

Repairing crashes in Android apps

**Shin Hwei
Tan**

**Zhen
Dong**

**Xiang
Gao**

**Abhik
Roychoudhury**



SUSTech
Southern University
of Science and Technology



NUS
National University
of Singapore

Prevalence of Mobile Apps

App Store

Launched with 500 apps.
By July 17, 10 millions
apps downloaded.

Google Play

3.5 millions apps
available for download
in December 2017.



IBM Simon

First device that
features apps like
calendar & calculator.

Android Market

Launched with
50 apps.

Apps Popularity

Total app downloads
number will jump to
352 billion.

- Mobile apps are indispensable
- Android app maintenance is a long-standing problem
- **62% of app users experienced a crash, freeze/error**

Problem: Fixing Crashes in Android Apps

Challenges for repair techniques:

- Source code for Android apps is often unavailable.
- Positive UI tests that encode program functionality are often unavailable.
- Negative UI test that encode the crashing behavior may be flaky.

Challenges for human developers:

- Inadequate understanding of Android development
 - Complexity of Android Activity lifecycle

Problem: Fixing Crashes in Android Apps

Challenges for repair techniques:

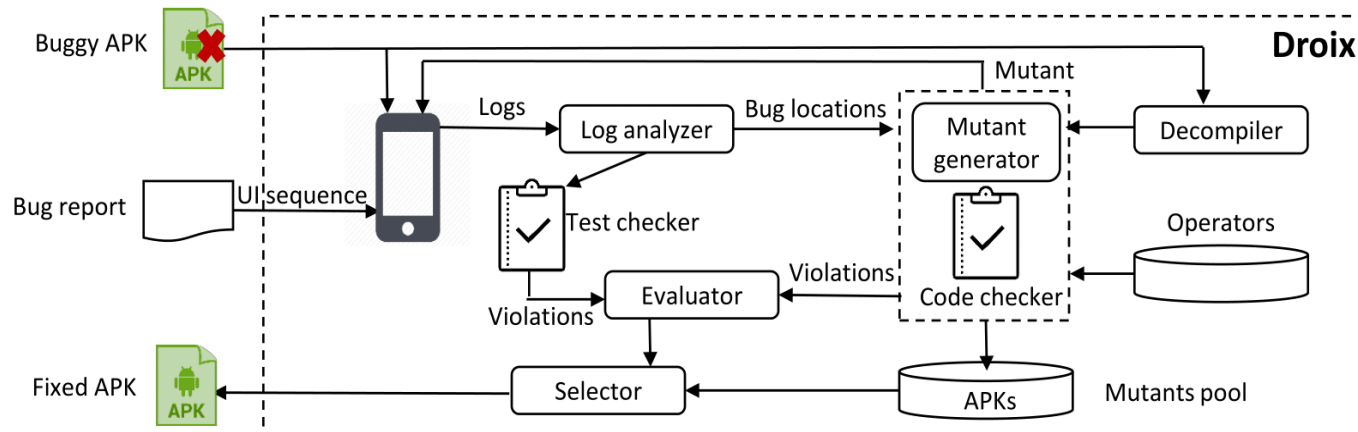
- Source code for Android apps is often unavailable.
- Positive UI tests that encode program functionality are often unavailable.
- Negative UI test that encode the crashing behavior may be flaky.

Challenges for human developers:

- Inadequate understanding of Android development
 - Complexity of Android Activity lifecycle

Automated Repair of Crashes in Android app: Given a buggy app and UI sequence S that leads to a crash, find a app that does not crash on S .

Summary of Droix



- Evolves Android apps using a single UI sequence
- Property violations checker at multiple levels:
 - Pre-execution (Code level)
 - Encode exception handling best practices
 - Static analysis
 - During Execution (Test level)
 - Encode Android Activity/Fragment lifecycle models.
- New lifecycle-aware repair operators for fixing Android apps

(E) The result is equal to or better than the most recent human-created solution to a long-standing problem for which there has been a succession of increasingly better human-created solutions.

- ✓ Long-standing problem
- ✓ A succession of increasingly better human-created solutions
- ✓ Result is equal to or better than human-created solution

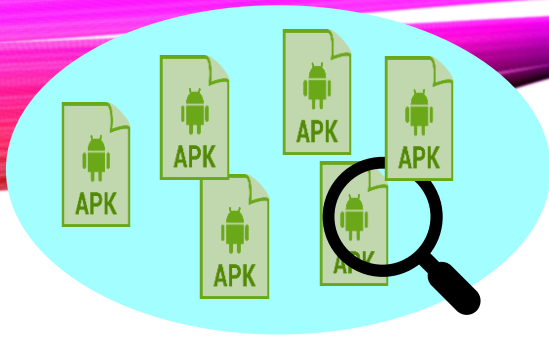


High Commercial Value: Patent In Progress

(A) The result was patented as an invention in the past, is an improvement over a patented invention, or would qualify today as a patentable new invention.

