

# Towards Detecting Inconsistent Comments in Java Source Code Automatically

*Nataliia Stulova, Arianna Blasi, Alessandra Gorla, Oscar Nierstrasz*

SCAM'20: 20th IEEE International Working Conference on  
Source Code Analysis and Manipulation

27 September 2020

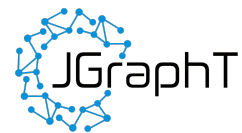


# Problem: commits introducing inconsistencies

— — —

```
/**
 * Checks if one of the graphs is from unsupported graph type and throws
 * IllegalArgumentException if it is. The current unsupported types are
 * graphs with multiple-edges.
 *
 * @param graph1
 * @param graph2
 *
 * @throws IllegalArgumentException
 */
protected static void assertUnsupportedGraphTypes(
    Graph g1,
    Graph g2)

    throws IllegalArgumentException
{...}
```



# Problem: commits introducing inconsistencies

---

```
/**
 * Checks if one of the graphs is from unsupported graph type and throws
 * IllegalArgumentException if it is. The current unsupported types are
 * graphs with multiple-edges.
 *
 * @param graph1
 * @param graph2
 *
 * @throws IllegalArgumentException
 */
protected static void assertUnsupportedGraphTypes(
-   Graph g1,
-   Graph g2)
+   Graph g)
    throws IllegalArgumentException
{...}
```



March 2006  b4805f5

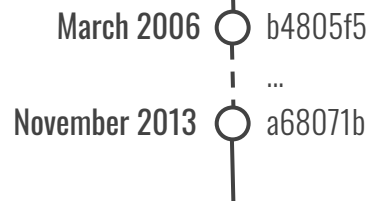
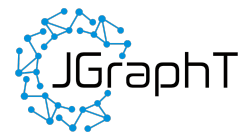
code and  
comment  
diverged!

# Problem: commits introducing inconsistencies

---

```
/**
 * Checks if one of the graphs is from unsupported graph type and throws
 * IllegalArgumentException if it is. The current unsupported types are
 * graphs with multiple-edges.
 *
 * - * @param graph1
 * - * @param graph2
 * + * @param g
 *
 * * @throws IllegalArgumentException
 */
protected static void assertUnsupportedGraphTypes(
    Graph g)

    throws IllegalArgumentException
{...}
```

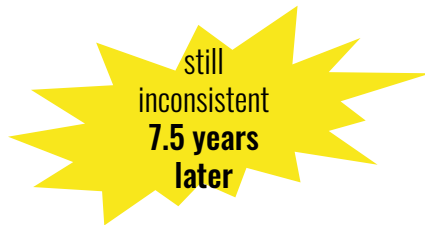


# Problem: commits introducing inconsistencies

— — —

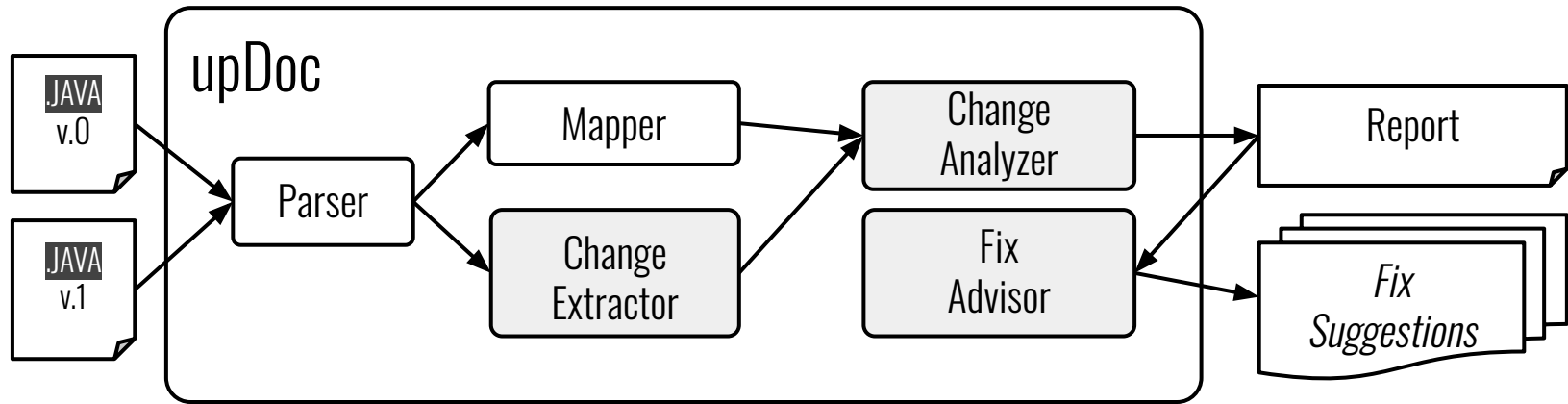
```
/**
 * Checks if one of the graphs is from unsupported graph type and throws
 * IllegalArgumentException if it is. The current unsupported types are
 * graphs with multiple-edges.
 *
 *
 *
 * @param g
 *
 * @throws IllegalArgumentException
 */
protected static void assertUnsupportedGraphTypes(
    Graph g)

    throws IllegalArgumentException
{...}
```



