



Analysis and Detection of Duplicate Issues

Group 3

Ahmet Berk Taş

Mert Türe

Mizbah Çelik



Flutter Repository Github Issues Example

10,418 Open ✓ 55,764 Closed

Author ▾

Label ▾

Some of our environments prefer sized deletes, others don't support it engine

#100327 opened 1 hour ago by flar

Icon tree shaking removes ligatures

#100325 opened 2 hours ago by michaelspiss

"Vertical viewport was given unbounded height" error with markdown and sliding up panel

#100324 opened 2 hours ago by httpedo13

Crash: -[FlutterEngine sendOnChannel:message:binaryReply:] customer: money (g3) engine P2 platform-ios

#100322 opened 3 hours ago by cyanglaz

It's possbile to drag Drawers with the mouse

#100321 opened 3 hours ago by justinmc

Flutter throws an exception when typing something while holding selection endpoint

#100319 opened 3 hours ago by zxcpsd

Cannot run the project in the Android studio

#100318 opened 4 hours ago by disburden



What is the motivation?

- Open source project repositories has various issues opened by users of these projects. While opening new issues, duplicate issues can occur which can take the time of these users while searching for solutions to their errors.
- The aim is to analyze these issues and detect duplicates to prevent users from wasting time.
- For the open source project part, **Flutter** is selected because we all have used Flutter before in our projects, faced these duplicate issue problems and while developing Flutter projects, we saw that the community of Flutter is very active. Hence, Flutter open source repository will be used for the project.



Research Questions

How can creation and handling of issues be made free of duplication?



Research Questions

Is there a state of the art way to detect and close duplicate issues and is it viable for the github repositories of open source projects?



What are the artifacts selected in the study?





We are planning to use issues of the Flutter repository such as bug reports, feature requests.

Our main focus will be issues which are titled 'duplicate' to have a good start.







Duplicate Issue Examples

- Duplicate Issue 1 & 2

-  [tool_crash] FileSystemException: Cannot create link, OS Error: Incorrect function., errno = 1 r: duplicate  2
#99378 by dannnzzzlll was closed 16 days ago
-  [tool_crash] FileSystemException: Cannot create link, OS Error: Incorrect function., errno = 1 r: duplicate  2
#98992 by lemillioncorp was closed 21 days ago

- Duplicate Issue 3 & 4

-  [tool_crash] FileSystemException: Cannot open file, OS Error: No such file or directory, errno = 2 r: duplicate  3
#98821 by esperaking81 was closed 25 days ago
-  [tool_crash] FileSystemException: Cannot open file, OS Error: No such file or directory, errno = 2 r: duplicate  2
#99555 by mishrhm was closed 11 days ago



Expected Outcomes

Preventing creation of duplicate issues by checking if those issues already exist in the repository.



Expected Outcomes

Report duplicate issues if they exist. So that the team can delete these issues.



Expected Outcomes

Preventing unnecessary work and time spent on duplicates.



Expected Outcomes

Increasing efficiency on important issues by allocating time and effort on these important issues rather than duplicate ones.