- Collaboration with Singapore Telecommunications Limited (Singtel)
 - Largest mobile network operator in Singapore
- Prior art search report by our patent attorney: “No previous invention offers the same features”





EA for Android apps

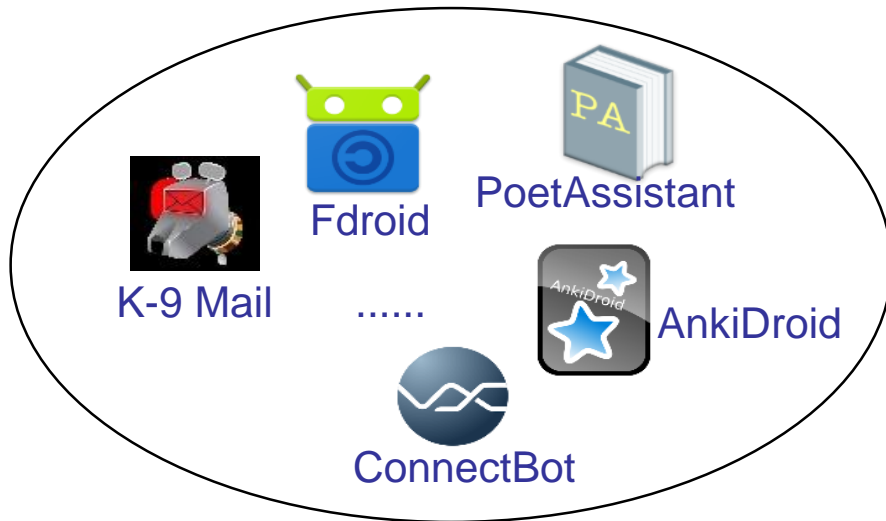
(D) The result is publishable in its own right as a new scientific result independent of the fact that the result was mechanically created

- $(\mu + \lambda)$ Evolutionary Algorithm, $\mu = 40$ and $\lambda = 20$
- Constraints for Code-level & Test-level Properties:
 - **Hard** Constraints: eliminate patches if property is violated
 - **Soft** Constraints: possible property violation
- Fitness Function f :

$f = \text{Minimize \# of violations for soft constraints}$

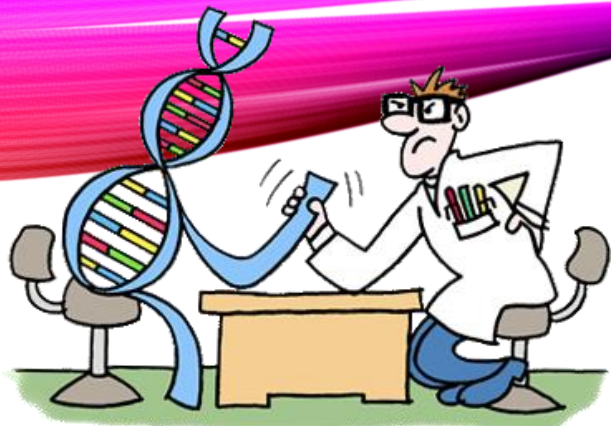
Empirical Evaluation

Benchmark: *Droixbench*

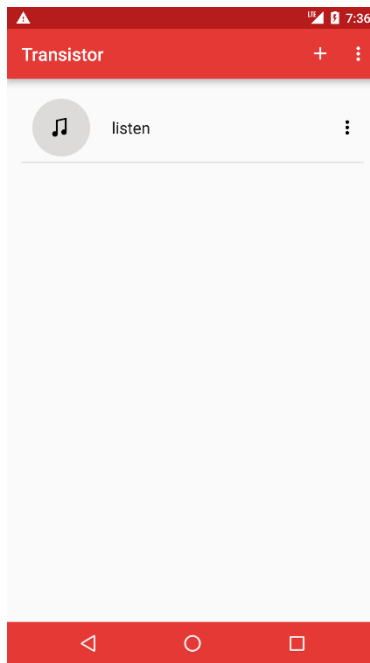


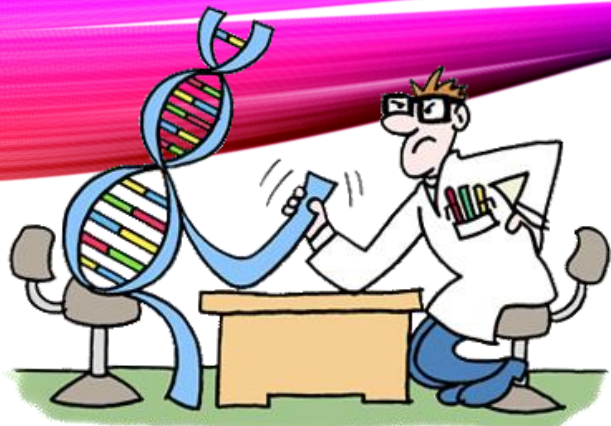
24 crashes in **15** mobile apps

- Fixes 15 crashes: **>50%**
- 12 fixes are of **comparable quality** to human-created fixes
 - *Sometimes better than human-crafted fix!*
- 3 fixes are incorrect and they are complex to fix
 - Potential future improvement



