ISCAS 中国科学院软件研究所学术年会'2023
暨计算机科学国家重点实验室开放周

学术论文

Locating Framework-specific Crashing Faults with Compact and Explainable Candidate Set

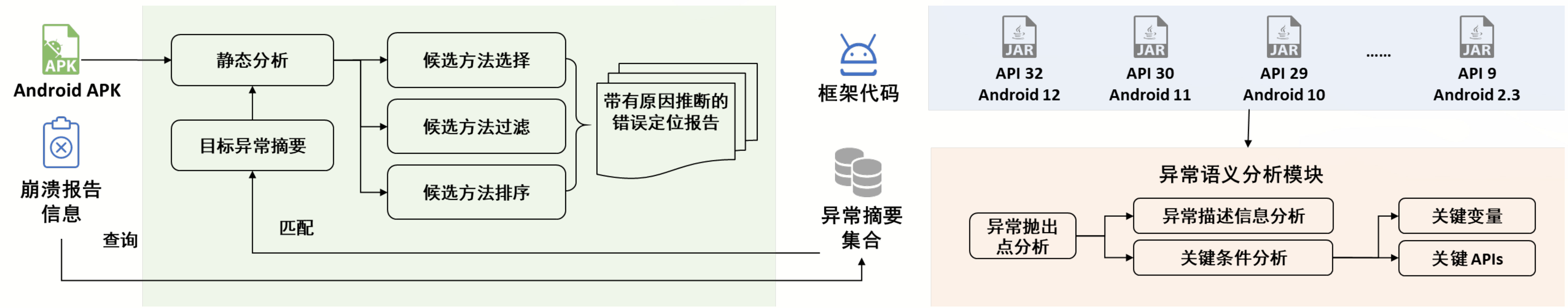
45th International Conference on Software Engineering, ICSE 2023.

