

HERMES: Using Commit-Issue Linking to Detect Vulnerability-fixing Commit

Truong-Giang Nguyen, Hong Jin Kang, David Lo
Singapore Management University

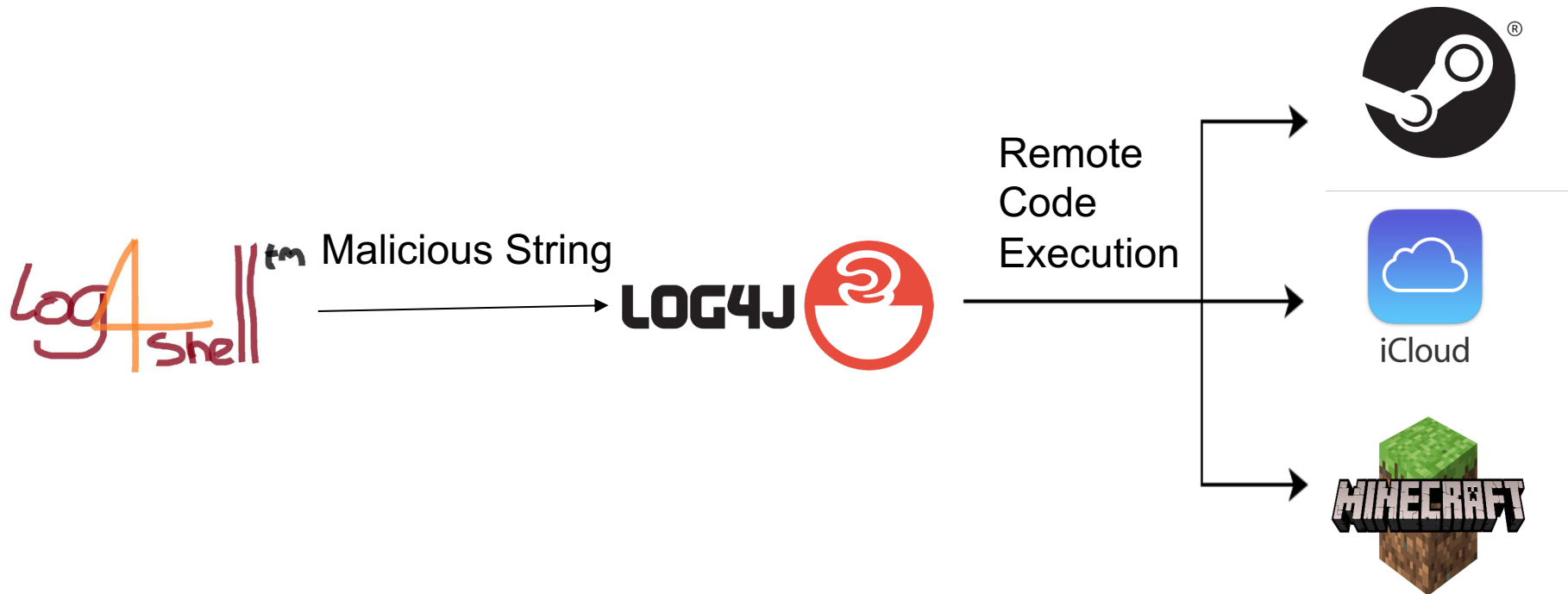
Abhishek Sharma, Andrew E. Santosa, Asankhaya Sharma, Ming Yi Ang
Veracode

Content

- Motivation and Challenge
- Approach
- Experimental result

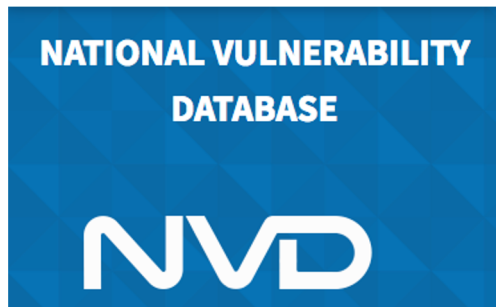
Motivation

- Modern software relies on third-party libraries
- Open Source Software (OSS) users are exposed to vulnerabilities (e.g., Log4Shell)