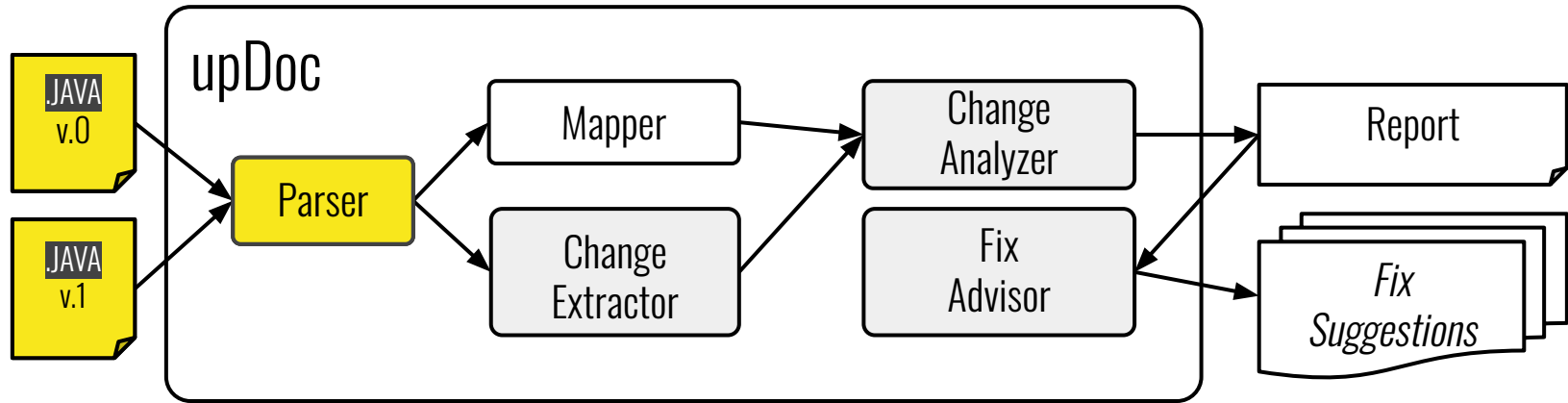
# Detecting inconsistent comments on change with NLP

---



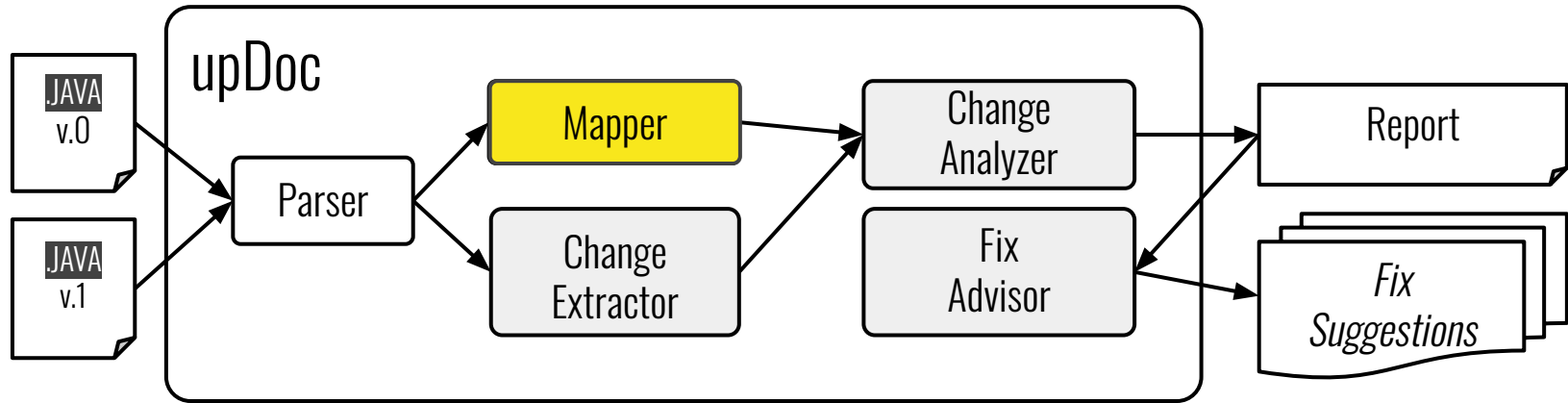
# Detecting inconsistent comments on change with NLP

