Bug Report Analytics

David Lo
School of Information Systems
Singapore Management University
davidlo@smu.edu.sg

38th ACM/IEEE International Conference on Software Engineering, Austin, Texas, USA
Problems with Bug Reports

- Bugs are reported in bug tracking systems
- The number of bug reports are often too many for developers to handle (Anvik et al., ETX 2005)
- Management of bugs is an expensive process (NIST, 2002)
Bug Report Management Process

1. Check for Duplicates
2. Assign Severity and Priority Level
3. Assign Suitable Developer
4. Locate Buggy Program Elements
5. Repair Buggy Program Elements
How Analytics Can Help?

- Automation
- Recommendation
Structure of This Talk

1. Duplicate Bug Report Detection
2. Priority/Severity Prediction
3. Developer Assignment
4. Bug Localization
5. Automated Repair
Duplicate Bug Report Detection

- Bug reporting is inherently a distributed and uncoordinated process.
- Similar people (users, testers) report the same bug in different reports.
Duplicate Bug Report Detection

1. Compute Similarities

2. Output ranked list of similar bug reports

Historical Bug Reports

Ranked List
Duplicate Bug Report Detection

1. Learn Model
2. Apply Model

Historical Bug Reports

New Bug Report
Duplicate Bug Report Detection

- **Similarity Based**
  - Anh Tuan Nguyen, Tung Thanh Nguyen, Tien N. Nguyen, David Lo, Chengnian Sun: Duplicate bug report detection with a combination of information retrieval and topic modeling. ASE 2012: 70-79
  - Chengnian Sun, David Lo, Siau-Cheng Khoo, Jing Jiang: Towards more accurate retrieval of duplicate bug reports. ASE 2011: 253-262
  - Chengnian Sun, David Lo, Xiaoyin Wang, Jing Jiang, Siau-Cheng Khoo: A discriminative model approach for accurate duplicate bug report retrieval. ICSE (1) 2010: 45-54

- **Classification Based**
  - Anahita Alipour, Abram Hindle, Eleni Stroulia: A contextual approach towards more accurate duplicate bug report detection. MSR 2013: 183-192
Severity/Priority Prediction

- Developers have limited time
- Some reports are more important than others
- **Severity** of reports need to be estimated
- Bug reports need to be **prioritized**

300 reports to triage daily!

Duplicate Check,
**Priority Assignment**
Developer Assignment
Severity/Priority Prediction

1. Compute Similarities
   - Historical Bug Reports
   - Most Similar Reports

2. Estimate Severity/Priority
Severity/Priority Prediction

1. Historical Bug Reports
2. Learn Model
3. Apply Model
4. 1 - 4
Severity/Priority Prediction

(DRONE, Tian et al., 2012)
Severity/Priority Prediction

- Severity Prediction
  - Ahmed Lamkanfi, Serge Demeyer, Quinten David Soetens, Tim Verdonck: Comparing Mining Algorithms for Predicting the Severity of a Reported Bug. CSMR 2011: 249-258
  - Ahmed Lamkanfi, Serge Demeyer, Emanuel Giger, Bart Goethals: Predicting the severity of a reported bug. MSR 2010: 1-10

- Priority Prediction
Developer Assignment

- Many projects have a large number of contributors
- Each contributor have different expertise
- How to assign the right contributor to a suitable bug report?
Developer Assignment

1. Compute Similarities

2. Output ranked list of developers who fixed similar bug reports

Historical Bug Reports

Ranked List
Developer Assignment

1. Learn Model
2. Apply Model

Historical Bug Reports
Developed Assignment

- **Similarity Based**

- **Classification Based**
  - John Anvik, Lyndon Hiew, Gail C. Murphy: Who should fix this bug? ICSE 2006: 361-370
  - Xin Xia, David Lo, Xinyu Wang, Bo Zhou: Accurate developer recommendation for bug resolution. WCRE 2013: 72-81
Bug Localization

How to locate the buggy files?

Bugs

Bug Localization!

