Guided, Stochastic Model-Based GUI Testing of Android Apps

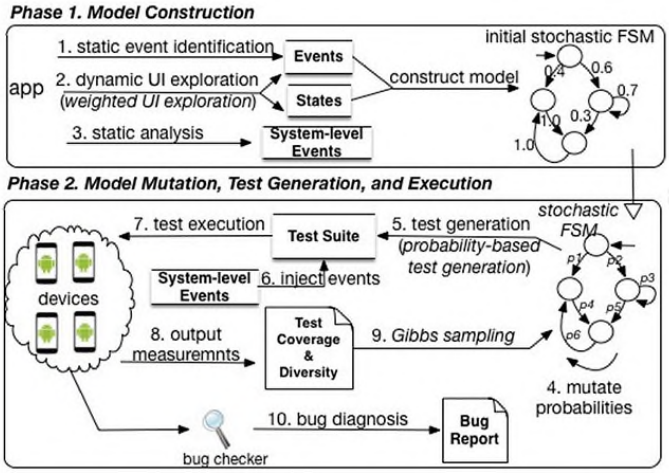
Background

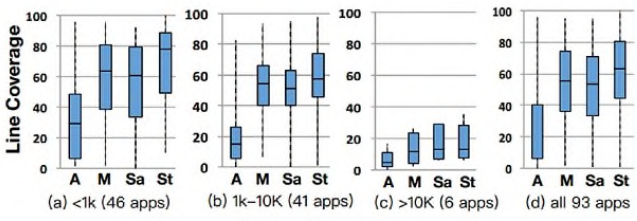
Mobile apps have become ubiquitous and drastically increased in number over the recent years. However, it is challenging to guarantee their quality. First, they are event-centric programs with rich graphical user interfaces (GUIs), and interact with complex environments (e.g., users, devices, and other apps). Second, they are typically developed under the time-to-market pressure, thus may be inadequately tested before releases. When performing testing, developers tend to exercise those functionalities or usage scenarios that they believe to be important, but may miss bugs that their designed tests fail to expose.

Project Description

Stoat (**STO**chastic model **A**pp **T**ester) is a novel guided approach to perform stochastic model-based testing on Android Apps. The idea is to thoroughly test the functionalities of an app from its GUI model, and validate the app's behavior by enforcing various user/system interactions.

Stoat operates in a unique two-phase process to test an app: (1) construct a stochastic app model (in the form of stochastic finite state machine); and (2) iteratively mutate/refine the stochastic model (by perturbing the probability values of the model transitions), and guide test generation toward achieving high code, model coverage as well as exhibiting diverse event sequences.



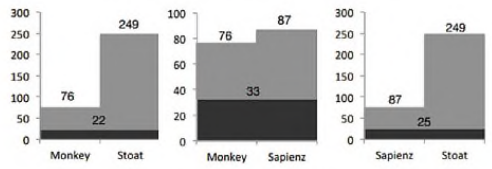


Code Coverage

Experiment Results

Stoat was evaluated on 93 open-source apps. The results show (1) the models produced by Stoat cover 17~31% more code than those by existing modeling tools; (2) Stoat detects 3X more unique crashes than two state-of-the-art testing tools, Monkey and Sapienz. Furthermore, Stoat tested 1661 most popular Google Play apps, and detected 2110 previously unknown and unique crashes.

Tool	#Buggy Apps	#Unique Crashes
A^{3}E	8	8
Monkey	40	76
Sapienz	43	87
Stoat	68	249



Fault Detection

Meng Guozhu and Su Ting
Cyber Security Lab, SCSE, NTU
GZMeng@ntu.edu.sg