

FourD: “Do Developers Discuss Design?” Revisited

Abbas Shakiba

Robert Green

Robert Dyer

Bowling Green State University

supported in part by the US National Science Foundation under CCF-15-18776 and CNS-15-12947

Do developers discuss design decisions?

2

- Are design decisions only happening before implementation?
- Do design discussions/decisions show in the commit logs?

Prior work

3

- Brunet, João, et al. **"Do developers discuss design?"**
11th Working Conference on Mining Software Repositories, 2014
 - Selected set of 5 projects for analysis
 - Analyzed:
 - commit logs
 - bug reports
 - discussions

Our Study

4

- Data from 2 software repositories
 - GitHub, SourceForge
- For each, 5 randomly selected projects
- Focus on commit logs
 - 200 randomly selected non-empty commits per project
- $2 \times 200 \times 5 = 2,000$ commits total
- Train ML classifiers to identify commits discussing design

Tools Used

5

- Boa Language and Infrastructure
 - A language for analyzing ultra-large-scale software repositories
- Weka
 - Data Mining Tool written in Java
- Ruby on Rails
 - A web application framework written in Ruby

Approach

6

Getting Data
(Boa)

Manual
Classification
(survey)

Pre-
Processing
(Weka)

Build
Models
(Weka)

Test
Models
(Weka)

Analyze
Results

Approach (Cont'd)

7

- Boa queries
 - Randomly pick 5 projects (not shown)
 - Randomly pick 200 commits (shown)

```
COMMITTS: output top(200) [string] of string weight float;
ids := {"6176545", "6150849", "209281", "13151128", "1019785"};

isempty := function(s: string) : bool {
  s2 := trim(s);
  if (match(`^\s*$`, s2))
    return true;
  if (match(`^no message$`, lowercase(s2)))
    return true;
  if (match(`^\s*\s*\s* empty log message \s*\s*\s*$`, lowercase(s2)))
    return true;
  return false;
};

exists (i: int; input.id == ids[i])
  visit(input, visitor {
    before rev: Revision ->
      if (!isempty(rev.log))
        COMMITTS[input.id] << rev.log weight rand();
  });
```

Approach (Cont'd)

8

- Survey website for crowdsourcing
- Each log shown to 2-3 users
- Required 2 YES or 2 NO

Does this commit discuss design?

Added some extra protective code around the end of the Power Allocation Phase. Updated scenarios to give Renegade ships some names

This commit:

Yes. It discusses design

I can not decide.

No. It does not discuss design

I am done with this survey

Does this commit discuss design?

Updated ArrayReference to work with primitive types.

This commit:

Yes. It discusses design

I can not decide.

No. It does not discuss design

I am done with this survey

Approach (Cont'd)

9

- Convert data to ARFF format

- e.g., data1:

“swapping the position of the input function <</>>”

Classified: **no**

| | | |
|-------------|----------|---|
| class: No | Swapping | 1 |
| attributes: | the | 2 |
| | position | 1 |
| | of | 1 |
| | the | 1 |
| | input | 1 |
| | function | 1 |
| | <</>> | 1 |

- e.g., data2:

“reorganized a package structure to better reflect a layered approach”

Classified: **yes**

| | | |
|-------------|-------------|---|
| class: Yes | reorganized | 1 |
| attributes: | package | 1 |
| | structure | 1 |
| | to | 1 |
| | better | 1 |
| | reflect | 1 |
| | a | 2 |
| | layered | 1 |
| | approach | 1 |

Approach (Cont'd)

10

- Convert data to ARFF format
- Tokenization
 - Remove tokens without letters
- Stemming
 - Remove stop words
 - a, an, the, to, etc.
 - Eliminate prefix and suffix
 - -ing, -ed, -ly, etc.

```
class: No      Swapping      1
attributes:   the                2
              position     1
              of           1
              the          1
              input        1
              function     1
              <</>>       1
```

```
class: Yes      reorganized      1
attributes:    package      1
              structure    1
              to           1
              better       1
              reflect      1
              a            2
              layered      1
              approach     1
```

Approach (Cont'd)

11

- Convert data to ARFF format
- Tokenization
 - Remove tokens without letters
- Stemming
 - Remove stop words
 - a, an, the, to, etc.
 - Eliminate prefix and suffix
 - -ing, -ed, -ly, etc.

```

class: No
attributes:
  Swapping 1
  the 2
  position 1
  of 1
  the 1
  input 1
  function 1
  <</> 1

