Hunting Software Bugs Using Machine Learning

The growing pervasiveness and complexity of software stands to expose masses of users to software failures and cyber-attacks. Existing automated approaches that are commonly used to find software bugs are based on logical reasoning. While logical reasoning provides important benefits such as correctness guarantees and human interpretability, however, it lacks the ability to handle uncertainty and noise arising from imprecise specifications, missing program parts, and imperfect environment models.

This talk presents a new methodology to overcome this limitation via machine learning. The methodology seamlessly incorporates probabilistic reasoning into existing approaches that are based on logical reasoning. It thereby achieves the best of both worlds without suffering the limitations of either: the probabilistic part enables to leverage a rich collection of beliefs learnt over time, while the logical part enables to leverage a rich literature of program reasoning rules to make accurate predictions by propagating the beliefs. I will demonstrate how the resulting approach enables to discover deep semantic bugs in widely-used Java programs comprising a million lines of code at a fraction of the human effort required by past approaches.