---



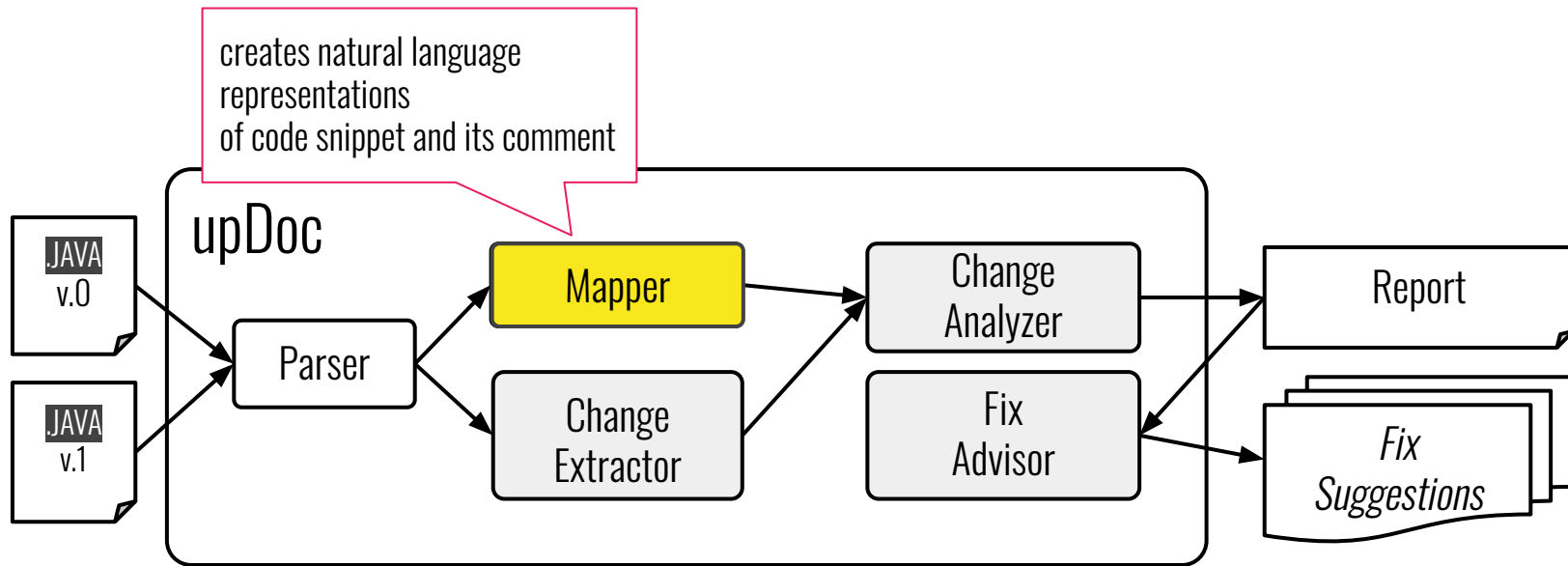
# Detecting inconsistent comments on change with NLP