```

```

class: Yes
attributes:
  reorganized 1
  package 1
  structure 1
  to 1
  better 1
  reflect 1
  a 2
  layered 1
  approach 1

```

Approach (Cont'd)

12

- Convert data to ARFF format
- Tokenization
 - Remove tokens without letters
- Stemming
 - Remove stop words
 - a, an, the, to, etc.
 - Eliminate prefix and suffix
 - -ing, -ed, -ly, etc.

| | | |
|-------------|----------|---|
| class: No | Swap | 1 |
| attributes: | input | 1 |
| | pos | 1 |
| | function | 1 |

| | | |
|-------------|----------|---|
| class: Yes | organ | 1 |
| attributes: | pack | 1 |
| | struc | 1 |
| | better | 1 |
| | flect | 1 |
| | layer | 1 |
| | approach | 1 |

Approach (Cont'd)

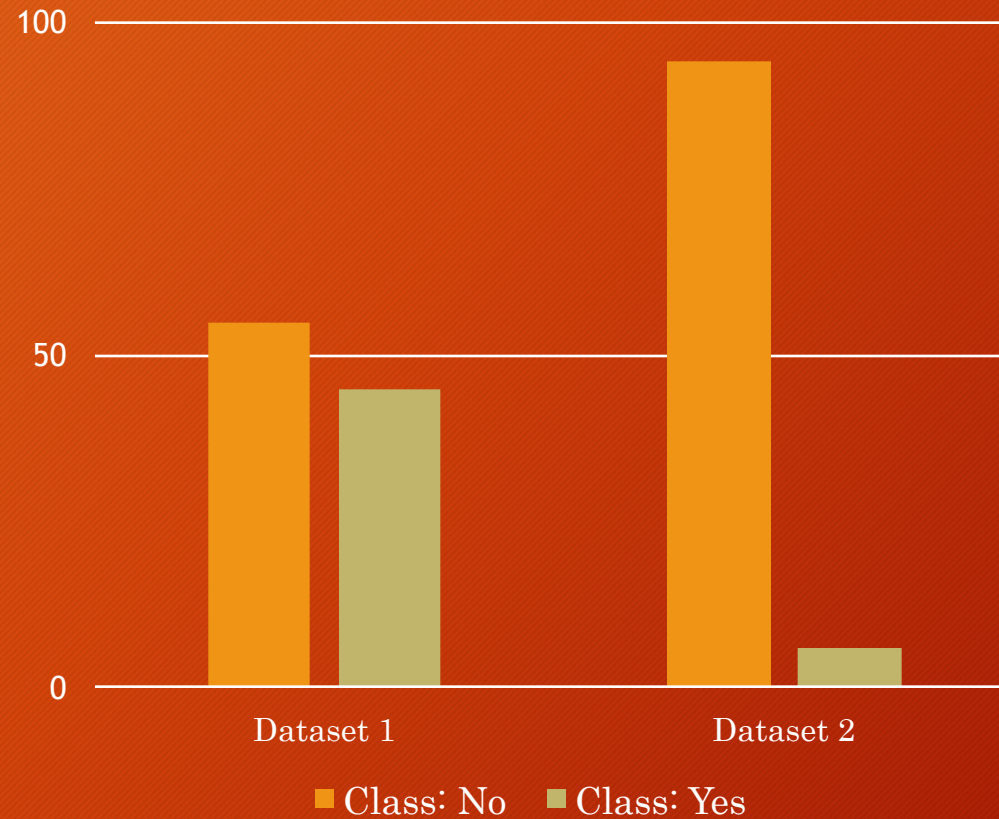
13

- Machine Learning Algorithms in Weka
 - Decision Tree
 - Random Forest
 - Naïve Bayes
 - Multinomial Bayes
 - Support Vector Machines
 - K-Nearest Neighbor

Difficulties

14

Different Data Distributions



Difficulties (Cont'd)