Motivation

- OSS users must keep up-to-date with vulnerabilities by monitoring public vulnerabilities advisories

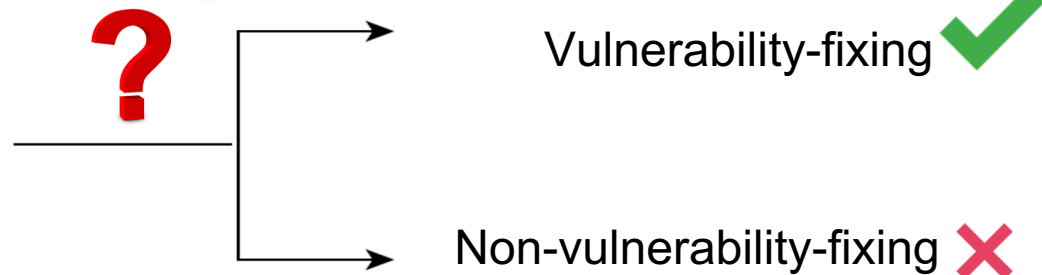


Motivation

- Vulnerability disclosure can vary from days to years
 - CVE-2018-11766 was disclosed two months later after patching
- Solution for in-time vulnerability monitoring
=> Automatically identify vulnerability-fixing commits

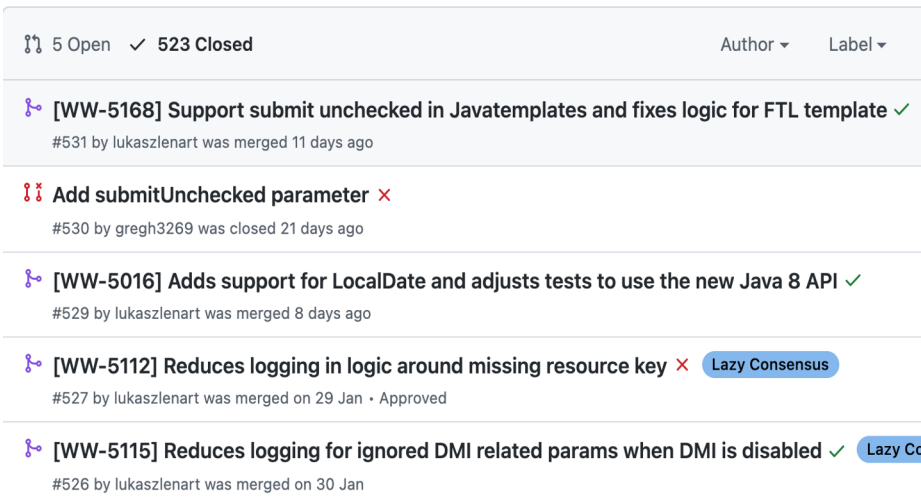


Commit



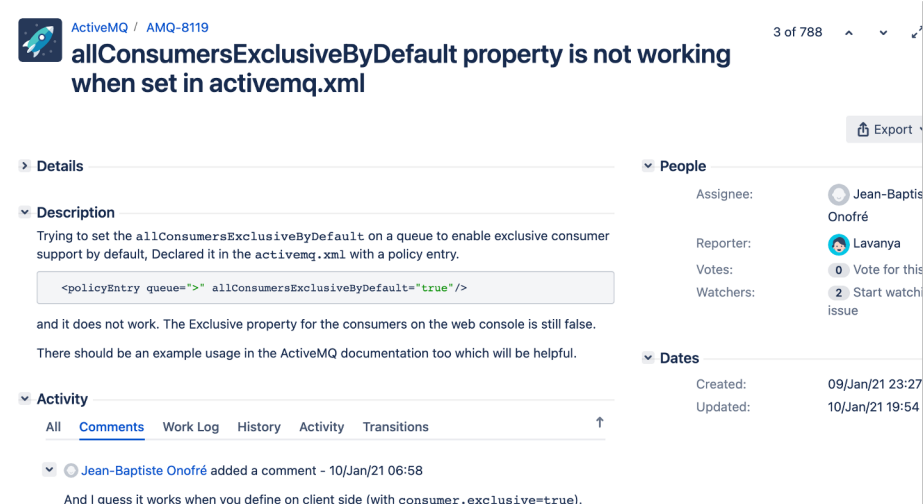
Motivation

- Propose techniques rely on commit messages and code changes
- Issue trackers contain rich source of information
 - GitHub issue
 - JIRA