---

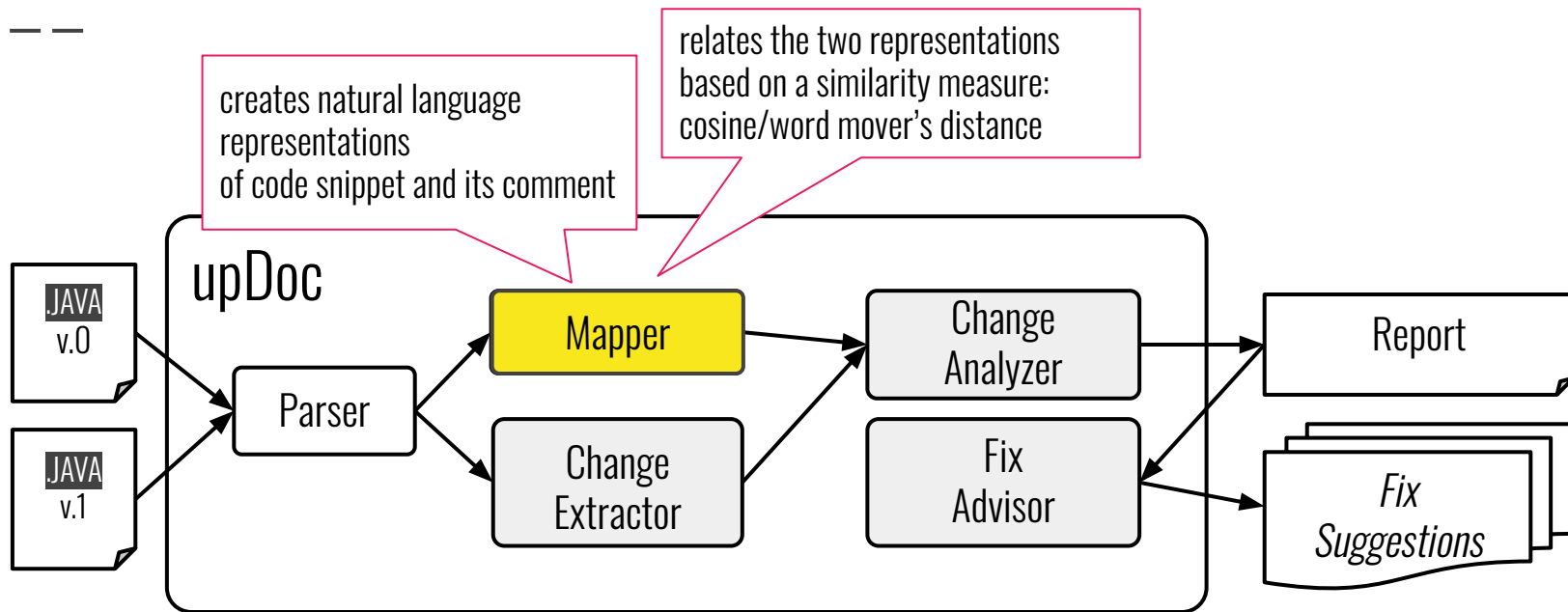




# Detecting inconsistent comments on change with NLP

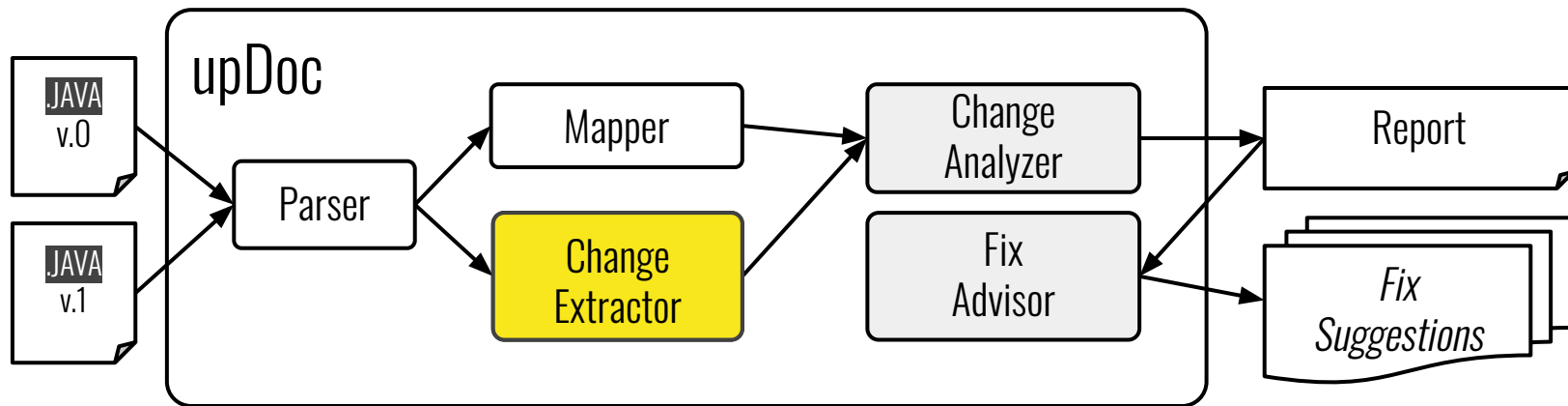


