SCAN: an Approach to Label and Relate Execution Trace Segments

Soumaya Medini, Giuliano Antoniol, Yann-Gaël Guéhéneuc, Massimiliano Di Penta, Paolo Tonella

SOCCER Lab. & Ptidej Team, École Polytechnique de Montréal, Québec, Canada
Fondazione Bruno Kessler (FBK) Trento, Italy
University of Sannio, Italy

October 15, 2012
CONTEXT

- Program comprehension take half the effort devoted to software maintenance.[Fjeldstad and Hamlen, 1983]

- Software maintenance can be up to 80% of software cost.[Arthur, 1999]

- Concept location is an important task during Program comprehension.[Rajlich, 2002]

- Concept location aims at identifying user-observable features and locating their implementation within code regions.
CONCEPT LOCATION AS TRACE SEGMENTATION

- Use textual content of methods to split execution trace into segments that implement concepts related to the features of interest.

- Asadi et al. [Asadi et al., 2010]: Identify concepts in execution trace by finding cohesive and decoupled fragments of the trace.

- Medini et al. [Medini et al., 2011]: Identify concepts by computing the exact split of an execution trace into segments.
PROPOSED APPROACH (1)

- These approaches create segments but do not assign a label to the identified segments.
- SCAN (Segment Concept AssigNer) approach: Assign meaningful labels to chunks of segmented traces.
  - Used VSM to label identified segments.
  - Used FCA to relate segments.
SCAN accepts as input one or more execution traces.

Trace segmentations obtained using the tool by Medini et al. [Medini et al., 2011].

Labeling of identified trace segments.

FCA module to identify relations between segments.
APPROACH DETAILS

- Multi-threading: Induces variability in traces collected for a given scenario.
- To limit this effect in assigning labels to segments, SCAN is able to merge segments obtained in multiple executions of the same scenario.
- Recognize similarities between segments belonging to multiple execution traces and merge them.
Source of information: Terms contained in the signature of invoked methods.

Hypothesis: A term appearing often in a particular segment, but not in other segments, provides linguistic information important for that given segment.

SCAN ranks the terms of the segment terms by tf-idf and keeps the topmost ones.
FORMAL CONCEPT

- Used to identify relations between concepts identified in different segments.
- Groups objects that have common attributes: Objects are segments and attributes are terms extracted for the segments.
- A concept: Maximal collection of objects that has common attributes. Grouping of all cohesive set of segments sharing terms.
FCA EXAMPLE

- FCA lattice produced for two executions of the scenario “New Package”. All segments and concepts are similar between the two traces, except for segment 10 of “NewPackage1”.

- SCAN’s ability to recognize the occurrence of the same concepts in different executions.
## CASE STUDY

<table>
<thead>
<tr>
<th>Systems</th>
<th>Scenarios</th>
<th>Original Size</th>
<th>Compressed Size</th>
<th>Number of Segments</th>
</tr>
</thead>
<tbody>
<tr>
<td>JHotDraw 5.1</td>
<td>Draw Rectangle</td>
<td>15,706</td>
<td>930</td>
<td>54</td>
</tr>
<tr>
<td></td>
<td>Draw Rectangle, Delete Rectangle</td>
<td>5,960</td>
<td>554</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td>Draw Ellipse</td>
<td>5,252</td>
<td>562</td>
<td>33</td>
</tr>
<tr>
<td></td>
<td>Draw Ellipse, Delete Ellipse</td>
<td>10,760</td>
<td>953</td>
<td>53</td>
</tr>
<tr>
<td></td>
<td>Draw Rectangle, Draw Ellipse</td>
<td>8,790</td>
<td>690</td>
<td>30</td>
</tr>
<tr>
<td>ArgoUML 0.19.8</td>
<td>New Class</td>
<td>82,579</td>
<td>2,785</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td>New Package</td>
<td>21,423</td>
<td>1,642</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>New Class, New Package</td>
<td>38,940</td>
<td>1,220</td>
<td>13</td>
</tr>
</tbody>
</table>

- For each scenario we collected a number (2-3) of traces.
RESEARCH QUESTIONS

- RQ1. How effective is SCAN in assigning labels to segments?
- RQ2. Does SCAN help to discover relations between segments? Does it help to discover the macro phases in a trace?
**RQ1: VALIDATION**

- We manually built labels for each segment and validated the SCAN results.

