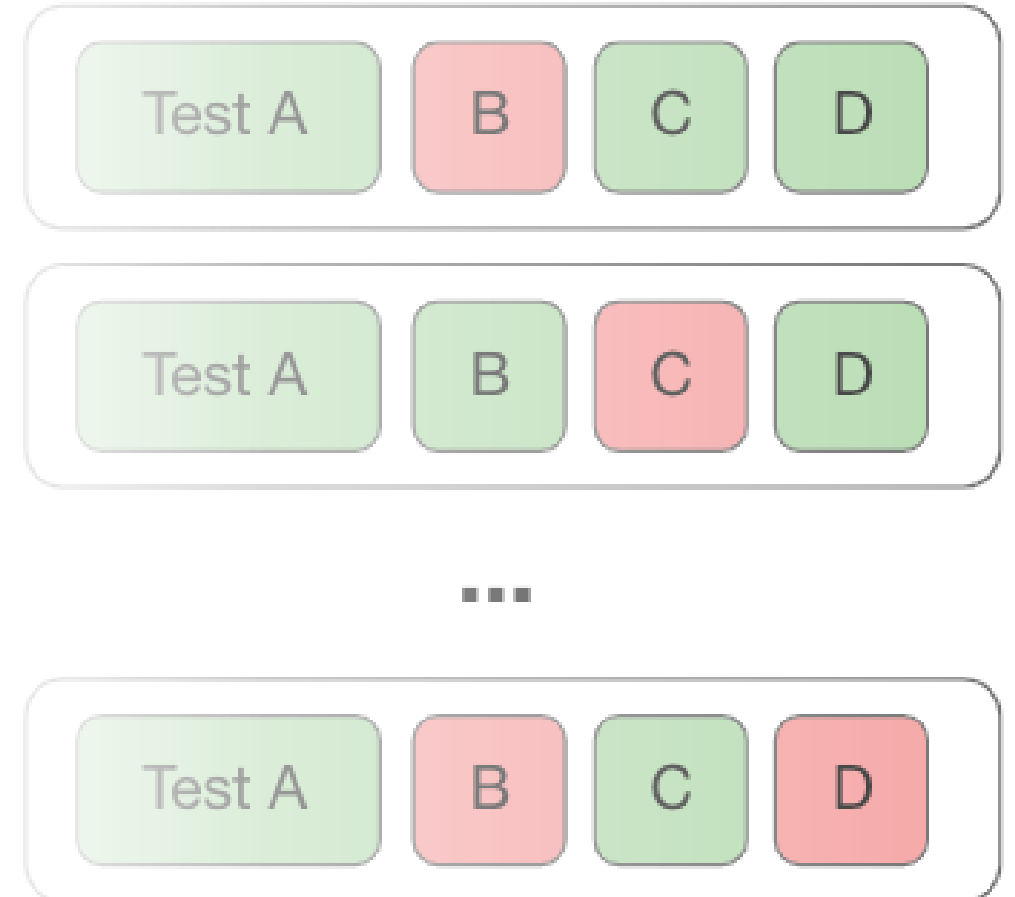


A Combined Approach to Detect Flaky Tests

Haya Samaana

Khaled Badran



Collect Flaky Tests

- Python projects
- ~22,000 Repositories
- ~800,000 Test Cases



Approach



Collect Flaky Tests



Extract Test Smells



Extract Vocabulary



Train and Test a Classifier



Analyze the Results

Collect Flaky Tests

- Python projects
- ~22,000 Repositories
- ~800,000 Test Cases



Extract Test Smells

- Download the test files
- PyNose for test smell extraction
- 20 different test smells



[Assertion Roulette](#)

[Conditional Test Logic](#)

[Constructor Initialization](#)

[Default Test](#)

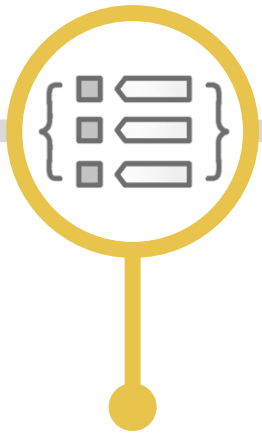
[Duplicate Assert](#)