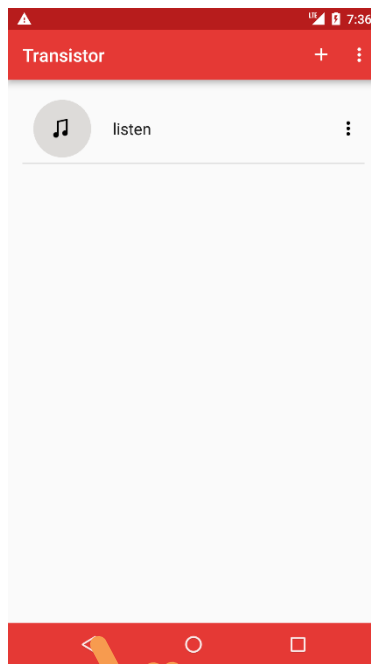
Human Competitive





Patch generated by Droix

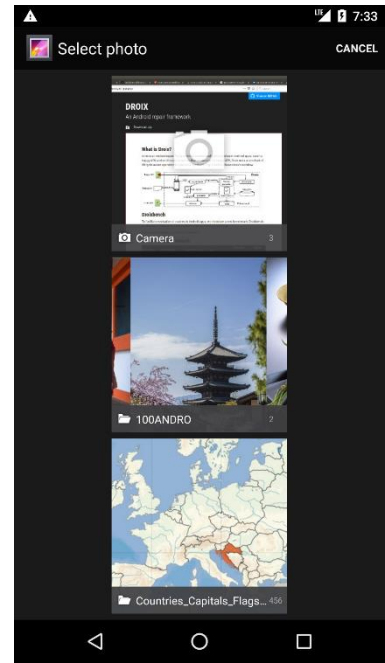
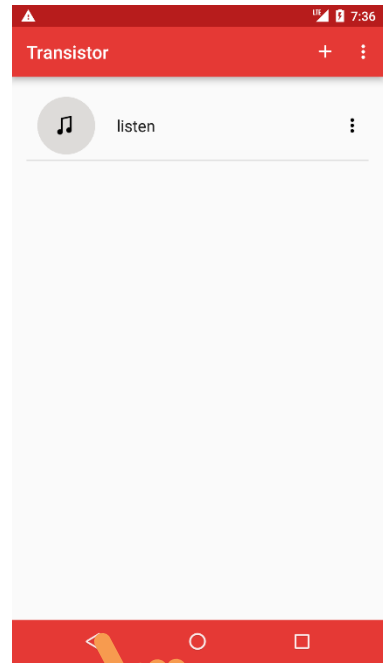
Human Competitive

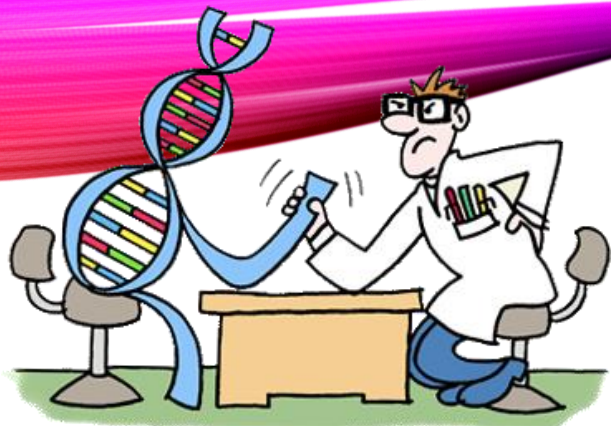




Human Competitive

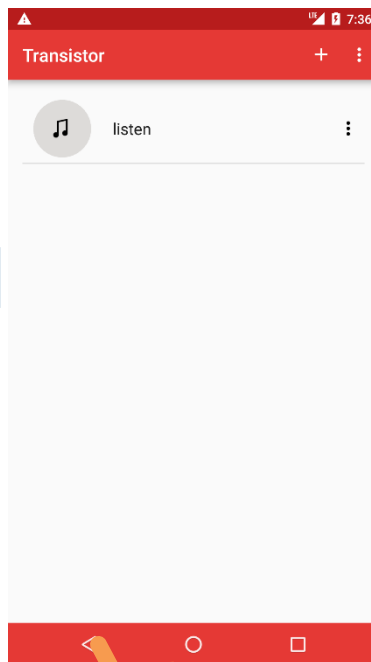
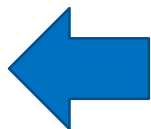
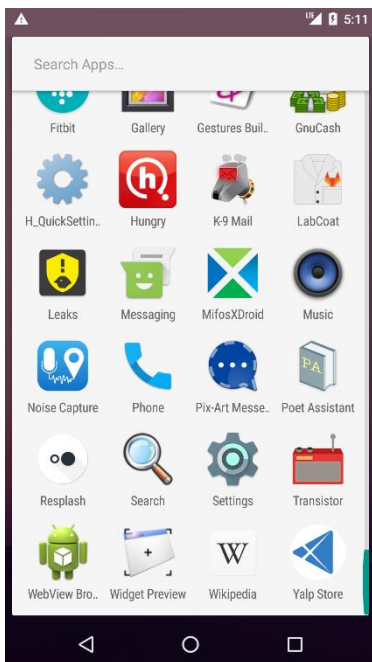
Patch generated by Droix



