Code Duplication on Stack Overflow

ICSE 2020 NIER

Sebastian Baltes

@s_baltes

empirical-software.engineering

Sheep artwork taken from: https://wsc2020.github.io/
Thanks to my co-author!

Code Duplication on Stack Overflow

Sebastian Baltes
sebastian.baltes@adelaide.edu.au
The University of Adelaide, Australia

Christoph Treude
christoph.treude@adelaide.edu.au
The University of Adelaide, Australia

ABSTRACT
Despite the unarguable importance of Stack Overflow (SO) for the daily work of many software developers and despite existing knowledge about the impact of code duplication on software maintainability, the prevalence and implications of code clones on SO have not yet received the attention they deserve. In this paper, we motivate why studies on code duplication within SO are needed and how existing studies on code reuse differ from this new research direction. We present similarities and differences between code clones in general and code clones on SO and point to open questions that need to be addressed to be able to make data-informed decisions about how to properly handle clones on this important platform. We present results from a first preliminary investigation, indicating that clones on SO are common and diverse. We further point to specific challenges, including incentives for users to clone successful answers and difficulties with bulk edits on the platform, and conclude with possible directions for future work.

CCS CONCEPTS
• Software and its engineering → Maintaining software;

it is only recently that researchers started investigating them. Studies have shown that developers utilise code snippets from SO in their software projects, regardless of maintainability, security, and licensing implications [5–14]. The main focus of that previous work was, however, to study how and why developers (re-)use SO code snippets outside of the question-and-answer platform. While researchers worked on identifying duplicate questions [15–17], their main goal was to replace or support the manual moderator process for marking duplicate questions rather than supporting the maintenance and evolution of code on SO. Considering the importance that SO has today for the daily work of many software developers worldwide and the fact that in many posts, non-trivial code snippets are collected and maintained, it is surprising that SO does not have proper features for code versioning and bug tracking. Text and code are versioned together as Markdown content [18], making it hard to identify changes to the code snippets in the provided revision view. Furthermore, there is no language-specific syntax highlighting or error checking in SO’s online Markdown editor, leading to many snippets that are not parseable, compilable, or even runnable [2]. Finally, there is no way to report bugs in SO code snippets other than posting a comment or an alternative answer.
Yet another study about Stack Overflow?

Interestingly, there is one important aspect that previous research largely ignored...
Code Duplication on Stack Overflow from a Community Perspective
Scenario 1: Using clones to increase reputation

I could **re-post** a rather successful snippet wherever it fits **without referencing** the original answer to **accumulate** views and upvotes 😐
Scenario 1: Using clones to increase reputation

• First usage of snippet in September 2016

• Overall 31 copies of the same snippet, almost exclusively posted by the same user

• Now imagine someone finds a bug...
Scenario 2: Maintaining code copied into SO

To make it more likely that my post gets upvoted, I could copy code from external documentation resources into my answer instead of just referencing it 🤔
Scenario 2: Using clones to increase reputation

- Happens frequently
- Potential licensing issues
- What if the authoritative source gets updated?
Issues when addressing Scenarios 1 & 2

- Authors can **reject edits** linking the SO-internal clones

- SO’s **rate limiting** prevents users from bulk editing posts (they would get reverted)
Community Involvement

How to handle code clones on Stack Overflow?

As part of our research, we've reconstructed and analyzed the evolution of Stack Overflow posts. In the course of that research, I extracted clones of code blocks from the official Stack Exchange data dump released 2018-09-05 and built a small website listing clones with at least 20 lines of code that are present in at least 20 different Stack Overflow threads.

I qualitatively analyzed the top 50 clones in that list and was able to identify the source (or at least a source) of the snippets in most of the cases. Moreover, I checked if the Stack Overflow posts refer to each other, which was rarely the case. In one example, a Java snippet was found in 45 different Stack Overflow threads and could have been copied from this website. It is not clear if this external source or this Stack Overflow post is the original source of the snippet. The post on androidhive was created around May 2012, the first occurrence on Stack Overflow was mid April 2012. Assuming that androidhive is the original source, the usage on Stack Overflow could be problematic (see their terms of service). If the original source is Stack Overflow, the androidhive author did not adhere to Stack Overflow's CC BY-SA license. I identified four more variants of this snippet on Stack Overflow (1, 2, 3, 4).

Another example is this VBA script, used in 21 different Stack Overflow posts and copied from here. The first occurrence attributes this source, but I didn't find any indication on the linked website that redistribution of that snippet is allowed.