Extract Vocabulary

- Extract Test Body
- Remove Stop Words
- Tokenize



Train and Test a Classifier

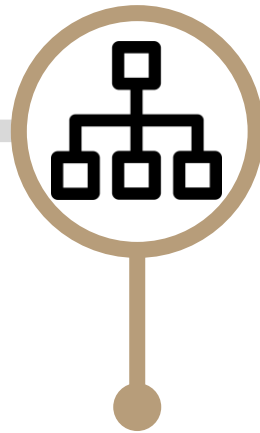


Binary

Count

TF-IDF

Frequency



Random
Forest

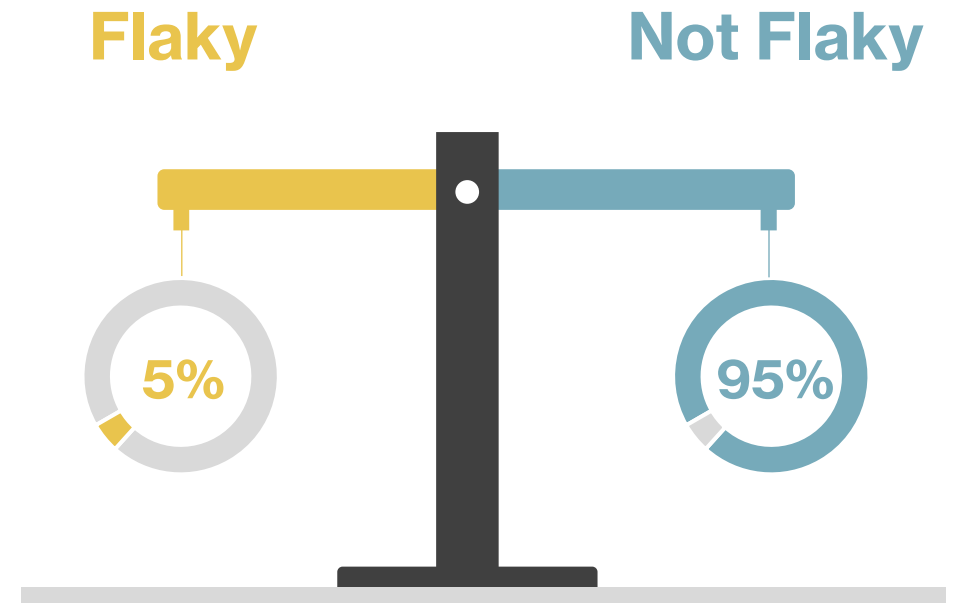
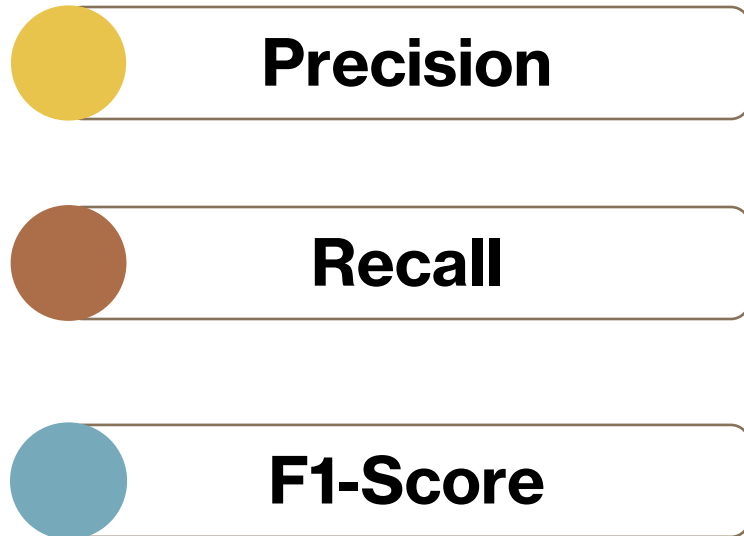


Mode = TF-IDF
words = 2000
trees = 100
...

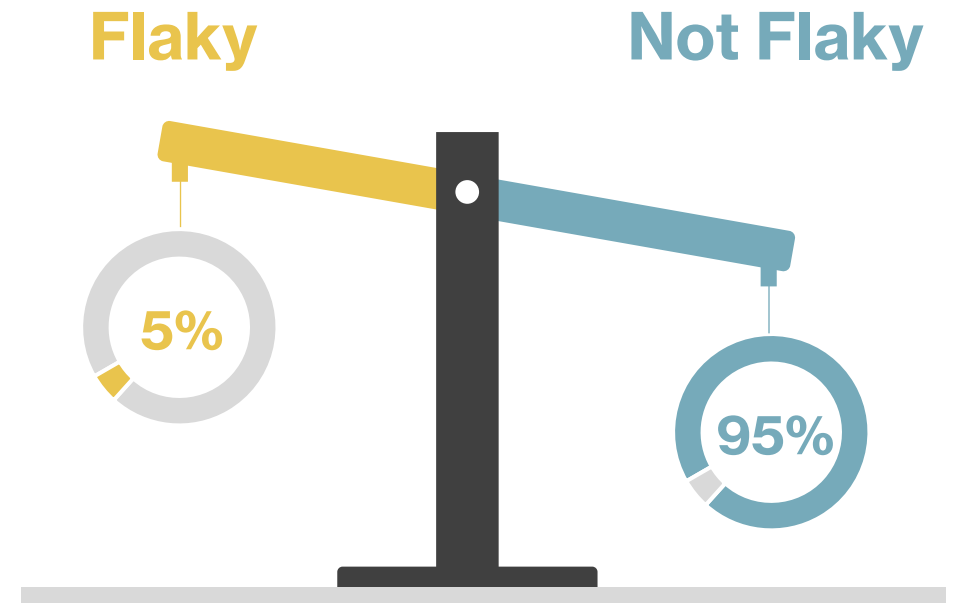
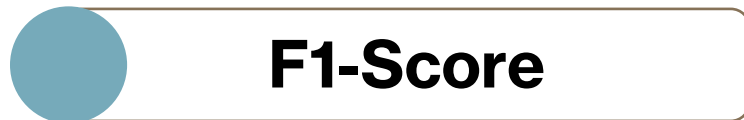