Developer

Software

Manually

Automatically

File • …

File • …

File • …

ICSE 2016
IR-Based Bug Localization

Bug Report → IR-Based Bug Localization Technique → Ranked List of Files

Ranked List of Files:
- File 1
- File 2
- File 3

(Thousands of) Source Code Files
### Spectrum-Based Bug Localization

<table>
<thead>
<tr>
<th>Block ID</th>
<th>Program Elements</th>
<th>T15</th>
<th>T16</th>
<th>T17</th>
<th>T18</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>int count;</td>
<td><img src="Image1" alt="Mark" /></td>
<td><img src="Image1" alt="Mark" /></td>
<td><img src="Image1" alt="Mark" /></td>
<td><img src="Image1" alt="Mark" /></td>
</tr>
<tr>
<td></td>
<td>int n;</td>
<td><img src="Image2" alt="Mark" /></td>
<td><img src="Image2" alt="Mark" /></td>
<td><img src="Image2" alt="Mark" /></td>
<td><img src="Image2" alt="Mark" /></td>
</tr>
<tr>
<td></td>
<td>Ele *proc;</td>
<td><img src="Image3" alt="Mark" /></td>
<td><img src="Image3" alt="Mark" /></td>
<td><img src="Image3" alt="Mark" /></td>
<td><img src="Image3" alt="Mark" /></td>
</tr>
<tr>
<td></td>
<td>List *src_queue, *dest_queue;</td>
<td><img src="Image4" alt="Mark" /></td>
<td><img src="Image4" alt="Mark" /></td>
<td><img src="Image4" alt="Mark" /></td>
<td><img src="Image4" alt="Mark" /></td>
</tr>
<tr>
<td></td>
<td>if (prio &gt;= MAXPRIO) /<em>maxprio=3</em>/</td>
<td><img src="Image5" alt="Mark" /></td>
<td><img src="Image5" alt="Mark" /></td>
<td><img src="Image5" alt="Mark" /></td>
<td><img src="Image5" alt="Mark" /></td>
</tr>
<tr>
<td>2</td>
<td>{return;}</td>
<td><img src="Image6" alt="Mark" /></td>
<td><img src="Image6" alt="Mark" /></td>
<td><img src="Image6" alt="Mark" /></td>
<td><img src="Image6" alt="Mark" /></td>
</tr>
<tr>
<td>3</td>
<td>src_queue = prio_queue[prio];</td>
<td><img src="Image7" alt="Mark" /></td>
<td><img src="Image7" alt="Mark" /></td>
<td><img src="Image7" alt="Mark" /></td>
<td><img src="Image7" alt="Mark" /></td>
</tr>
<tr>
<td></td>
<td>dest_queue = prio_queue[prio+1];</td>
<td><img src="Image8" alt="Mark" /></td>
<td><img src="Image8" alt="Mark" /></td>
<td><img src="Image8" alt="Mark" /></td>
<td><img src="Image8" alt="Mark" /></td>
</tr>
<tr>
<td></td>
<td>count = src_queue-&gt;mem_count;</td>
<td><img src="Image9" alt="Mark" /></td>
<td><img src="Image9" alt="Mark" /></td>
<td><img src="Image9" alt="Mark" /></td>
<td><img src="Image9" alt="Mark" /></td>
</tr>
<tr>
<td></td>
<td><em>Bug</em> /* expected : count&gt;0*/</td>
<td><img src="Image10" alt="Mark" /></td>
<td><img src="Image10" alt="Mark" /></td>
<td><img src="Image10" alt="Mark" /></td>
<td><img src="Image10" alt="Mark" /></td>
</tr>
<tr>
<td>4</td>
<td>n = (int) (count*ratio + 1);</td>
<td><img src="Image11" alt="Mark" /></td>
<td><img src="Image11" alt="Mark" /></td>
<td><img src="Image11" alt="Mark" /></td>
<td><img src="Image11" alt="Mark" /></td>
</tr>
<tr>
<td></td>
<td>proc = find_nth(src_queue, n);</td>
<td><img src="Image12" alt="Mark" /></td>
<td><img src="Image12" alt="Mark" /></td>
<td><img src="Image12" alt="Mark" /></td>
<td><img src="Image12" alt="Mark" /></td>
</tr>
<tr>
<td></td>
<td>if (proc) {</td>
<td><img src="Image13" alt="Mark" /></td>
<td><img src="Image13" alt="Mark" /></td>
<td><img src="Image13" alt="Mark" /></td>
<td><img src="Image13" alt="Mark" /></td>
</tr>
<tr>
<td></td>
<td>src_queue = del_ele(src_queue, proc);</td>
<td><img src="Image14" alt="Mark" /></td>
<td><img src="Image14" alt="Mark" /></td>
<td><img src="Image14" alt="Mark" /></td>
<td><img src="Image14" alt="Mark" /></td>
</tr>
<tr>
<td></td>
<td>proc-&gt;priority = prio;</td>
<td><img src="Image15" alt="Mark" /></td>
<td><img src="Image15" alt="Mark" /></td>
<td><img src="Image15" alt="Mark" /></td>
<td><img src="Image15" alt="Mark" /></td>
</tr>
<tr>
<td></td>
<td>dest_queue = append_ele(dest_queue, proc);</td>
<td><img src="Image16" alt="Mark" /></td>
<td><img src="Image16" alt="Mark" /></td>
<td><img src="Image16" alt="Mark" /></td>
<td><img src="Image16" alt="Mark" /></td>
</tr>
<tr>
<td></td>
<td>}</td>
<td><img src="Image17" alt="Mark" /></td>
<td><img src="Image17" alt="Mark" /></td>
<td><img src="Image17" alt="Mark" /></td>
<td><img src="Image17" alt="Mark" /></td>
</tr>
</tbody>
</table>

