

Code Readability Assessment and Improvement: Human and Automated Perspectives



Carlos Eduardo C. Dantas
carlooseduardodantas@iftm.edu.br



Marcelo de Almeida Maia
marcelo.maia@ufu.br

<http://carlooseduardoxp.github.io>

<http://isel.ufu.br/>

Code Snippet

Developers often search for high-quality, reusable code snippets online

How to insert an item into an array at a specific index?

You want the `splice` function on the native array object.

5652 `arr.splice(index, 0, item);` will insert `item` into `arr` at the specified `index` (deleting `0` items first, that is, it's just an insert).

In this example we will create an array and add an element to it into index 2:


```
var arr = [];  
arr[0] = "Jani";  
arr[1] = "Hege";  
arr[2] = "Stale";  
arr[3] = "Kai Jim";  
arr[4] = "Borge";  
  
console.log(arr.join()); // Jani,Hege,Stale,Kai Jim,Borge  
arr.splice(2, 0, "Lene");  
console.log(arr.join()); // Jani,Hege,Lene,Stale,Kai Jim,Borge
```

 Run code snippet

 Expand snippet



CROKAGE: effective solution recommendation for programming tasks by leveraging crowd knowledge

Rodrigo Fernandes Gomes da Silva¹ · Chanchal K. Roy² ·
Mohammad Masudur Rahman² · Kevin A. Schneider² · Klérisson Paixão¹ ·
Carlos Eduardo de Carvalho Dantas¹ · Marcelo de Almeida Maia¹ 

<https://doi.org/10.1007/s10664-020-09863-2>

Improved Retrieval of Programming Solutions With Code Examples Using a Multi-featured Score

Rodrigo F. Silva^a, Mohammad Masudur Rahman^b, Carlos Eduardo Dantas^{a,c},
Chanchal Roy^d, Foutse Khomh^e, Marcelo A. Maia^a

^a*Faculty of Computing, Federal University of Uberlândia, Brazil*

^b*Faculty of Computer Science, Dalhousie University, Canada*

^c*Instituto Federal do Triângulo Mineiro, Brazil*

^d*Department of Computer Science, University of Saskatchewan, Canada*

^e*École Polytechnique de Montréal, Canada*

Describe the Java programming task

Task:

Search

Clean

Num. of Answers: ☐ 10 ☒ 5 ☐ 1

How do you like the overall result?



Post id: 17846344

java arrays sum

In java-8 you can use streams:

```
int[] a = {10,20,30,40,50};
int sum = IntStream.of(a).sum();
System.out.println("The sum is " + sum);
```

Output:

The sum is 150.

It's in the package `java.util.stream`

```
import java.util.stream.*;
```

Post id: 22078933

java arrays sum

I use this:

```
public static long sum(int[] i_arr)
{
    long sum;
    int i;
    for(sum= 0, i= i_arr.length - 1; 0 <= i; sum+= i_arr[i--]);
    return sum;
}
```

<http://isel.ufu.br:9000/>

 AUGUST 14, 2019

CROKAGE: A New Way to Search Stack Overflow



Ben Popper ›

Staff



<https://stackoverflow.blog/2019/08/14/crokage-a-new-way-to-search-stack-overflow/>



Users from 80 countries garnered 15,865 queries



APISonar: Mining API usage examples

Andre Hora 

Department of Computer Science, Federal University of Minas Gerais, Belo Horizonte, Brazil

Correspondence

Andre Hora, Department of Computer Science, Federal University of Minas Gerais, Belo Horizonte, Brazil.
Email: andrehora@dcc.ufmg.br

DOI: 10.1002/spe.2906

3 Ranking API usage examples (for each API)

Compute rank

API similarity
Readability
Reusability



Rank API examples



Contents lists available at [ScienceDirect](#)

The Journal of Systems & Software

journal homepage: www.elsevier.com/locate/jss



Characterizing top ranked code examples in Google[☆]

Andre Hora

Department of Computer Science, Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, Brazil





Code Readability

It is a human judgment of how easy a text is to understand (Buse et al., 2010)





Readability Models

The readability models aim to measure the effort required to read code on single snapshots

A Comprehensive Model for Code Readability

Simone Scalabrino¹, Mario Linares-Vásquez², Rocco Oliveto¹, and Denys Poshyvanyk³

¹ *University of Molise, Pesche (IS), Italy*

² *Universidad de los Andes, Bogotá, Colombia*

³ *The College of William and Mary, Williamsburg, Virginia, USA*

<https://doi.org/10.1002/smr.1958>

A Model to Detect Readability Improvements in Incremental Changes

Devjeet Roy
Washington State
University
Washington, USA
devjeet.roy@wsu.edu

Sarah Fakhoury
Washington State
University
Washington, USA
sarah.fakhoury@wsu.edu

John Lee
Washington State
University
Washington, USA
john.k.lee@wsu.edu

Venera Arnaoudova
Washington State
University
Washington, USA
venera.arnaoudova@wsu.edu

<https://dl.acm.org/doi/10.1145/3387904.3389255>

ASAT

sonarlint

Set of metrics
that can identify potential
readability violations in code

Rule

Array designators "[]" should be on the type, not the variable

Analyze your code

Code Smell Minor Quick Fix convention


Array designators should always be located on the type for better code readability. Otherwise, developers must look both at the type and the variable name to know whether or not a variable is an array.

Noncompliant Code Example

```
int matrix[][];    // Noncompliant
int[] matrix[];   // Noncompliant
```

Compliant Solution

```
int[][] matrix;   // Compliant
```

Automated Code Readability Improvements

Using Deep Learning to Automatically Improve Code Readability

Antonio Vitale, Valentina Piantadosi, Simone Scalabrino, Rocco Oliveto
STAKE Lab @ University of Molise, Italy

doi: 10.1109/ASE56229.2023.00112



Limitations

Most readability models don't accurately reflect the real-world code improvements made by developers.

We presented an empirical study investigating the correlation between code understandability and 121 metrics related to the code itself, to the documentation available for it, and to the developer understanding it. We asked 46 developers to understand 50 Java snippets, and we gathered a total of 324 evaluations. We assessed the *perceived* and *actual* participants' understanding for each snippet they inspected and the time they needed for the comprehension process. Our results demonstrate that, in most of the cases, there is no correlation between the considered metrics and code understandability.

DOI: [10.1109/ASE.2017.8115654](https://doi.org/10.1109/ASE.2017.8115654)

We found that, more often than not, the considered quality metrics are not able to capture the quality improvement as perceived by developers.

DOI: [10.1109/ICSME.2018.00017](https://doi.org/10.1109/ICSME.2018.00017)



Limitations

How Does Code Readability Change During Software Evolution?

Citation for published version (APA):

Piantadosi, V., Fierro, F., Scalabrino, S., Serebrenik, A., & Oliveto, R. (2020). How Does Code Readability Change During Software Evolution? *Empirical Software Engineering*, 25(6), 5374-5412.
<https://doi.org/10.1007/s10664-020-09886-9>

Summary of RQ_1 . The tool by Scalabrino et al. (2018) is not reliable when the output is in the range [0.416, 0.600].

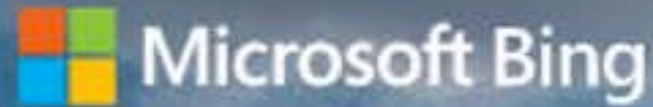
Summary of RQ_2 . There is a loss of accuracy when using readability prediction tools for transitions instead of single versions.

Find maximum element of ArrayList in java



```
public class ArrayListMaxValue {  
  
    public static void main(String[] args) {  
  
        // creating list.  
        List<Integer> intValues = new ArrayList<>();  
  
        // adding values  
        intValues.add(6);  
        intValues.add(64);  
        intValues.add(45);  
        intValues.add(60);  
        intValues.add(9);  
        intValues.add(2);  
  
        // calling max() method.  
        Integer max = Collections.max(intValues);  
        System.out.println("ArrayList values : " + intValues);  
        System.out.println("ArrayList max value : " + max);  
    }  
}
```

0.67



```
// Finding Maximum Element of Java ArrayList  
import java.util.ArrayList;  
import java.util.Collections;  
  
class MinElementInArrayList {  
  
    public static void main(String[] args)  
    {  
        // ArrayList of Numbers  
        ArrayList<Integer> myList  
            = new ArrayList<Integer>();  
  
        // adding elements to Java ArrayList  
        myList.add(16);  
        myList.add(26);  
        myList.add(3);  
        myList.add(52);  
        myList.add(70);  
        myList.add(12);  
  
        int maximum = myList.get(0);  
        for (int i = 1; i < myList.size(); i++) {  
            if (maximum < myList.get(i))  
                maximum = myList.get(i);  
        }  
        System.out.println("Maximum Element in ArrayList = "  
            + maximum);  
    }  
}
```

0.51

Crokage

Describe the Java programming task

```
public class MaxList {  
    public static void main(String[] args) {  
        List l = new ArrayList();  
        l.add(1);  
        l.add(2);  
        l.add(3);  
        l.add(4);  
        l.add(5);  
        System.out.println(Collections.max(l)); // 5  
        System.out.println(Collections.min(l)); // 1  
    }  
}
```

0.55



Limitations

Are Static Analysis Violations Really Fixed? A Closer Look at Realistic Usage of SonarQube

Diego Marcilio^ϕ, Rodrigo Bonifácio^{ϕψ}, Eduardo Monteiro^δ, Edna Canedo^ϕ, Welder Luz^ϕ and Gustavo Pinto^ω

^ϕComputer Science Department, University of Brasília, Brasília, Brazil

^ψPaderborn University, Paderborn, Germany

^δDepartment of Statistics, University of Brasília, Brasília, Brazil

^ωFaculty of Computing, Federal University of Pará, Belém, Brazil

<https://doi.org/10.1109/ICPC.2019.00040>

Surprisingly, we found a low resolution rate per project in all organizations—on average, 13% of the issues have been solved in the systems.



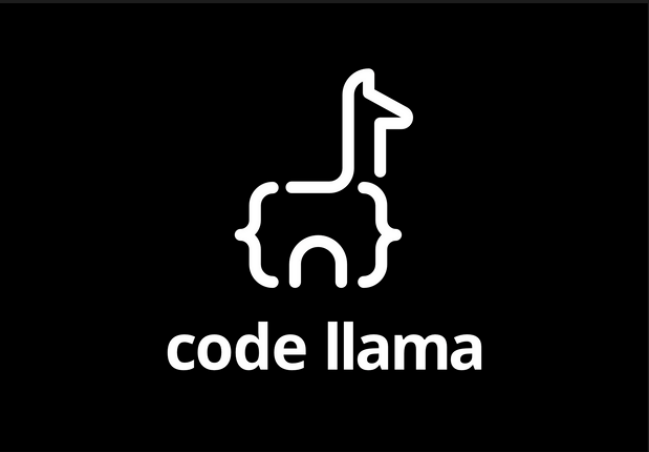
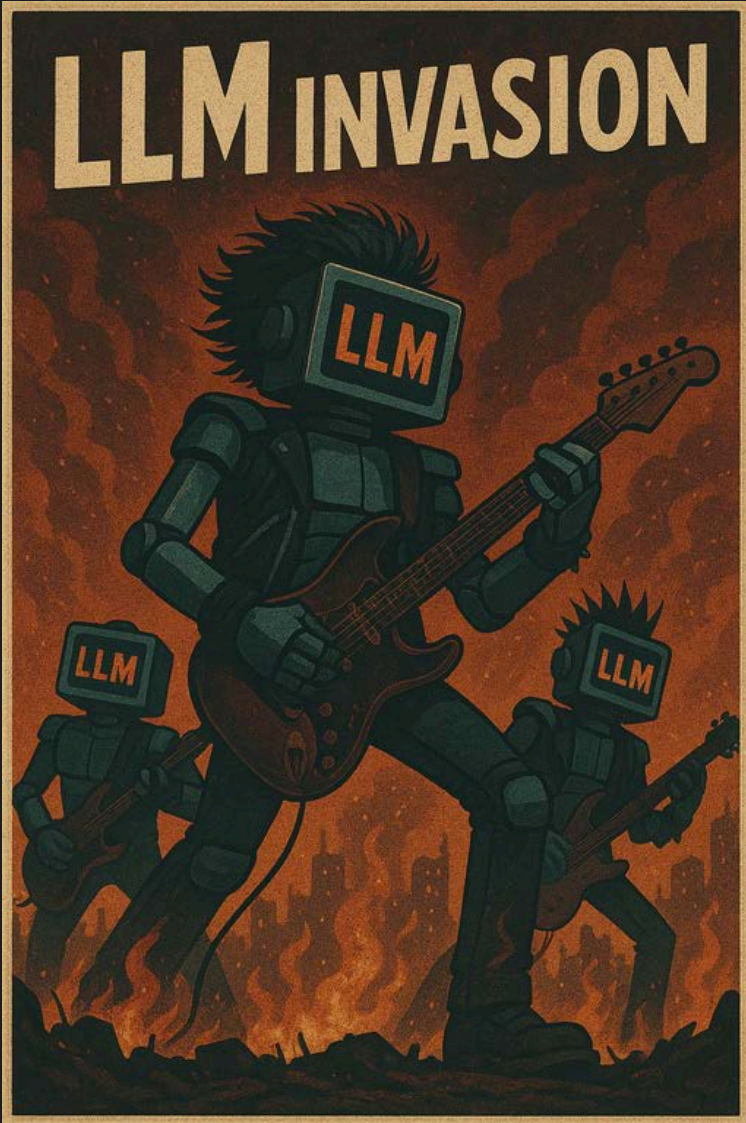
Limitations