<table>
<thead>
<tr>
<th>Segment Number</th>
<th>Automatic Labels</th>
<th>Manual Labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>draw iconkit creat palett text tool button line imag icon</td>
<td>Create drawing palette button tool and create icons kit.</td>
</tr>
<tr>
<td>2</td>
<td>draw cut transfer figur command view</td>
<td>Execute draw figure command.</td>
</tr>
<tr>
<td>3</td>
<td>draw menu copi shortcut past add command transfer duplic view</td>
<td>Add a command with the given shortcut to the menu.</td>
</tr>
<tr>
<td>4</td>
<td>draw transfer delet figur command view</td>
<td>Execute draw figure command.</td>
</tr>
</tbody>
</table>

\[
Precision_{i,j} = \frac{|M_{i,j} \cap S_{i,j}|}{|S_{i,j}|} \quad Recall_{i,j} = \frac{|M_{i,j} \cap S_{i,j}|}{|M_{i,j}|}
\]
### RQ1: RESULTS (JHOTDRAW)

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Precision</th>
<th></th>
<th></th>
<th>Recall</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Q1 median</td>
<td>Q3 median</td>
<td>mean</td>
<td>Q1 median</td>
<td>Q3 median</td>
<td>mean</td>
</tr>
<tr>
<td>Draw Rectangle</td>
<td>0.50</td>
<td>0.60</td>
<td>0.83</td>
<td>0.75</td>
<td>0.83</td>
<td>1.00</td>
</tr>
<tr>
<td>Draw Rectangle, Delete Rectangle</td>
<td>0.50</td>
<td>0.60</td>
<td>0.72</td>
<td>0.70</td>
<td>0.80</td>
<td>1.00</td>
</tr>
<tr>
<td>Draw Rectangle, Draw Ellipse</td>
<td>0.40</td>
<td>0.60</td>
<td>0.70</td>
<td>0.67</td>
<td>0.80</td>
<td>1.00</td>
</tr>
</tbody>
</table>

- The mean Precision varies between 0.56 and 0.65
- The mean Recall is stable around 0.81-0.82
RQ1: RESULTS (ARGO-UML)

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Precision</th>
<th>Recall</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Q1 median</td>
<td>Q3 mean</td>
<td>Q1 median</td>
<td>Q3 mean</td>
<td>Q1 median</td>
<td>Q3 mean</td>
<td>Q1 median</td>
<td>Q3 mean</td>
</tr>
<tr>
<td>New Class</td>
<td>0.29</td>
<td>0.40</td>
<td>0.50</td>
<td>0.40</td>
<td>0.50</td>
<td>0.67</td>
<td>0.75</td>
<td>0.64</td>
</tr>
<tr>
<td>New Package</td>
<td>0.29</td>
<td>0.33</td>
<td>0.50</td>
<td>0.36</td>
<td>0.50</td>
<td>0.71</td>
<td>0.50</td>
<td>0.54</td>
</tr>
<tr>
<td>New Class, New Package</td>
<td>0.20</td>
<td>0.33</td>
<td>0.50</td>
<td>0.38</td>
<td>0.25</td>
<td>0.67</td>
<td>0.50</td>
<td>0.48</td>
</tr>
</tbody>
</table>

- Performances are relatively lower than those obtained for JHotDraw.
RQ2: RESULTS (ARGO-UML)

- FCA lattice for the execution trace of the scenario “New Class”
- S1: System start-up and S2-S7: “prepare creation” and “addition” of a new UML Class.
RQ2: RESULTS (ARGOUML)

- FCA lattice for the execution trace of the scenario “New Class”
- S1: System start-up and S2-S7: “prepare creation” and “addition” of a new UML Class.
S4, 10 and 16 implement the same concept.