# Detecting inconsistent comments on change with NLP



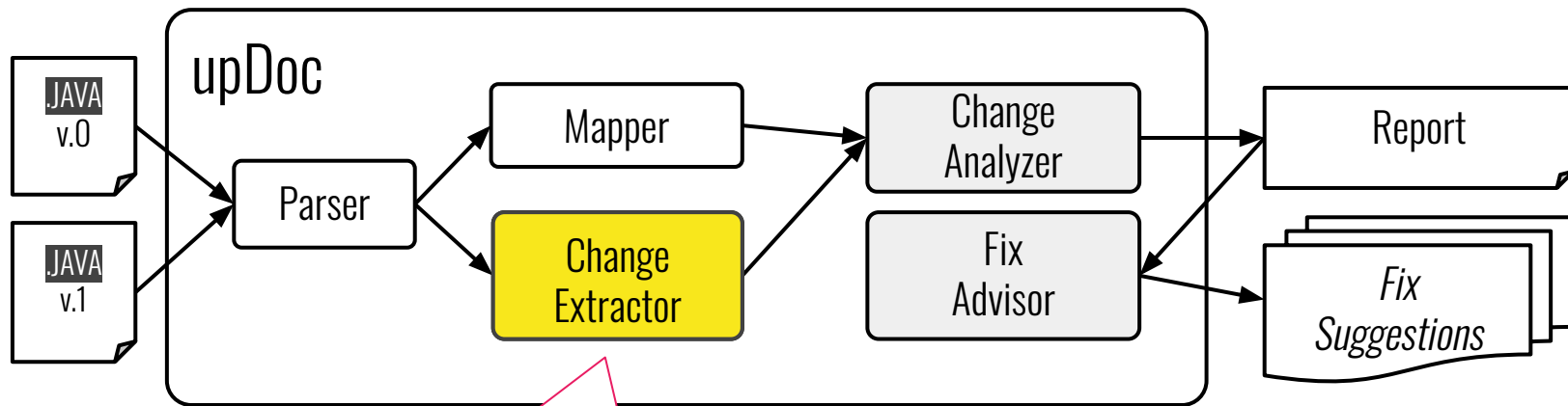
# Detecting inconsistent comments on change with NLP