Understanding Code Understandability Improvements in Code Reviews

Delano Oliveira, Reydne Santos, Benedito de Oliveira, Martin Monperrus, Fernando Castor, and Fernanda Madeiral

doi: 10.1109/TSE.2024.3453783

Answer to RQ6. We found 89 out of 323 occurrences of understandability smells that could be detected by the linters. Some of the remaining 234 occurrences of understandability smells could be detected by creating new rules in these linters, while other occurrences require a deeper and context-dependent analysis that goes beyond what linters typically do.



Unveiling ChatGPT's Usage in Open Source Projects: A Mining-based Study

Rosalia Tufano*
SEART @ Software Institute
Università della Svizzera italiana
Switzerland

Antonio Mastropaolo*
SEART @ Software Institute
Università della Svizzera italiana
Switzerland

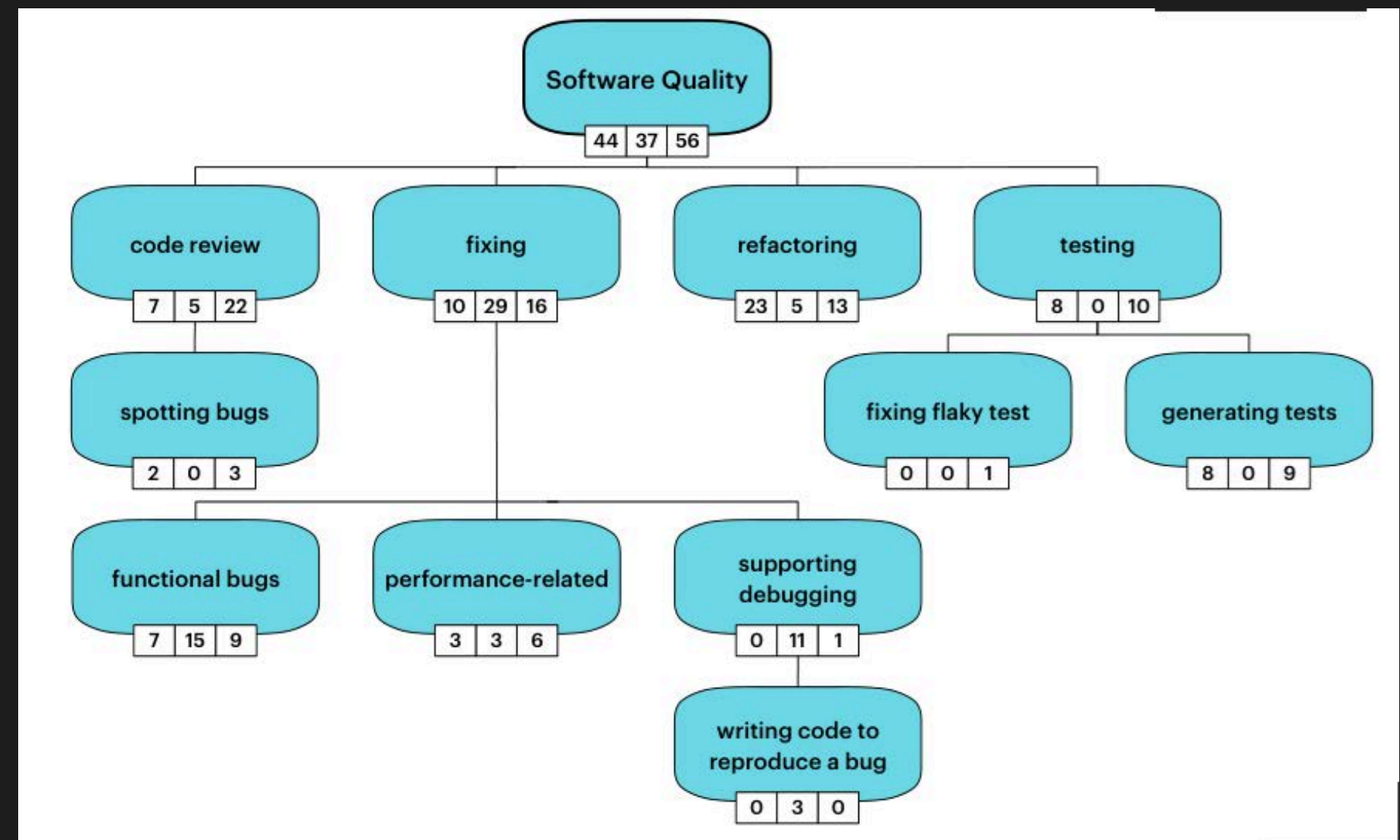
Federica Pepe
Department of Engineering
University of Sannio
Italy

Ozren Dabić
SEART @ Software Institute
Università della Svizzera italiana
Switzerland

Massimiliano Di Penta
Department of Engineering
University of Sannio
Italy

Gabriele Bavota
SEART @ Software Institute
Università della Svizzera italiana
Switzerland

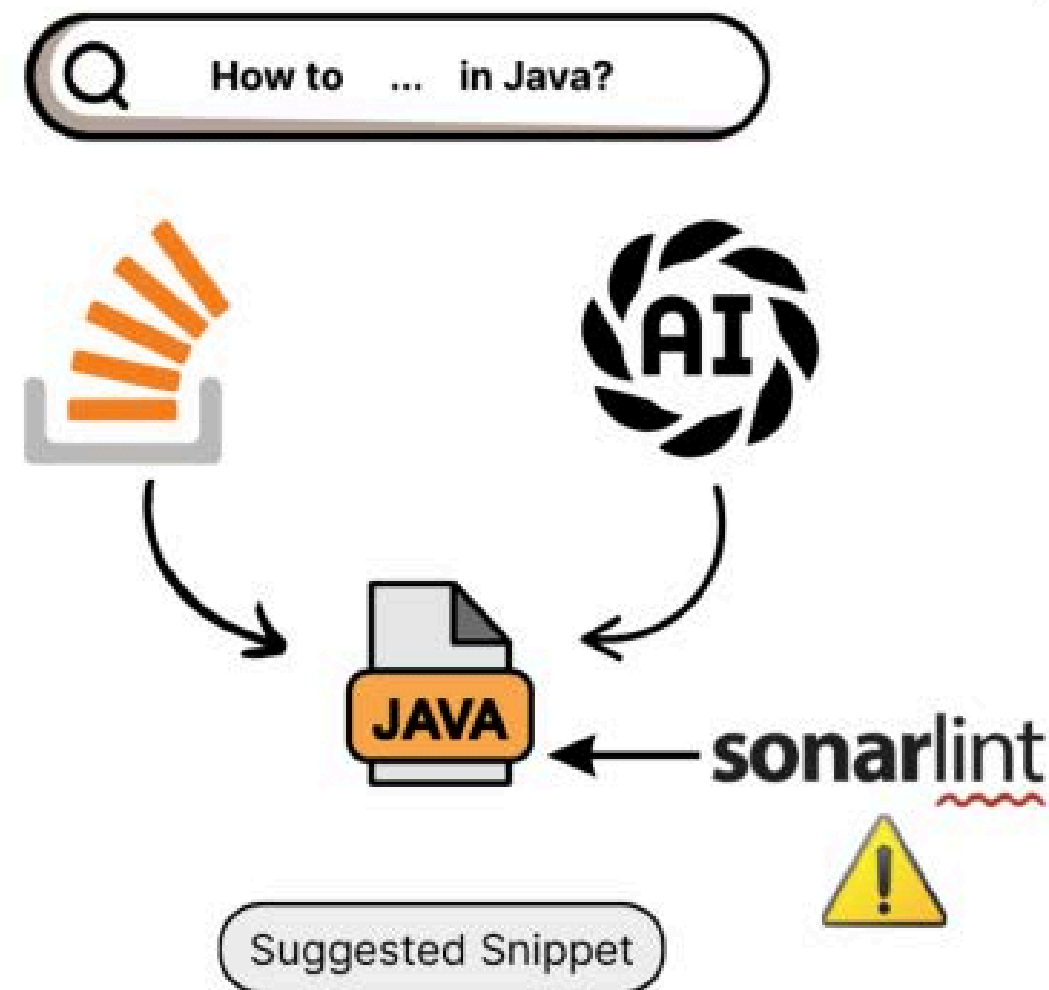
<https://doi.org/10.1145/3643991.3644918>



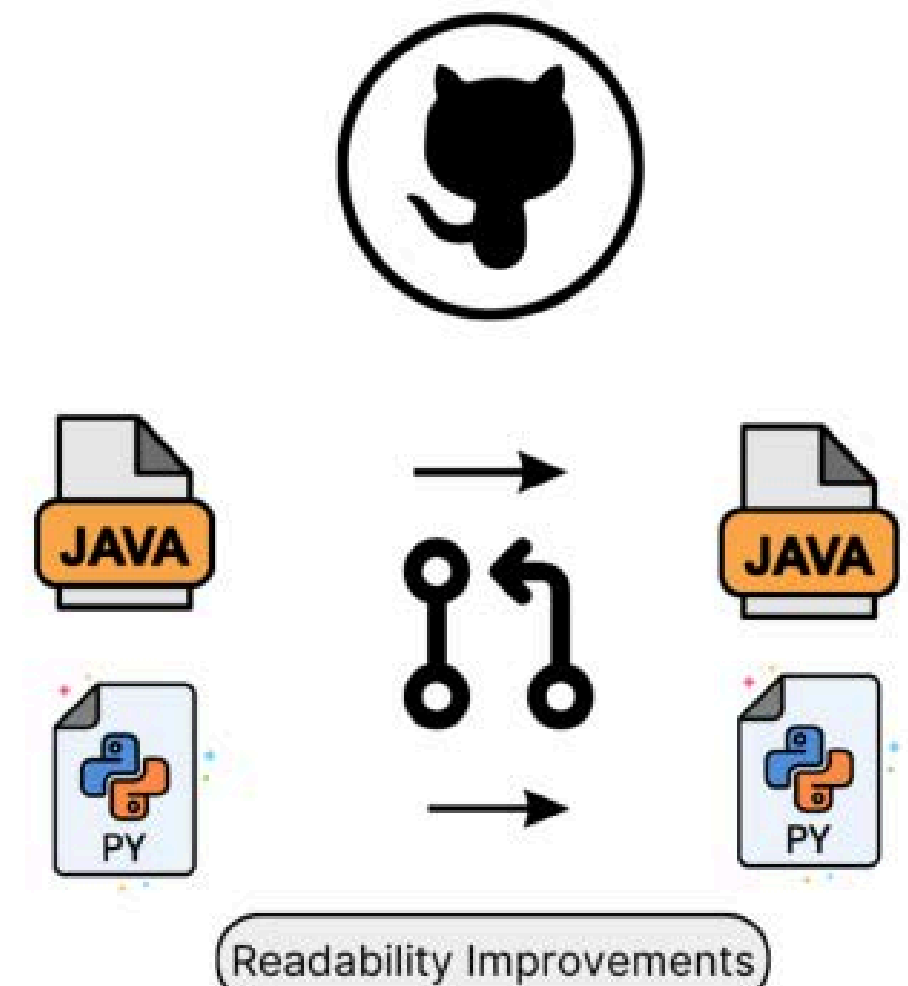
Objectives

Code Readability Perspectives

What developers are recommended to write



What developers actually write and improve



Large Language Models

LLMs can also be used to generate code snippets for the developers.



how to insert an item into an array at a specific index in java?