15

- Confusion Matrix
 - Add weight to cells
- Statistical measurements
 - F-Measure
 - G-Mean

| Confusion Matrix | | Predicted | |
|------------------|-----|---------------------|---------------------|
| | | Yes | No |
| Actual | Yes | True Positive (TP) | False Negative (FN) |
| | No | False Positive (FP) | True Negative (TN) |

$$Precision = \frac{TP}{TP + FP} \quad Recall = \frac{TP}{TP + FN}$$

$$Accuracy^{Yes} = \frac{TP}{TP + FN} \quad Accuracy^{No} = \frac{TN}{TN + FP}$$

$$F_1 score = \frac{Precision \times recall}{Precision + recall}$$

$$G_{Mean} = \sqrt{Accuracy^{Yes} \times Accuracy^{No}}$$

All Results

16

Decision Tree

| Data | Acc. | %TP | %TN | % F-M | G-mean |
|---------------------------------|-------|------|------|-------|--------|
| GitHub | 86.18 | 98.3 | 12.2 | 82.8 | 68.50 |
| sForge | 87.39 | 97.6 | 25 | 84.7 | 74.81 |
| Both | 86.99 | 96.5 | 20.8 | 85.2 | 65.82 |
| Increased False Negative Weight | | | | | |
| Github | 82.98 | 90.2 | 38.1 | 82.9 | 58.94 |
| sForge | 82.58 | 87.3 | 53.6 | 83.5 | 61.26 |
| Both | 82.13 | 87.8 | 47.3 | 69.70 | 59.32 |

Random Forest

| Data | Acc. | %TP | %TN | %F-M | G-mean |
|---------------------------------|-------|------|------|------|--------|
| GitHub | 86.38 | 99.0 | 8.6 | 86.4 | 71.14 |
| sForge | 86.98 | 99.8 | 8.5 | 82.0 | 87.23 |
| Both | 86.44 | 99.5 | 5.7 | 81.4 | 75.01 |
| Increased False Negative Weight | | | | | |
| GitHub | 87.58 | 96.4 | 33.1 | 86.0 | 73.31 |
| sForge | 87.49 | 95.7 | 37.1 | 86.3 | 72.68 |
| Both | 87.78 | 96.5 | 34.0 | 86.3 | 74.21 |

Naive Bayes

| Data | Acc. | %TP | %TN | %F-M | G-mean |
|---------------------------------|-------|------|------|------|--------|
| GitHub | 75.17 | 76.2 | 69 | 78.6 | 54.72 |
| sForge | 81.48 | 84.2 | 65 | 83.2 | 61.33 |
| Both | 79.63 | 82.4 | 62.3 | 81.4 | 58.28 |
| Increased False Negative Weight | | | | | |
| Github | 69.70 | 68.8 | 74.9 | 74.2 | 51.38 |
| sForge | 75.18 | 74.5 | 79.3 | 78.6 | 56.73 |
| Both | 67.01 | 65.3 | 77.4 | 72.0 | 50.17 |

Multinomial Bayes

| Data | Acc. | %TP | %TN | %F-M | G-mean |
|---------------------------------|-------|------|------|------|--------|
| GitHub | 74.27 | 74.8 | 71.2 | 78.0 | 54.33 |
| sForge | 70.57 | 70.0 | 73.5 | 85.5 | 51.85 |
| Both | 4.32 | 74.7 | 72.0 | 87.0 | 54.57 |
| Increased False Negative Weight | | | | | |
| Github | 81.28 | 86.7 | 47.5 | 82.3 | 57.74 |
| sForge | 78.98 | 82.9 | 55.0 | 80.9 | 56.23 |
| Both | 83.74 | 86.4 | 67.4 | 85.0 | 64.81 |

Support Vector Machine

| Data | Acc. | %TP | %TN | %F-M | G-mean |
|---------------------------------|-------|------|------|------|--------|
| GitHub | 85.48 | 93.7 | 34.5 | 84.7 | 64.95 |
| sForge | 86.27 | 93.7 | 40.7 | 85.7 | 68.21 |
| Both | 86.38 | 95.3 | 31.2 | 84.9 | 68.14 |
| Increased False Negative Weight | | | | | |
| GitHub | 85.58 | 92.1 | 45.3 | 85.4 | 66.25 |
| sForge | 85.69 | 92.0 | 47.1 | 85.6 | 72.68 |
| Both | 85.34 | 90.1 | 50.8 | 85.5 | 64.61 |

K-nearest neighbor

| Data | Acc. | %TP | %TN | %F-M | G-mean |
|---------------------------------|-------|------|------|------|--------|
| GitHub | 85.89 | 99.3 | 2.9 | 80.3 | 58.85 |
| sForge | 85.99 | 99.7 | 2.1 | 80.1 | 67.81 |
| Both | 86.18 | 99.9 | 1.8 | 80.1 | 80.16 |
| Increased False Negative Weight | | | | | |
| GitHub | 82.89 | 89.9 | 40.3 | 83.0 | 59.50 |
| sForge | 83.38 | 90.3 | 40.7 | 83.4 | 60.59 |
| Both | 85.43 | 95.6 | 21.9 | 83.1 | 62.81 |