---



# Detecting inconsistent comments on change with NLP

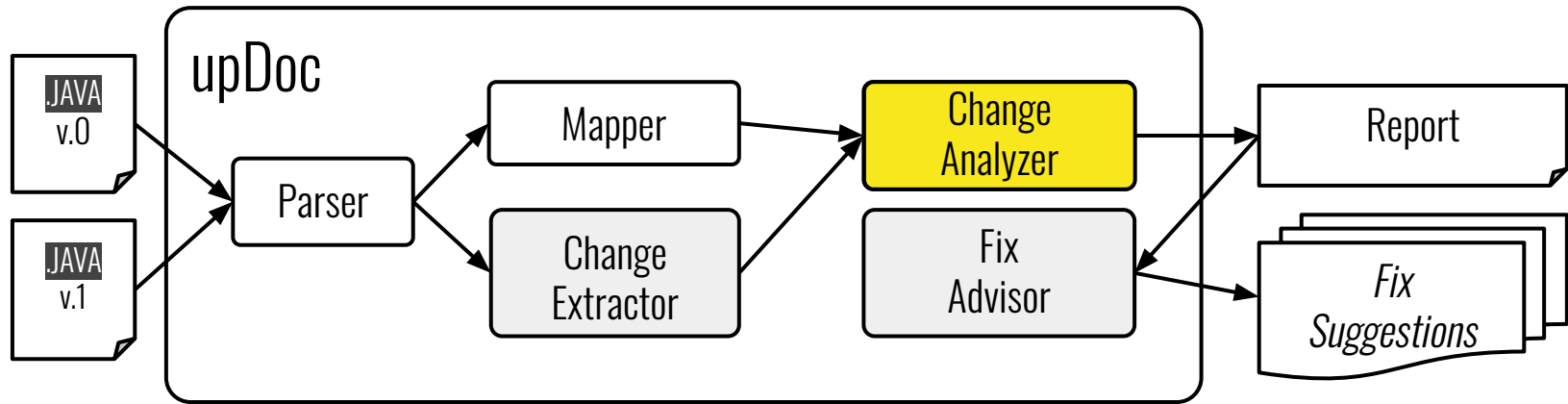
---



works with AST-based diffs for clear code snippet scopes

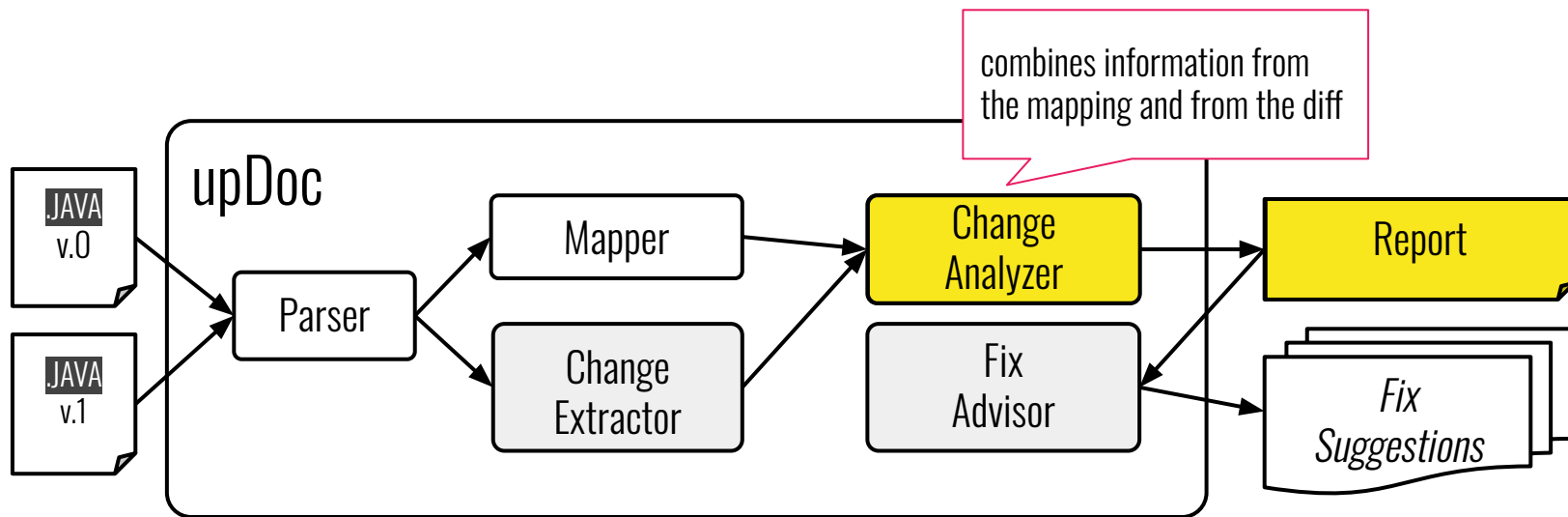
# Detecting inconsistent comments on change with NLP

---



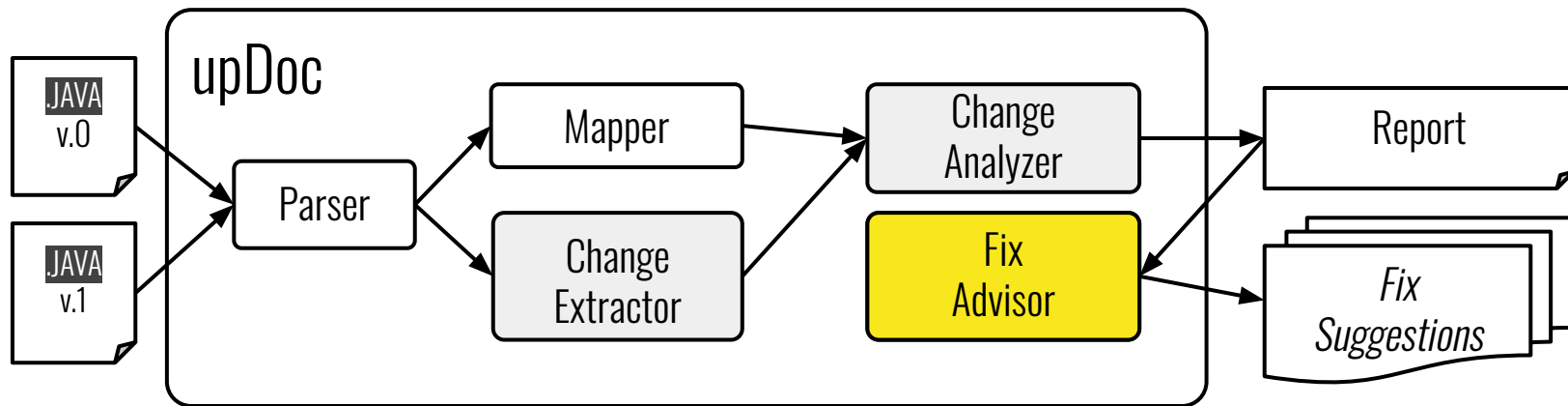
# Detecting inconsistent comments on change with NLP

