Analyzing and Supporting Adaptations of Online Code Examples

Tianyi Zhang,¹ Di Yang,² Crista Lopes,² Miryung Kim¹
¹University of California, Los Angeles
²University of California, Irvine

Dataset and Tool: https://github.com/tianyi-zhang/ExampleStack-ICSE-Artifact

* Both the first author and the second author contributed significantly.
Modern Programming Workflow

Interpret Problem

Search Online

Browse & Assess

Modify Code

“how to connect to MySQL in Java?”

Brandt et al. Two studies of opportunistic programming: interleaving web foraging, learning, and writing code. 2009.
Modern Programming Workflow

Interpret Problem → Search Online → Browse & Assess → Modify Code

“how to connect to MySQL in Java?”

Brandt et al. Two studies of opportunistic programming: interleaving web foraging, learning, and writing code. 2009.
Modern Programming Workflow

Interpret Problem → Search Online → Browse & Assess → Modify Code

“how to connect to MySQL in Java?”

Brandt et al. Two studies of opportunistic programming: interleaving web foraging, learning, and writing code. 2009.
Modern Programming Workflow

Interpret Problem → Search Online → Browse & Assess → Modify Code

“how to connect to MySQL in Java?”

Brandt et al. Two studies of opportunistic programming: interleaving web foraging, learning, and writing code. 2009.
Modern Programming Workflow

Interpret Problem

Search Online

Browse & Assess

Modify Code

Test cases [CodeGenie]
I/O types [ParseWeb, Hunter]
I/O examples [Stolee et al., 2014, FlashFill]
Multimodal [Reiss, 2009]

... 

StrathCona [Homles and Murphy, 2005]
Sourcerer [Bajracharya et al., 2006]
Exemplar [McMillan et al., 2012]
FaCoy [Kim et al., 2018]

... 

Prompter [Ponzanelli et al., 2014]
AnswerBot [Xu et al., 2017]
Deprecation Watchter [Zhou et al., 2017]
ExampleCheck [Zhang et al., 2018]
Examplore [Glassman et al., 2018]
...
What we have known so far ...

• Online Code Reuse Behavior
  • Copy and paste with adaptations [Wu et al., 2018]
  • Seldom attribute to the sources of online code [Baltes and Diehl, 2018]

• Code Adaptation & Integration Support
  • Rename variables and port relevant program statements [SnipMatch, Jigsaw]
What we don’t know yet...

• RQ1. What kinds of adaptations do developers make in practice?
• RQ2. Are these adaptations done repetitively?
• RQ3. How can we provide effective tool support?
Outline

1. A Comprehensive Dataset
2. Qualitative Analysis
3. Quantitative Analysis
4. Tool Design & User Study
Outline

1. A Comprehensive Dataset
2. Qualitative Analysis
3. Quantitative Analysis
4. Tool Design & User Study
Identify Reused Stack Overflow Examples

- **Challenge**: the lack of attribution [Baltes and Diehl, 2018]

312K Stack Overflow code snippets (>= 3 LOC)

50K GitHub repos (>= 5 stars)

Identify Reused Stack Overflow Examples

- **Challenge**: the lack of attribution [Baltes and Diehl, 2018]

312K Stack Overflow code snippets (>= 3 LOC)

50K GitHub repos (>= 5 stars)

**Variation Set: Over-approximation**
- Clone Detection → Timestamp Analysis → Variation Dataset
- 14,124 potentially reused SO examples

**Adaptation Set: Under-approximation**
- Clone Detection → Timestamp Analysis → Scan SO Links → Adaptation Dataset
- 629 explicitly attributed SO examples
Outline

1. A Comprehensive Dataset
2. Qualitative Analysis
3. Quantitative Analysis
4. Tool Design & User Study
Qualitative Analysis

- Randomly sample 200 pairs of clones from each dataset
- Manually inspect their differences using GumTree [Falleri et al., 2014]
- Label program changes with short descriptions and group similar ones.
24 Frequent Adaptation Types in 6 Categories

- Code Hardening
- Resolve Compilation Error
- Exception Handling
- Logic Customization
- Refactoring
- Miscellaneous
24 Frequent Adaptation Types in 6 Categories

- Code Hardening
- Resolve Compilation Error
- Exception Handling

Insert/delete a try-catch block
Insert/delete a thrown exception in a method header
Update an exception type
Change statements in a catch/finally block
Outline

1. A Comprehensive Dataset
2. Qualitative Analysis
3. Quantitative Analysis
4. Tool Design & User Study
Automated Rule-based Classification

- Codify each adaptation type as a logic rule
  - e.g., `Insert(t_1, t_2, i) \land \text{NodeType}(t_1, \text{TryStatement}) \Rightarrow \text{Insert\_Try\_Catch\_Block}`

- 98% precision and 96% recall on another 100 clone pairs
Distribution of Common Adaptation Types

(a) Adaptations: 629 explicitly attributed SO examples

(b) Variations: 14,124 potentially reused SO examples
Finding 1. Variation patterns resemble adaptation patterns

(a) Adaptations: 629 explicitly attributed SO examples

(b) Variations: 14,124 potentially reused SO examples
Finding 2. Different GitHub clones of the same example share common adaptation types.