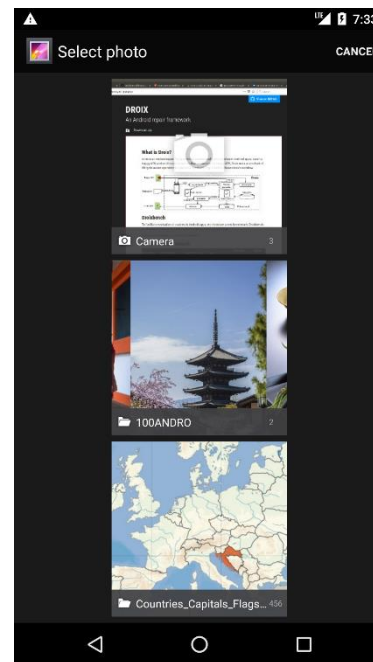


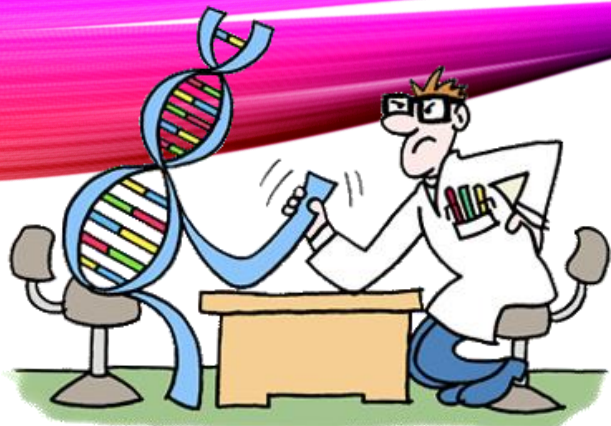
Human Competitive

Patch generated by Droix



Patch crafted by human

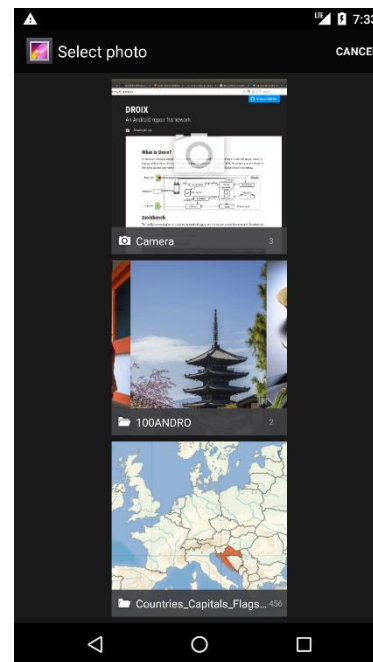
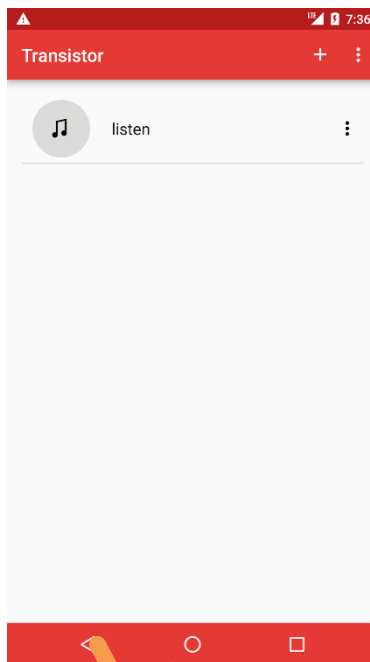
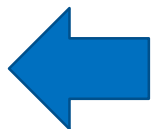
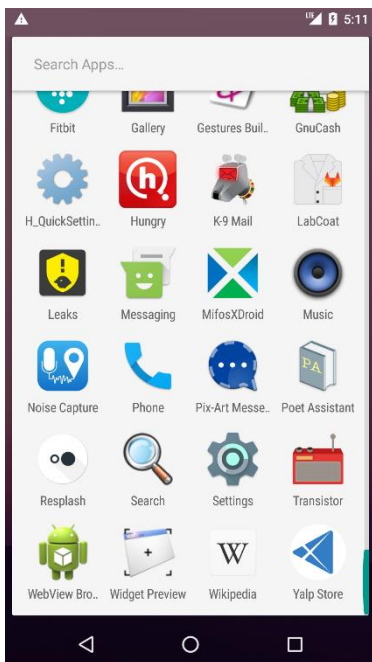