**Status of Test Case Execution:**
- T15: Pass
- T16: Pass
- T17: Pass
- T18: Fail
Spectrum-Based Bug Localization

Table 1: Raw Statistics for Program Element $e$

<table>
<thead>
<tr>
<th></th>
<th>$e$ is executed</th>
<th>$e$ is not executed</th>
</tr>
</thead>
<tbody>
<tr>
<td>unsuccessful test</td>
<td>$n_f(e)$</td>
<td>$n_f(\bar{e})$</td>
</tr>
<tr>
<td>successful test</td>
<td>$n_s(e)$</td>
<td>$n_s(\bar{e})$</td>
</tr>
</tbody>
</table>

$$Tarantula(e) = \frac{n_f(e)}{n_f} + \frac{n_s(e)}{n_s}$$
Bug Localization

- IR-Based Bug Localization
  - Shaowei Wang, David Lo, Julia Lawall: Compositional Vector Space Models for Improved Bug Localization. ICSME 2014: 171-180
  - Xin Ye, Razvan C. Bunescu, Chang Liu: Learning to rank relevant files for bug reports using domain knowledge. SIGSOFT FSE 2014: 689-699
Bug Localization

- Spectrum-Based Bug Localization
  - Tien-Duy B. Le, David Lo, Claire Le Goues and Lars Grunske. A Learning-to-Rank Based Fault Localization Approach using Likely Invariants. ISSTA 2016 (to appear)

- Combination
Automatic Repair

Mutates buggy program to create repair candidates

Candidate passing all test cases

E.g., GenProg, PAR, etc
History Driven Repair (Le et al., SANER’16)

Test Cases

Mutates buggy program
to create repair candidates

Candidates:
- frequently occur in knowledge base
- pass negative test cases

Knowledge base: Learned bug fix behaviors from history

Fast

Avoid nonsensical patches
Automatic Repair

- Xuan-Bach D. Le, David Lo, and Claire Le Goues. History Driven Program Repair. 23rd IEEE International Conference on Software Analysis, Evolution, and Reengineering (SANER) 2016
- Xuan-Bach D. Le, Tien-Duy B. Le, David Lo: Should fixing these failures be delegated to automated program repair? ISSRE 2015: 427-437
- Chen Liu, Jinqiu Yang, Lin Tan, Munawar Hafiz: R2Fix: Automatically Generating Bug Fixes from Bug Reports. ICST 2013: 282-291
- Sergey Mechtaev, Jooyong Yi, Abhik Roychoudhury: DirectFix: Looking for Simple Program Repairs. ICSE (1) 2015: 448-458
- Shin Hwei Tan, Abhik Roychoudhury: relifix: Automated Repair of Software Regressions. ICSE (1) 2015: 471-482
- Fan Long, Martin Rinard: Staged program repair with condition synthesis. ESEC/SIGSOFT FSE 2015: 166-178
Future Opportunities on Bug Report Analytics

- Achieve higher accuracy
  - Technical innovation
  - Additional data sources
- AI-Human interaction
  - Incorporating incremental user feedback
- Tool support
  - Integration with standard IDEs/bug trackers
- Field study
  - Deploying bug report analytics techniques live and get feedback
Thank you!

Questions? Comments?
davidlo@smu.edu.sg