5 Open ✓ 523 Closed Author ▾ Label ▾

- [\[WW-5168\] Support submit unchecked in Javatemplates and fixes logic for FTL template](#) ✓
#531 by lukaszlenart was merged 11 days ago
- [Add submitUnchecked parameter](#) ✗
#530 by gregh3269 was closed 21 days ago
- [\[WW-5016\] Adds support for LocalDate and adjusts tests to use the new Java 8 API](#) ✓
#529 by lukaszlenart was merged 8 days ago
- [\[WW-5112\] Reduces logging in logic around missing resource key](#) ✗ **Lazy Consensus**
#527 by lukaszlenart was merged on 29 Jan • Approved
- [\[WW-5115\] Reduces logging for ignored DMI related params when DMI is disabled](#) ✓ **Lazy Co**
#526 by lukaszlenart was merged on 30 Jan



ActiveMQ / AMQ-8119 3 of 788

allConsumersExclusiveByDefault property is not working when set in activemq.xml

Export

Details

Description

Trying to set the allConsumersExclusiveByDefault on a queue to enable exclusive consumer support by default, Declared it in the activemq.xml with a policy entry.

```
<policyEntry queue="*" allConsumersExclusiveByDefault="true"/>
```

and it does not work. The Exclusive property for the consumers on the web console is still false. There should be an example usage in the ActiveMQ documentation too which will be helpful.

Activity

All **Comments** Work Log History Activity Transitions ↑

Jean-Baptiste Onofré added a comment - 10/Jan/21 06:58

And I guess it works when you define on client side (with consumer.exclusive=true).

People

Assignee: Jean-Baptiste Onofré

Reporter: Lavanya

Votes: 0 Vote for this issue

Watchers: 2 Start watching this issue

Dates

Created: 09/Jan/21 23:27

Updated: 10/Jan/21 19:54

Motivation - Example

× Fix #486

master
release-1.13.1 ... release-1.12.0

decebals committed on 12 Dec 2018

```
@@ -25,6 +25,9 @@
25     import javax.xml.bind.JAXBException;
26     import javax.xml.bind.Marshaller;
27     import javax.xml.bind.Unmarshaller;
28 +   import javax.xml.stream.XMLInputFactory;
29 +   import javax.xml.stream.XMLStreamException;
30 +   import javax.xml.stream.XMLStreamReader;
31     import java.io.StringReader;
32     import java.io.StringWriter;
33
34 @@ -69,10 +72,16 @@ public String toString(Object object) {
72     public <T> T fromString(String content, Class<T> classOfT) {
73         try (StringReader reader = new StringReader(content)) {
74             JAXBContext jaxbContext = JAXBContext.newInstance(classOfT);
75             -   Unmarshaller jaxbUnmarshaller = jaxbContext.createUnmarshaller();
76
77             -   return (T) jaxbUnmarshaller.unmarshal(reader);
78             -   } catch (JAXBException e) {
79
80                 XMLInputFactory xmlInputFactory = XMLInputFactory.newFactory();
81                 xmlInputFactory.setProperty(XMLInputFactory.IS_SUPPORTING_EXTERNAL_ENTITIES, true);
82                 xmlInputFactory.setProperty(XMLInputFactory.SUPPORT_DTD, true);
83                 XMLStreamReader xmlStreamReader = xmlInputFactory.createXMLStreamReader(content);
84                 Unmarshaller unmarshaller = jaxbContext.createUnmarshaller();
85
86                 return (T) unmarshaller.unmarshal(xmlStreamReader);
87             } catch (JAXBException | XMLStreamException e) {
88                 throw new PippoRuntimeException(e, "Failed to deserialize content to '{content}'");
89             }
90         }
91     }
92 }
```