Human Competitive

Patch generated by Droix

Patch crafted by human

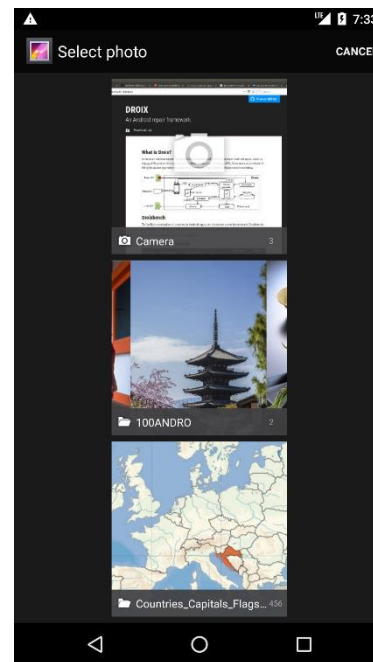
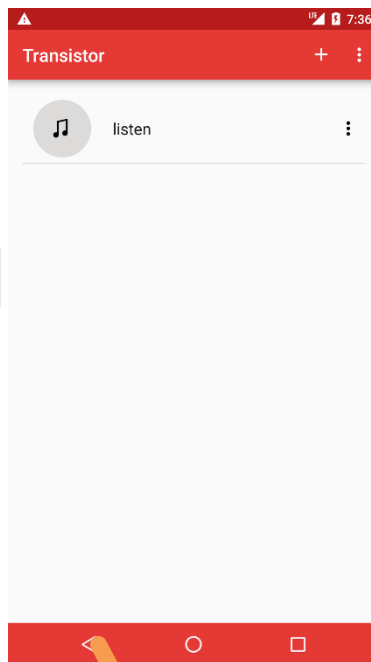
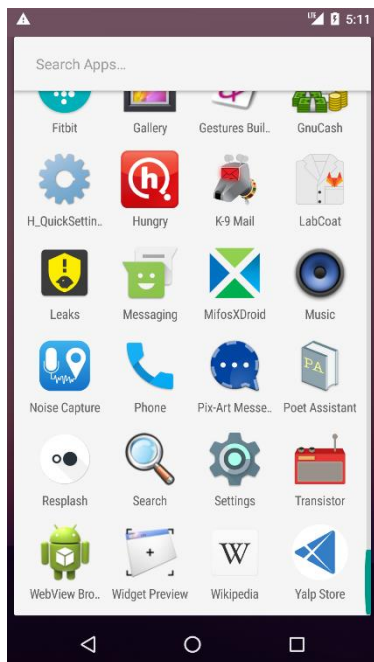




Human Competitive

Patch generated by Droix

Patch crafted by human

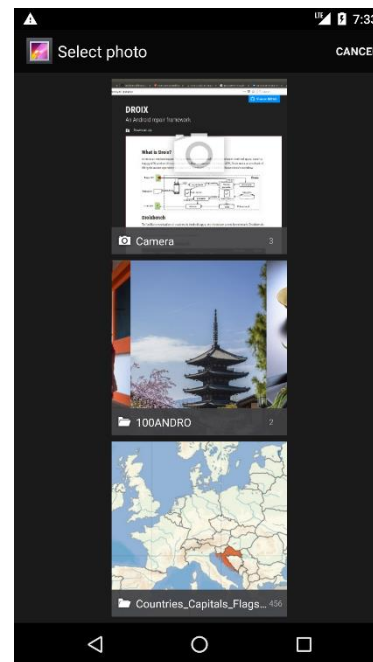
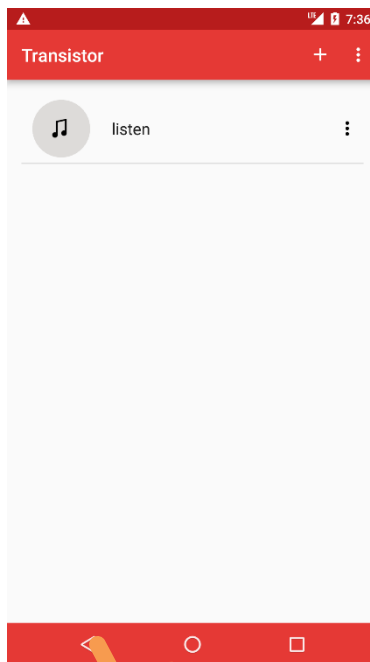
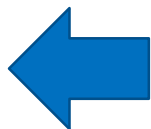
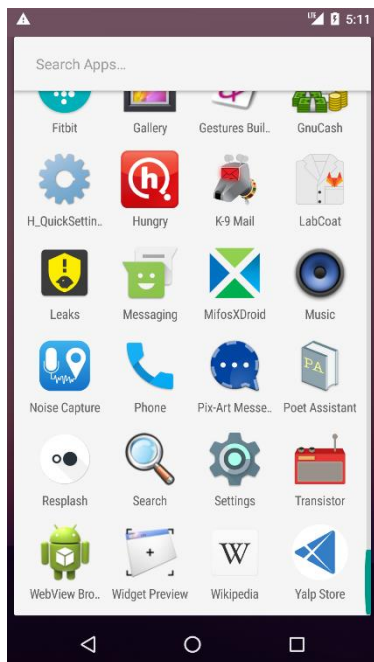


