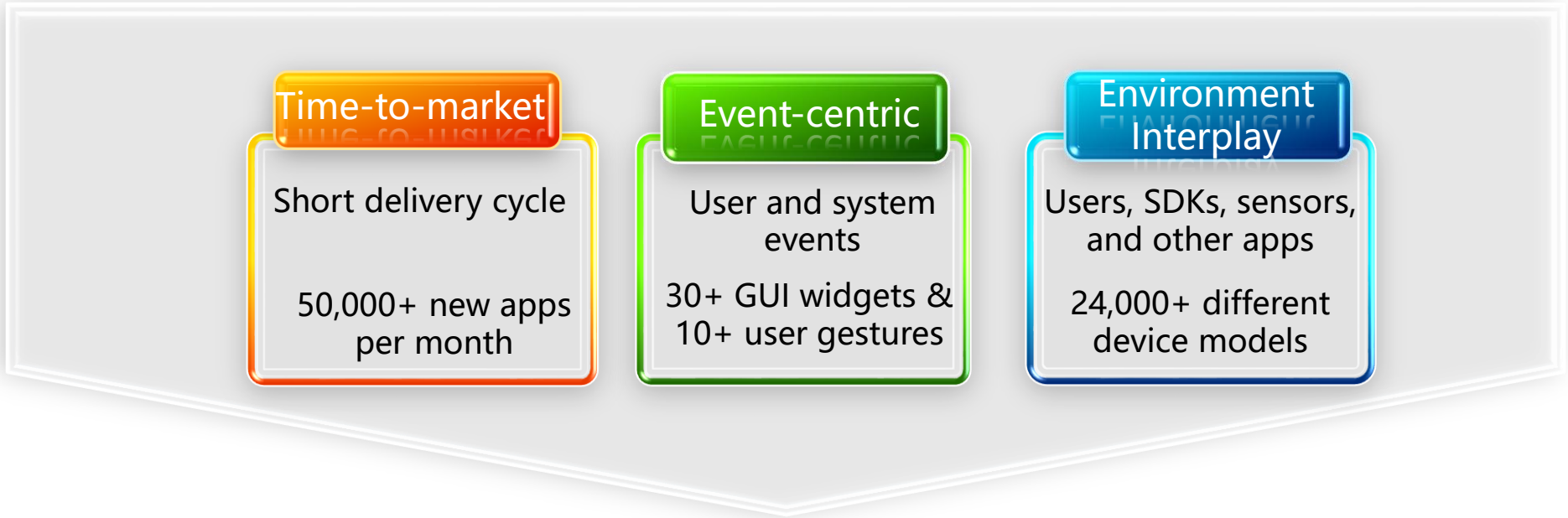
Stoat: Guided, Stochastic Model-based GUI Testing of Android Apps

Ting Su

Research Fellow@NTU

Mobile Apps (Android)



Ensuring app quality is *challenging*

Existing Mobile App Testing Techniques

Approach	Tools
Random Testing/Fuzzing	Google Monkey, WCTester[FSE'16-ind] Dynodroid[FSE'13]
Symbolic Execution	ACTeve[FSE'12], JPF-Android[SSEN'12]
Evolutionary (Genetic) Algorithm	Evodroid[FSE'14] , Sapienz[ISSTA'16]
Model-based Testing (MBT)	GUIRipper[ASE'12], ORBIT[FASE'13], A3E[OOPSLA'13], SwiftHand[OOPSLA'13], PUMA[MobiSys'14], MobiGuitar[IEEE Software'15], AMOLA [ASE'16], DroidBot [ICSE'17 –tool]
Other Approaches	MonkeyLab[MSR'15], CrashScope[ICST'16], TrimDroid[ICSE'16], EHBDroid [ASE'17]

