中国科学院软件研究所学术年会’2020
暨计算机科学国家重点实验室开放周

学术论文

基于启动上下文构造的Android应用多入口探索方法

Multiple-Entry Testing of Android Applications by Constructing Activity Launching Contexts
(ICSE 2020: 457-468)

燕季薇 (yanjw@ios.ac.cn)，刘昊，潘临杰，严俊，张健，梁彬
软件工程技术研究开发中心

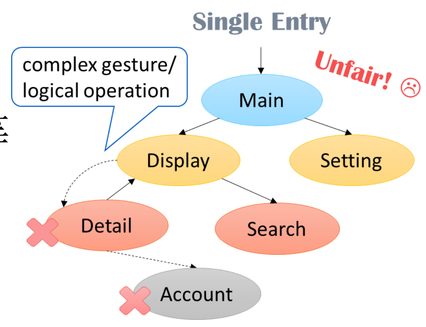
Fax on GitHub

背景介绍

安卓应用测试需要指定测试入口，虽然安卓是多入口的应用程序，但现有的安卓应用测试生成方法大都默认从单一主入口进行探索，这导致了远离主入口的边缘组件很难被覆盖。而且，由于部分组件的访问需要经过多次组件跳转或涉及复杂的用户操作，单个组件访问失败可能导致被该组件支配的其他组件均不能被访问。已有的模糊测试方法对测试上下文的构造不够全面、准确，且没有考虑包含复杂对象的上下文，因此很难对每个组件进行充分测试。本文作者在研究中，提出了一种基于组件启动上下文构造的多入口测试方法 *Fax* 来解决这些问题。*Fax* 首先通过精确的静态分析构建活动启动模型并生成完整的启动上下文，使用构造的上下文直接启动被测组件，而无需依赖图形界面上事件序列的执行；其次，*Fax* 提出了一个可在探索时为每个启动上下文动态分配探索权重的自适应探索框架，以实现应用的深度、公平探索。

单入口探索

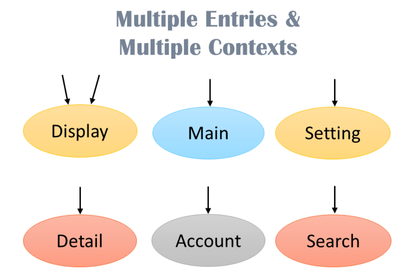
- **组件覆盖不均匀**
  - 每个组件距离主入口距离不等
  - 组件访问受限于其支配组件



- **上下文测试不充分**
  - 难以覆盖完整的上下文环境
  - 动态探索时，启动上下文的覆盖情况难以评估
  - 未考虑不同启动上下文的权重分配

多入口探索

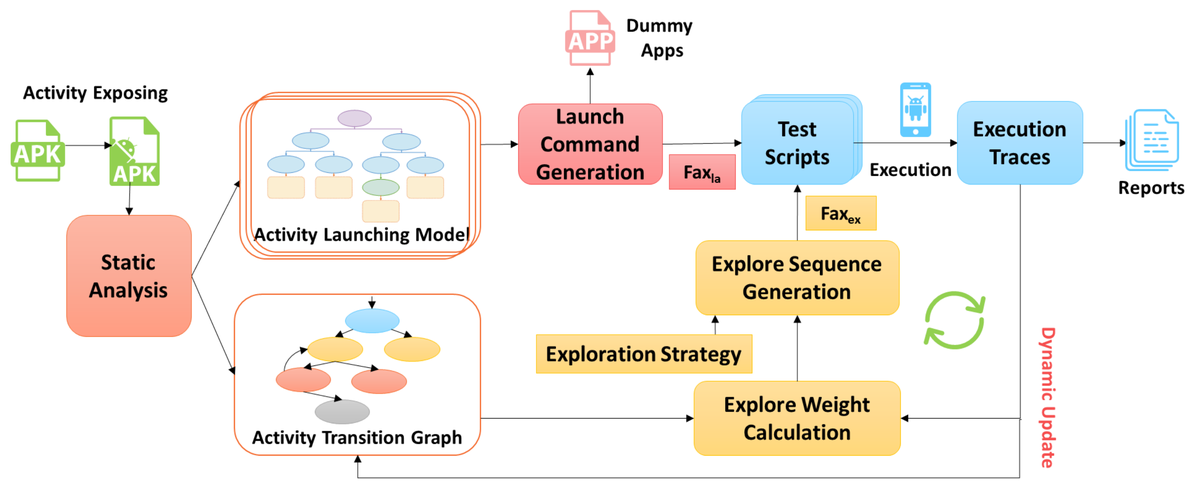
- **组件覆盖均匀化**
  - 内部组件转变为暴露组件
  - 每个组件都将被主动启动



- **上下文测试充分**
  - 根据组件上下文消息接收代码反向构造有效的上下文启动消息
  - 支持复杂的数据类型
  - 自适应分配探索过程中对启动上下文的探索权重

方法框架

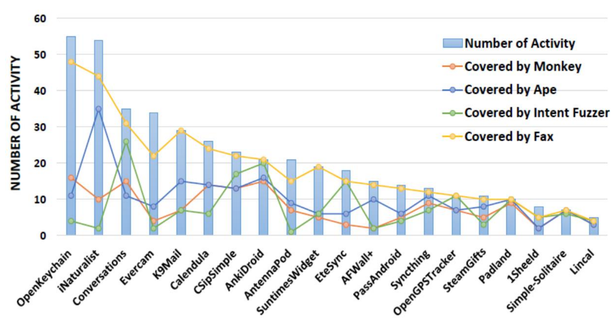
- **预处理模块：**插桩+暴露内部组件
- **静态分析模块：**构造活动启动模型(ALM)和静态活动迁移模型(ATG)
- **Fax启动策略：**根据ALM构造启动上下文并执行
- **Fax探索策略：**根据上下文执行结果和ATG分配探索权重，动态更新ATG结构，动态更新探索权重
- **结果分析模块：**统计探索过程的代码覆盖情况和错误触发情况，生成测试报告



结果展示

- **测试设置：**20个广泛使用的开源应用程序
- **RQ1: Activity启动情况**

Fax: 377/391	APE: 208/391
IntentFuzzer: 158/391	Monkey: 147/391



- **RQ2: 应用探索的有效性**

Fax: 22.51 %	APE: 22.82%
IntentFuzzer: 6.44 %	Monkey: 18.80 %

Name	Target	Entry	Strategy
IntentFuzzer	Intent Fuzzing	Multiple	None
Fax	Both	Multiple	Random
Monkey	GUI Exploration	Single	Random
Ape	GUI Exploration	Single	Model-based

- **RQ3: 唯一崩溃检测的有效性**

Fax: 180/719	APE: 12/12
IntentFuzzer: 18/81	Monkey: 8/8