RUHR UNIVERSITÄT BOCHUM RUB

Master Thesis Topic

Automated Repair of Neural Networks under Data Poisoning Attacks

Motivation and Background:
While ML components are integrated into modern applications nowadays, we also need to ensure the quality of these ML models. Automated testing [1] and automated repair [2] techniques have a pivotal role in maintaining these components since re-training is often too expensive. Models like neural networks (NN) have deficiencies that can lead to the generation of low-quality, incorrect, or insecure code [3]. This thesis project aims to develop a technique to repair a poisoned model with realistic (minimal) assumptions. I.e., instead of relying on a test suite or training/test data that can be used to infer correctness constraints [4] or to re-train the model, we only assume access to the model and the observation of one poisoned input that is misclassified. This setup is similar to security repair in conventional software, where we only have a small number of failing test cases and no comprehensive test suite.

Student Task and Responsibilities:

- Make yourself familiar with the state-of-the-art in ML repair.
- Systematically explore techniques to repair poisoned, feedforward NNs under minimal assumptions on the existing test data.
- Design/select evaluation metrics and conduct a thorough evaluation of your approach on poisoned models, e.g., based on MNIST and CIFAR.
- Analyze the results and document your findings.

Deliverables:

- Concept of repairing poisoned ML models under realistic assumptions.
- Prototypical implementation of your concept.
- Evaluation artifacts (dataset, tools, etc.)
- Documented findings of the conducted experiments

Pre-Requisites: (Programming Languages, OS, Skills, Papers, etc)
Strong knowledge in Java, neural networks, and data poisoning attacks is helpful for this project.

[1] G. Fraser and A. Arcuri, “EvoSuite: Automatic Test Suite Generation for Object-Oriented Software,” in *ESEC/FSE’11*. New York, NY, USA: ACM, 2011, pp. 416–419. https://doi.org/10.1145/2025113.2025179

[2] C. Le Goues, M. Pradel, A. Roychoudhury and S. Chandra, "Automatic Program Repair," in *IEEE Software*, vol. 38, no. 4, pp. 22-27, July-Aug. 2021. https://doi.org/10.1109/MS.2021.3072577

[3] H. Pearce, B. Ahmad, B. Tan, B. Dolan-Gavitt, and R. Karri, “Asleep at the Keyboard? Assessing the Security of GitHub Copilot’s Code Contributions,” in *SP'22*, San Francisco, CA, USA: IEEE, May 2022. https://doi.org/10.1109/SP46214.2022.9833571

[4] M. Usman, D. Gopinath, Y. Sun, Y. Noller, and C. S. Păsăreanu, “NNrepair: Constraint-Based Repair of Neural Network Classifiers,” in CAV'21. https://doi.org/10.1007/978-3-030-81685-8_1.

Contacts
Prof. Dr. Yannic Noller (sq-office@rub.de)
Software Quality group, Faculty of Computer Science, Ruhr University of Bochum