Closed

xxe vulnerabilities #486

QiAnXinCodeSafe opened this issue on 11 Dec 2018 · 3 comments

Hello, I am a member of the 360 Code Guard team. In our open source project code audit, we found that Pippo has xxE vulnerabilities. Details are as follows.
pippo/pippo-content-type-parent/pippo-jaxb/src/main/java/ro/pippo/jaxb/JaxbEngine.java

Because the XML parser does not disable dtd, xxE attacks can occur when content parameters are controlled by malicious attackers



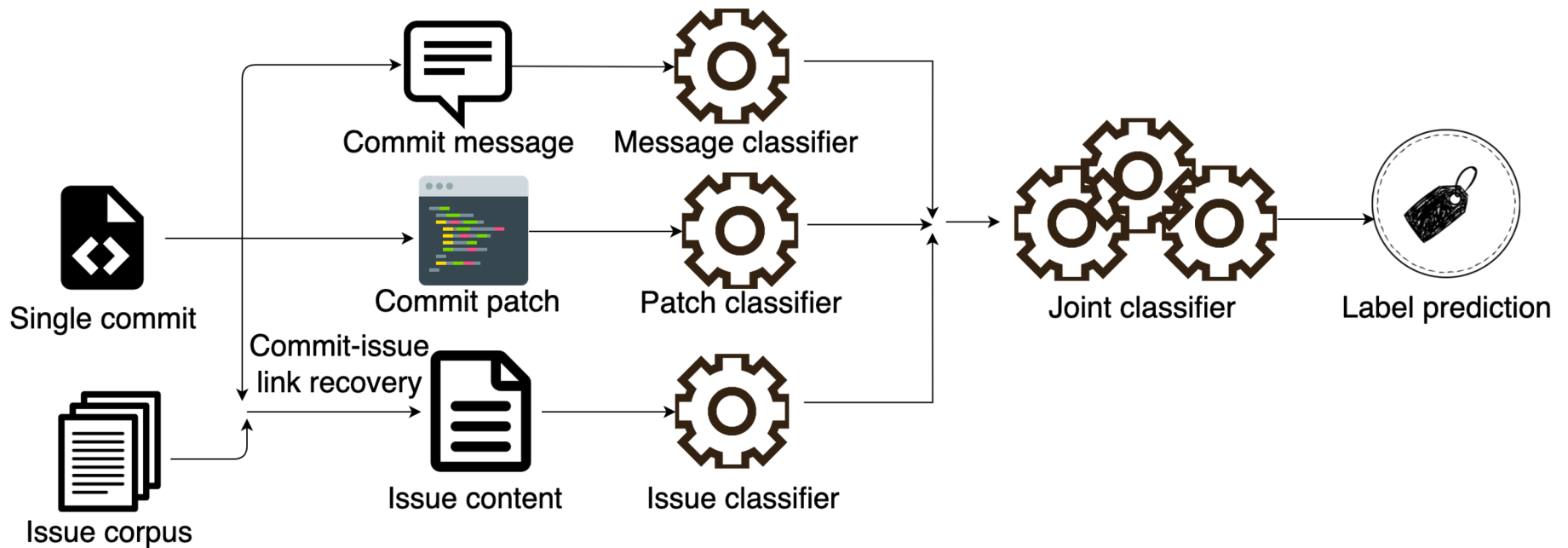
Challenge

- However, many commits are not linked to issues
 - The proportion of unlinked commits have been reported from 35% to 40%
 - In our dataset, nearly 63% of commits are unlinked
- Solution: HERMES uses existing commit-issue link recovery technique to infer links between each unlinked commit and an issue that best matches it