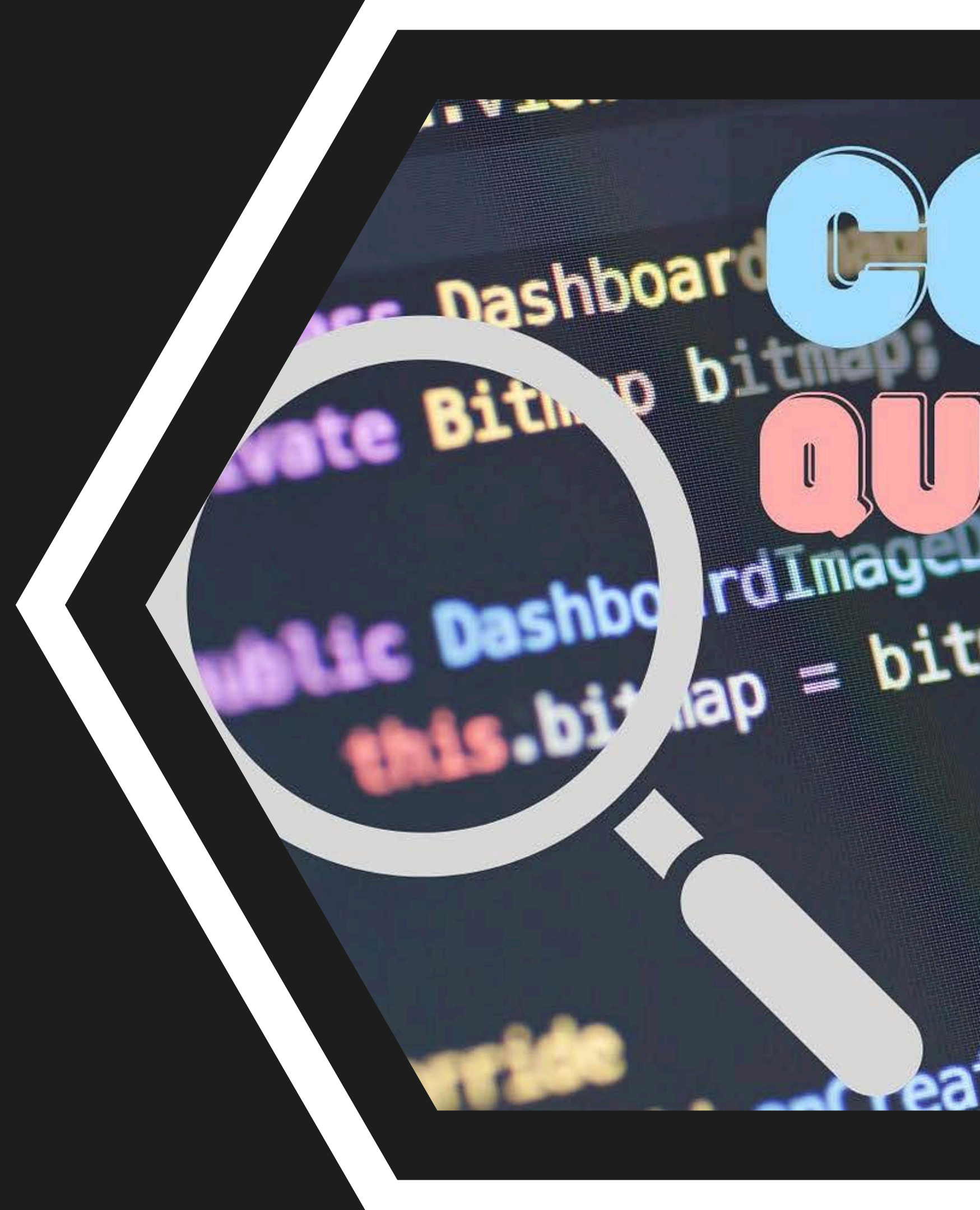
java

Copy code

```
public class Main {  
    public static void main(String[] args) {  
        // Original array  
        int[] originalArray = {1, 2, 3, 4, 5};  
  
        // Element to insert  
        int elementToInsert = 99;  
  
        // Position to insert the element  
        int position = 2;  
  
        // Create a new array with size one more than the original array  
        int[] newArray = new int[originalArray.length + 1];  
  
        // Copy elements from original array to the new array  
        for (int i = 0, j = 0; i < originalArray.length; i++, j++) {  
            // If we reach the position to insert, insert the new element at  
            if (i == position) {  
                newArray[j] = elementToInsert;  
                j++; // Move to next position in newArray  
            }  
            newArray[j] = originalArray[i];  
        }  
  
        // If the position to insert is at the end of the array  
        if (position == originalArray.length) {  
            newArray[newArray.length - 1] = elementToInsert;  
        }  
    }  
}
```


However

The quality of its code suggestions is still uncertain and requires further evaluation





Objectives

Compare the readability warnings identified by SonarLint in code snippets **generated by LLMs** with those **authored by human developers** on Stack Overflow



Human Generated
Content

VS



AI Generated
Content



Selecting Developer Input Queries

Users from 80 countries garnered 15,865 queries

349 SELECTED QUERIES



Crokage

Describe the Java programming task

Task:

Num. of Answers: ☒ 10 ☐ 5 ☐ 1

How do you like the overall result?

★★★★★

Post id: 19233594 java iterator iteration

If you need to just remove the last element of an array list, you could just do:

```
arrayList.remove(arrayList.size()-1);
```

If you wanted to remove all elements from back to front, you could use this for loop:

```
for(int i = arralist.size()-1; i >= 0; i--){
    arrayList.remove(i);
}
```

For more information on Array Lists, you can go here: http://www.tutorialspoint.com/java/java_arraylist_class.htm

Hope this helps.

Post id: 11094123 java filtering duplicates

Use a List `toRemove` to record element at the first time `iterator` stumble into it, afterwards when meet again with the recorded element, remove it using `iterator.remove()`

```
private void removeDups(List list) {
    List toRemove = new ArrayList();
    for(Iterator it = list.iterator(); it.hasNext();){
        Object next = it.next();
        if(!toRemove.contains(next)) {
            toRemove.add(next);
        }
    }
}
```


02



Code Snippet Extraction from Stack Overflow and LLM Models

349 SELECTED QUERIES



SO post IDs with code snippet recommended by CROKAGE



ChatGPT 3.5



ChatGPT 4o



deepseek

Gemini

+ New chat

Query + "in Java"