燕季薇 (yanjiwei@otcaix.iscas.ac.cn)，王苗苗，刘烨庞，严俊，张龙
软件工程技术研究开发中心

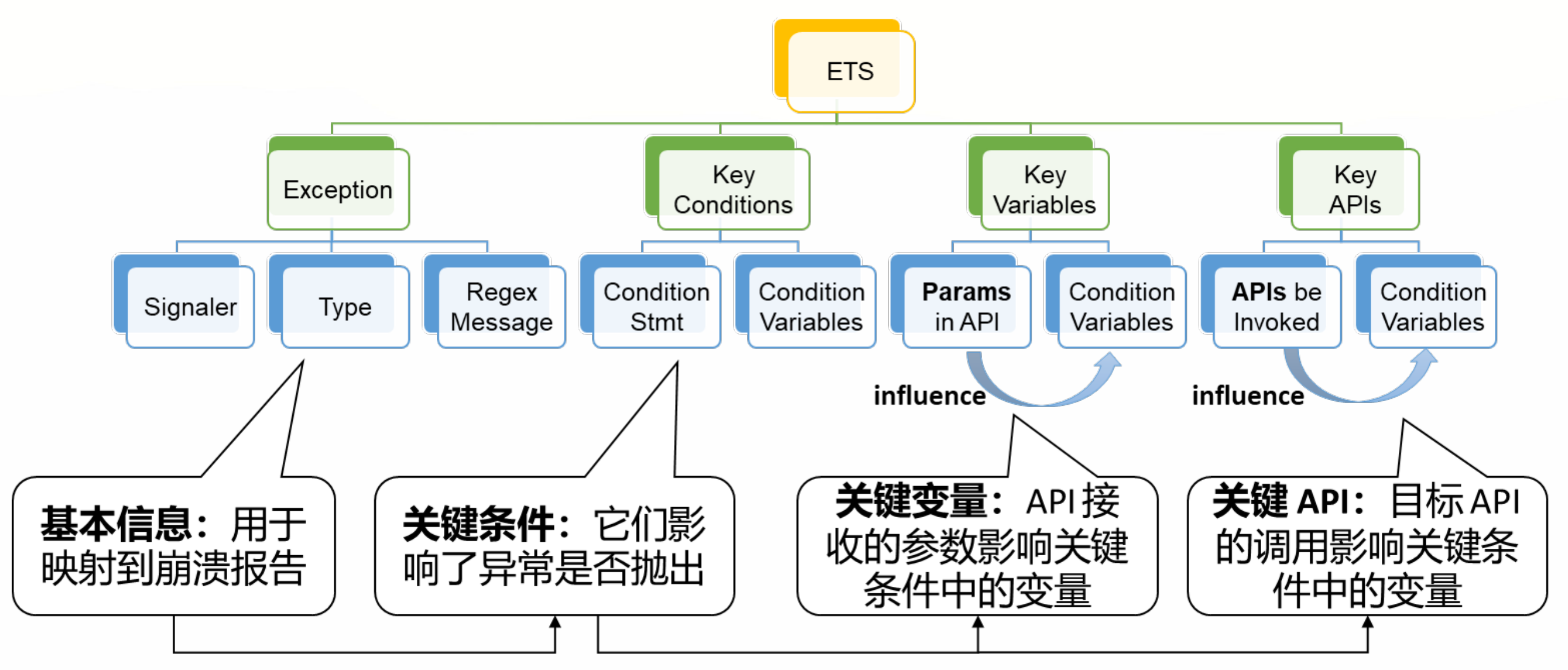
背景介绍

大量应用程序依赖于各种框架或库存在，但对复杂底层代码的误用导致崩溃现象普遍发生。基于用户崩溃报告，已有方法大多从崩溃堆栈路径开始，沿着函数调用图搜寻曾调用过的函数，或构建包含大量崩溃修复记录的数据集用于训练和预测。然而，这些方法受限于调用图的完整性或依赖于历史修复记录。为了实现对第三方框架或库所抛出崩溃的调试，提出了一种基于静态摘要提取和崩溃堆栈信息追踪的错误定位方法。其关键思路是，每个崩溃栈信息应都对应着框架中确定的异常抛出点，对该抛出点代码的语义分析可以帮助找出崩溃触发的根本原因。基于这个想法，本文为所有框架级抛出的异常预先构建可重用的摘要信息，并通过对应用程序代码中的关键变量和 API 进行数据追踪，获得排序后的候选错误函数列表。与相关方法比，CrashTracker在仅生成少量候选集的前提下，有效提高了框架级崩溃错误定位的精度。