Interesting Results

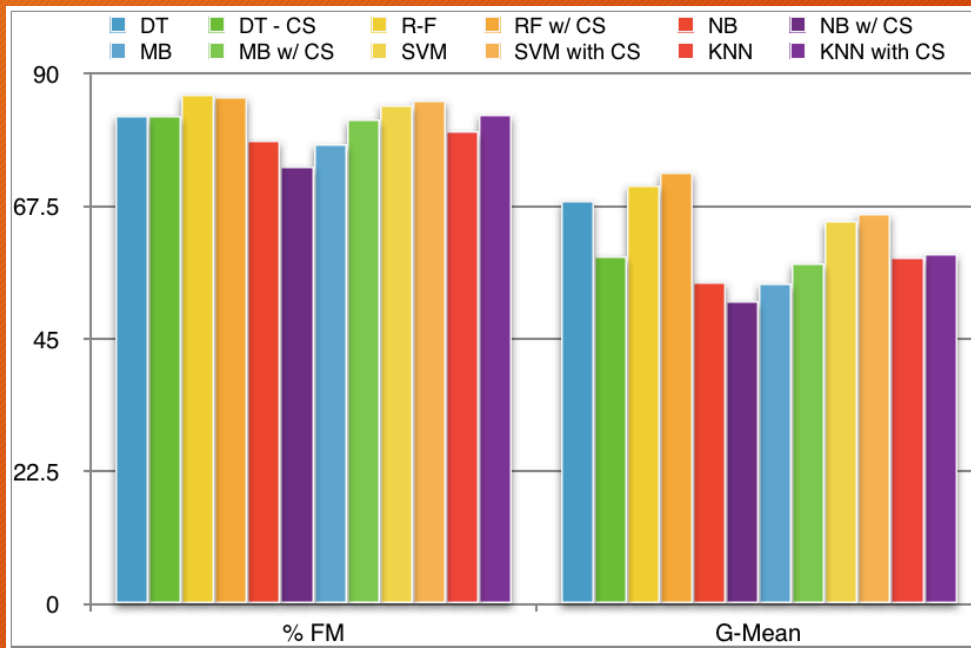
17

| Random Forest | | | | | |
|---------------------------------|-------|------|------|------|--------|
| Data | Acc. | %TP | %TN | %F-M | G-mean |
| GitHub | 86.38 | 99.0 | 8.6 | 86.4 | 71.14 |
| sForge | 86.98 | 99.8 | 8.5 | 82.0 | 87.23 |
| Both | 86.44 | 99.5 | 5.7 | 81.4 | 75.01 |
| Increased False Negative Weight | | | | | |
| GitHub | 87.58 | 96.4 | 33.1 | 86.0 | 73.31 |
| sForge | 87.49 | 95.7 | 37.1 | 86.3 | 72.68 |
| Both | 87.78 | 96.5 | 34.0 | 86.3 | 74.21 |