Stack Overflow Example

<table>
<thead>
<tr>
<th>GitHub Counterparts</th>
<th>Adaptations</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Add an if check, renaming</td>
</tr>
<tr>
<td>B</td>
<td>Add an if check, change a method call</td>
</tr>
<tr>
<td>C</td>
<td>Add an if check, renaming</td>
</tr>
<tr>
<td>D</td>
<td>Change a method call, renaming</td>
</tr>
</tbody>
</table>

![Adaptation Dataset](chart1)

- **Adaptation Dataset**
  - At least one common adaptation type: 126
  - Different adaptation types: 54
  - Percentage: 70%

- **Variation Dataset**
  - At least one common adaptation type: 6548
  - Different adaptation types: 2314
  - Percentage: 74%
Implications and Hypothesis Development

• **Implications**
  - Variations in similar code resemble real adaptations made by developers
  - Different GitHub developers make similar adaptations independently

• **Hypothesis:** Displaying variations in similar GitHub code can inspire more careful reasoning when adapting code
Outline

1. A Comprehensive Dataset
2. Qualitative Analysis
3. Quantitative Analysis
4. Tool Design & User Study
“How to calculate the distance between two coordinates?”

Based on another question on stackoverflow, I got this code. This calculates the result in meters, not in miles :)

```java
public static float distFrom(float lat1, float lng1, float lat2, float lng2) {
    double earthRadius = 6371000; //meters
    double dLat = Math.toRadians(lat2-lat1);
    double dLng = Math.toRadians(lng2-lng1);
    double a = Math.sin(dLat/2) * Math.sin(dLat/2) +
               Math.cos(Math.toRadians(lat1)) * Math.cos(Math.toRadians(lat2)) *
               Math.sin(dLng/2) * Math.sin(dLng/2);
    double c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));
    float dist = (float) (earthRadius * c);

    return dist;
}
```

[Answer by Espen Herseth Halvorsen](https://stackoverflow.com/users/39627) at May 8 '09 at 2:11
“How to calculate the distance between two coordinates?”

Based on another question on stackoverflow, I got this code. This calculation is in miles. :)

```java
public static float distFrom(float lat1, float lng1, float lat2, float lng2) {
    double earthRadius = 3958.75; //meters
    double dLat = Math.toRadians(lat2 - lat1);
    double dLng = Math.toRadians(lng2 - lng1);
    double a = Math.sin(dLat/2) * Math.sin(dLat/2) +
               Math.cos(Math.toRadians(lat1)) * Math.cos(Math.toRadians(lat2)) *
               Math.sin(dLng/2) * Math.sin(dLng/2);
    double c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1 - a));
    double dist = earthRadius * c;

    int meterConversion = 1609;

    return new Double(dist * meterConversion).floatValue();
}
```

```java
public static int distFrom(float lat1, float lng1, float lat2, float lng2) {
    double earthRadius = 6371; //kilometers
    double dLat = Math.toRadians(lat2 - lat1);
    double dLng = Math.toRadians(lng2 - lng1);
    double a = Math.sin(dLat/2) * Math.sin(dLat/2) +
               Math.cos(Math.toRadians(lat1)) * Math.cos(Math.toRadians(lat2)) *
               Math.sin(dLng/2) * Math.sin(dLng/2);
    double c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1 - a));

    return (int) (earthRadius * c);
}
```
public static float distFrom(float lat1, float lng1, float lat2, float lng2) {
    double earthRadius = 6371000; // meters
    double dLat = Math.toRadians(lat2-lat1);
    double dLng = Math.toRadians(lng2-lng1);
    double a = Math.sin(dLat/2) * Math.sin(dLat/2) +
               Math.cos(Math.toRadians(lat1)) * Math.cos(Math.toRadians(lat2)) *
               Math.sin(dLng/2) * Math.sin(dLng/2);
    double c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));
    float dist = (float) (earthRadius * c);
    return dist;
}
Within-Subjects User Study

- Sixteen students from UCLA Computer Science
- Two code reuse tasks
  - Control: view a code example and search online
  - Experiment: view similar code in GitHub using ExampleStack

<table>
<thead>
<tr>
<th>Task</th>
<th>Task Description</th>
<th>LOC</th>
<th>GitHub Clone#</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task I</td>
<td>compute the distance between two coordinates on earth</td>
<td>12</td>
<td>2</td>
</tr>
<tr>
<td>Task II</td>
<td>get the relative path of a given file and a root folder</td>
<td>74</td>
<td>2</td>
</tr>
<tr>
<td>Task III</td>
<td>encode an array of bytes to a hexadecimal string</td>
<td>12</td>
<td>17</td>
</tr>
<tr>
<td>Task IV</td>
<td>add animation to an Android view</td>
<td>29</td>
<td>4</td>
</tr>
</tbody>
</table>
Finding 1. Viewing variations in similar GitHub code inspires new adaptations that are otherwise overlooked.
Finding 1. Viewing variations in similar GitHub code inspires new adaptations that are otherwise overlooked.
Finding 2. Seeing similar code is more useful than overwhelming.

P5: “It highlights the best practices followed by the community and prioritizes the changes that I should make first”

P6: “Super nice, it seems like the fast path to reach consensus on a particular operation”

P9: “[It is] reassuring to know that the same code is used in production systems and to know the common pitfalls”

P14: “I would have completely forgotten about the null check without seeing it in a couple of [GitHub] examples”
Contributions

1. Make available a large-scale dataset of reused code between SO and GitHub.
2. Rigorously codify common adaptation patterns and create a taxonomy
3. Quantify the frequencies of common adaptations
4. Build a prototype and conduct a user study

Dataset and Tool: [https://github.com/tianyi-zhang/ExampleStack-ICSE-Artifact](https://github.com/tianyi-zhang/ExampleStack-ICSE-Artifact)