方法架构



异常抛出摘要 ETS



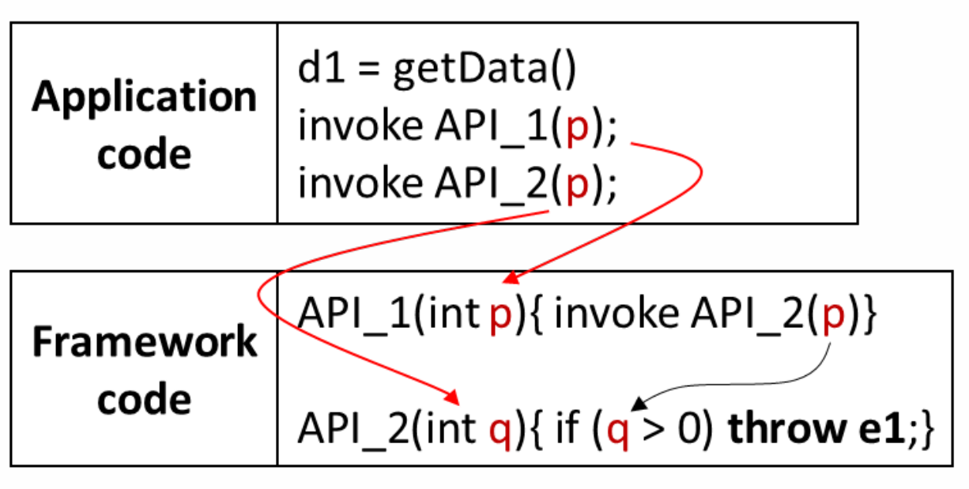
基本信息：用于映射到崩溃报告

关键条件：它们影响了异常是否抛出

关键变量：API 接收的参数影响关键条件中的变量

关键 API：目标 API 的调用影响关键条件中的变量

- **关键变量 (*keyVars*)**
  - 这些参数的取值可能被错误地传入到框架层代码中



Application code

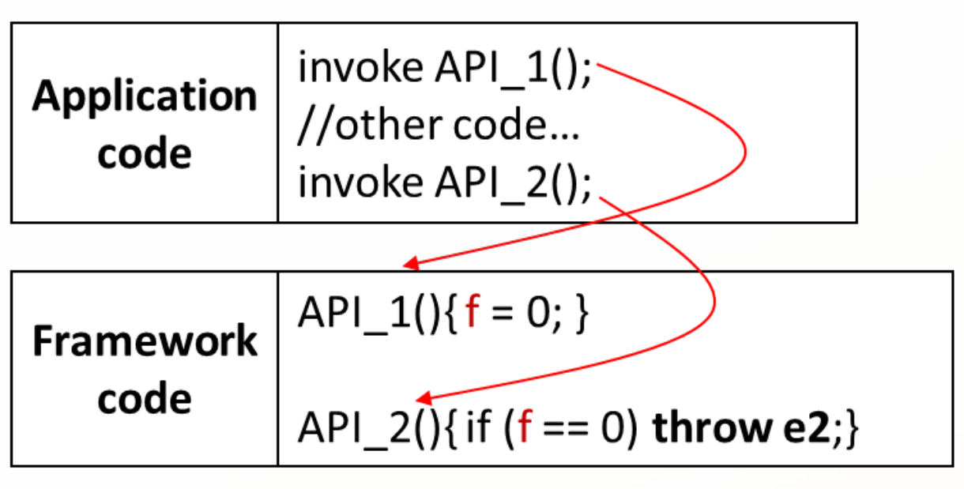
```
d1 = getData()
invoke API_1(p);
invoke API_2(p);
```

Framework code

```
API_1(int p){ invoke API_2(p)}
API_2(int q){ if (q > 0) throw e1;}
```

API_1 和 API_2 的第 1 个参数和异常触发相关

- **关键 API (*keyAPIs*)**
  - 这些 API 的调用会影响全局变量，进而改变了异常抛出前置检查的执行结果



Application code

```
invoke API_1();
//other code...
invoke API_2();
```

Framework code

```
API_1(){f = 0; }
API_2(){if (f == 0) throw e2;}
```

API_1 本身的调用和异常触发相关

应用层错误定位

对于给定的错误报告，先在框架摘要集合中匹配对应的**ETS**信息，在根据**ETS**类型选择应用层错误定位策略。

ETS 类型	错误定位策略
T_1: 没有条件变量 CondVar	S_1: 在子类中进行 Override 分析
T_2: 没有外部变量 OutsideVar	S_2: 对 crashAPI 调用语句中的变量进行数据追踪
T_3: 只有关键变量 keyVar	S_3: 对关键变量中的变量进行数据追踪
T_4: 只有关键API keyAPI	S_4: 对关键 API 的调用方法进行追踪
T_5: 同时有关键变量和关键 API	S_3: 对关键变量中的变量进行数据追踪 + S_4: 对关键 API 的调用方法进行追踪