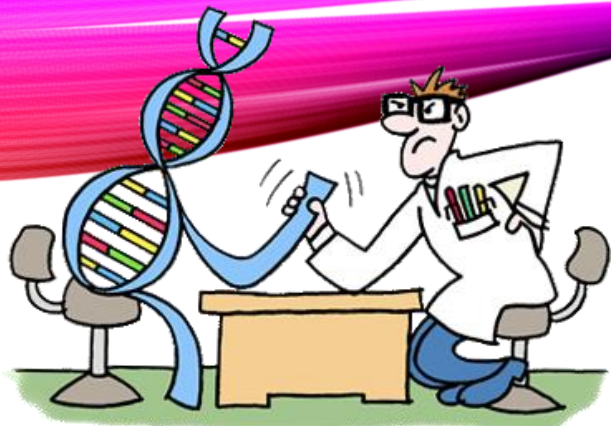


Human Competitive

Patch generated by Droix

Patch crafted by human

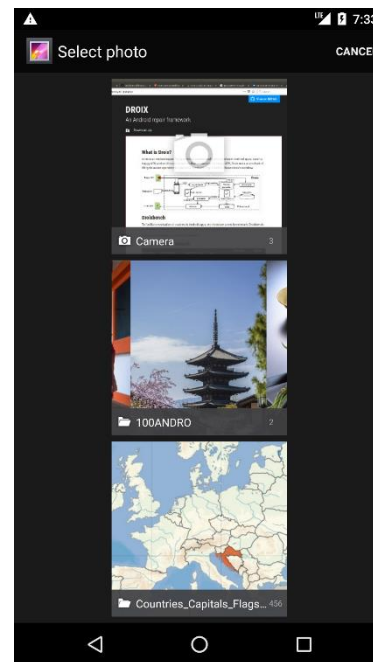
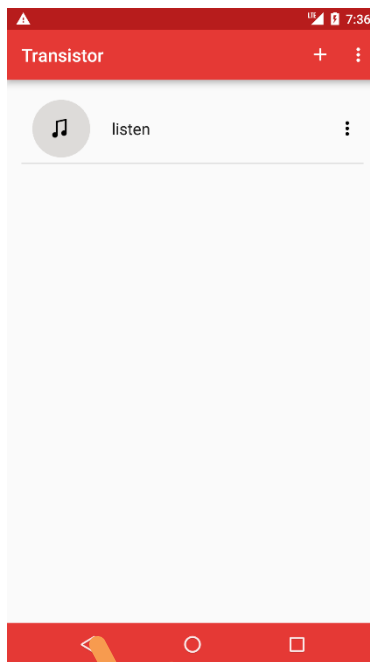
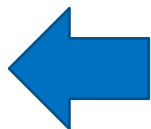
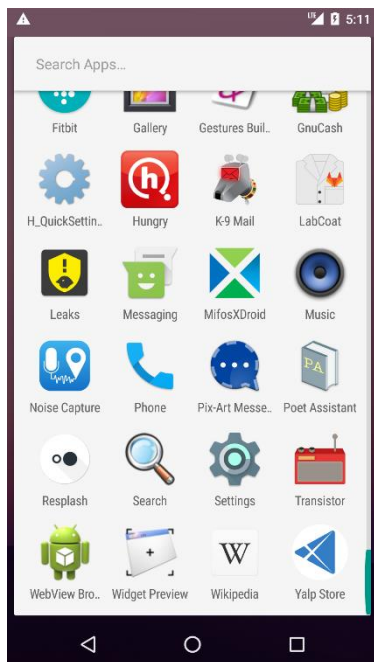