Approach

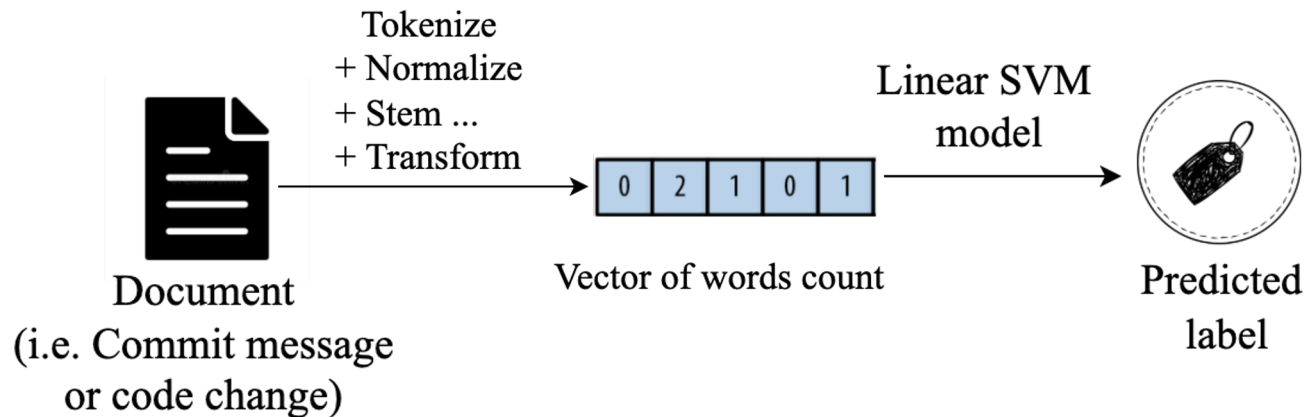
- High-level design of HERMES consists of:
 - An issue linker
 - Three base classifiers (message classifier, code change classifier, issue classifier)
 - A joint classifier

Approach



Approach

- Message Classifier + Patch Classifier
 - Proposed by Sabetta et al. [1]



[1] Sabetta, A., & Bezzi, "A practical approach to the automatic classification of security-relevant commits", ICSME 2018