03








Assessing Semantic Adherence of ChatGPT Code Snippets with Input Queries



Query + Code Snippet



-  **1 UNRELATED**
-  **2 WEAKLY RELATED**
-  **3 RELATED**
-  **4 UNDERSTANDABLE**
-  **5 STRAIGHTFOWARD**



03



Assessing Semantic Adherence of ChatGPT Code Snippets with Input Queries



327 OUT OF 349 CODE
SNIPPETS ARE
SEMANTICALLY ALIGNED
WITH THE
DEVELOPER'S QUERY



Query + Code Snippet



04



Identifying Code Readability Issues

sonarlint



ChatGPT



deepseek

Gemini



107 Code Smell Rules
5,910 warnings detected
1,635 code snippets



04



Identifying Code Readability Issues

107 Code
Smell Rules



77 Readability
Rules



✓ **1 READABILITY**

✗ **2 NON-READABILITY**

Syntactic Scope

METHODS

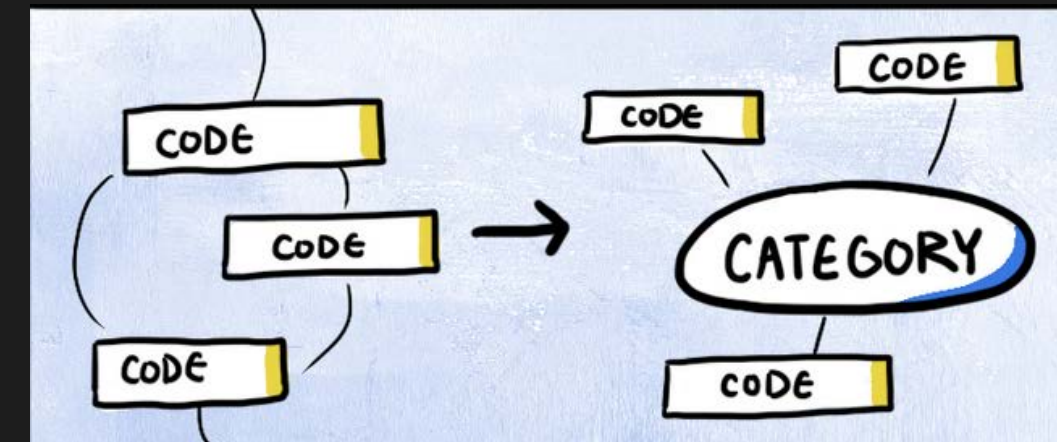
CLASSES

ATTRIBUTES

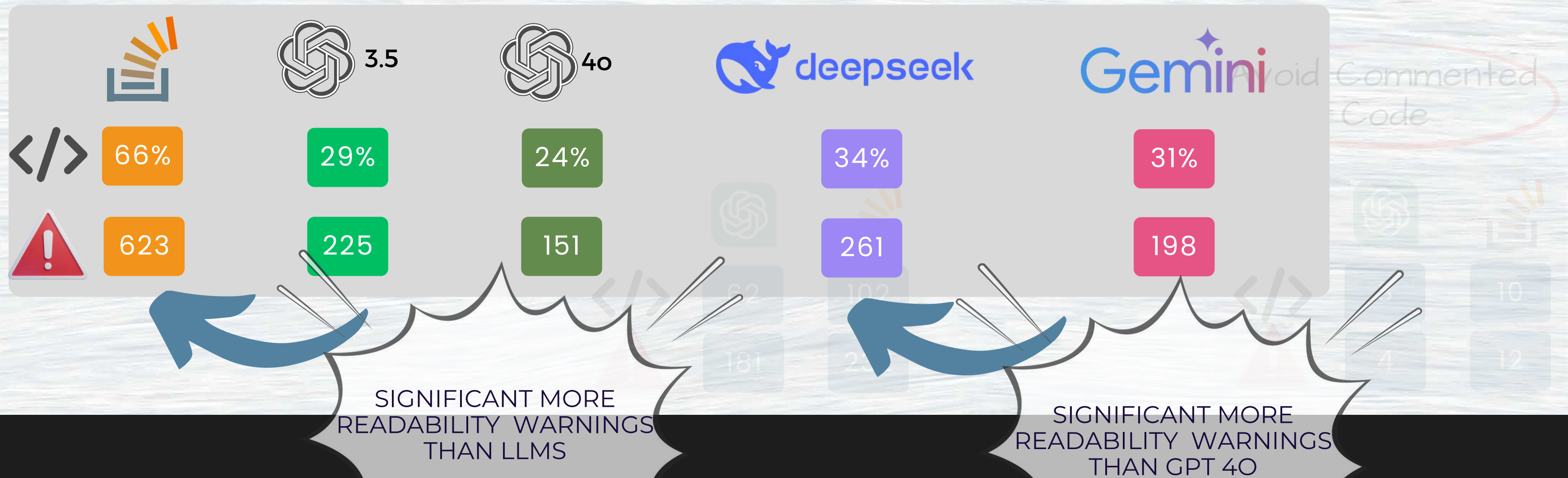
EXCEPTIONS

COMMENTS

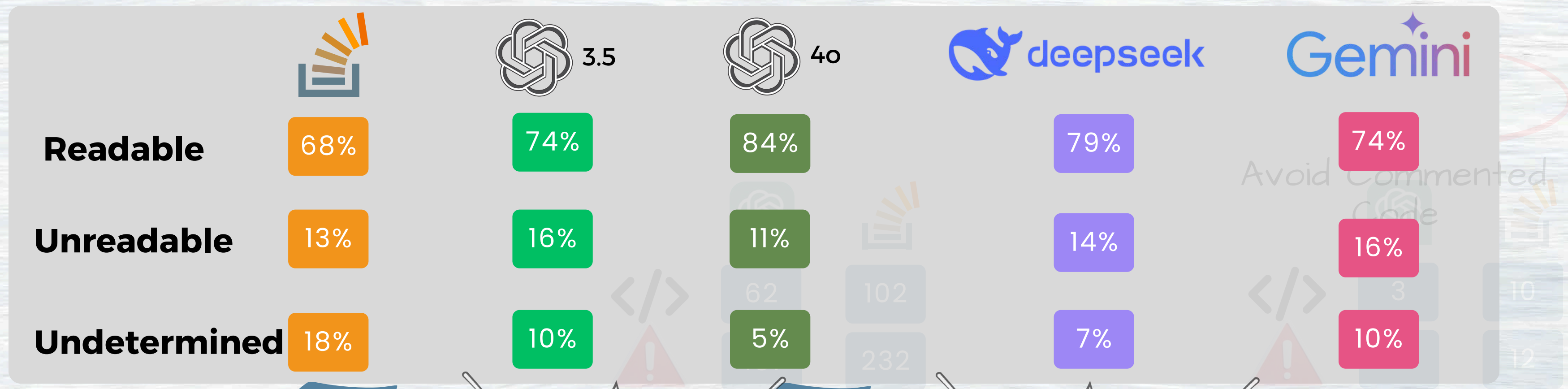
Semantic Labels



RESULTS



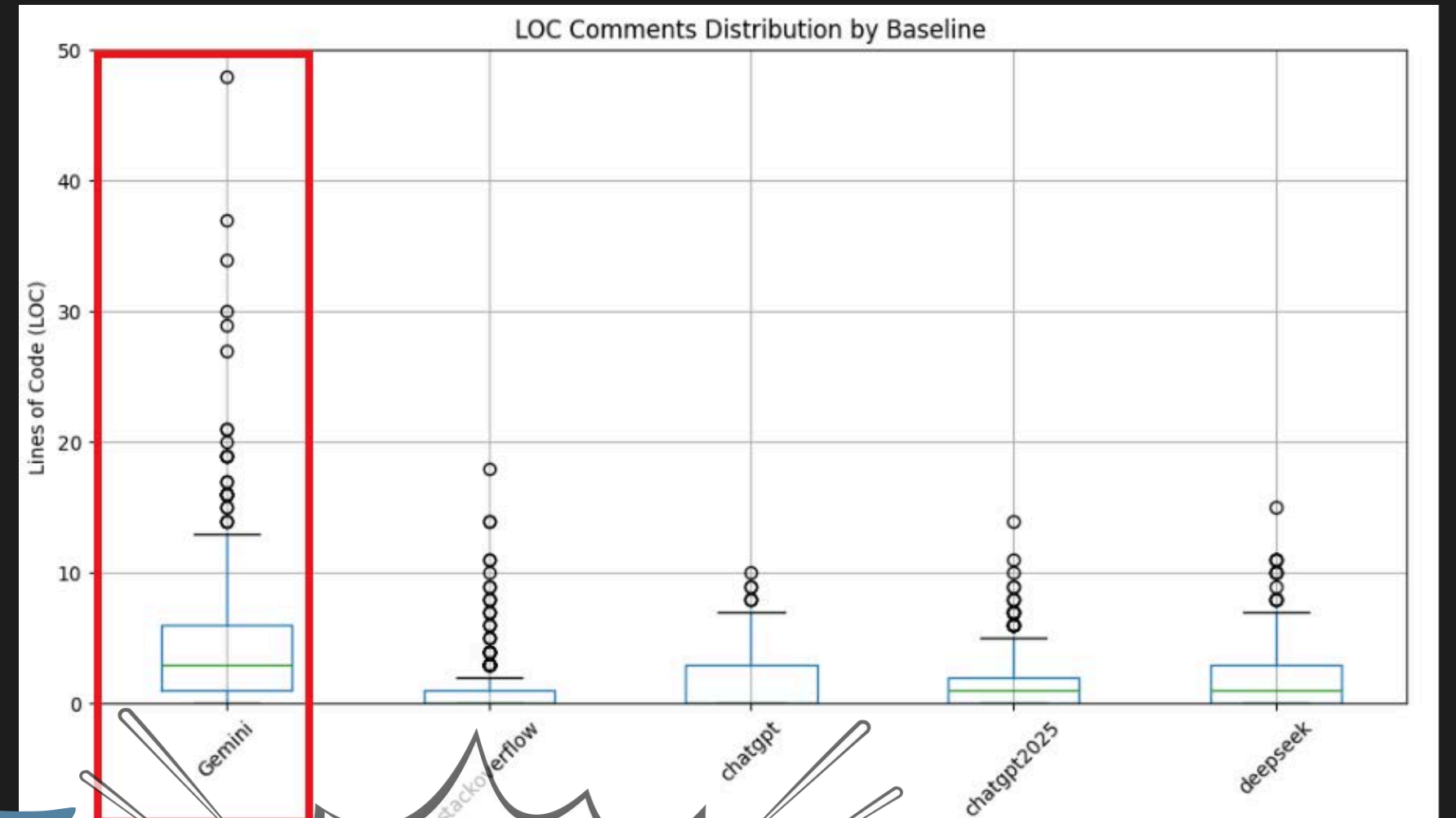
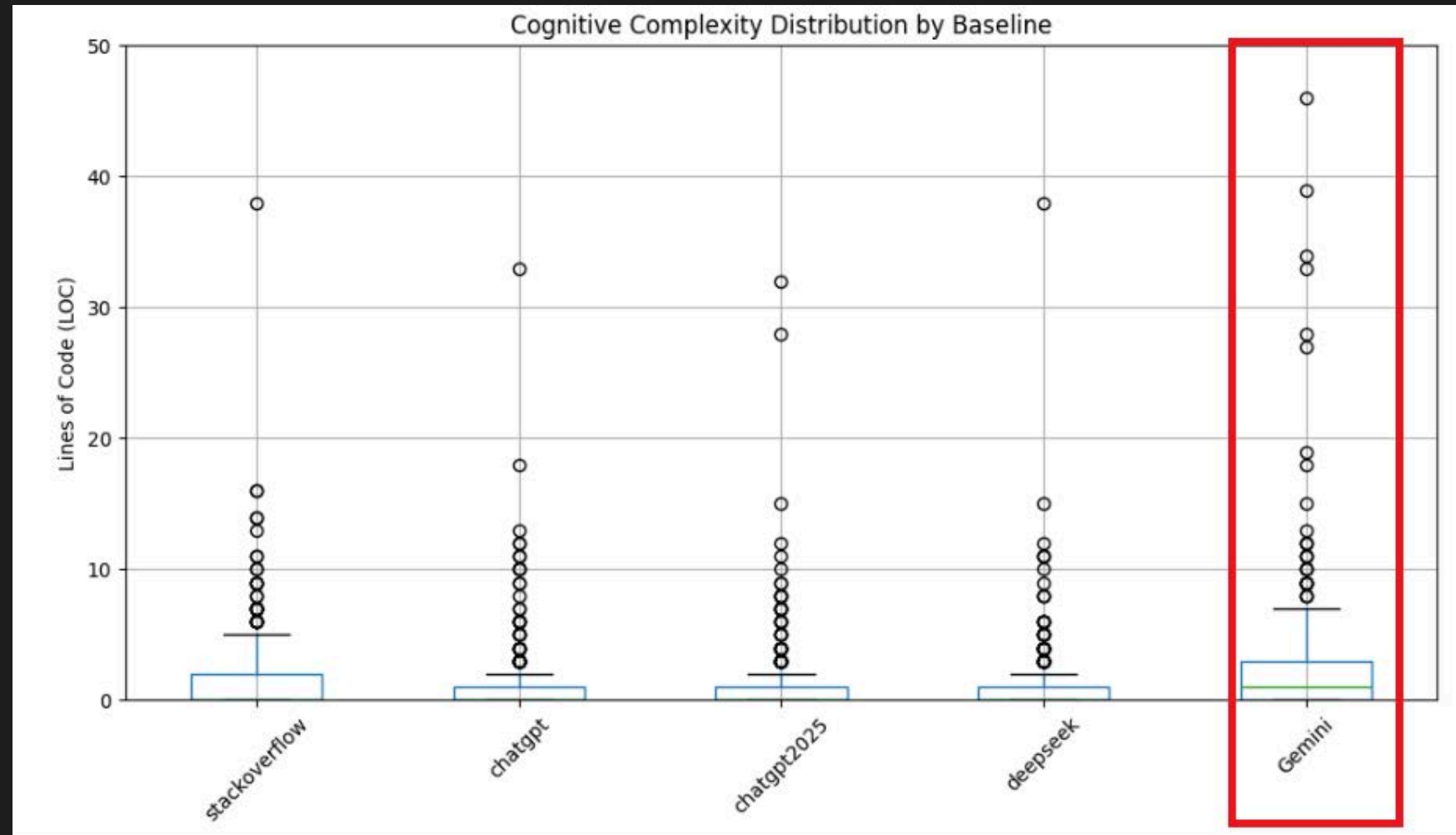
RESULTS



GENERATE MORE
UNREADABLE CODE
SNIPPETS THAN
EXPECTED

GENERATE FEWER
UNREADABLE CODE
SNIPPETS THAN
EXPECTED

RESULTS



HIGHER LOC AND
COGNITIVE COMPLEXITY
DO NOT NECESSARILY
MEAN UNREADABLE
CODE.

RESULTS

77 Readability
Rules



Clarify Code Intent

23 Rules

Improve
Regex

4 Rules

Replace
Magic Literals

2 Rules

Avoid
Shadowing

3 Rules

Use Proper
Modifiers

4 Rules

Use Static
Properly

5 Rules

Improve
Naming

2 Rules

Avoid Nested
Structures

2 Rules

Use Specific
Exceptions

1 Rule

Prefer Concise Code

19 Rules

Eliminate
Redundancy

9 Rules

Simplify
Lambdas

4 Rules

Merge
Redundant
Blocks

2 Rules

Use try with
resources

1 Rule

Prefer Built-in
Methods

3 Rules

Clean Up

15 Rules

Remove
Useless Code

13 Rules

Remove
Duplicate Code

1 Rule

Remove
Commented
Code

1 Rule

Convention

15 Rules

Use Naming
Conventions

6 Rules

Follow Code
Style
Conventions

4 Rules

Follow
Conditional
Conventions

5 Rules

Improve Code Modularity

5 Rules

Refactor
Complex Logic

4 Rules

Limit Variable
Scope

1 Rule

RESULTS

Clarify
Code
Intent



3.5



4o



deepseek

Gemini



16%

6%

6%

7%

12%



90

24

26

31

54

List list = new ArrayList();
[0-9]*\$

@Override
hardcoded string literals

RESULTS

Prefer
Concise
Code



3.5



4o



deepseek

Gemini



21%

9%

3%

4%

6%



91

35

12

17

26

<> diamond operator

<> diamond operator

anonymous class

anonymous class

RESULTS

Clean Up



40%

303

useless code



3.5

18%

155

useless code



4o

13%

98

useless code



deepseek

24%

209

useless code

Gemini








16%

97

useless code
commented code

RESULTS

Convention

		 3.5	 4o	 deepseek	
	16%	0,1%	0,2%	0,1%	0,2%
	130	7	11	3	14

Naming Convention
Style Convention

RESULTS

Improve
Code
Modularity



0,1%

9



3.5

0,1%

4



4o

0,1%

4



deepseek

0,1%

1

Gemini

0,2%

7

RESULTS

77 Readability
Rules



Conditions

§§ 12 Rules

**Attributes
(Fields)**

§§ 10 Rules

Methods

§§ 7 Rules

Statements

§§ 7 Rules

**Local
Variables**

§§ 6 Rules

Exceptions

§§ 5 Rules

Loops

§§ 4 Rules

Assignment

§§ 4 Rules

Regex

§§ 4 Rules

Generics

§§ 4 Rules

Lambda

§§ 4 Rules

**Modifiers and
Annotations**

§§ 3 Rules

Classes

§§ 3 Rules

Parameters

§§ 2 Rules

Comments

§§ 1 Rule

Constants

§§ 1 Rule

RESULTS



Local Variables

36%

Assignment

30%

Generics

14%



3.5

Local Variables

15%

Assignment

15%

Generics

3%



4o

Local Variables

10%

Assignment

9%

Attributes

5%



deepseek

Assignment

19%

Local Variables

18%

Attributes

5%

Gemini

Comments

8%

Local Variables

8%

Statements

6%

RESULTS

Assessing the Readability of ChatGPT Code Snippet Recommendations: A Comparative Study

Carlos Eduardo C. Dantas
carloseduardodantas@iftm.edu.br
Federal University of Uberlândia
Brazil

Adriano M. Rocha
adriano.rocha@ufu.br
Federal University of Uberlândia
Brazil

Marcelo A. Maia
marcelo.maia@ufu.br
Federal University of Uberlândia
Brazil

<https://doi.org/10.1145/3613372.3613413>



<> diamond operator

anonymous classes



method reference

Discussion

1



CROKAGE frequently suggests Stack Overflow posts that are linked to the specific context of their corresponding thread.