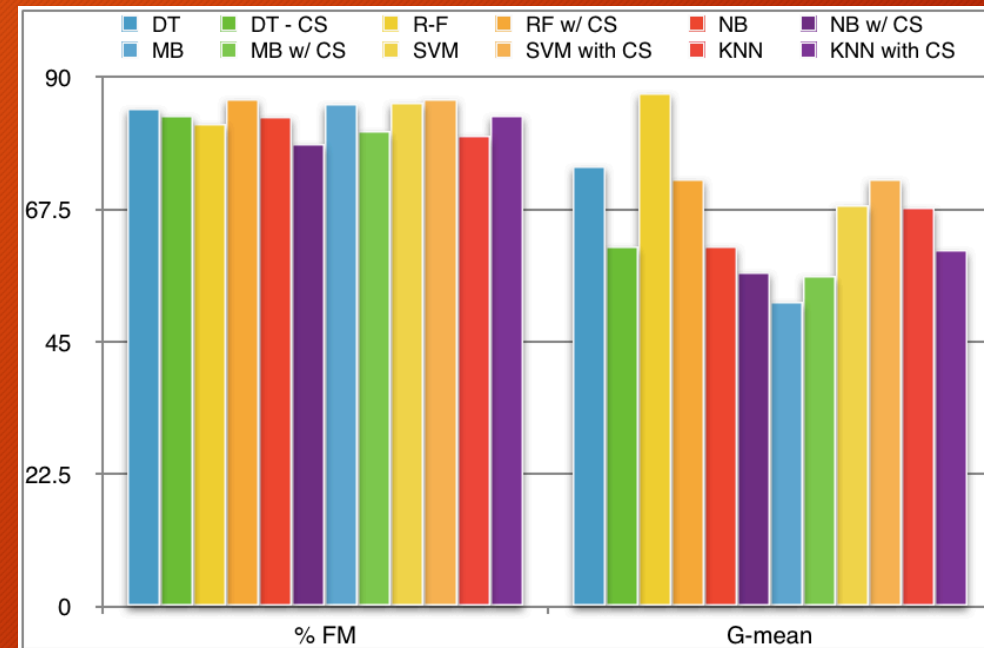
| Support Vector Machine | | | | | |
|---------------------------------|-------|------|------|------|--------|
| Data | Acc. | %TP | %TN | %F-M | G-mean |
| GitHub | 85.48 | 93.7 | 34.5 | 84.7 | 64.95 |
| sForge | 86.27 | 93.7 | 40.7 | 85.7 | 68.21 |
| Both | 86.38 | 95.3 | 31.2 | 84.9 | 68.14 |
| Increased False Negative Weight | | | | | |
| GitHub | 85.58 | 92.1 | 45.3 | 85.4 | 66.25 |
| sForge | 85.69 | 92.0 | 47.1 | 85.6 | 72.68 |
| Both | 85.34 | 90.1 | 50.8 | 85.5 | 64.61 |

F-measure and G-mean

18



GitHub



SourceForge

Future Work

19

- Move analysis completely into Boa
 - Pre-processing tasks
 - Machine learning models
- Do developers discuss other topics?
 - testing
 - debugging
 - etc.

```

COMMITTS: output top(200) [string] of string weight float;
ids := {"6176545", "6150849", "209281", "13151128", "1019785"};
isempty := function(s: string) : bool {
  s2 := trim(s);
  if (match(`^s*$`, s2))
    return true;
  if (match(`^no message$`, lowercase(s2)))
    return true;
  if (match(`^\\*\\* empty log message \\*\\*$`, lowercase(s2)))
    return true;
  return false;
};
exists (i: int; input.id == ids[i])
visit(input, visitor {
  before rev: Revision ->
  if (!isempty(rev.log))
    COMMITTS[input.id] << rev.log weight rev.log;
});
    
```

Survey Questions

Does this commit discuss design?

Added some extra protective code around the end of the Power Allocation Phase. Updated scenarios to give Renegade ships some names

This commit:

class: No 1
 attributes: input 1
 pos 1
 function 1

class: Yes 1
 attributes: organ 1
 pack
 struc
 better
 flect
 layer
 approach

20

| Likelihood Table | | Design | | |
|------------------|----------|--------|--------|--------|
| | | Yes | No | |
| Data | Refactor | 16/39 | 4/61 | 20/100 |
| | Fix | 3/39 | 16/61 | 19/100 |
| | Test | 2/39 | 31/61 | 33/100 |
| | struct | 18/39 | 31/61 | 49/100 |
| | | 39/100 | 61/100 | |

To summarize...

| Confusion Matrix | | Predicted | |
|------------------|-----|-----------|----|
| | | Yes | No |
| Actual | Yes | TP | FN |
| | No | FP | TN |

| Decision Tree | | | | | | |
|---------------------------------|-------|------|------|------|--------|--|
| Random Forest | | | | | | |
| Naive Bayes | | | | | | |
| Support Vector Machine | | | | | | |
| Data | Acc. | %TP | %TN | %F-M | G-mean | |
| GitHub | 85.48 | 93.7 | 34.5 | 84.7 | 64.95 | |
| sForge | 86.27 | 93.7 | 40.7 | 85.7 | 68.21 | |
| Both | 86.38 | 95.3 | 31.2 | 84.9 | 68.14 | |
| Increased False Negative Weight | | | | | | |
| GitHub | 85.58 | 92.1 | 45.3 | 85.4 | 66.25 | |
| sForge | 85.69 | 92.0 | 47.1 | 85.6 | 72.68 | |
| Both | 85.34 | 90.1 | 50.8 | 85.5 | 64.61 | |