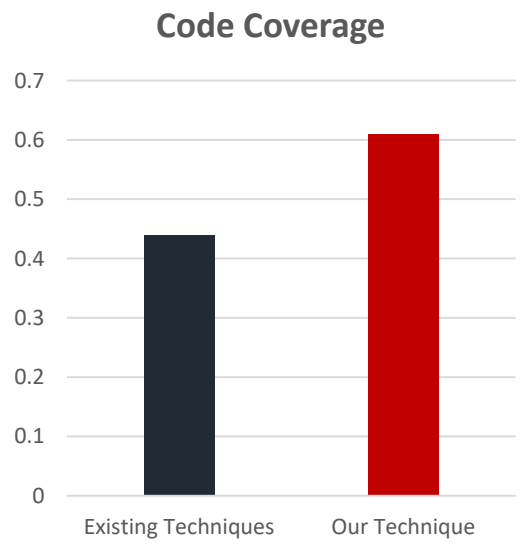
Our Approach --- Stoat

- Stoat (**Sto**chastic model **A**pp **T**ester)
  - *A guided*, *stochastic model-based* GUI testing approach
  - A *fully-automatic* tool for testing/fuzzing Android apps

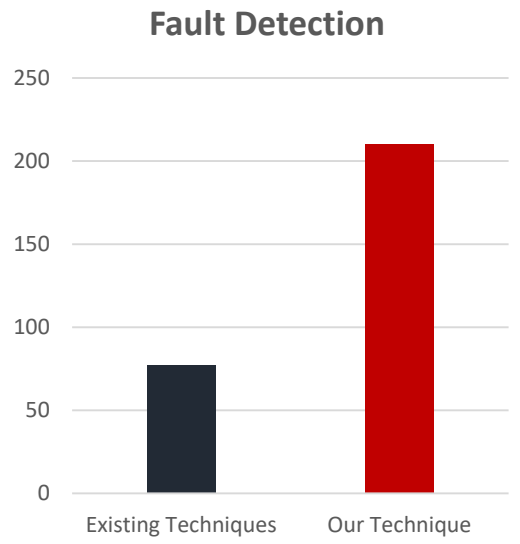
- Given an app as input,
  1. Model Construction
     - Use *dynamic/static analysis* to learn a stochastic model
  2. Test Generation and Optimization
     - Adopt *Gibbs sampling* to iteratively mutate/refine the model
     - Validate apps with various *user/system-level* events

Evaluation & Effectiveness

- Subjects
  - 93 benchmark apps 
- Outperform existing techniques



17~31% higher code coverage



3X more unique crashes

Evaluation & Effectiveness

- Subjects
  - 93 benchmark apps
  - 1661 Google Play apps

- Contribute to real-world apps

ID	Exception Type	Number
1	NullPointerException	1226
2	Windows Leaked Exception	255
3	ActivityNotFoundException	191
4	SQLite Related Exception	71
5	IllegalStateException	47
6	IllegalArgumentException	37
7	RuntimeException	21
8	ClassCastException	9
9	UnsatisfiedLinkError	8
10	WindowManager$BadTokenException	4
11	Other Exceptions	233

2110 unique previously-unknown crashes from 691 apps

1 bug 2 bugs 1 bug ...

Contribute to the apps with *billions of users*

Evaluation & Effectiveness

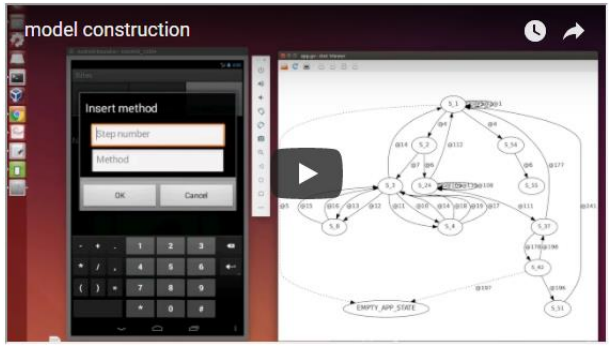
- Subjects
  - 93 benchmark apps 
  - 1661 Google Play apps 
  - 2104 F-droid apps (total 4560 versions) 

- Effective bug detection
  - Detected 3535 unique app crashes
  - Categorized into 75 types of errors