10 folds cross
validation

Analyze the Results



Analyze the Results

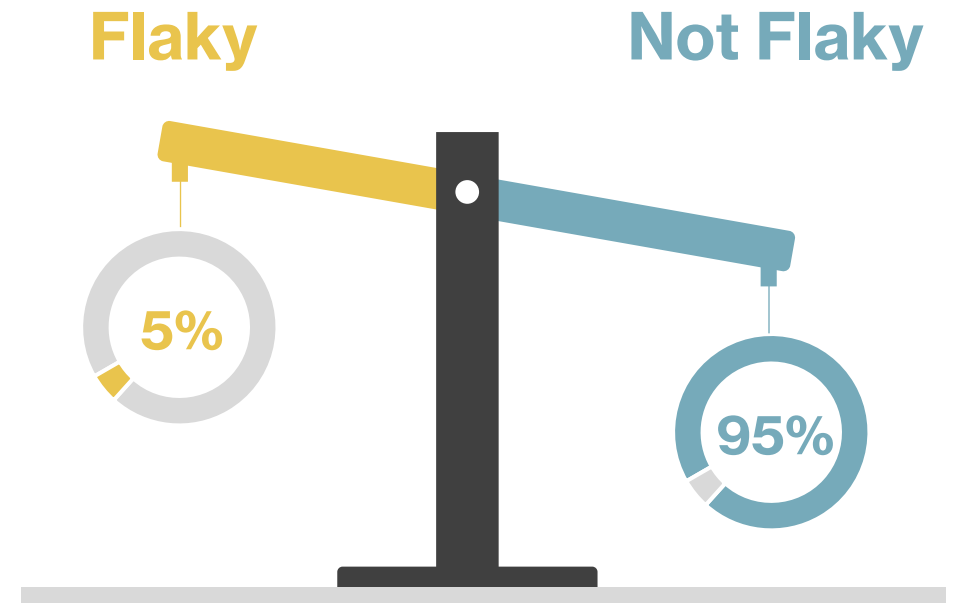


Analyze the Results

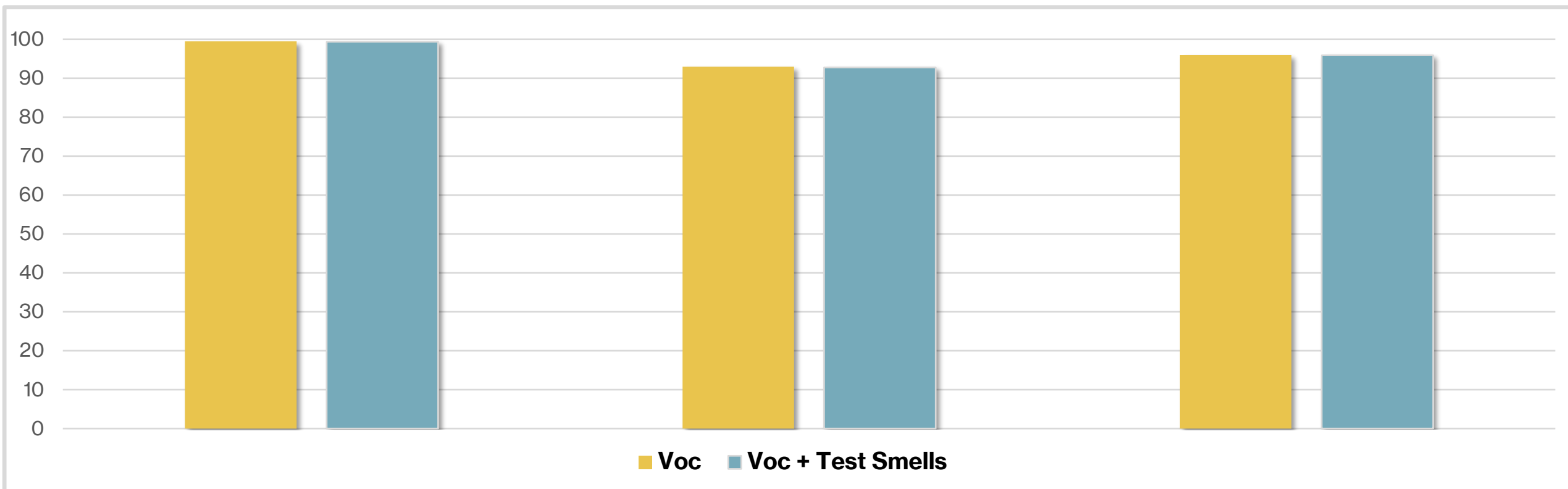
 **Macro Precision**

 **Macro Recall**

 **Macro F1-Score**



Results



Conclusion



Adding test smells to the vocabulary information does not provide a substitutional improvement to the classifier.