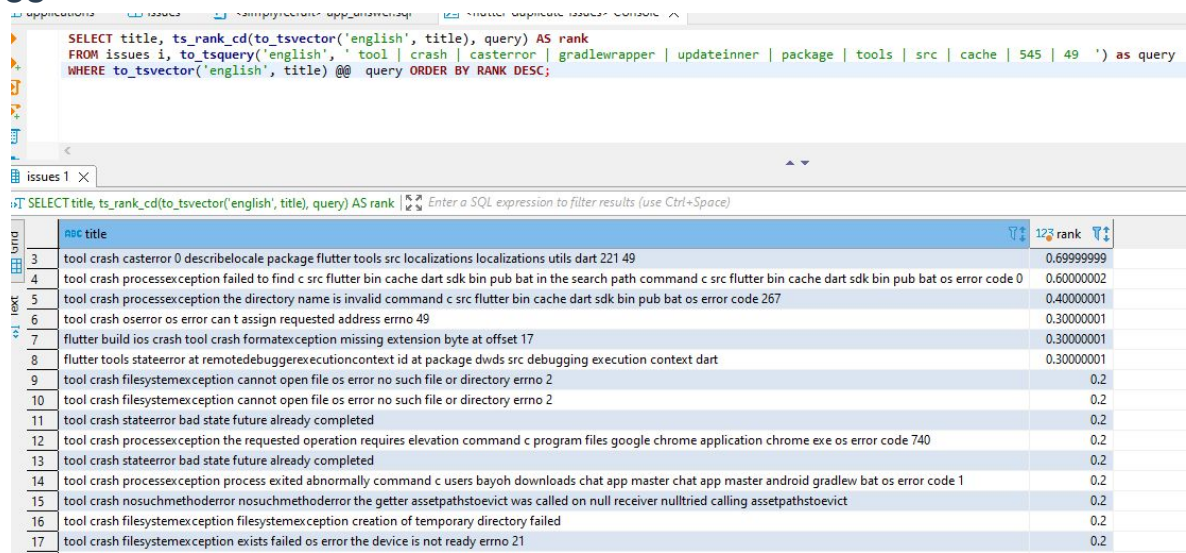


ML Models

Models for Duplicate Issue Detection

Full Text Search:

- Used PostgreSQL's helper functions
- Tokenize words
- Added two stop words; Flutter, Dart
- Rank based on term frequencies
- Eliminates ~60% of the issues



The screenshot shows a PostgreSQL query editor with a query that uses the `ts_rank_cd` function to rank issues based on a full-text search. The query filters for issues containing the words 'tool', 'crash', 'casterror', 'gradlewrapper', 'updateinner', 'package', 'tools', 'src', 'cache', '545', '49', and 'as'.

```
SELECT title, ts_rank_cd(to_tsvector('english', title), query) AS rank
FROM issues i, to_tsquery('english', ' tool | crash | casterror | gradlewrapper | updateinner | package | tools | src | cache | 545 | 49 ') as query
WHERE to_tsvector('english', title) @@ query ORDER BY RANK DESC;
```

The results table shows the top 17 ranked issues. The first three results have a rank of 0.69999999, while the remaining 14 results have a rank of 0.40000001 or 0.30000001.

rank	title
0.69999999	tool crash casterror 0 describelocale package flutter tools src localizations localizations utils dart 221 49
0.60000002	tool crash processexception failed to find c src flutter bin cache dart sdk bin pub bat in the search path command c src flutter bin cache dart sdk bin pub bat os error code 0
0.40000001	tool crash processexception the directory name is invalid command c src flutter bin cache dart sdk bin pub bat os error code 267
0.30000001	tool crash oserror os error can't assign requested address errno 49
0.30000001	flutter build ios crash tool crash formatexception missing extension byte at offset 17
0.30000001	flutter tools stateerror at remotedebuggerexecutioncontext id at package dwds src debugging execution context dart
0.2	tool crash filesystemexception cannot open file os error no such file or directory errno 2
0.2	tool crash filesystemexception cannot open file os error no such file or directory errno 2
0.2	tool crash stateerror bad state future already completed
0.2	tool crash processexception the requested operation requires elevation command c program files google chrome application chrome exe os error code 740
0.2	tool crash stateerror bad state future already completed
0.2	tool crash processexception process exited abnormally command c users bayoh downloads chat app master chat app master android gradlew bat os error code 1
0.2	tool crash nosuchmethoderror nosuchmethoderror the getter assetpathstoevict was called on null receiver nulltried calling assetpathstoevict
0.2	tool crash filesystemexception filesystemexception creation of temporary directory failed
0.2	tool crash filesystemexception exists failed os error the device is not ready errno 21

Models for Duplicate Issue Detection

TF-IDF Implementation: Term frequency-inverse document frequency.

Evaluates how relevant a word is to a document in a collection of documents by multiplying two metrics: how many times a word appears in a document, and the inverse document frequency of the word across a set of documents.

```
Issue:  tool crash  filesystemexception  cannot create link  os error  incorrect function  errno  1
```

Similar Issues:

```
Issue:  tool crash  filesystemexception  cannot create link  os error  access denied  errno  5
Euclidean Distance : 0.78330183648161
```

```
Issue:  tool crash  filesystemexception  cannot create link  os error  fun  o incorreta  errno  1
Euclidean Distance : 0.8469544979742823
```

```
Issue:  tool crash  filesystemexception  cannot delete file  os error  access denied  errno  5
Euclidean Distance : 1.0587995325341772
```

**But we have a
problem with
TF-IDF.**

Models for Duplicate Issue Detection

Word2Vec: It converts words to vectors. We used pre-trained google's embedded model for word2vec. Model is below

```
W2V_PATH="/content/drive/MyDrive/Google Word2vec/GoogleNews-vectors-negative300.bin"  
model_w2v = gensim.models.KeyedVectors.load_word2vec_format(W2V_PATH, binary=True)
```

It creates new embedding matrix with the model

Combines TF-IDF weights with embedding matrix weights to more accurate results.

