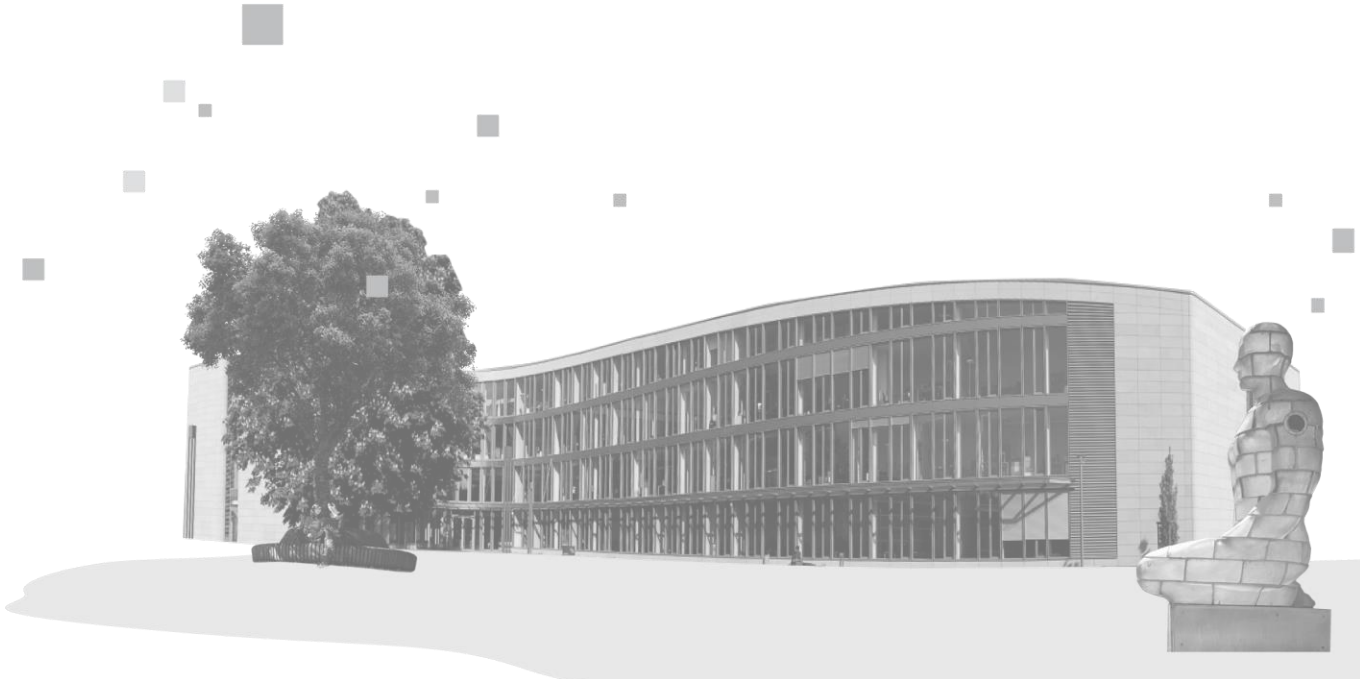
Hasso Plattner Institut
Digital Engineering · Universität Potsdam

Mini project 2: Bug reports

Henok Lachmann, Noel Bastubbe



Design IT.
Create Knowledge.