ETS 类型和相应的错误定位策略

CrashTracker

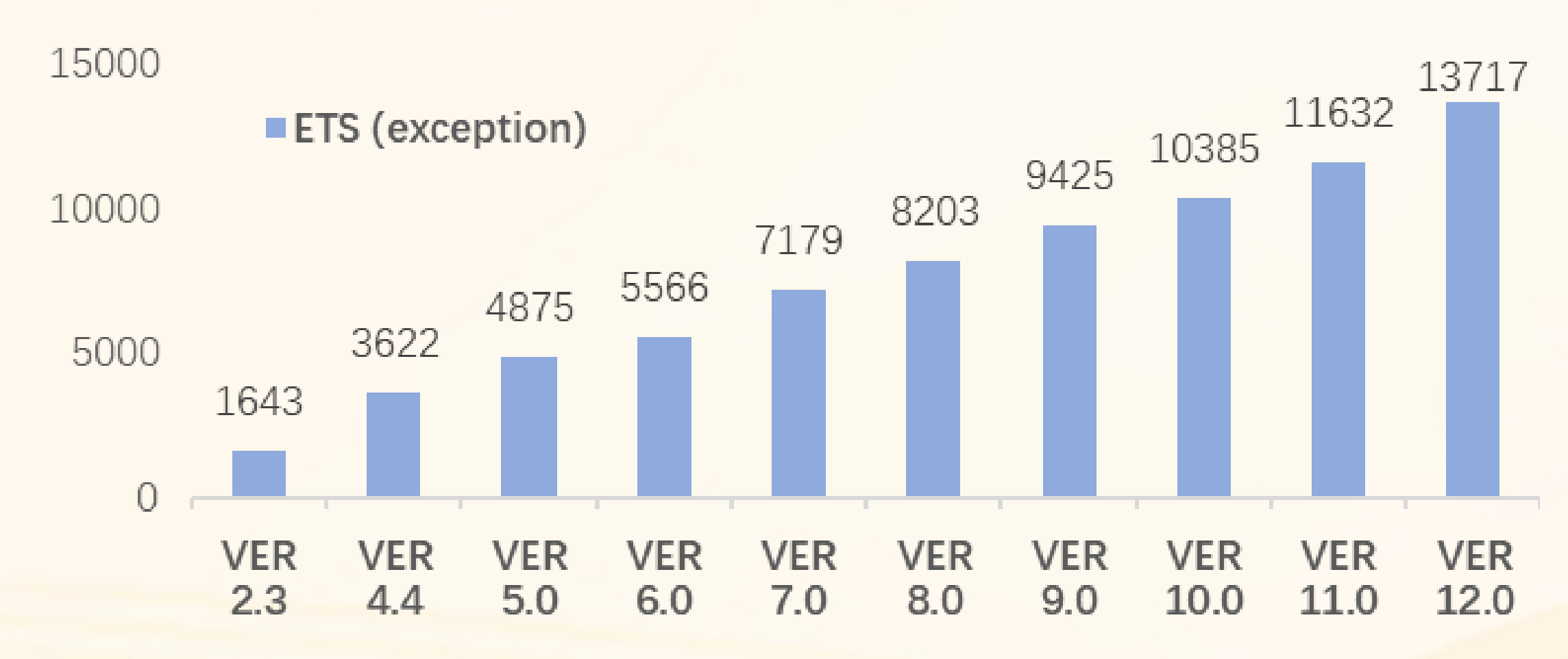
CrashTracker：框架级崩溃错误自动定位原型工具

- •基于广泛使用的静态分析框架 Soot
- •提供框架代码异常摘要构造功能模块
- •提供基于异常摘要的应用错误定位功能模块

工具 GitHub 链接:
https://github.com/hanada31/CrashTracker

CrashTracker

实验评估结果



从各版本框架级代码中提取出大量的 ETS 信息

第1个候选结果为出错函数的数量

错误报告数量	定位成功数量	平均候选方法个数	#R@1	#R@5	#R@10
580	568	6.35	500	562	567

CrashTracker 定位的有效性

共569个崩溃报告，其中出错点不在栈上的56个

相关工作 Anchor，基于学习+分析的方法：

	Anchor			CrashTracker		
错误报告数量	#R@1	#R@5	#R@10	#R@1	#R@5	#R@10
569	463	493	494	493	551	556
56	21	32	33	14	38	43

与相关工作 Anchor 的对比结果

分析效率

- 为10 个版本的安卓框架提取摘要，共计1 小时。
- 对 580 个应用做错误定位，8线程，共计1.5小时。