https://meta.stackoverflow.com/q/375761/1974143
### Code Snippet

```java
// Load CAs from an InputStream
// (could be from a resource or ByteArrayInputStream)
CertificateFactory cf = CertificateFactory.getInstance("X509Certificate");
InputStream caInput = new FileInputStream(new File("cacert.txt"));
try {
    Certificate ca = cf.generateCertificate(caInput);
    System.out.println("ca: " + (X509Certificate) ca);
} finally {
    caInput.close();
}

// Create a KeyStore containing our trusted CAs
String keyStoreType = "PKCS12";
KeyStore keyStore = KeyStore.getInstance(keyStoreType);
keyStore.load(null, null);
keyStore.setCertificateEntry("ca", ca);

// Create a TrustManager that trusts the CAs in our KeyStore
TrustManager tmf = new TrustManager() {
    public void checkClientTrusted(X509Certificate[] chain, String authType)
    {
    }
    public void checkServerTrusted(X509Certificate[] chain, String authType)
    {
    }
    public java.security.cert.X509Certificate[] getAcceptedIssuers()
    {
    return null;
    }
};

// Create an SSLContext that uses our TrustManager
SSLContext sslContext = SSLContext.getInstance("TLS");
sslContext.init(null, tmf.getTrustManagers(), null);

// Tell the URLConnection to use a SocketFactory from the SSLContext
URL url = new URL("https://certs.cac.washington.edu/Certificates/CA/CoR/CA-Root.pem");
HttpsURLConnection urlConnection = (HttpsURLConnection) url.openConnection();
urlConnection.setSSLSocketFactory(sslContext.getSocketFactory());
InputStream in = urlConnection.getInputStream();
copyInputStreamToOutputStream(in, System.out);
```

---

**Number of Posts:** 14

<table>
<thead>
<tr>
<th>Snippet</th>
<th>Posts (CreationDate)</th>
<th>LinkedPosts (Count)</th>
<th>ExternalLinks (Count)</th>
</tr>
</thead>
<tbody>
<tr>
<td>a21834622 (2014-02-17)</td>
<td><a href="http://developer.android.com/training">http://developer.android.com/training</a>... (4)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a2624205 (2014-10-07)</td>
<td><a href="http://developer.android.com/training">http://developer.android.com/training</a>... (2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a25557171 (2015-09-16)</td>
<td><a href="http://developer.android.com/training">http://developer.android.com/training</a>... (2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a334191878 (2015-10-29)</td>
<td><a href="http://developer.android.com/training">http://developer.android.com/training</a>... (1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a34694143 (2016-01-09)</td>
<td><a href="http://developer.android.com/training">http://developer.android.com/training</a>... (1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a54997438 (2016-01-25)</td>
<td><a href="http://developer.android.com/training">http://developer.android.com/training</a>... (1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a5161639 (2016-02-03)</td>
<td><a href="http://developer.android.com/training">http://developer.android.com/training</a>... (1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a6404405 (2016-04-04)</td>
<td><a href="http://developer.android.com/training">http://developer.android.com/training</a>... (1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a41896413 (2017-01-25)</td>
<td><a href="http://developer.android.com/training">http://developer.android.com/training</a>... (1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a4921966 (2017-02-14)</td>
<td><a href="http://developer.android.com/training">http://developer.android.com/training</a>... (1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a43093311 (2017-04-29)</td>
<td><a href="http://developer.android.com/training">http://developer.android.com/training</a>... (1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a44435159 (2017-09-08)</td>
<td><a href="http://developer.android.com/training">http://developer.android.com/training</a>... (1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a44619566 (2017-09-13)</td>
<td><a href="http://developer.android.com/training">http://developer.android.com/training</a>... (1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a50226623 (2018-05-07)</td>
<td><a href="http://developer.android.com/training">http://developer.android.com/training</a>... (1)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**Hash Value:** 8243950880252242804

**Number of Posts:** 14

---

Sebastian Baltes – Code Duplication on Stack Overflow (ICSE 2020 NIER)
Code Duplication on Stack Overflow

Sebastian Baltes
sebastian.baltes@adelaide.edu.au
The University of Adelaide, Australia

Christoph Treude
christoph.treude@adelaide.edu.au
The University of Adelaide, Australia

ABSTRACT
Despite the unarguable importance of Stack Overflow (SO) for the daily work of many software developers and despite existing knowledge about the impact of code duplication on software maintainability, the prevalence and implications of code clones on SO have not yet received the attention they deserve. In this paper, we motivate why studies on code duplication within SO are needed and how existing studies on code reuse differ from this new research direction. We present similarities and differences between code clones in general and code clones on SO and point to open questions that need to be addressed to be able to make data-informed decisions about how to properly handle clones on this important platform. We present results from a first preliminary investigation, indicating that clones on SO are common and diverse. We further point to specific challenges, including incentives for users to clone successful answers and difficulties with bulk edits on the platform, and conclude with possible directions for future work.

CCS CONCEPTS
- Software and its engineering → Maintaining software.

It is only recently that researchers started investigating them. Studies have shown that developers utilise code snippets from SO in their software projects, regardless of maintainability, security, and licensing implications [5–16]. The main focus of that previous work was, however, to study how and why developers (re-)use SO code snippets outside of the question-and-answer platform. While researchers worked on identifying duplicate questions [15–17], their main goal was to replace or support the manual moderator process for marking duplicate questions rather than supporting the maintenance and evolution of code on SO. Considering the importance that SO has today for the daily work of many software developers worldwide and the fact that in many posts, non-trivial code snippets are collected and maintained, it is surprising that SO does not have proper features for code versioning and bug tracking. Text and code are versioned together as Markdown content [18], making it hard to identify changes to the code snippets in the provided revision view. Furthermore, there is no language-specific syntax highlighting or error checking in SO’s online Markdown editor, leading to many snippets that are not parsable, compilable, or even runnable [2]. Finally, there is no way to report bugs in SO code snippets other than posting a comment or an alternative answer.