**But again we have a
problem with Word2Vec**



**It does not look “directly”
semantics**

Models for Duplicate Issue Detection

PROBLEM: Does not have Semantic Relationship. If more words are common, more similarity there is.

Issue: fonts icon request wallet

Similar Issues:

Issue: fonts icon request move down
Euclidean Distance : 0.963018461522383

Issue: fonts icon request electric bolt
Euclidean Distance : 0.9704961800449186

Issue: fonts icon request satellite alt
Euclidean Distance : 0.9908752678586373

Issue: fonts icon request lock person
Euclidean Distance : 0.9908752678586373



Only icon type is different but meaning is totally different.



Should use different models for Semantic Relationships

Models for Duplicate Issue Detection

Semantic similarity with transformers:

There have been a lot of approaches for Semantic Similarity. **The most straightforward and effective method now is to use a powerful model (e.g. transformer) to encode sentences to get their embeddings and then use a similarity metric (e.g. cosine similarity) to compute their similarity score.**

Model Selection and Initialization

SentenceTransformers supports a variety of pretrained models fine-tuned for different tasks out of the box. To find a list of models optimized for semantic textual similarity,

stsb-roberta-large, which uses [ROBERTA-large](#) as the base model and mean-pooling, is the best model for the task of semantic similarity. Thus, we use this model to demonstrate.

Semantic similarity with transformers

```
1 # encode list of sentences to get their embeddings
2 embedding1 = model.encode(title_list, convert_to_tensor=True)
3 embedding2 = model.encode(title_list, convert_to_tensor=True) # compute similarity scores of two embeddings
4 cosine_scores = util.pytorch_cos_sim(embedding1, embedding2)
5
6
```

Python

```
1 similars_map = {}
2 for i in range(len(title_list)):
3     similars_map[(issues_df.iloc[i]["id"], issues_df.iloc[i]["title"])] = []
4     for j in range(len(title_list)):
5         cosine_similarity = cosine_scores[i][j].item()
6         if cosine_similarity > 0.6 and i != j:
7             similars_map[(issues_df.iloc[i]["id"], issues_df.iloc[i]["title"])] .append((issues_df.iloc[j], cosine_similarity))
8
```

Python

Semantic similarity with transformers

```
* issue id: 102068 Issue Title: [tool_crash] NoSuchMethodError: NoSuchMethodError: The getter 'assetPathsToEvict' was called on null.Receiver: nullTried calling: assetPathsToEvict
```

```
Similar Issues:
```

```
1 - Similarity Score: 0.9999991655349731 -> issue id: 101606 Issue Title: [tool_crash] NoSuchMethodError: NoSuchMethodError: The getter 'assetPathsToEvict' was called on null.Receiver: nullTried calling: assetPathsToEvict
```

```
* issue id: 101975 Issue Title: [tool_crash] ArgumentError: Invalid argument(s): Cannot find executable for /Users/wangyao/development/flutter/bin/cache/artifacts/engine/android-x64-release/darwin-x64/gen_snapshot.
```

```
Similar Issues:
```

```
1 - Similarity Score: 0.6019042730331421 -> issue id: 101764 Issue Title: [tool_crash] FileSystemException: Failed to decode data using encoding 'utf-8', null
```

```
2 - Similarity Score: 0.6420588493347168 -> issue id: 101714 Issue Title: [tool_crash] FileSystemException: Cannot create link, OS Error: Incorrect function., errno = 1
```

```
3 - Similarity Score: 0.6019043922424316 -> issue id: 101678 Issue Title: [tool_crash] FileSystemException: Failed to decode data using encoding 'utf-8', null
```

```
4 - Similarity Score: 0.7096668481826782 -> issue id: 101661 Issue Title: [tool_crash] ProcessException: Process exited abnormally:FAILURE: Build failed with an exception.* What went wrong:Project 'app' not found in root project 'android'.* Try:> Run with --stacktrace option to get the stack trace.> Run with --info or --debug option to get more log output.> Run with --scan to get full insights.* Get more help at https://help.gradle.orgBUILD FAILED in 3s Command: C:\Users\asadbek\Desktop\projects\ohang\android\gradlew.bat, OS error code: 1
```

