Our Vision:

ML4Nets as a Community Effort

Walter Willinger

NIKSUN, Inc.

wwillinger@niksun.com

Scenario

- Consider an advanced persistent threat (APT) scenario faced by two networks A and B.
    - A and B are differently configured, use different vendors' products, etc.
    - Combined telemetry collected from both networks is substantially richer than the information that each network can observe separately

- Two researchers (R^A from A and R^B from B) want to collaboratively develop a ML4Nets solution to detect APTs
    - The researchers cannot exchange trained models or private datasets due to privacy concerns
    - The researchers can share domain knowledge (e.g., which features they find to be critical for detecting an APT)

Step-wise Approach (Steps 1 and 2)

- **Step 1:** Assuming R^A leads this effort, with the help of A's network operators, R^A will collect and label network data from network A that closely matches A's production traffic.

- **Step 2:** R^A applies the closed-loop ML pipeline to train an effective model that has seemingly no underspecification issues and is therefore a prime candidate for being generalizable. However, because of R^A's inability to use its own, let alone B's production network, for model training and testing, it is not known whether the model trained this way will indeed generalize (i.e., perform as expected in either of the two production networks).

Step-wise Approach (Steps 3 and 4)

- **Step 3:** To check if the trained model generalizes, R^A wants to examine the trained model's efficacy on network B's data and requests that R^B and B's operator generate data similar to Step 1. However, due to privacy concerns, neither R^A's trained model can be directly sent to R^B nor can the data from B be sent directly to A.

- **Step 4:** R^B informs R^A how to generate privacy-preserving synthetic data that closely matches the traffic that R^B and B's operator collected from B's network in Step 3, mimics network B's production traffic, and can be used by R^A to evaluate A's model and check whether or not it generalizes.

Enabler for Scaling this Approach: New Community-Wide Infrastructure

- Facilitates flexible and high-quality data generation and collection efforts that are easy to replicate in different networks
- Provides an innovative framework for collaborative and privacy-preserving knowledge sharing
- Informs the foundational design of generalizable learning models for networking problems
- Establishes practical roadmaps for deploying ML4Nets solutions in production networks both within and across the networks of the different participating networks.

Some Unresolved Issues

- How powerful (or limited) is the use of the DT that Trustee generates to explain the decision-making of a given black-box model?
- Are there examples where a Trustee's extracted DT teaches the domain experts new decision-making strategies?
- How many different types of model underspecifications are there (beyond shortcut learning, vulnerabilities to ood samples, spurious correlation), and are there principled approaches to detecting (and subsequently mitigating) them?

Takeaway I:
Less Explorimentation, More Science

- The standard ML pipeline
  - Akin to "explorimentation" (Forde and Paganini, 2019), *the practice of poking around to see what happens*
  - Obfuscates training data and provides no understanding of how trained model work, why they work, or when they don't work (and why not)
  - Contributes to a "dumbing down of networking research" …
- The closed-loop ML pipeline
  - Employs tools of the scientific method
  - Treats training data as 1st-class citizen and provides a basic understanding of how trained models, why they work, or when they don't work (and why not)
  - Contributes to an "opening up of networking research" …

Takeaway II:
Less Hubris, More Humility

- The standard ML pipeline
  - Emphasis is on ensuring “effect” rather than understanding “cause”
  - Extols the virtues and “magic” of black-box models
  - Obfuscates the critical role that the utilized data play in training the models
- The closed-loop pipeline
  - Emphasis is on understanding “cause” rather than ensuring “effect”
  - Suggests an evolving but natural “division of labor” between machines and humans
    - Use the algorithmic power of ML to let machines do the grunt or ”dirty” work
    - Use the intelligence of humans to identify and mitigate underspecification issues
  - Enables closing the gap between training data that we **can** collect vs training data that we **would like** to collect (i.e., real-world production traffic from third-party networks)

References

- W. Willinger, A. Gupta, A. Jacobs, R. Beltiukov, R. Ferreira, and L. Granville. *A NetAI Manifesto (Part I): Less Explorimentation, More Science. In:* ACM Sigmetrics Performance Evaluation Review, Sept. 2023 (to appear)

- W. Willinger, A. Gupta, R. Beltiukov, and W. Guo. A NetAI Manifesto (Part II): Less Hubris, More Humility. In: ACM Sigmetrics Performance Evaluation Review, Sept. 2023 (to appear)