Human Competitive

Patch generated by Droix

Patch crafted by human

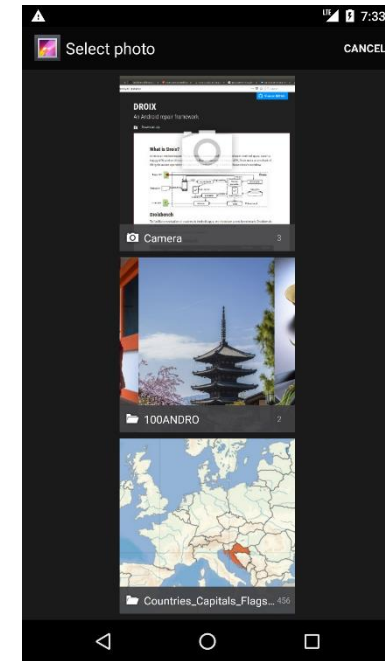
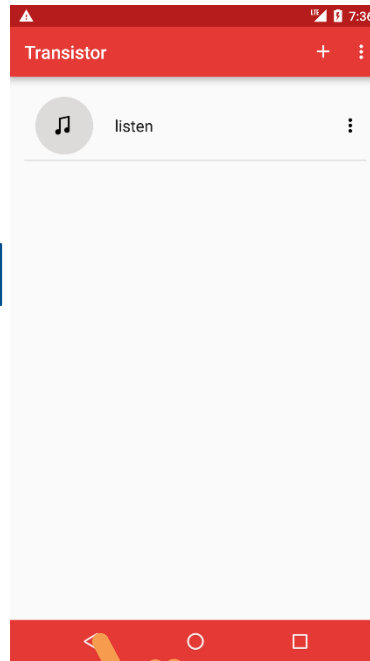
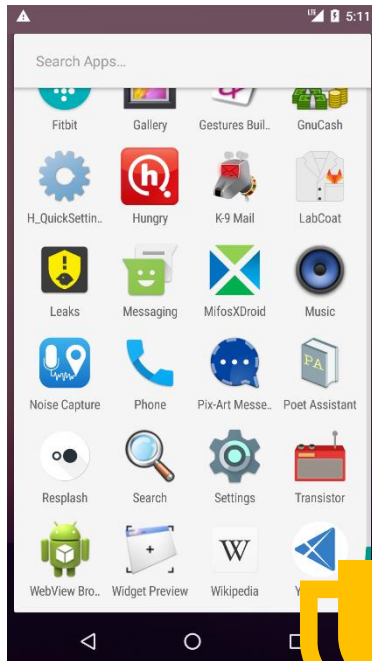




Human Competitive

Patch generated by Droix

Patch crafted by human



➤ <17minutes to fix

➤ 2 days to fix

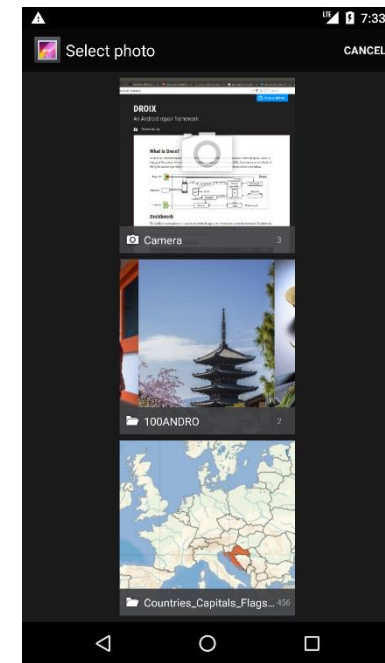
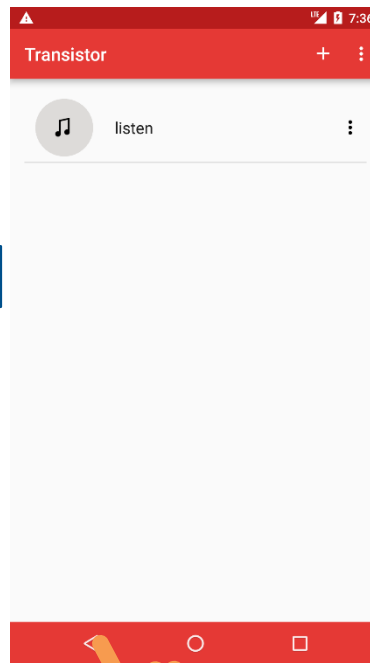
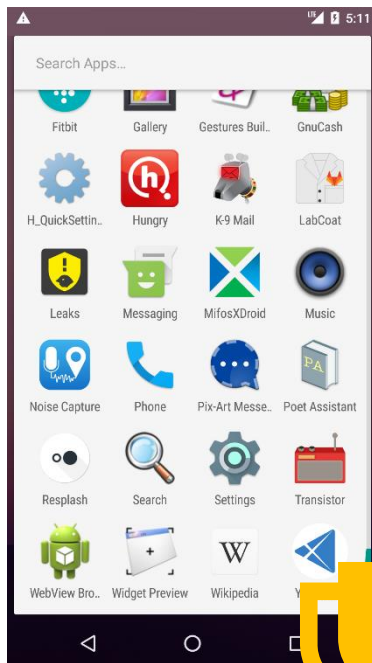
➤ Resolved 9 months later



