StoryDroid:
Automated Generation of Storyboard for Android Apps

ICSE 2019
Montréal, QC, Canada

Sen Chen, Lingling Fan,
Chunyang Chen, Ting Su, Wenhe Li,
Yang Liu, Lihua Xu
Over 3.8 million Android apps are available at Google Play Store. One single app category usually contains many similar Android apps.
Searching apps from Google Play Store

Mobile apps are facing fierce competitions.
Before developing a new mobile app

Development team usually endeavors painstaking efforts to review many existing apps (i.e., competitive analysis) with similar purposes.
Similar purposes of competitive analysis

- Helps understand the competitors’ strengths and weaknesses
- Helps developers gain more insights on the actual implementation
- Reduces market risks before development
Reviewing apps for competitive analysis

Eve: Product Manager

Alice: UX/UI Designer

Bob: App Developer

Functionalities

UI design and layout code

Logic code

Login

Main

Layout Code

Activity Code
However…

Manual exploration of hundreds of existing similar apps by different roles is ineffective.

✔ It is a time-consuming task.
✔ It is difficult to explore all the functionalities.

How to improve the reviewing efficiency?
Storyboard

Storyboard of movies

Storyboard of apps
Storyboard of Android Apps

*ATG: Activity Transition Graph

ATG with UI pages

Layout Code

Activity Code

Method Hierarchy

Profile

Password Recovery

Login

Registration

Help

Main

Friend View

Connection
Challenges

✔ ATGs* are usually incomplete due to the limitation of current static analysis tools (e.g., IC3 [1])

✔ A pure static approach may miss parts of UIs that are dynamically rendered.

✔ The obfuscated activity names lack the semantics of the corresponding functionalities.

Our Solution - StoryDroid

1. ATG Extraction
2. UI Page Rendering
3. Semantic Name Inferring
StoryDroid – Activity Transition Extraction

StoryDroid Enhances the ATG extraction ability of IC3, especially for fragments and inner classes.

StoryDroid leverages control- and data-flow analysis to obtain relatively complete ATG.
Activity Transition Extraction

(a) Transitions between activities
Main -> Reg

(b) Transitions with Fragment
Main -> PrefEditor -> AdvancedPrefEditor

(c) Transitions with Inner Class
SearchPanel -> PartList

Main -> Reg
Start Activity
PrefEditor
Start Fragment
PrefEditor Fragment
Advanced PrefEditor
Start Activity
SearchPanel
Use Inner Class
SearchBy PartName
PartList
Start Activity
StoryDroid – UI Page Rendering

StoryDroid translates *dynamic* and *hybrid layouts* to static layout (if needed) to render UI pages that users interact with.

![Diagram](image_url)
UI Page Rendering

Layout Type

- Checking
- Translating

UI Page Rendering

ListView

```
// Static Layout
<TextView
    android:id="@+id/text"
    android:text="Password">
</TextView>

// Dynamic Layout
RelativeLayout r = new RelativeLayout();
TextView tv = new TextView();
tv.setText("Password");
r.addView(tv);
```

// Hybrid Layout

```
// main_act refers to a static XML file
LayoutInflater li = LayoutInflater.form();
View view = li.inflate(R.layout.main_act);
```
StoryDroid – Semantic Name Inferring

StoryDroid infers the semantic name for the obfuscated activity names by layout comparison.
StoryDroid – Semantic Name Inferring

Semantic Name Inferring → ATG

Layout Tree DB:
- 4,426 F-Droid apps
- 13,792 layout trees

Layout Tree:
- ScrollView
- LinearLayout
- TextView
- EditView
- Button

DB Construction

Large-Scale XML Files → Layout Tree DB

XML File
XML Name

Ranking & Matching

Top 10 candidates → Semantic Act. Name
Automated Generation of Storyboard
Effectiveness Evaluation - Research Questions

- **RQ1:** Can StoryDroid extract a more complete ATG for an app, and achieve better activity coverage than the dynamic testing tool (i.e., Stoat)?
- **RQ2:** Can StoryDroid render UI pages with high similarity compared with the real screenshots?
- **RQ3:** Can StoryDroid infer semantic names for obfuscated activities?
Effectiveness Evaluation - RQ1

Can StoryDroid extract a more complete ATG for an app, and achieve better activity coverage than the dynamic testing tool (i.e., Stoat)?
Effectiveness Evaluation - RQ1