Semantic similarity with transformers

```
* issue id: 99625 Issue Title: flutter doctor error
```

```
Similar Issues:
```

```
1 - Similarity Score: 0.8706974983215332 -> issue id: 100561 Issue Title: Flutter doctor gives a strange error
```

```
* issue id: 99413 Issue Title: intellij idea community edition (flutter doctor crashed)
```

```
Similar Issues:
```

```
1 - Similarity Score: 0.8068115711212158 -> issue id: 100479 Issue Title: IntelliJ IDEA Community Edition (the doctor check crashed)
```

```
2 - Similarity Score: 0.8068115711212158 -> issue id: 100287 Issue Title: IntelliJ IDEA Community Edition (the doctor check crashed)
```

```
3 - Similarity Score: 0.8068115711212158 -> issue id: 99599 Issue Title: IntelliJ IDEA Community Edition (the doctor check crashed)
```

Models for Duplicate Issue Detection

BERT

BERT is the MVP of NLP. And a big part of this is down to BERT's ability to embed the meaning of words into densely packed vectors.

We call them *dense* vectors because every value within the vector has a value and has a reason for being that value — this is in contrast to *sparse* vectors, such as one-hot encoded vectors where the majority of values are **0**.

BERT is **great** at creating these dense vectors, and each encoder layer (there are several) outputs a set of dense vectors.

The easiest approach for us to implement everything we just covered is through the sentence-transformers library — which wraps most of this process into a few lines of code.

BERT

```
1 sbert_model = SentenceTransformer('bert-base-nli-mean-tokens')
```

```
1 document_embeddings = sbert_model.encode(issues_df['title_cleaned'])
```

```
1 document_embeddings
```

```
1 pairwise_similarities=cosine_similarity(document_embeddings)
2 pairwise_differences=euclidean_distances(document_embeddings)
3
```

BERT

Issue: platform executable return null flutter desktop application linux

Similar Issues:

Issue: ally scaffold bottomsheet

Cosine Similarity : 0.3663930892944336

Issue: using physical keyboard ipad erased characters reappear

Similar Issues:

Issue: ally scaffold bottomsheet

Cosine Similarity : 0.38358503580093384



Testing and Results


```
similars_map = {}
for i in range(len(title_list)):
    # print("issue id:", issues_df.iloc[i]["id"], " Issue Title:", title_list[i])
    # print("Similar Issues:\n")
    similars_map[(issues_df.iloc[i]["id"], issues_df.iloc[i]["title"])] = []
    for j in range(len(original_title_list)):
        cosine_similarity = cosine_scores[i][j].item()
        if cosine_similarity > 0.6:
            similars_map[(issues_df.iloc[i]["id"], issues_df.iloc[i]["title"])].append(
                (issues_df.iloc[j], cosine_similarity))
        # print("issue id:", issues_df.iloc[j]["id"], " Issue Title:", title_list[j] )
        # print("Similarity Score:", cosine_scores[i][j].item())
        # print("\n")
```

```
original_count = 0
for k, v in original_similars_map.items():
    if len(v) > 0:
        print("* issue id:", k[0], " Issue Title:", k[1])
        print("  Similar Issues:")
        i = 1
        v.sort(key=lambda a: a[1], reverse=True)
        entered = False
        for index in range(len(v)):
            if v[index][0]["originalId"] != k[0]:
                entered = True
                print("    ", i, "- Similarity Score:", v[index][1], " -> , original issue id:",
                    v[index][0]["originalId"], " Issue Title:",
                    v[index][0]["originalTitle"])
                i += 1
        if entered:
            original_count += 1
        print("\n")
original_count
```

```
* issue id: 102068 Issue Title: [tool_crash] NoSuchMethodError: NoSuchMethodError: The getter 'assetPathsToEvict' was called on null.Receiver: nullTried calling: assetPathsToEvict
```

```
Similar Issues:
```

