A Vision for Helping Developers
Use APIs by Leveraging Temporal Patterns

Erick Raelijohn, Michalis Famelis, Houari Sahraoui

FormaliSE 2019
IDEs are Awesome!

- Automatic code generation
- Organize imports
- “On the fly” compilation
But not so much as it could be

- Read and understand the documentation
- Look for example
- Seeking community support

Constraints

Productivity!

Feedback from other sources (e.g., execution, bug reports)
Our Vision

- **Constraints**
  - Read and understand the documentation
  - Seek example
  - Seeking community support

- **Productivity!**

- **Safer Usage**

- **IDE**

- **Knowledge about the API**

- **Code**

Adele
Temporal API Recommender (TAPIR)

- Towards Assisting Developers in API Usage by Automated Recovery of Complex Temporal Patterns, in GECCO ‘18
- Automated API property inference techniques, in IEEE Transactions on Software Engineering ‘13
Recommendation

How much Tapir knows about Adele’s method

- Nothing yet
- Incomplete code
- Everything

Goal

<table>
<thead>
<tr>
<th></th>
<th>Before coding</th>
<th>During coding</th>
<th>After coding</th>
</tr>
</thead>
<tbody>
<tr>
<td>Improve available documentation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Instant contextual feedback</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conformity check</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
After coding a complete method

**Strategy**

**Feedback** about potential API misuse

**Challenges**

- Does the complete client code conform to API patterns?
- What if overall client still incomplete and we don’t have execution logs?

**Current Approach**

- Model check execution traces to find errors
- Model check pseudo-traces (static analysis) at compile time to produce warnings
Recommendation

How much Tapir knows about Adele’s method

- Nothing yet
- Incomplete code
- Everything

Before coding

During coding

After coding

Goal

- Improve available documentation
- Instant contextual feedback
- Conformity check
During coding a method

**Strategy**

- Generate contextual **help** (e.g., tell user that a pattern applies)
- Propose pattern-based code **auto-completion**

**Challenges**

- How do we check if a pattern **applies** to **incomplete** code?
- How to generate auto-completion recommendations?

**Current Approach**

Recommendation

How much Tapir knows about Adele’s method:

- Nothing yet
- Incomplete code
- Everything

Before coding:

- Improve available documentation

During coding:

- Instant contextual feedback

After coding:

- Conformity check
Before coding a method

Strategy

Improve existing **documentation** with API usage patterns

Challenges

• How to translate LTL to natural language?

• Can we visualize patterns for documentation?

Current Approach

• Based on LTL semantics
• Based on examples of pattern translations, e.g., using statistical machine translation
• Represent patterns by adapting Groums or CFGs
Summary

**Recommendation Phase** | **Goal**
--- | ---
Before coding | Documentation improvements
During coding | Instant contextual feedback
After coding | Conformity check

**Constraints**
- Read and understand the documentation
- Look for example
- Seeking community support

Erick Raelijohn @ErickFifali
Michalis Famelis @Mfamelis
Houari Sahraoui @sahraouh

Tapir sketch courtesy of Sasha Kopf [CC BY-SA 3.0](https://creativecommons.org/licenses/by-sa/3.0)