Automated Classification of Static Code Analysis Alerts: A Case Study

29th IEEE International Conference on Software Maintenance
26 September 2013 - Eindhoven, The Netherlands

Ulaş Yüksel\textsuperscript{1,2}
Hasan Sözer\textsuperscript{2}

\textsuperscript{1} Vestel Electronics, Manisa, Turkey
\textsuperscript{2} Özyegin University, Istanbul, Turkey
Problem

Static code analysis tools (SCATs) complement manual reviews and testing...

but

if you have 1500 KLOC legacy software ...
Problem (cont.)

1500 KLOC code...

- May generate up to 3000 alerts to be inspected.
- 30% - 100%* false positives.
- May need 250 hours of manual inspection.

- Is automatic classification feasible?

Evaluated approach

- Post-process of SCAT alerts to classify with *machine-learning algorithms*
- Based on set of artifact characteristics.

Alerts artifact chars:
- Severity
- Alert code
- Lifetime
  ..
  ..
Outline

✓ Introduction: Problem definition
  • Industrial case: Digital TV software
  • Data set
  • Case study and results
  • Future work
Industrial Case: Digital TV Software

- **VESTEL** Electronics: Leading company in digital TV market
- Supports: Digital video broadcast (DVB), recording, media and web browsing, and so on...
- Embedded
- Over 1500KLOC C/C++ distributed over 8000 files
Data set and characteristics

- 19 weekly run of SCAT
- 1147 Alerts with unique ID and set of artifact characteristics:
  - Severity, Alert code, Life time, Developer idea, File name, Module name, Open alerts, Total alerts, Total alerts in module, Total alerts in file
- Manually inspected alert distribution:

<table>
<thead>
<tr>
<th>Resolution</th>
<th>Manual Inspection Result</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>True Positive</td>
</tr>
<tr>
<td>Resolved</td>
<td>203</td>
</tr>
<tr>
<td>Not resolved</td>
<td>295</td>
</tr>
<tr>
<td>Total</td>
<td>498</td>
</tr>
</tbody>
</table>
Case Studies on Data Set

- (a) Attribute Evaluation
- (b) Classification with 10-folds cross validation
- (c) Classification during the development life-cycle
Case Study- (a): Attribute Evaluation

- Aimed to see which of the characteristics are relevant for classification.
- With 10 attribute selection evaluator tools of Weka*

* WEKA: Data mining software http://www.cs.waikato.ac.nz/ml/weka/
Case Study- (b): Classification with 10-folds cross validation

- 34 machine learning algorithms available in Weka tested.

- Top classifiers with respect to accuracy:

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Accuracy</th>
<th>Precision&lt;sup&gt;a&lt;/sup&gt;</th>
<th>Recall&lt;sup&gt;a&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random forest</td>
<td>86.1%</td>
<td>86.1% [0.84,0.88]</td>
<td>86.1% [0.84,0.88]</td>
</tr>
<tr>
<td>Random committee</td>
<td>86.4%</td>
<td>86.4% [0.85,0.88]</td>
<td>86.4% [0.84,0.88]</td>
</tr>
<tr>
<td>DTNB*</td>
<td>83.6%</td>
<td>83.8% [0.80,0.87]</td>
<td>83.6% [0.84,0.84]</td>
</tr>
</tbody>
</table>

<sup>a</sup> Values in brackets are particular calculations for true positive and false positive classifications respectively.

* DTNB: M. A. Hall and E. Frank, “Combining naive bayes and decision tables,”
Case Study- (c): Classification during the development life-cycle

- First 5 run of SCAT for training set (92% of whole).
- 3 groups of test sets created with the rest:

<table>
<thead>
<tr>
<th>Group</th>
<th>SCAT run #</th>
<th>True Positive</th>
<th>False Positive</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st</td>
<td>[6,10]</td>
<td>2</td>
<td>13</td>
<td>15</td>
</tr>
<tr>
<td>2nd</td>
<td>[11,16]</td>
<td>9</td>
<td>11</td>
<td>20</td>
</tr>
<tr>
<td>3rd</td>
<td>[17,19]</td>
<td>34</td>
<td>12</td>
<td>46</td>
</tr>
</tbody>
</table>

- Developer idea and lifetime are modified to reflect the real development life cycle.
- 34 machine learning algorithms available in Weka tested..
Case Study- (c): Classification during the development life-cycle (cont.)

- Best classifier with respect to accuracy: DTNB

<table>
<thead>
<tr>
<th>Group</th>
<th>Accuracy</th>
<th>Precision&lt;sup&gt;a&lt;/sup&gt;</th>
<th>Recall&lt;sup&gt;a&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>1&lt;sup&gt;st&lt;/sup&gt;</td>
<td>66.7%</td>
<td>80.7% [0.20,0.90]</td>
<td>66.7% [0.5,0.69]</td>
</tr>
<tr>
<td>2&lt;sup&gt;nd&lt;/sup&gt;</td>
<td>80.0%</td>
<td>86.2% [0.69,1.00]</td>
<td>80.0% [1.00,0.64]</td>
</tr>
<tr>
<td>3&lt;sup&gt;rd&lt;/sup&gt;</td>
<td>89.1%</td>
<td>90.7% [0.97,0.73]</td>
<td>89.1% [0.88,0.92]</td>
</tr>
</tbody>
</table>

<sup>a</sup> Values in brackets are particular calculations for true positive and false positive classifications respectively.
Conclusion & Future work

• We conclude:
  – Machine learning techniques can be a viable approach for automated classification of SCAT alerts...

• In future:
  – Extend data set with the consequent SCAT runs...
  – Incorporate additional artifact characteristics...
Thank you for your attention

Any Questions?

Contact:
ulas.yuksel@ozu.edu.tr
hasan.sozer@ozu.edu.tr