Human Competitive

Patch generated by Droix

Patch crafted by human



➤ <17minutes to fix

➤ 2 days to fix

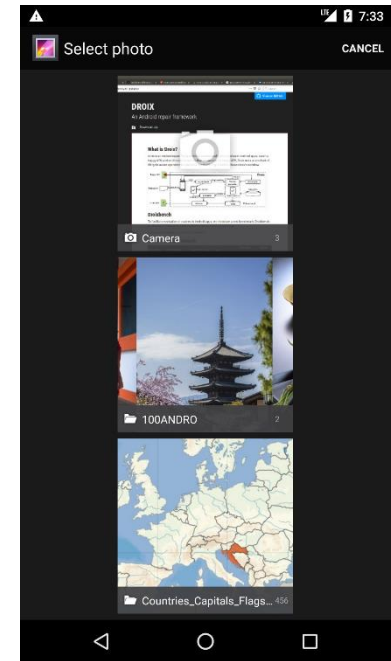
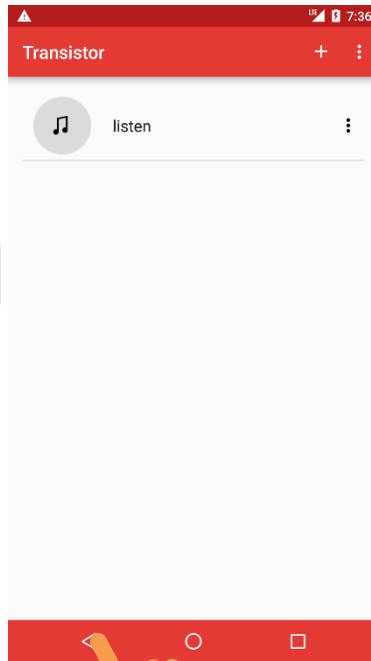
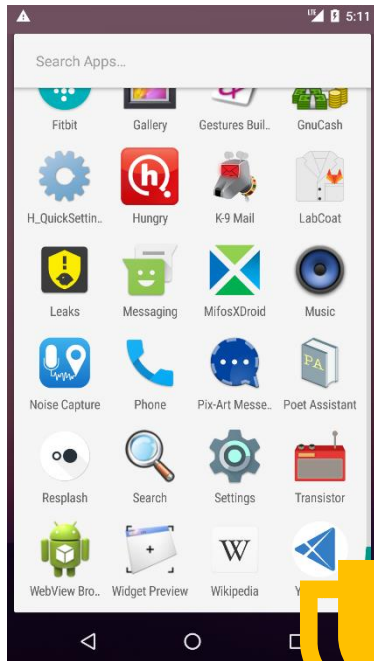
➤ Resolved 9 months later



Human Competitive

Patch generated by Droix

Patch crafted by human

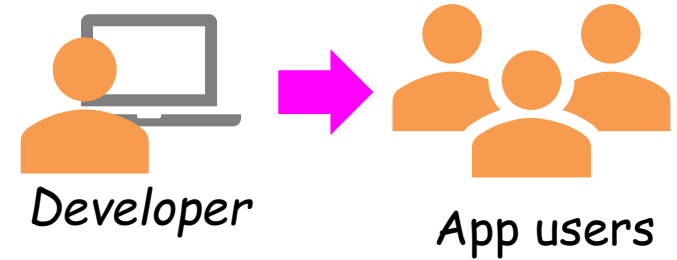


➤ <17minutes to fix

➤ 2 days to fix

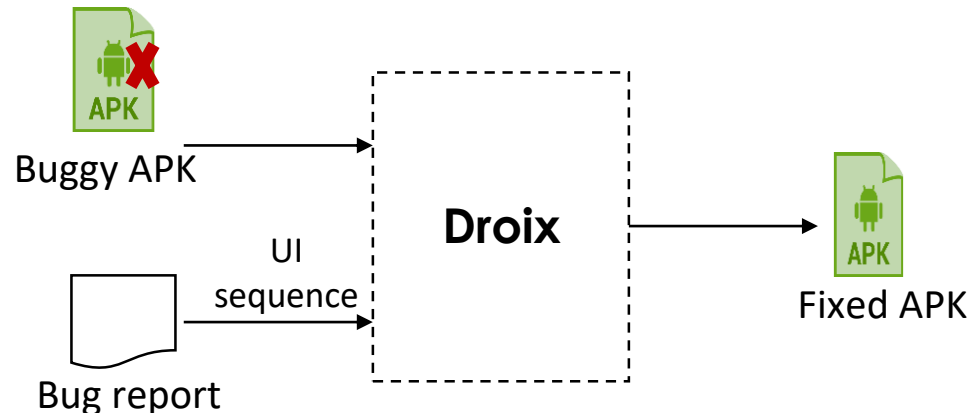
➤ Resolved 9 months later

Wide applicability of our results



Repair for everyone

- ✓ Could be used by any smartphone user
 - ✓ Doesn't require source code
 - ✓ No prior knowledge of Android app
 - ✓ No prior experience of running tests



Conclusion

- **Technical Contributions:**
 - EA for fixing Android apps without source code & heavy test suite
 - Specialized operators for Android apps
 - *Droixbench*: systematic evaluation
- **High Commercial Value:** Patent-filling with Singtel
- **Practicality:** Crashes in Android apps affect all app users.
- **Human-competitive:** Our approach could generate patches that are not only of comparable quality to human patches, but could also outperform human patch.