www.hpi.de

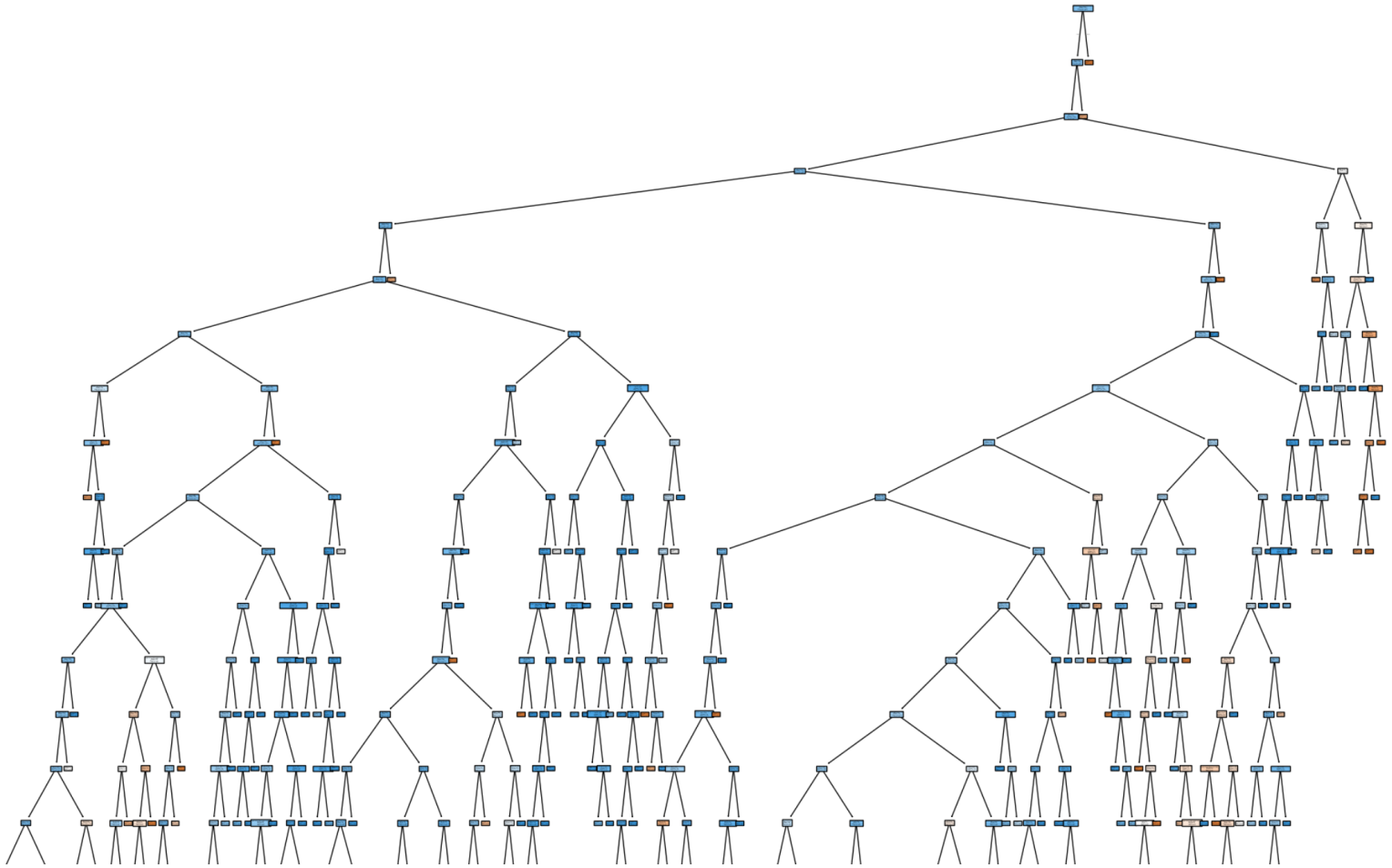
Train model

Decision-tree model with nominal features

Tree size: 205

Number of leaves: 115

Random Forest ensemble model with 5-fold cross validation



Categorize answers

We chose bug categories 1 and 4 for the holdout set.

Category 1:

Precision: 0.8021

Recall: 0.9625

Overall training:

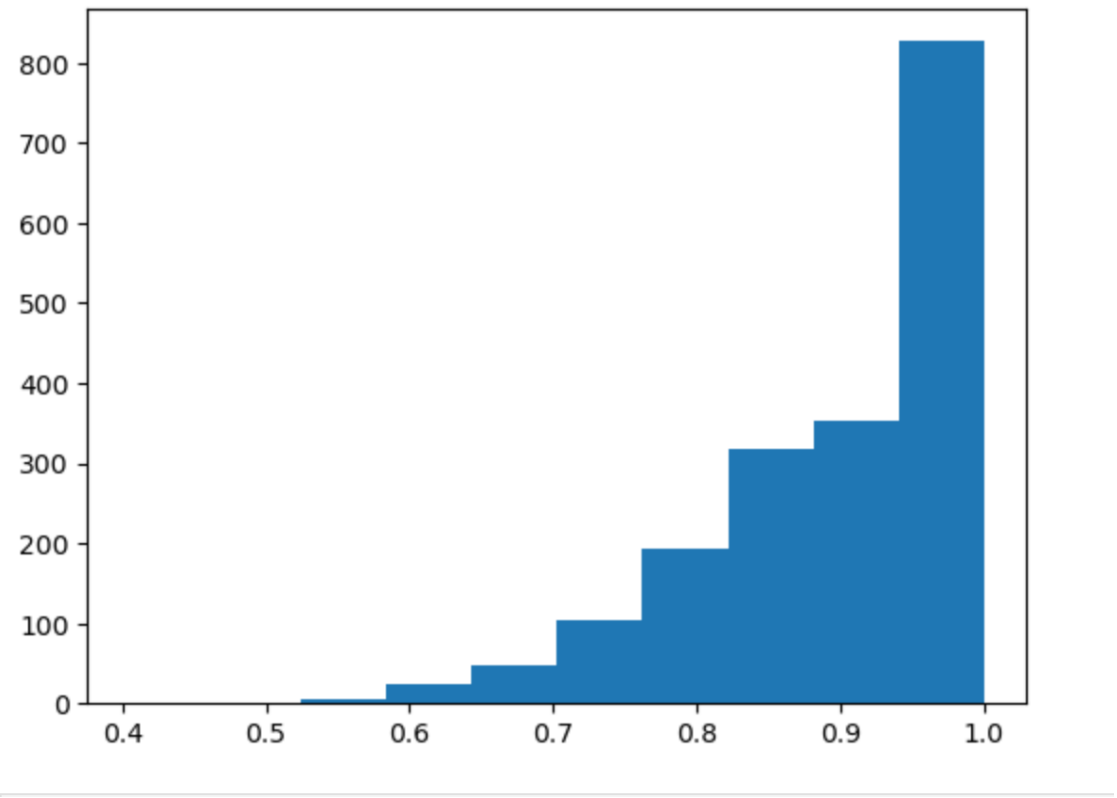
F1: 0.8544

Category 4:

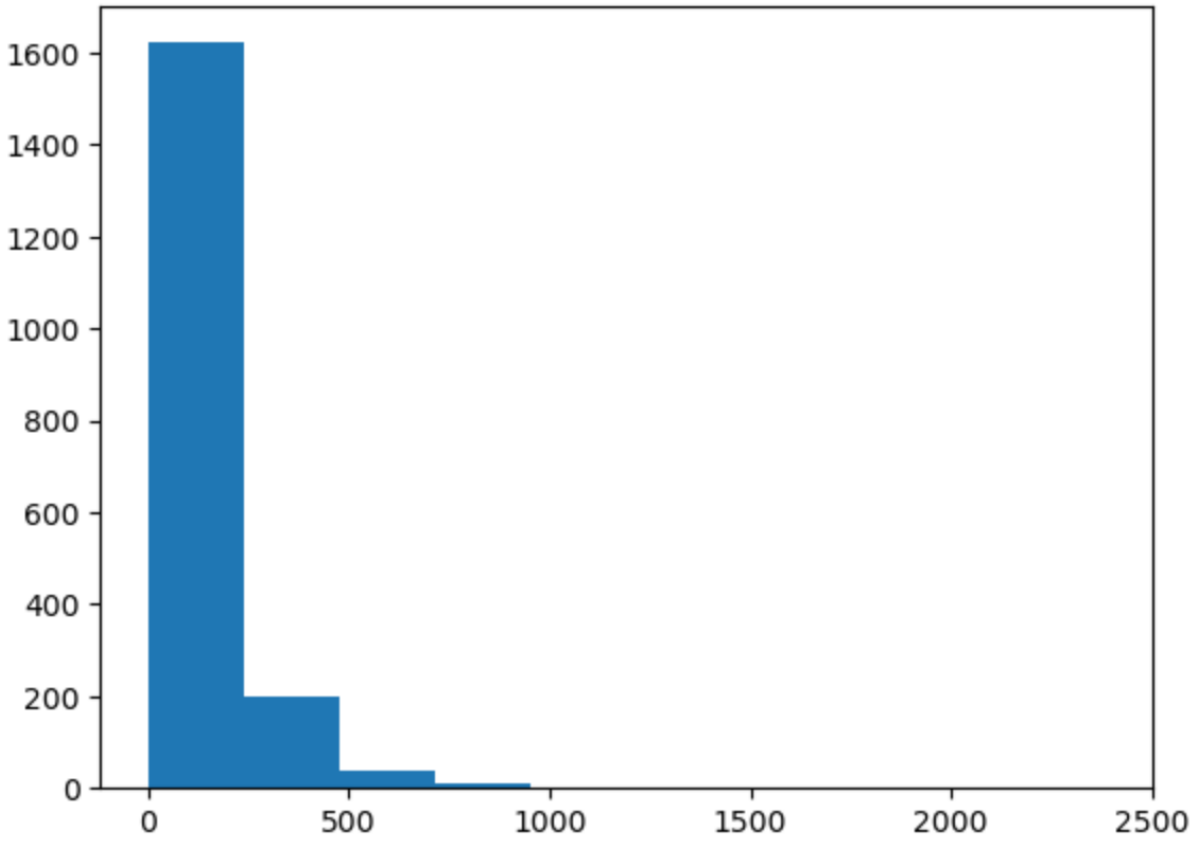
Precision: 0.8711

Recall: 0.9188

Frequency of correct examples by TTR



Frequency of correct examples by explanation length



Consolidate explanations

Prompt:

The attached csv file contains 40 explanations for the same bug. I want you to generate a single explanation by merging them in a way that minimizes

redundant information, while keeping the information that would be necessary for someone else to fix the

bug. Types of information that, if present in the explanation, should be preserved are how the program works,

how the failure is Happening and what the problem in the code is.