Approach

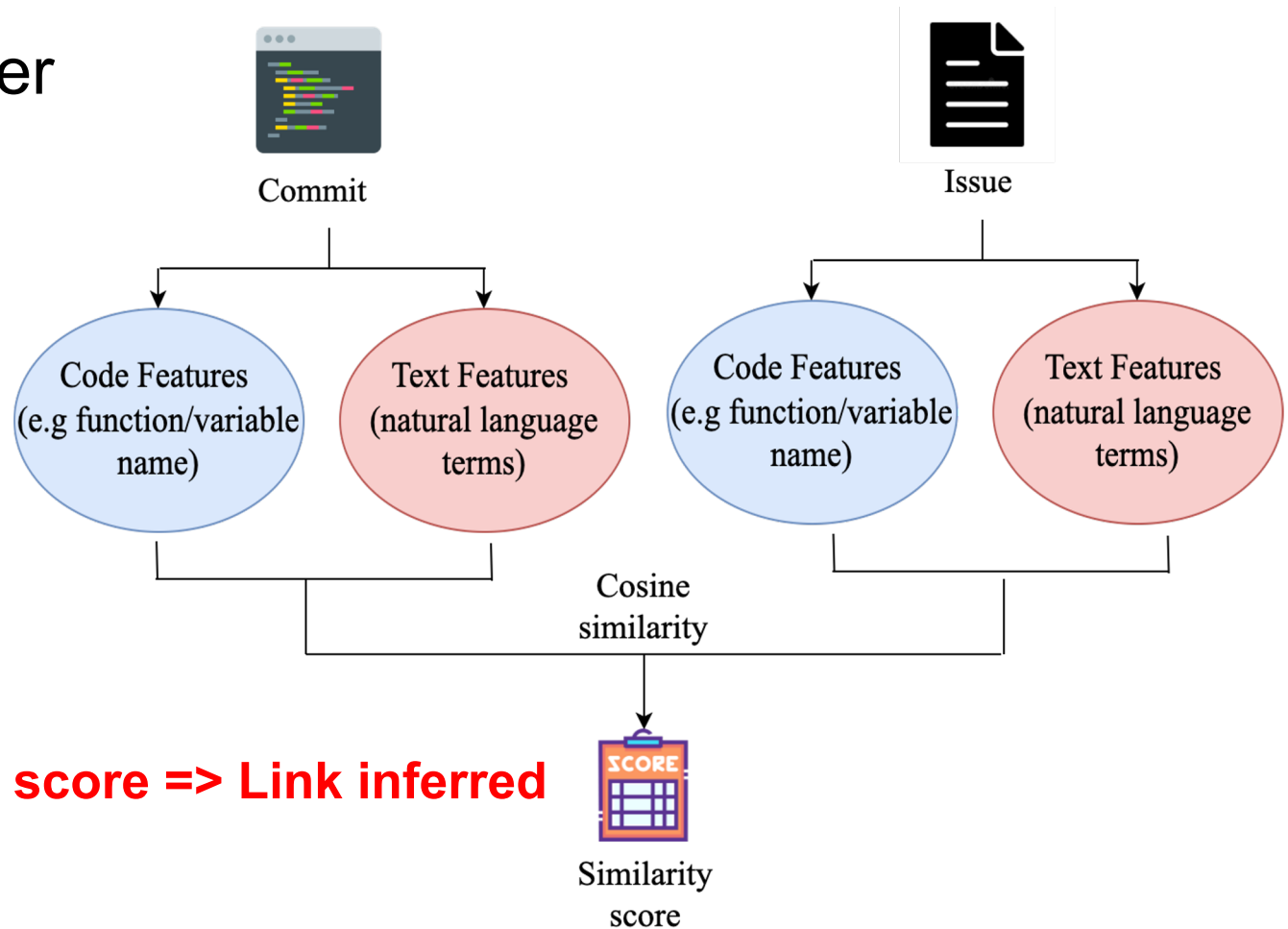
- Issue Classifier

- Commit with explicit links to issues
 - Manually identify relevant issue trackers
 - Use regular expression for matching issue ID in commit message (e.g. "CAMEL-16527", "WW-4348", "STS-262")
- Commit without explicitly link to any issue
 - Build a corpus of over 290k issues from multiple projects in the dataset
 - Implement FRLink[1] as an issue linker

[1] Sun, Yan and Wang, Qing and Yang, Ye, "Frlink: Improving the recovery of missing issue-commit links by revisiting file relevance", Information and Software Technology 2017

Approach

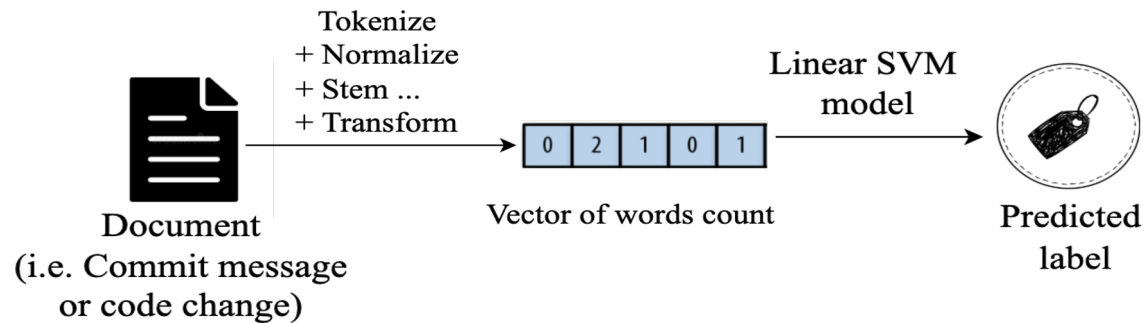
- Issue linker



Highest similarity score => Link inferred

Approach

- Issue Classifier (follow up)
 - Issue content extraction
 - GitHub issue: Title, Body, Comment(s)
 - JIRA issue: Summary, Description, Comment(s)



- Joint classifier
 - Employs a Logistic Regression Classifier to combine outputs of three base classifier

Experimental Result

- How effective is HERMES for commits with explicit links?
 - Evaluate HERMES on subset of dataset where commit and issue are linked by commit authors

Model	Precision	Recall	F1
Sabetta et al. [1]	0.54	0.82	0.64
HERMES	0.8	0.67	0.72

Experimental Result

- How effective is HERMES when leveraging commit-issue link recovery technique?
 - Evaluate HERMES on the full dataset after performing commit-issue linking

Model	Precision	Recall	F1
Sabetta et al. [1]	0.52	0.81	0.63
HERMES	0.74	0.66	0.70

Thank for watching