- Self-developed 10 apps
- Capability of handling fragments and inner classes

<table>
<thead>
<tr>
<th>App ID</th>
<th>Feature</th>
<th>#Transition pairs</th>
<th>#Identified by IC3</th>
<th>#Identified by StoryDroid</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Activity</td>
<td>14</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>13</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>3</td>
<td>Inner Class</td>
<td>13</td>
<td>-</td>
<td>13</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>13</td>
<td>-</td>
<td>13</td>
</tr>
<tr>
<td>5</td>
<td>Fragment</td>
<td>13</td>
<td>-</td>
<td>13</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>13</td>
<td>-</td>
<td>13</td>
</tr>
<tr>
<td>7</td>
<td>Activity &amp; Inner Class</td>
<td>13</td>
<td>1</td>
<td>13</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>13</td>
<td>1</td>
<td>13</td>
</tr>
<tr>
<td>9</td>
<td>Activity &amp; Fragment</td>
<td>10</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td>10</td>
<td>1</td>
<td>10</td>
</tr>
</tbody>
</table>
Effectiveness Evaluation - RQ1

- **100 apps** (i.e., 50 F-Droid and 50 Google Play)
- Transitions pairs and activity coverage

StoryDroid **outperforms IC3** on ATG extraction and **covers 2 times more activities** than Stoat with less time.
Effectiveness Evaluation - RQ2

Can StoryDroid render UI pages with high similarity compared with the real screenshots?
Effectiveness Evaluation - RQ2
- Similarity compared with the real screenshots
- Metrics: MSE (Mean Squared Error) & MAE (Mean Absolute Error)

(a) Real page (b) Our rendered (c) Real page (d) Our rendered

StoryDroid can render UI pages with **high similarity (84%)** to the real ones in our experiments.
Effectiveness Evaluation - RQ3

Can StoryDroid infer semantic names for obfuscated activities?
Effectiveness Evaluation - RQ3
- 92 out of 100 activity names

<table>
<thead>
<tr>
<th>Ground Truth Act. Name</th>
<th>Rank in Candidates</th>
<th>Corresponding XML Name</th>
<th>Inferred by StoryDroid</th>
</tr>
</thead>
<tbody>
<tr>
<td>AboutAct.</td>
<td>1</td>
<td>about</td>
<td>AboutAct.</td>
</tr>
<tr>
<td>PersonInfoAct.</td>
<td>3</td>
<td>content_extended_title</td>
<td>WizardAct.</td>
</tr>
<tr>
<td>LoginAct.</td>
<td>3</td>
<td>login</td>
<td>LoginAct.</td>
</tr>
<tr>
<td>ContactAct.</td>
<td>1</td>
<td>contact_list</td>
<td>ContactListAct.</td>
</tr>
<tr>
<td>SearchAct.</td>
<td>4</td>
<td>grid_base</td>
<td>Searcher</td>
</tr>
<tr>
<td>SettingAct.</td>
<td>1</td>
<td>setting_container</td>
<td>SettingAct.</td>
</tr>
<tr>
<td>ShareAct.</td>
<td>1</td>
<td>activity_share</td>
<td>ShareAct.</td>
</tr>
<tr>
<td>SplashAct.</td>
<td>3</td>
<td>activity_splash</td>
<td>SplashAct.</td>
</tr>
<tr>
<td>TrackListAct.</td>
<td>1</td>
<td>list_view</td>
<td>TrackListAct.</td>
</tr>
</tbody>
</table>

StoryDroid can infer semantic names with high accuracy for obfuscated activity names.
Usefulness Evaluation

Whether StoryDroid can help explore and understand the functionalities of apps effectively?
Usefulness Evaluation – User Study

1. 4 apps with different number of activities (12-15 activities) from 2 categories (i.e., finance and tool), each category contains 2 apps

2. 8 participants including post-docs, Ph.D., and masters

3. For each category, each participant explored one app with StoryDroid, and the other without StoryDroid.

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Manual Exploration</th>
<th>StoryDroid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time (min)</td>
<td>5.2</td>
<td>2.5</td>
</tr>
<tr>
<td>Coverage</td>
<td>40.8%</td>
<td>86.5%</td>
</tr>
<tr>
<td>Satisfactoriness</td>
<td>4.2</td>
<td>4.4</td>
</tr>
</tbody>
</table>

Compared with manual exploration, StoryDroid achieves 2 times more activity coverage with less time cost to help understand the app functionalities.
Future applications based on StoryDroid

✔ Recommendation of UI design and code

✔ Guiding regression testing of apps

✔ Extracting commonalities across apps
Summary

Storyboard of Android Apps

Our Solution - StoryDroid

Usefulness Evaluation

Whether StoryDroid can help explore and understand the functionalities of apps effectively?

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Manual Exploration</th>
<th>StoryDroid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time (min)</td>
<td>5.2</td>
<td>2.5</td>
</tr>
<tr>
<td>Coverage</td>
<td>40.8%</td>
<td>86.5%</td>
</tr>
<tr>
<td>Satisfactoriness</td>
<td>4.2</td>
<td>4.4</td>
</tr>
</tbody>
</table>

Future applications based on StoryDroid

- Recommendation of UI design and code
- Guiding regression testing of apps
- Extracting commonalities across apps
Thanks and Questions?

Sen Chen
Nanyang Technological University
Contact: ecnuchensen@gmail.com
Homepage: https://sen-chen.github.io/