- 1 - Similarity Score: 0.9999991655349731 -> issue id: 98174 , original issue id: 98174 Issue Title: [tool_crash] NoSuchMethodError: NoSuchMethodError: The getter 'assetPathsToEvict' was called on null.Receiver: nullTried calling: assetPathsToEvict
- 2 - Similarity Score: 0.6396397948265076 -> issue id: 102068 , original issue id: 89738 Issue Title: NoSuchMethodError: The getter 'assetPathsToEvict' was called on null.
- 3 - Similarity Score: 0.6396397948265076 -> issue id: 101606 , original issue id: 89738 Issue Title: NoSuchMethodError: The getter 'assetPathsToEvict' was called on null.
- 4 - Similarity Score: 0.6396397948265076 -> issue id: 97624 , original issue id: 89738 Issue Title: NoSuchMethodError: The getter 'assetPathsToEvict' was

Precision: 0.46

Recall: 0.40350877192982454

Accuracy: 0.57

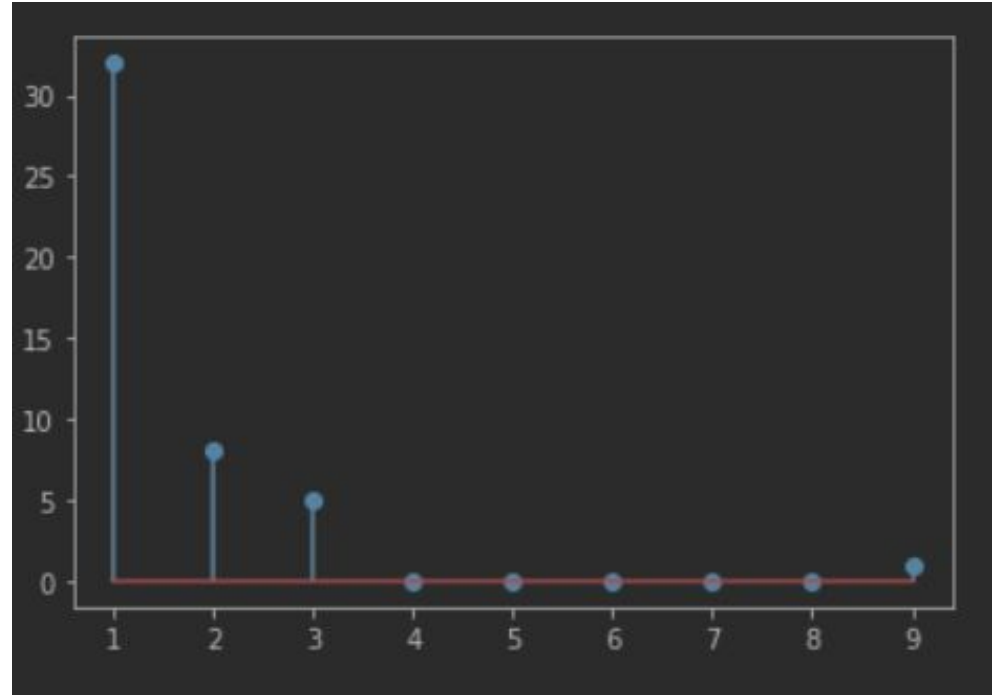
N1 - N9



First duplicate: 32 out of 100

Second most similar duplicate: 8 out of 100

Third most similar duplicate: 5 out of 100



Testing with a specific issue by taking input

```
issue = input("Enter issue")
inputEmbedding = model.encode([issue], convert_to_tensor=True)
originalEmbedding = model.encode(title_list, convert_to_tensor=True) # compute similarity scores of two embeddings
input_cosine_scores = util.pytorch_cos_sim(inputEmbedding, originalEmbedding)
```

```
print("Input: ", issue, "\n")
for j in range(len(title_list)):
    cosine_similarity = input_cosine_scores[0][j].item()
    if cosine_similarity > 0.6:
        print("issue id:", issues_df.iloc[j]["id"], " Issue Title:", title_list[j] )
        print("Similarity Score:", input_cosine_scores[0][j].item())
        print("\n")
```

Input: When using the physical keyboard on the ipad, erased characters

issue id: 102135 Issue Title: using physical keyboard ipad erased characters reappear
Similarity Score: 0.6863803863525391

issue id: 101273 Issue Title: textfield ios use texteditingcontroller add delete emoji emoji displayed abnormally
Similarity Score: 0.672225832939148

issue id: 100945 Issue Title: windows keyboard show without textfield tablet mode
Similarity Score: 0.621269702911377



Thank you for your time !