LLMs tends to provide more generalized responses, not linked to a particular context.

Discussion



Stack Overflow, being a platform where **multiple users contribute to posting code snippets**, exhibits **more diversity** or variation in terms of coding styles, conventions, patterns, or structures.



The code snippets generated by ChatGPT often exhibit consistent patterns in terms of code readability issues.



How developers perceive Code Readability?

Linters rely on predefined rules and may not fully capture how developers perceive code readability in practice.

Octoverse: AI leads Python to top language as the number of global developers surges

<https://github.blog/news-insights/octoverse/octoverse-2024/>

Pull Requests (PRs)

1

PRs promote a peer code review process

2

Developers could detail code changes in PR comments to help reviewers understand.

The screenshot shows a GitHub Pull Request (PR) titled "Refactor listSplits operation when using maxSplits #2381". The PR is marked as "Merged" and was merged on Dec 13, 2021. It shows 5 conversations, 5 commits, 4 checks, and 2 files changed. A comment from the contributor, dated Dec 10, 2021, describes the changes: "Refactored listSplits method in TableOperationsImpl. This change affects the listSplits command which takes maxSplits as an option." The comment lists four bullet points: "Renamed variable names to enhance readability", "Added documentation for the method", "Replaced while-loop with if-loop after determining while-loop was only run at most once each time", and "Created IT test for method in ShellIT class". A link to "Detailed Description" is provided. Below the comment, the PR is shown as "approved these changes on Dec 10, 2021" with a green checkmark. The code diff is visible, showing changes to the `listSplits` method. The diff highlights the removal of a `while` loop and the addition of an `if` loop. Annotations point to the variable `endRows` before the commit and `existingSplits` after the commit.

Refactor listSplits operation when using maxSplits #2381

Merged on Dec 13, 2021

Conversation 5 Commits 5 Checks 4 Files changed 2

commented on Dec 10, 2021

Refactored listSplits method in TableOperationsImpl. This change affects the listSplits command which takes maxSplits as an option.

- Renamed variable names to enhance readability.
- Added documentation for the method.
- Replaced while-loop with if-loop after determining while-loop was only run at most once each time.
- Created IT test for method in ShellIT class.

Closes #2371

Refactor listSplits operation when using maxSplits ...

approved these changes on Dec 10, 2021

View changes

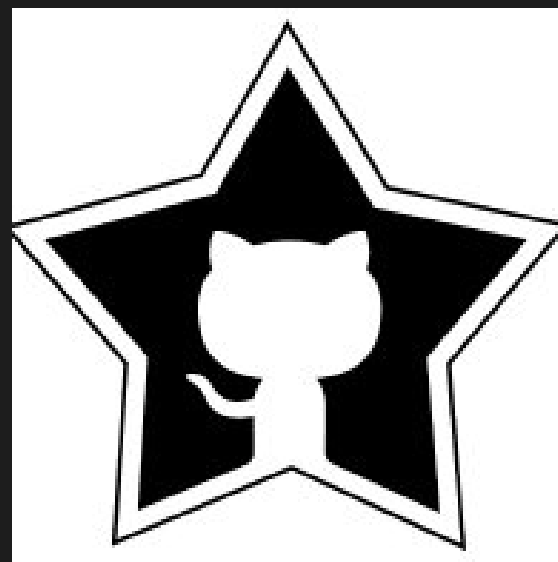
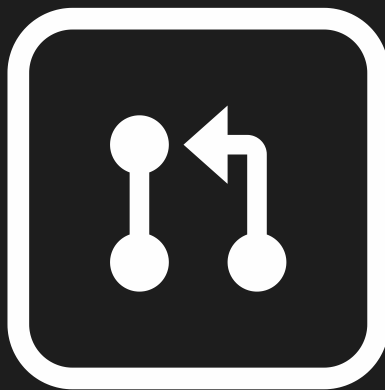
```
695 716 // tableName is validated in _listSplits
696 - List<Text> endRows = _listSplits(tableName);
697 - if (endRows.size() <= maxSplits)
698 -   return endRows;
699 -
717 + final List<Text> existingSplits = _listSplits(tableName);
718 +
719 + // As long as maxSplits is equal to or larger than the number of current splits, the existing
720 + // splits are returned and no additional processing is necessary.
721 + if (existingSplits.size() <= maxSplits) {
722 +   return existingSplits;
```

Variable name Before Commit

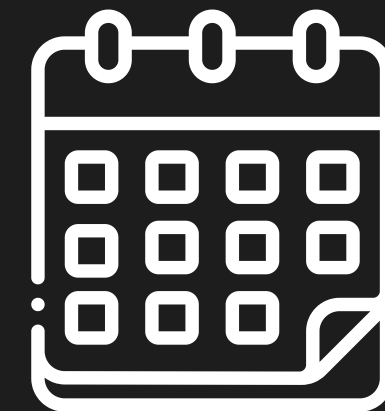
Variable name After Commit



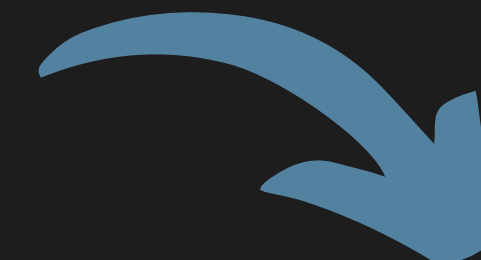
Mining Candidate Pull Requests



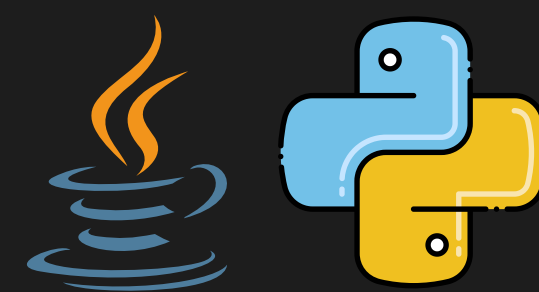
10 stars



Recently Modified



Candidate Pull Requests



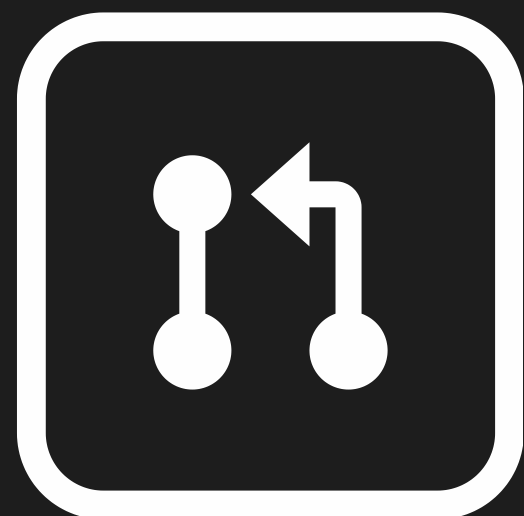
- ✓ 1 READABILITY
- ✓ 2 UNDERSTANDABILITY
- ✓ 3 LEGIBILITY



"READABILITY"
KEYWORD MENTIONED
IN MAJORITY OF THE
CASES



Selecting Readability-focused PRs from Candidates



Candidate
Pull Requests



Readability Focused PRs



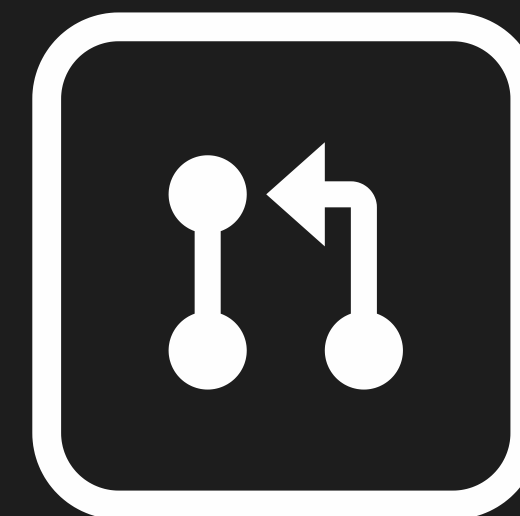
User-Facing readability PR



Unexplained Readability PR



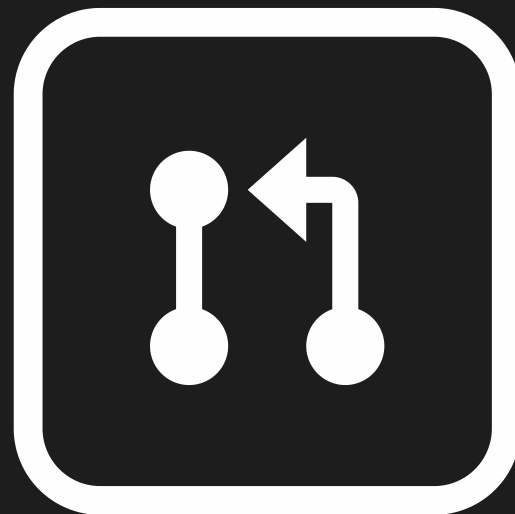
Tangled PR



Pull Requests focused on
Code Readability
Improvements



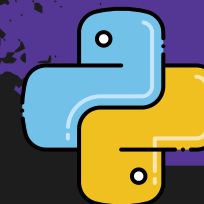
Collecting the Types of Code Readability Improvements



Pull Requests focused on
Code Readability
Improvements



- ✓ **1 PR HYPERLINK**
- ✓ **2 DETAILED DESCRIPTIONS**
- ✓ **3 CODE DIFF**



437 CODE READABILITY
IMPROVEMENTS



456 CODE READABILITY
IMPROVEMENTS



4

Classifying the Types of Code Readability Improvements



Prompt Templates used for readability improvements

System: "You are a software developer specialized in [PROGRAMMING_LANGUAGE], skilled in reading GitHub Pull Request texts written by other developers and labeling the code readability improvements described by them using one of the following labels: [LABELS]. You may also suggest new labels if none of the provided labels accurately represents the improvement described by the developer."

