality

There is no ML "magic"

This feedback loop is weak at best, and goes stale quickly

Prioritization is highly context specific

Internet

Alert queue

Cyber Threat Intelligence

Prioritized, filtered alert queue

sensors

sensors

sensors

Prioritized, filtered alert queue
AI is Not Magic
Can I Use ML?

Framing questions:

• Can you state your problem as:
  • I would like to use ______ data to predict ______?
• Is it a large scale problem?
• Have you done exploratory analysis on available data?
Problem Specification

Typically we start with an underspecified problem:

For network security we want to predict *malicious activity in a network* using a *combination of network sensor data*

What data are we using? What condition are we looking for?

- Network traffic data
- Host log data
- Indicators of compromise
- Statistically anomalous behavior
Build a Model

Historical Data → Trained model → predictions

Observations

Additional computation,viz, context

output
Build a Model

This is where the majority of the effort will occur.

If the observations do not “look like” the historical data, then predictions will not be reliable.

Sometimes the historical data is not very responsive to the question you want answer.
Large Scale Problem

- Network monitoring data can be in the terabyte and petabyte per month scale
- It contains observations from multiple different sensors that are placed at different locations
  - Sensors all have slightly different, overlapping formats
- Normalizing and tying together data from multiple collections / perspectives can be challenging & time consuming
Data Engineering

Looking for many types of things from many sources of data at many different timescales.
Exploratory Analysis

• Analytic techniques for identifying potentially anomalous network behavior are relatively well developed

• However, specific configurations, baseline assumptions, critical assets vary by network

• Exploratory analysis is required to ensure that an ML approach to network analysis is applicable

ML applications are higher impact where a task can be repeated and extended
Should I Use ML?

Framing questions:

- Can I apply the same test repeatably?
  Yes, but I have to apply many tests in parallel

- Do I have historical and / or ground truth data?
  Historical data, yes, but almost never labels or ground truth

- Can I validate the output of the model?
  Yes, but it requires specialized knowledge
ML for Network Security
State of Data Science for Network Security

Lots of products are selling ML/AI for network security

Under the hood many of these products are still narrow and don’t advance much beyond a signature
Problems – Model Fragility

What’s a model?

A conditional mean

If the baseline changes, then the detection criteria changes
Problems - Extensibility

Detection criteria typically work for some model on some network and if you change the network, then you need to train a new model.
State of Data Science – Detection Criteria

Only some fraction of the sensor data is germane to this model

Baked into the model is some detection criteria that is dependent on the network
Detect and Categorize Threats

Useful information to establish whether some observed behavior is bad exists in being able to “see” across

Sensors frequently work as a “stack” because different sensors are configured to look for different things
Even if you have the capability to place sensors inside and outside the network, tying the data together is frequently a challenging engineering effort.
The missing piece of ML for intrusion detection is **actionability**. Threat intelligence tells the analyst why they care.
Problems – Threat Intelligence

The desire to integrate threat intelligence into the detection process is to help automate the decision process, but the information goes stale quickly.
Anomaly Detection
Anomaly Based Methods

- Anomaly detection is based on the assumption that unusual traffic is “bad” and that typical traffic is “good”
What’s unusual?

First problem: Network defenders often don’t know what’s typical traffic on the network

Much of what ML for network security boils down to is constructing baselines for traffic
What’s expected?

Second problem: Network defenders often do not know if baseline traffic is consistent with desired behavior

The baseline can not be used for anomaly detection if it contains unknown malicious traffic
What’s malicious?

Third problem: Anomalous traffic is not always malicious and malicious traffic is not always anomalous

Even if the network defender knows what is anomalous, he may not know if some specific anomalous traffic is malicious
Longer Time Horizon Detection
Recall

Internet

Cyber Threat Intelligence

Prioritized, filtered alert queue

ML magic

Alert queue
Advancing Cyber Threat Hunting

Detection
find something new
find it earlier, and anticipate
find it faster
find it with less human effort
find combinations of indicators

Data Collection
collect the right data
share it
integrate context (right visibility)
triage data (store for reuse)
adapt based on detection/context

Chaining & Integrated Models
Adaptive feedback
collection ⇔ detection

Abstractions of real world
TTPs ⇔ data
How Can We Use ML

Detection
find something **new**
find it **earlier**, and **anticipate**
find it **faster**
find it with **less human effort**
find **combinations** of indicators
Defending Networks

Much of the current practice operates on a diagnose & treat model

Events are handled on an individual basis and patterns are hard to detect
Deriving Actionable Information From Text

Common representations & ontologies allow for abstracting observations from action

Having a knowledge representation that is broad enough to make high level connections & deep enough to resolve information is foundational for longitudinal analysis
Abstracting Threat Intelligence

- Rapidly updating hierarchical representations of observable types
- Constructing heuristic rules about combinations of observations
- Forming hypotheses about attack mechanisms

Hierarchies allow you to map observations back to a higher level
How Can We Use ML

- **Background / baseline data**
- **Cyber threat intelligence**
- **Sensors**
- **Data triage model**
- **Adaptive sensing**
- **Alert**
- **Secondary detection**

**Collect the right data**

**Triage data**

**Share it**
AI is Not Magic
What We Can Improve with ML

• Looking at increasingly larger volumes & time windows of data
• Graph methods for proximity to suspected bads
• Learning abstractions to improve shareability of CTI and longevity of CTI
What We Can not Improve with ML

• No amount of ML will make up for certain types of missing data
• Unknown unknowns will continue to be a challenge
• “Adversarial perspective” / smart hypothesis generation