Technical Innovation

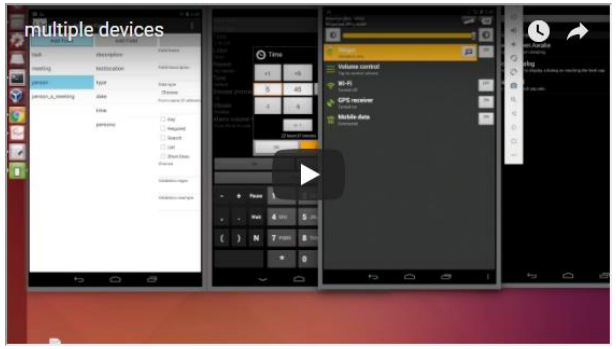
Key Technique

- Learn a behaviour model for an app
- Sample tests to optimize test generation
- Enforce various user/system interactions



Model learning

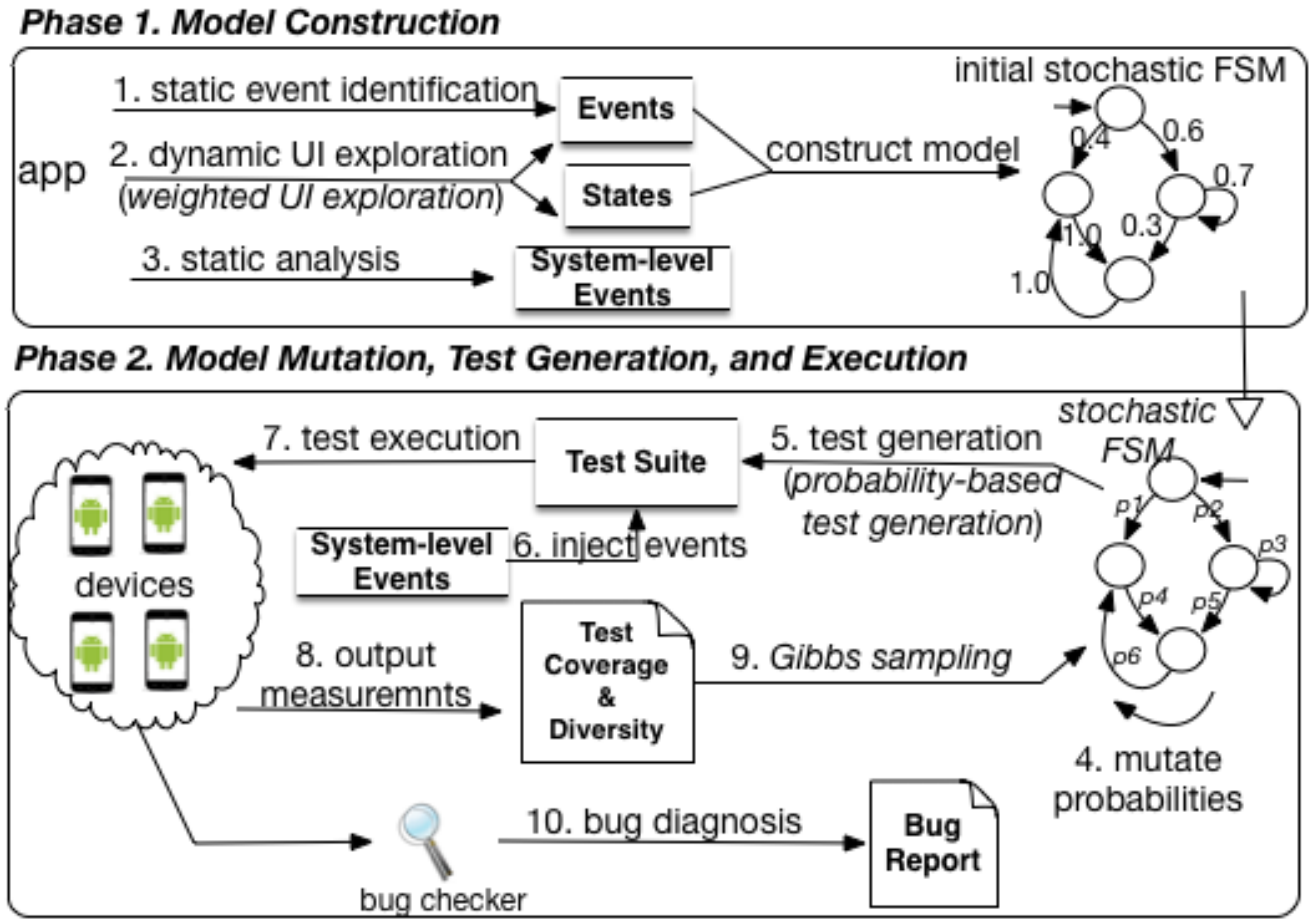
https://youtu.be/v4UkJgdcWDQ
(or https://v.youku.com/v_show/id_XMzA0Nzc4MTcyNA)