User: "Based on this pull request with the developer description: [DEVELOPER_DESCRIPTION], the code was changed by the developer from [CODE_BEFORE] to [CODE_AFTER], please, provide the appropriate label that represents the code readability improvement described and performed by the developer. No explanations, only the name of the appropriate label"

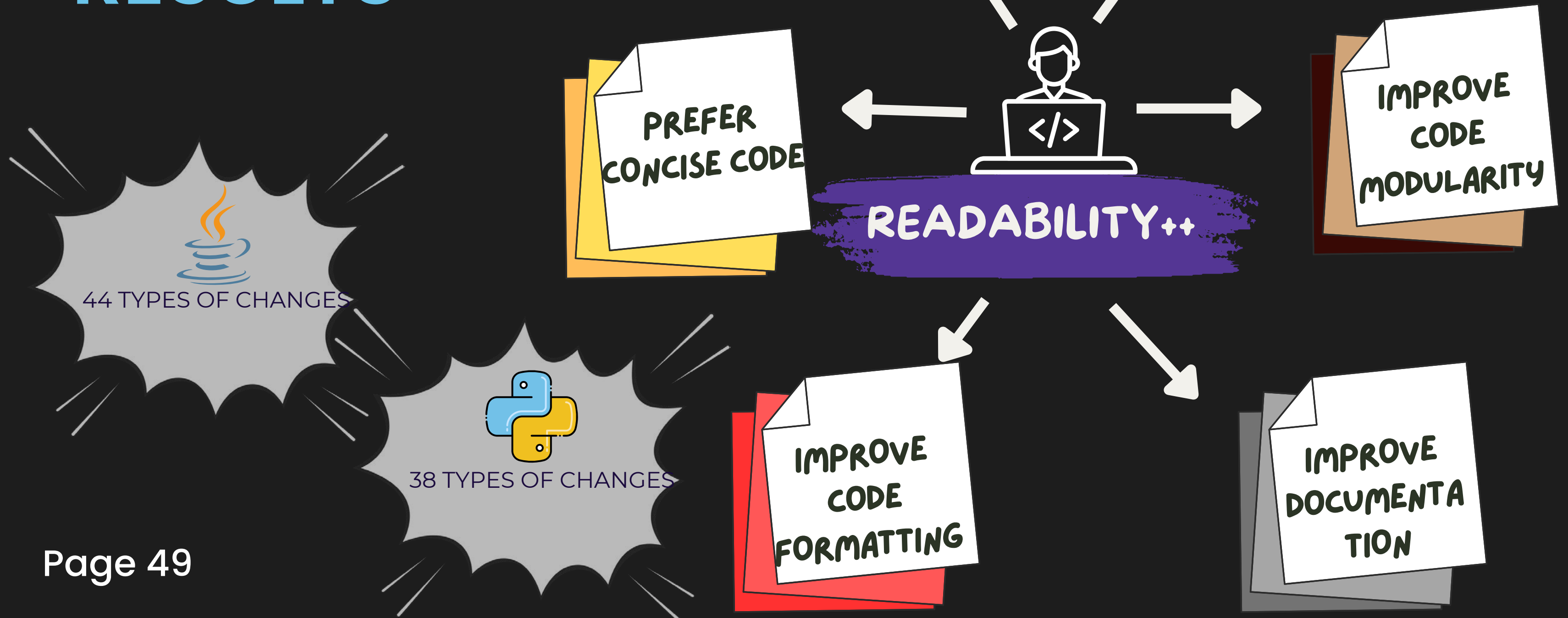


91% agreement
(74% first round)

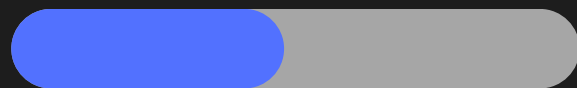
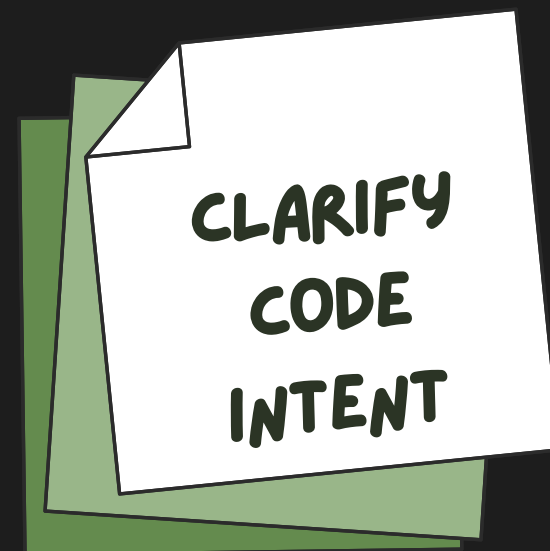


88% agreement
(76% first round)

RESULTS



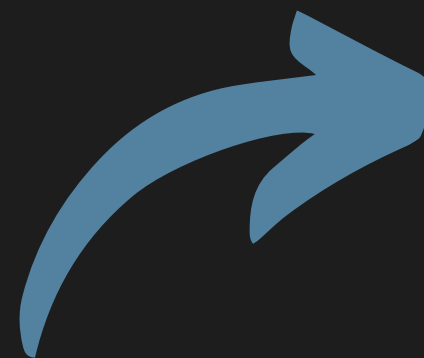
RESULTS



48%



45%



1

"Renamed variable names to enhance readability"

2

"This PR changes 'graphconverter.py' by using constants instead of hard-coded strings"

3

"Split up some single lines of code to make it easier to read"

RESULTS

**CLARIFY
CODE
INTENT**



Apply Early Return

Replace Ternary
With if/else

Extract Literals to
Enumerated Values

Use domain
specific types to
clarify intent

Improve Naming

Extract Variable

Include Modifiers and
Annotations

Extract Literals to
Constants

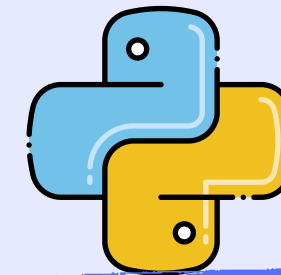
Replace Generic With
Specialized API

Clarify Numeric
Constants

Prefer Positive
Conditions

Encapsulate
Private Logic

Clarify Test Intent



Clarify Positional
Indexing

Replace Lookup
with Dict

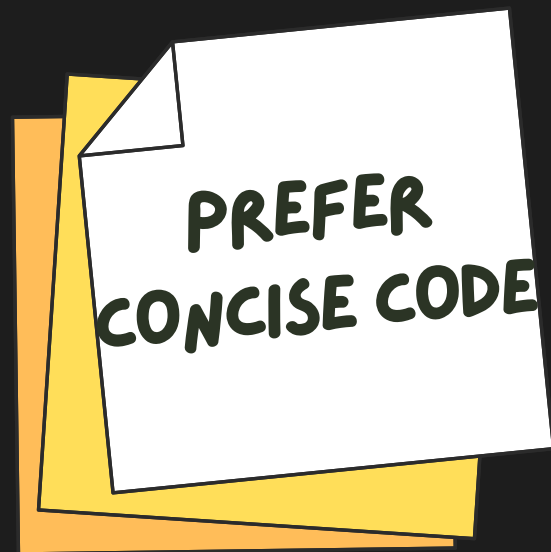
Include Type Hints
and Named
Parameters

Improve Regex

Add Parenthesis
for Clarity

Prefer Specific
Exception Handling

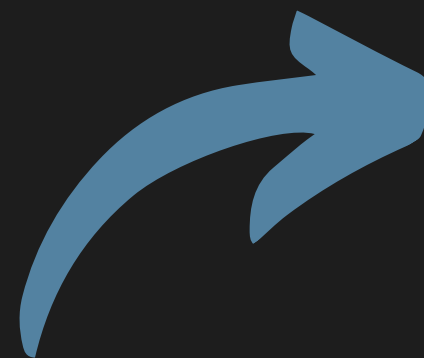
RESULTS



25%



31%



1

"Improve readability by removing boilerplate code."

2

"The anonymous classes were converted to lambda at the places where it improves readability and decreases the lines of codes"

3

"Split up some single lines of code to make it easier to read"

RESULTS

**PREFER
CONCISE CODE**



Favour Diamond
Operators

Replace Lambda
With Method
Reference

Replace Switch
With if-else

Prefer Structured
Resource Management

Replace Anonymous
Classes With
Functional Style

Replace if-else with
Switch

Prefer Built-in Methods

Improve String
Formatting

Simplify Object
Construction and
Conversion

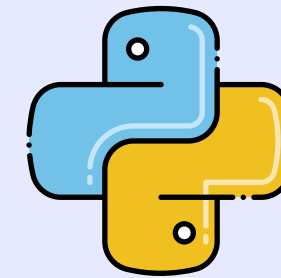
Unwrap Unnecessary
Else

Inline Temporary
Variables

Reduce Conditional
Nesting

Prefer Idiomatic
Iteration

Collapse Boolean
Return



Prefer Idiomatic
Conditions

Simplify Import
Statements

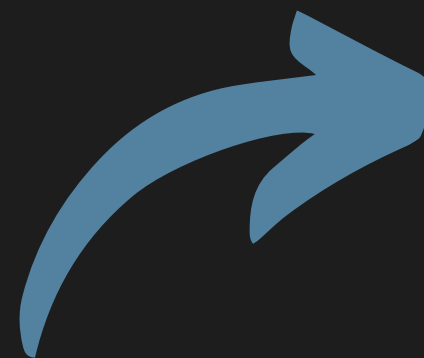
RESULTS



15%



19%



1

"We better clean up duplicated code and enhance code readability."

2

"removing unnecessary method calls indirections and resulting unused methods"

3

"Cleanup unused imports"

RESULTS

CLEAN UP



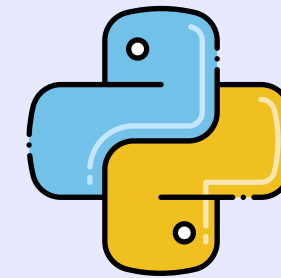
Remove
Commented Code

Remove
Unnecessary Throws

Remove Useless
Parameter

Remove Unused
Field of Variable

Remove Unnecessary
Annotation or Modifier

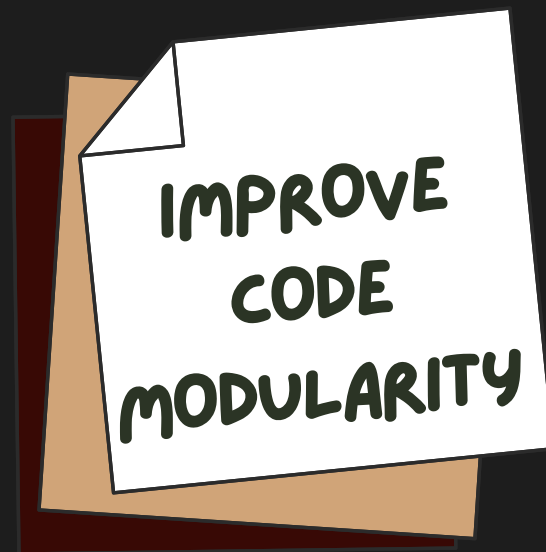


Remove Duplicate Code

Remove Unused
Method/Chunk of
Code

Remove Unused
Imports

RESULTS



15%



19%



1

"Split long functions into multiple logical units"