Compare original with synthetic explanations

Answer:

Program Functionality

The program processes time offsets using a function that takes minutesOffset as an argument. Specifically, the function DateTimeZone.forOffsetHoursMinutes(-2, -15) assigns the value -15 to minutesOffset. This value is used in a check to validate the time offset, ensuring it falls within an acceptable range.

Compare original with synthetic explanations

Failure Mechanism

The failure occurs because the code allows minutesOffset to be assigned a negative value. When minutesOffset is less than 0 (e.g., -15), the program throws an illegal argument exception. This happens because the logic checks do not properly restrict values of minutesOffset to the expected range between 0 and 59 or -59 and 0.

Compare original with synthetic explanations

Code Issue

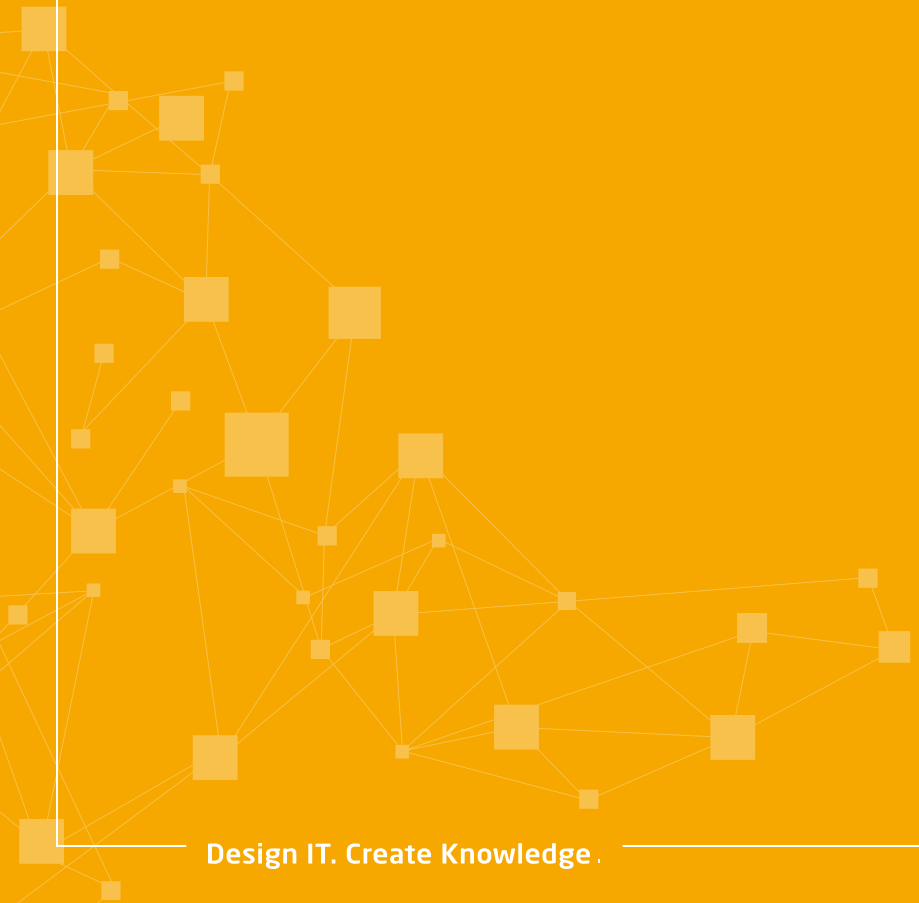
The root issue lies in incomplete or incorrect conditional checks for minutesOffset. Specifically:

The code neglects to enforce bounds for the minutesOffset variable.

While comments suggest the variable should be an "unsigned int," its actual behavior allows negative values, which causes exceptions.

The problematic condition appears on line 279, where an invalid range for minutesOffset is not properly handled.

Reflection



Quality of data

Bug explanations seemed to be rather elaborate and eloquent judging by the complexity metric (type-token-ratio).

An explanation is good if it leads to the fix.

Keeping up-to-date

There might be distribution shift in the occurrence of frequent bugs.

Development landscape is transforming rather quickly (new languages, IDEs, idioms, patterns)

Testing output

Distance metrics like BLEU and ROUGE can be misleading of the quality of summary explanations, e.g. when word tokenization is chosen for short explanations.

In general, metrics like explanation length and TTR can be used to stress-test classifier or LLM.

Quality of consolidated explanations

Consolidated explanations should be much more stable in terms of wrong details in the reports. Scores give some orientation, but can also be too rough.

Debugging integration

The classifier acts as a first gate. If we notice a decline in f1, we can look for a cause for that without having to consider the LLM. If the classifier performs decent, we can investigate problems in the consolidation done by the LLM.