---



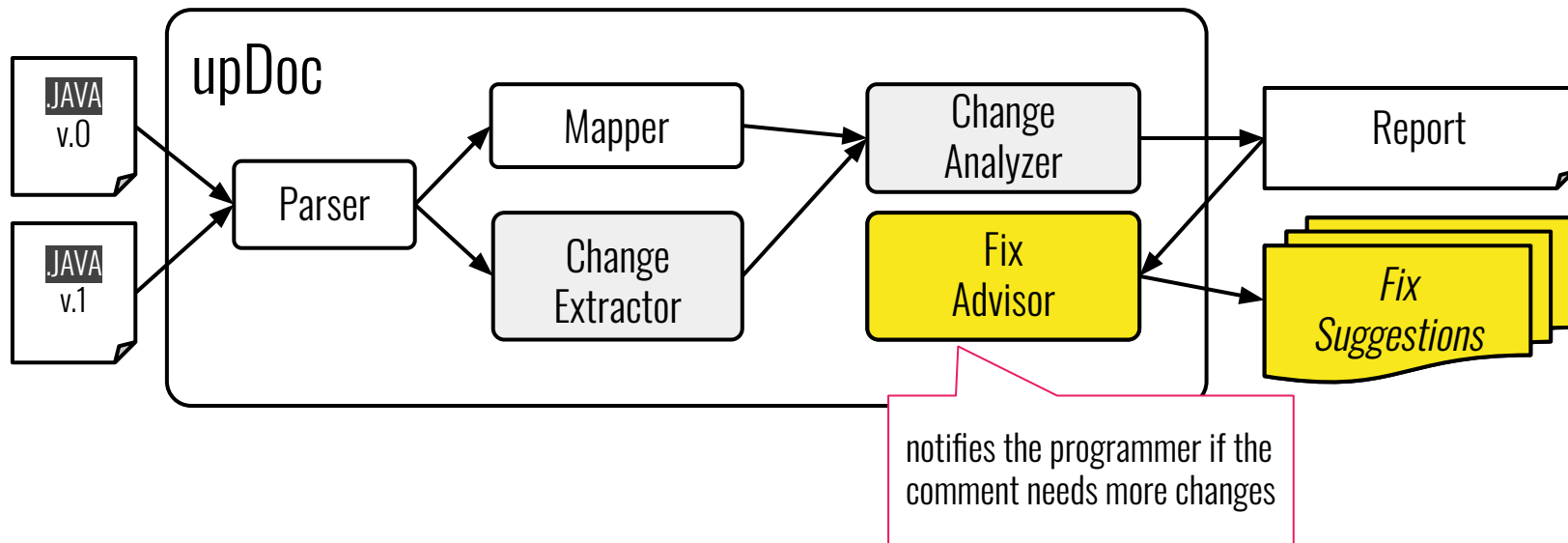
# Detecting inconsistent comments on change with NLP

---



# Detecting inconsistent comments on change with NLP

---





# The core part: code-comment mapping

---

stems

**Step 1:** extract *natural language cues* from both code and comments and store as a bag-of-words

**Step 2:** measure similarity between two bags-of-words, add a mapping link if it is high enough

comment { [1:check, 1:one, 2:graph, 1:unsupport, 1:type,  
1:throw, 1:illeg, 1:argument, 1:except ]

code v.0 { [1:void, 1:assert, 1:unsupport, 4:graph, 1:type,  
1:throw, 1:illeg, 1:argument, 1:except, 1:first, 1:second]

code v.1 { [1:void, 1:assert, 1:unsupport, 2:graph, 1:type,  
1:throw, 1:illeg, 1:argument, 1:except, 1:g,]

Word Mover's  
Distance

WMD similarity


70%

66% (-4%)