2

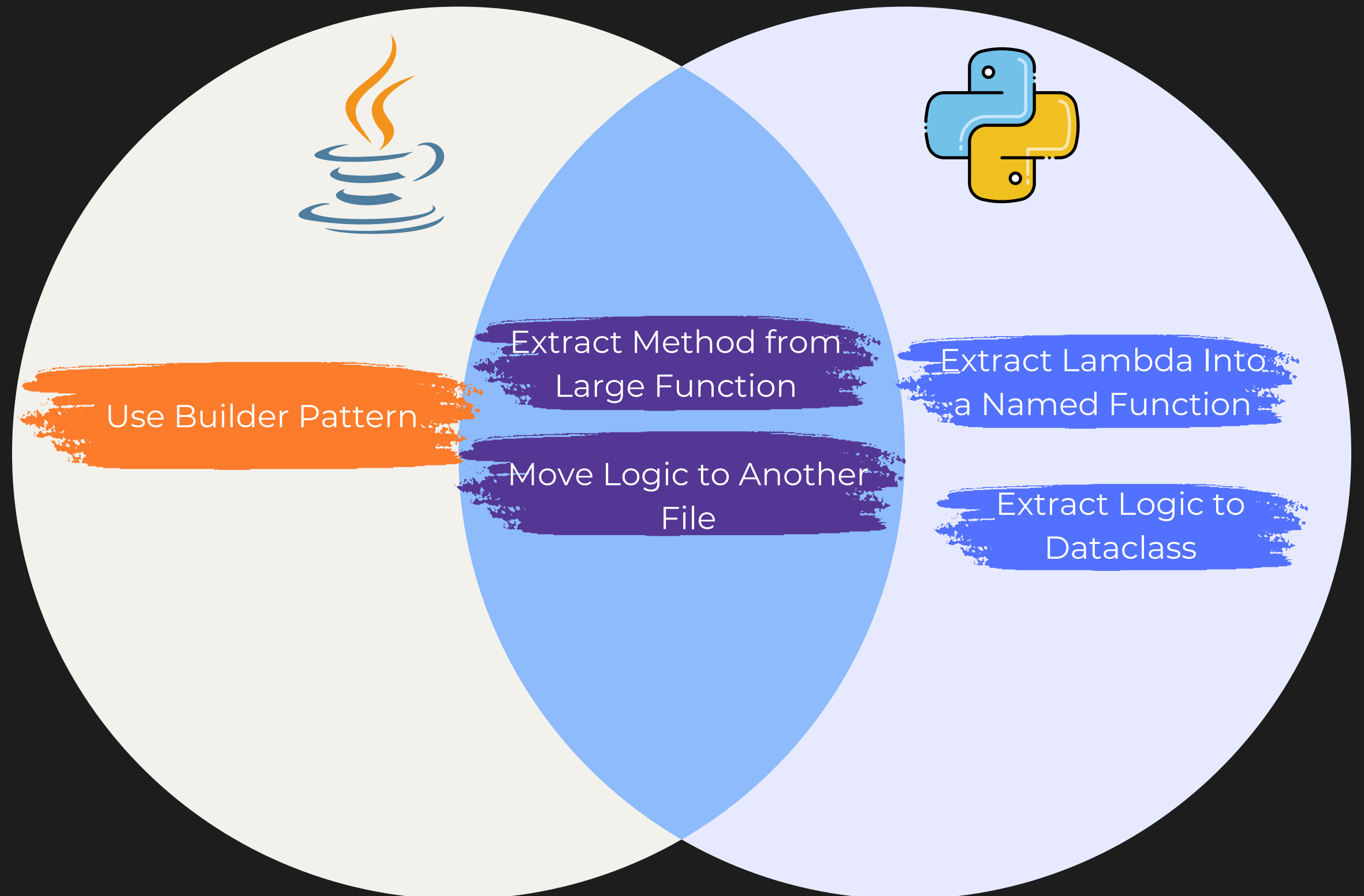
"Splitting the 'linter.py' and formatters in separate files"

3

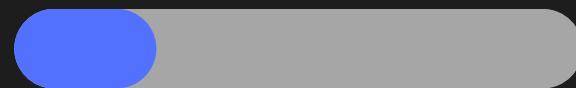
"Use a proper builder pattern for 'SourceTestStep' and 'SinkTestStep'"

RESULTS

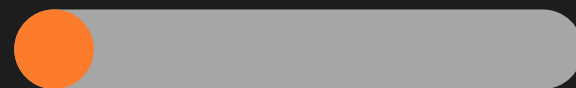
**IMPROVE
CODE
MODULARITY**



RESULTS



25%



14%



1

"Add comments between logical blocks"

2

"Rewrites comment to read better"

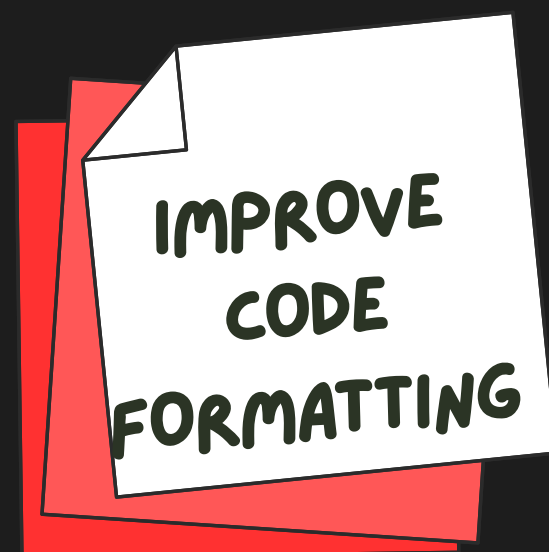
3

"fixing some comment typos"

RESULTS



RESULTS



16%



10%



1

"Moves functions around to be in a more intuitive place"

2

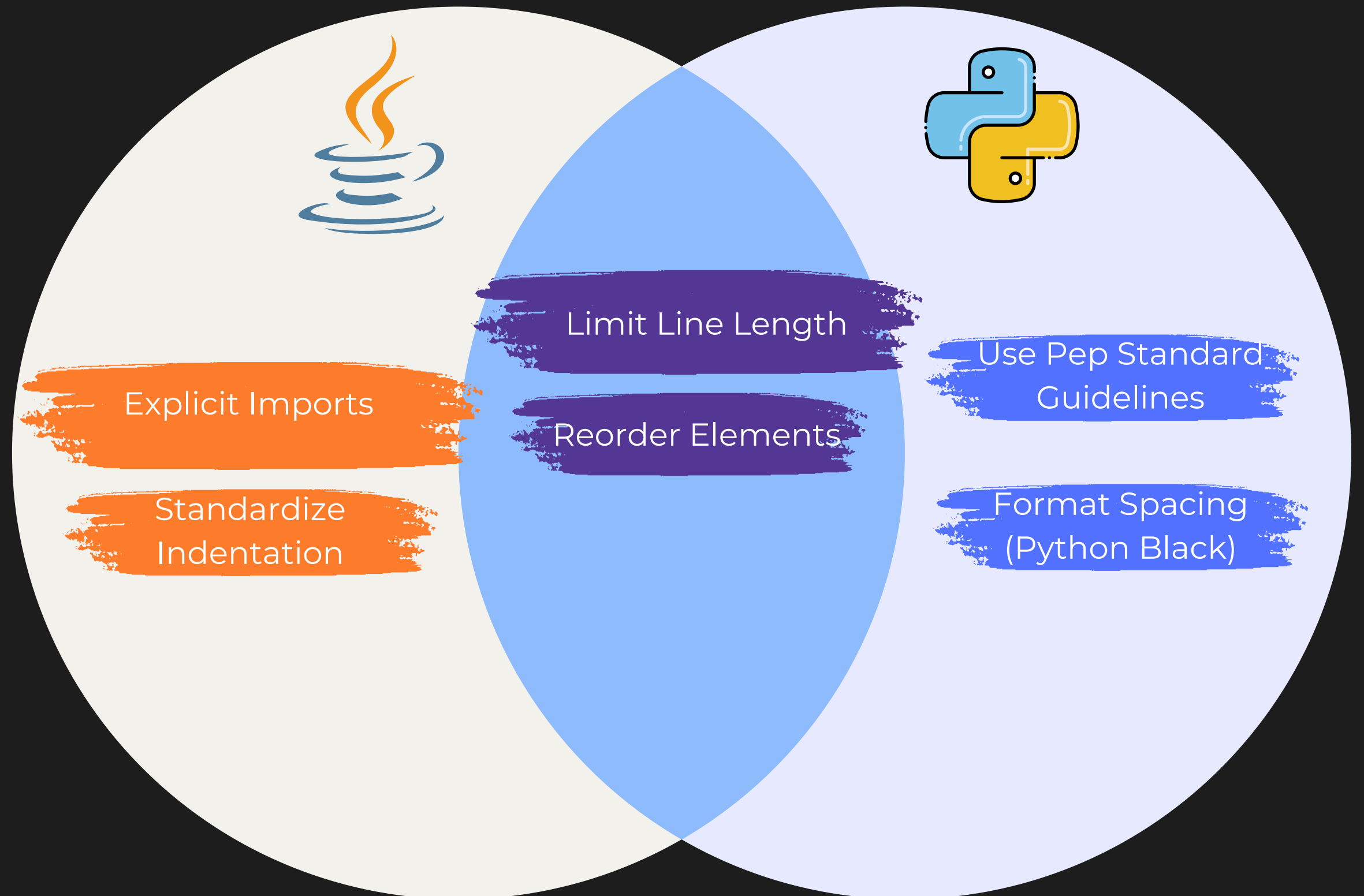
"use line-length=120 for better code readability"

3

"adding spaces where recommended by PEP8)."

RESULTS

**IMPROVE
CODE
FORMATTING**





Discussion

subjective
vs.
objective

1 BREAK LONG LINES

2 INDENTATION

3 IF/ELSE CODE STYLE

1 REPLACE GENERIC APIS

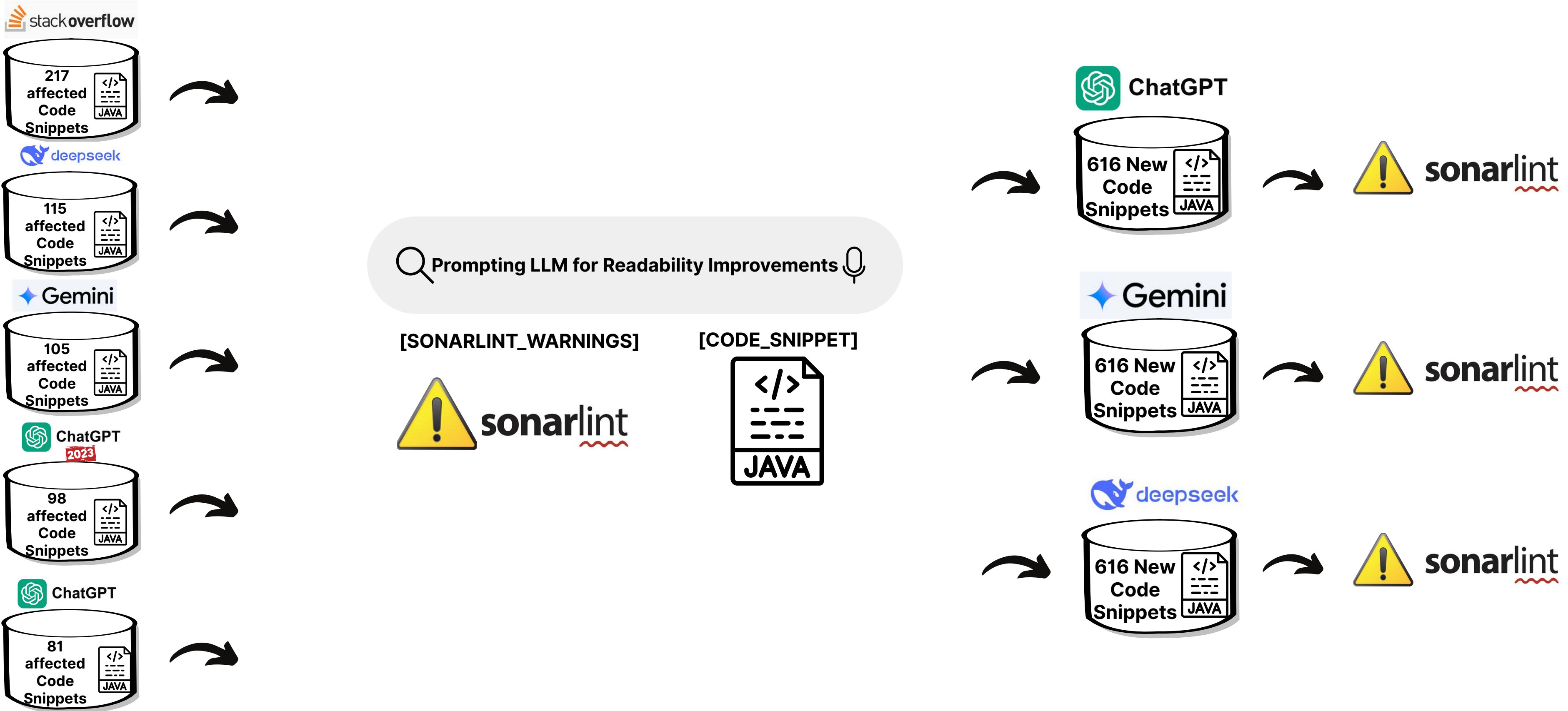
2 CONSUME LANGUAGE API

3 IMPROVE NAMING



Automated Improvements

Although significant progress has been made in assessing if the code is readable or not and in identifying code fragments that could potentially be improved, this does not guarantee that developers will actually improve the readability of the code.



Prompting LLMs

GPT-4.1
API



Gemini 2.5 Pro



deepseek
coder

Prompt Templates used for readability improvements

System: "You are a software developer specialized in the Java programming language, with expertise in improving the readability of the code based on a list of recommendations."