Model-based testing

https://youtu.be/Xk7A7wczLj0
(or https://v.youku.com/v_show/id_XMzA0Nzc4NjYxMg)

Workflow of Stoat



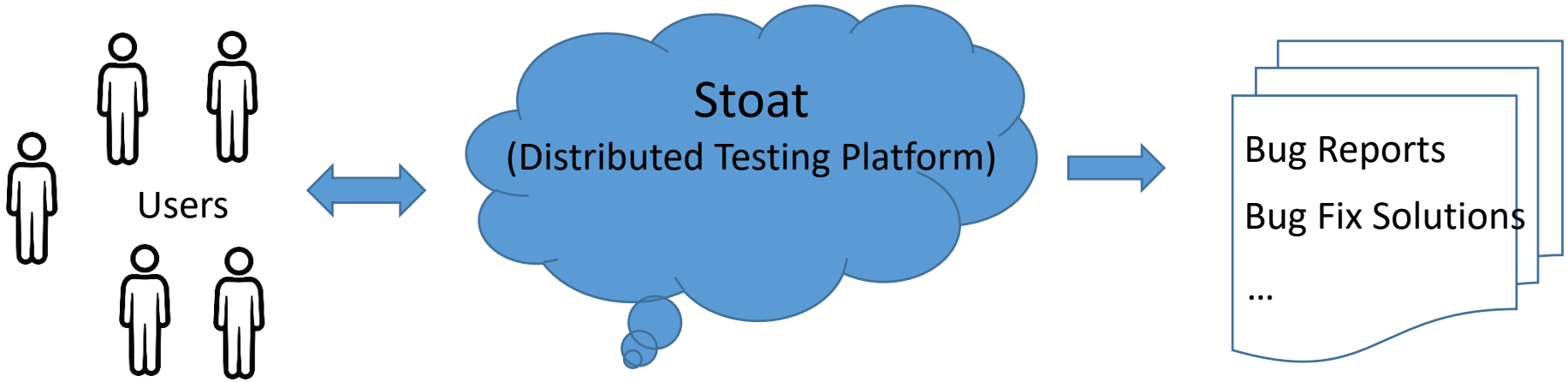
Test Optimization Goal

- ✓ Statement coverage
- ✓ Model coverage
- ✓ Event sequence diversity

- ✓ Inject 113+ user/system events

More details: https://tingsu.github.io/files/nasac2017-stoat.pdf

Features & Usability



✓*End-to-end, server-client* distributed testing;

✓Support *binary* and *open-source* apps on *real devices* and *emulators*

✓Generate model/class/*method*/*line* coverage reports; Bug-triggering tests/screenshots

Summary

- Tool: Stoat (**Sto**chastic model **A**pp **T**ester)
  - A ***Guided, Stochastic*** model-based GUI testing approach
  - Tested 6000+ APKs, detected 5800+ fatal crashes
- Goal
  - Thoroughly test various usage scenarios of an app;
  - Enforce environmental interplay
- Publication
  - Guided, Stochastic Model-Based GUI Testing of Android Apps (ESEC/FSE'17)
  - FSMdroid: Guided GUI Testing of Android Apps (*First Prize of ACM SRC*@ICSE 2016)
  - Large-Scale Analysis of Framework-Specific Exceptions in Android Apps (*ACM SIGSOFT Distinguished Paper Award*@ICSE 2018)
  - Efficiently Manifesting Asynchronous Programming Errors in Android Apps (ASE 2018)
- https://tingsu.github.io/files/stoat.html