The concept containing segments 3, 9 and 15 is a super-concept of the one containing segments 2, 8 and 14. It points to higher level concepts (generate key java module), while the sub-concept includes segments specific of the display functionality.
RQ2: RESULTS

S4, 10 and 16 implement the same concept.

The concept containing segments 3, 9 and 15 is a super-concept of the one containing segments 2, 8 and 14. It points to higher level concepts (generate key java module), while the sub-concept includes segments specific of the display functionality.
S4, 10 and 16 implement the same concept.

The concept containing segments 3, 9 and 15 is a super-concept of the one containing segments 2, 8 and 14. It points to higher level concepts (generate key java module), while the sub-concept includes segments specific of the display functionality.
The relations between cohesive sets of segments regarded as execution phases. One phase is built by repeated segments in a trace.

- S 2, 3, 4, 5 and 6 define an execution phase on the trace.
- This phase is repeated two times: S 8, 9, 10, 11 and 12, and S 14, 15, 16, 17 and 18.
- The rest of the segments are also converted to an execution phases.
The relations between cohesive sets of segments regarded as execution phases. One phase is built by repeated segments in a trace.

- S 2, 3, 4, 5 and 6 define an execution phase on the trace.
- This phase is repeated two times: S 8, 9, 10, 11 and 12, and S 14, 15, 16, 17 and 18.
- The rest of the segments are also converted to an execution phases.
After the phases we draw a higher level flow diagram of phases with labels, using the temporal relations between phases.

The “New Class” scenario, generating 32 segments, can be summarized into four macro execution phases.

- **Phase 1**: System startup
- **Phase 2**: Activity needed to create class and properties (e.g., state, composite, etc.)
DISCUSSION

- Quantitative results might be read as indicators of poor performance of the label assignment algorithm (recall/precision around 50% and above).

- Adequate to support program understanding tasks. We expect that developer with some knowledge about the application would find it relatively easy.
**RQ1: VALIDATION**

We manually built labels for each segment and validated the SCAN results.

<table>
<thead>
<tr>
<th>Segment Number</th>
<th>Automatic Label</th>
<th>Manual Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>draw icon open palette button tool-file menu</td>
<td>Create drawing palette button tool and create icon list</td>
</tr>
<tr>
<td>2</td>
<td>draw icon open palette affordance tool-file menu</td>
<td>Execute drawing command</td>
</tr>
<tr>
<td>3</td>
<td>draw menu open shortcut plus add command transfer dup command</td>
<td>Add a command with the given shortcut to the menu</td>
</tr>
<tr>
<td>4</td>
<td>draw menu open shortcut plus add command transfer dup command</td>
<td>Execute drawing command</td>
</tr>
</tbody>
</table>

\[
\text{Precision}_{ij} = \frac{|M_i \cap S_{ij}|}{|S_{ij}|} \quad \text{Recall}_{ij} = \frac{|M_i \cap S_{ij}|}{|M_i|}
\]

**RQ1: RESULTS (JHOTDRAW)**

* The mean Precision varies between 0.56 and 0.65
* The mean Recall is stable around 0.81-0.82

**RQ2: RESULTS (ARGOUML)**

* FCA lattice for the execution trace of the scenario “New Class”
* S1: System start-up and S2-S7: “prepare creation” and “addition” of a new UML Class.

**RQ2: IDENTIFYING PHASES**

* After the phases we draw a higher level flow diagram of phases with labels, using the temporal relations between phases.
* The “New Class” scenario, generating 32 segments, can be summarized into four macro execution phases.
  * Phase1: System startup
  * Phase2: Activity needed to create class and properties (e.g., state, composite, etc.)
FUTURE WORK

- Phase recognition: We plan to investigate how to automate it.
- Further validation of SCAN with a pool of independent developers.
- Application of SCAN to label multiple trace segmentations, i.e., segmentations of traces corresponding to different scenarios.