# Preliminary evaluation of mapping accuracy

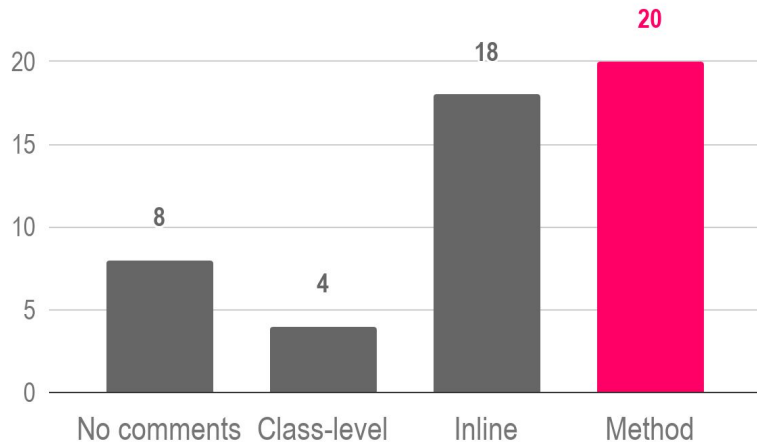
---

## Data

- use public inconsistent documentation fixes dataset from ICPC'19 produced by REVEAL group from USI
- take first 50 commits and filter out commits with fixes upDoc does not detect (yet) 
- analyze **67 fixes** in **20 commits** remaining

## Hypothesis

upDoc would report higher similarity scores in a mapping for the fixed version

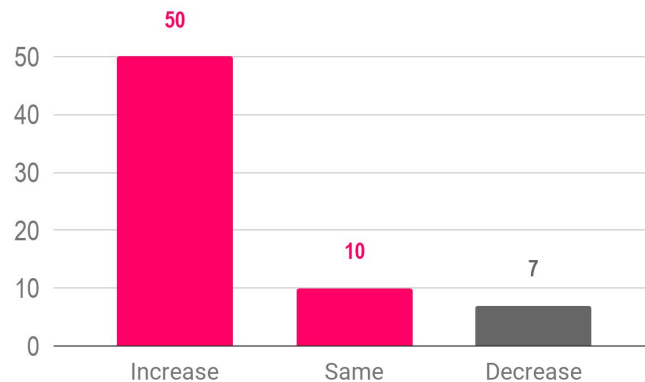


# Preliminary evaluation of mapping accuracy

— — —

## Results

- 50 fixes - similarity scores improve as expected
- 10 fixes - the similarity scores did not change (formatting fixes)
- 7 fixes - unexpected decreases in similarity scores (not supported by upDoc)

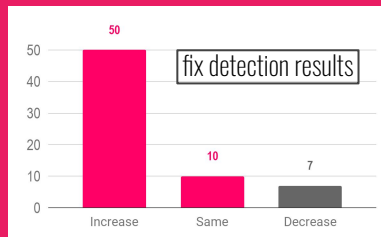
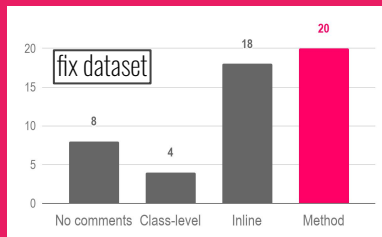
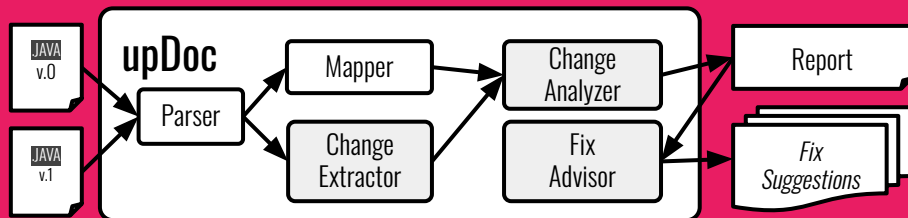


## Hypothesis **CONFIRMED**

upDoc would report higher similarity scores in a mapping for the fixed version

# Towards Detecting Inconsistent Comments in Java Source Code Automatically

```
/**  
 * Checks if one of the graphs is from unsupported graph type and throws  
 * IllegalArgumentException if it is. The current unsupported types are  
 * graphs with multiple-edges.  
 *  
 * @param g  
 *  
 * @throws IllegalArgumentException  
 */  
protected static void assertUnsupportedGraphTypes(  
    Graph g)  
    throws IllegalArgumentException  
{...}
```



## Further discussion:

**Q1:** there are many approaches for automatic comment generation, yet can we improve them with our technique to capture implementation faults earlier?

**Q2:** machine learning is used more and more in software engineering tasks, yet is it an alternative or more of a complement to our approach?