User: "Given the following Java code snippet: [CODE_SNIPPET], the following improvements are recommended: [SONARLINT_WARNINGS]. Please provide a revised version of the code snippet that applies the recommended improvements. Only the revised code snippet, without additional text."

RESULTS

GPT-4.1
API

 Gemini 2.5 Pro


deepseek
coder

Corrected

70%

77%

79%

Not Corrected

15%

11%

12%

Introduced

15%

12%

9%

GPT IS THE LEAST
EFFECTIVE LLM IN
CORRECTING
WARNING

RESULTS

Clean up

17~25% of code snippets not corrected or introduced new warnings

Remove unused code often introduce new warnings

RESULTS

Clarify
Code Intent

LLMs actually corrected the code or simply removed the portion that raised the SonarLint warning

Warnings addressed with single prompt

RESULTS

Prefer
Concise Code

Most of the Prefer Concise Code rules were successfully addressed using a single prompt

Some rules could apply sequentially

RESULTS

Convention

Improve Code
Modularity

LLMs often suggested names that follow conventions

Even large classes and long methods were refactored



Conclusions

1

LLMs can generate code with significant fewer readability warnings than human-written code from Q&A platforms such as Stack Overflow

2

56 documented types of code readability improvements (26 shared, 12 for Python and 18 for Java)

3

While LLMs proved to be useful as annotators, code generators, and for refactoring tasks, they are heavily dependent on well-crafted prompts and rich contextual information.

Publications

Assessing the Readability of ChatGPT Code Snippet Recommendations: A Comparative Study

Carlos Eduardo C. Dantas
carlooseduardodantas@iftm.edu.br
Federal University of Uberlândia
Brazil

Adriano M. Rocha
adriano.rocha@ufu.br
Federal University of Uberlândia
Brazil

Marcelo A. Maia
marcelo.maia@ufu.br
Federal University of Uberlândia
Brazil

<https://doi.org/10.1145/3613372.3613413>

Simpósio Brasileiro de Engenharia de Software (SBES)

Aluno de doutorado Carlos Dantas recebe premiação

por Sistemas de Informação - Monte Carmelo

Publicado: 29/09/2023 - 16:15

Última modificação: 29/09/2023 - 16:15

[Compartilhar 0](#) [Postar](#) [Whatsapp](#)



Publications

How do Developers Improve Code Readability? An Empirical Study of Pull Requests

1st Carlos Eduardo C. Dantas
Federal University of Uberlândia
Uberlândia, Brazil
carlooseduardodantas@iftm.edu.br

2nd Adriano M. Rocha
Federal University of Uberlândia
Uberlândia, Brazil
adriano.rocha@ufu.br

3rd Marcelo A. Maia
Federal University of Uberlândia
Uberlândia, Brazil
marcelo.maia@ufu.br

doi: 10.1109/ICSME58846.2023.00022





Publications

Readability and Understandability Scores for Snippet Assessment: an Exploratory Study

Carlos Eduardo C. Dantas
carloseduardodantas@iftm.edu.br
Federal University of Uberlândia
Brazil

Marcelo A. Maia
marcelo.maia@ufu.br
Federal University of Uberlândia
Brazil

<https://aesir-workshop.github.io/aesir2021/acceptedpapers.html>

Readability and Understandability of Snippets Recommended by General-purpose Web Search Engines: a Comparative Study

Carlos Eduardo C. Dantas
carloseduardodantas@iftm.edu.br
Federal University of Uberlândia
Brazil

Marcelo A. Maia
marcelo.maia@ufu.br
Federal University of Uberlândia
Brazil

DOI: <https://doi.org/10.5753/vem.2021.17217>

Other Academic Contributions

What Developers Ask to ChatGPT in GitHub Pull Requests? an Exploratory Study

Julyanara R. Silva¹, Carlos Eduardo C. Dantas¹, Marcelo A. Maia²

¹Instituto Federal de Ciência e Tecnologia do Triângulo Mineiro (IFTM)
Campus Uberlândia Centro – Uberlândia, MG – Brazil

²Universidade Federal de Uberlândia (UFU) – Uberlândia, MG – Brazil

Estudante do Campus Uberlândia Centro ganha prêmio por artigo apresentado no 15º Congresso Brasileiro de Software

Concorrendo com trabalhos até de pós-graduação, a aluna do Ensino Médio Integrado Julyanara Silva conquistou prêmio de melhor artigo

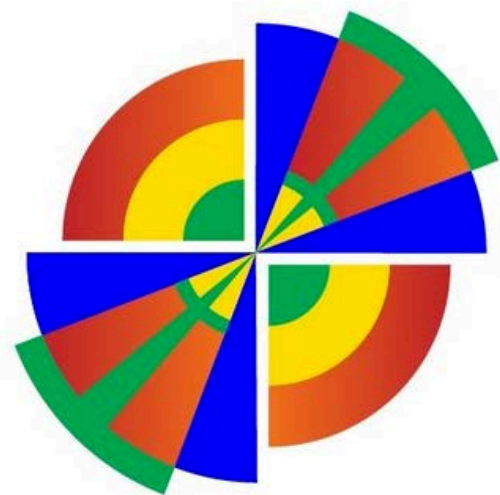
Publicado em 03/10/2024 13:43

Atualizado em 04/10/2024 09:58

Compartilhe: [f](#) [x](#) [e](#)



Other Academic Contributions



CBSOFT'25

XVI CONGRESSO BRASILEIRO DE SOFTWARE: TEORIA E PRÁTICA

22 A 26 DE SETEMBRO | RECIFE/PE

How Readable Is LLM-Generated Code Snippets? A Comparison of ChatGPT, DeepSeek, and Gemini

Giovanna Fernandes¹, Marcelo A. Maia², Carlos Eduardo C. Dantas¹

¹Instituto Federal do Triângulo Mineiro (IFTM) – Uberlândia, MG – Brazil

²Universidade Federal de Uberlândia (UFU) – Uberlândia, MG – Brazil

How Close Is ChatGPT to Developer Judgment? A Study on Stack Overflow Java Questions

Felipe Augusto Guimarães Reis¹, Marcelo A. Maia², Carlos Eduardo C. Dantas¹

¹Instituto Federal do Triângulo Mineiro (IFTM) – Uberlândia, MG – Brazil

²Universidade Federal de Uberlândia (UFU) – Uberlândia, MG – Brazil

felipeaggs@gmail.com, marcelo.maia@ufu.br, carloseduardodantas@iftm.edu.br

Do LLMs Suggest Consistent Identifiers? An Empirical Study on GitHub Pull Requests

Julyanara R. Silva
Instituto Federal do Triângulo
Mineiro (IFTM)
Uberlândia, Brazil
julyanara.silva@estudante.iftm.edu.br


Marcelo A. Maia
Universidade Federal de Uberlândia
(UFU)
Uberlândia, Brazil
marcelo.maia@ufu.br

Carlos Eduardo C. Dantas
Instituto Federal do Triângulo
Mineiro (IFTM)
Uberlândia, Brazil
carloseduardodantas@iftm.edu.br


Other Academic Contributions

Scores Submitted

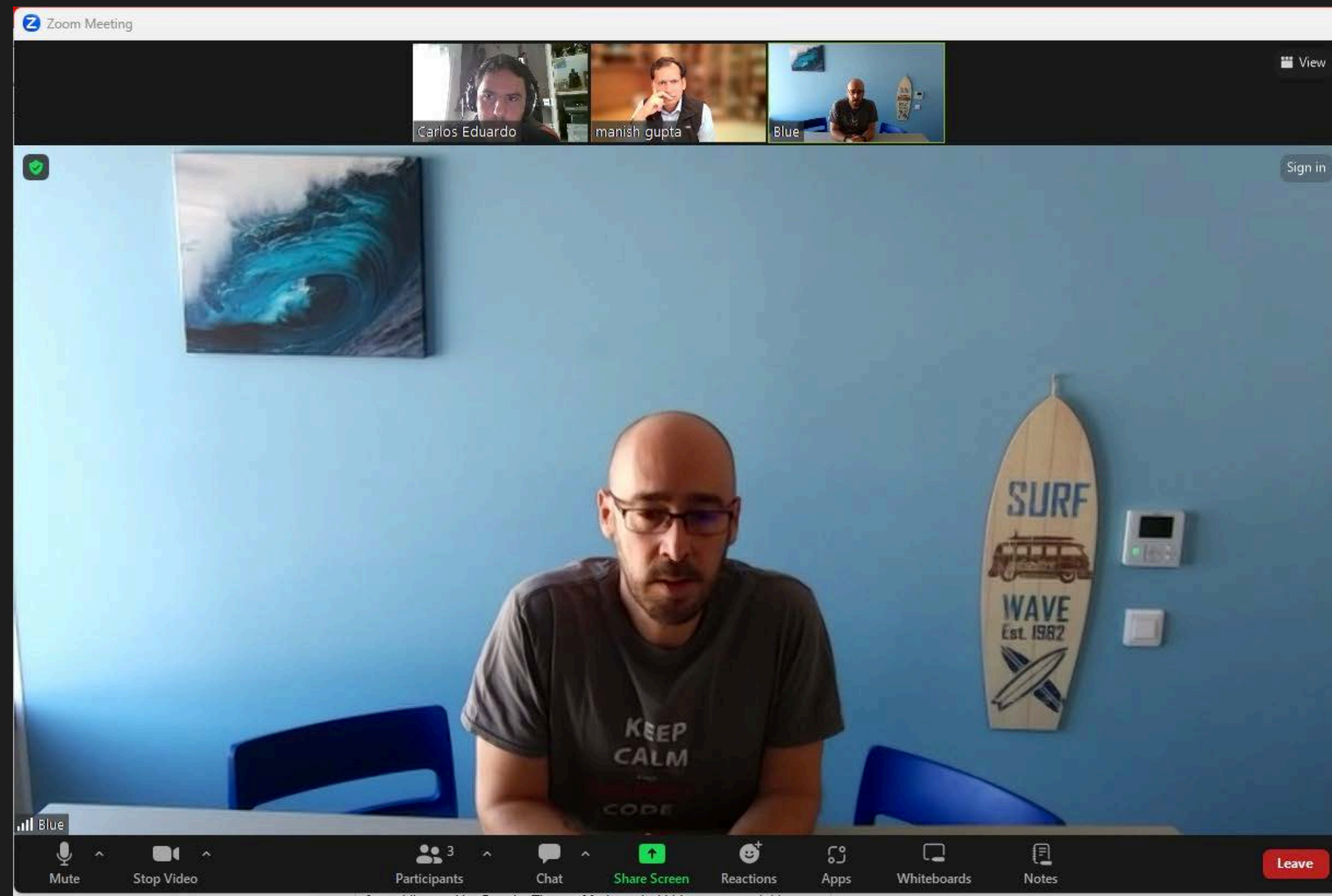
ACTION	COMPLETED	ID/TITLE	STATUS
<div>Select... ▾</div>	24-Apr-2024	<div>TSE-2023-08-0428.R1</div> <div></div>	<div>Minor Revisions Required (23-May-2024) a revision has been submitted</div> <div>Assignments:</div> <div>ADM: Gogia, Chahat</div>

 My Reviews


AF 2025 - Artifacts Festival

Id	Review status	Title / Authors
#15476		
#15845		

Other Academic Contributions



Other Academic Contributions



34th International Conference on Program Comprehension
Rio de Janeiro, Brazil, 12-13 April


Sun 12 - Mon 13 April 2026 Rio de Janeiro, Brazil

co-located with ICSE 2026


Attending Tracks Organization Search Series

ICSE 2026 (series) / ICPC 2026 (series)


Organizing Committee ICPC 2026




Marcelo De Almeida Maia General Chair
Federal University of Uberlândia
Brazil




Nicole Novielli Program Co-Chair
University of Bari
Italy



Felipe Fronchetti Louisiana State University, USA
Publicity and Social Media Chair



Carlos Eduardo Carvalho Dantas Web Co-Chair
Instituto Federal do Triângulo Mineiro
Brazil



Giuseppe Colavito Web Co-Chair
University of Bari, Italy
Italy



Code Readability Assessment and Improvement: Human and Automated Perspectives



Thank You