Paper Presentation

“Bayesian Analysis of Bug-Fixing Time using Report Data”

Presented by

Md Mynuddin
ID : ASH1825007M
NSTU, IIT

Published on

“16th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement”
Accepted: 18-23 May 2022

Overview

- ➔ Bug-fixing is the crux of software maintenance. There was a perception that bugs were "eating the world".
- ➔ To control changes in a project and manage backlogs of issues there are several available ITS(issue tracking system), such as **Bugzilla, YouTrack,** and **Jira**.
- ➔ Many studies explored information on bug reports available on different ITS such as
  - was this bug already registered?
  - who is the best person to fix this bug?
  - Is this a real bug?
  - Is this report good, and does it have enough information?
  - what is its priority?
  - How much time necessary to fix this bug

Problem

- Finding, reporting, and fixing bugs consumes 50% of the software developer’s time.
- Other studies or ITS does not offer out-of-the-box solutions like the statistical tests of the frequentist framework.
- Improve the bug fixing process.

Proposed framework

Bayesian Framework

➔ In this paper they show the process of using Bayesian statistics to analyze bug report data.

➔ Evaluate the relation between the **BFT**

- Bug report priority,
- Links between reports
- Code-churn size of bug-fix commits.

➔ Their results show that the existence of links and higher code-churn values lead to BFTs that are at least twice as long.

"Bayesian Analysis of Bug-Fixing Time using Report Data"

Dataset Characterization

Investigation answers the following research questions:

- RQ1: How does the existence of links in bug reports impacts the BFT ?
- RQ2: How does the priority level of a bug report impacts the BFT ?
- RQ3: How does the code-churn size of fixing commits relates to the BFT ?

The source to answer our research questions:

- The dataset comprises bug report information regarding 55 open-source projects from the Apache ecosystem
- All bugs reported in the dataset were identified and fixed between 2009 and 2018.
- They use three category(seven features) available in the dataset

Bug Report Links

InwardIssueLinks
OutwardIssueLinks fields.

Bug Report Priority

Low
Medium, High.

Bug Report Code-Churn Size

The sum of added lines
And removed lines

"Bayesian Analysis of Bug-Fixing Time using Report Data"

Dataset Preprocessing

Bug Report Links

Report without links (Rnl)
Report with link (Rwl)

Bug Report Priority

- Report with Low priority (Rlp)
- Report with Low Medium priority (Rmp)
- Report with Low High priority(Rhp)

Bug Report Code-Churn Size

- with higher code-churn values (hcc)
- with lower code-churn values (lcc)

"Bayesian Analysis of Bug-Fixing Time using Report Data"

Modeling Process and Models Description

We first define some sets, distributions, and variables that we use to describe the models:

- *days* / d: The time to fix the bugs in days, as non-negative real numbers. For all models, we considerer $log(days) \sim \mathcal{N}(\mu, \sigma^2)$, as *days* can not assume negative numbers.
- $\mathcal{G}$: the groups of bug fixing time in *days*. The groups of data are $\mathcal{G} = \{R_{nl}, R_{wl}, R_{lp}, R_{mp}, R_{hp}, R_{lcc}, R_{hcc}\}$, as presented in subsection 3.3.
- $\mathcal{P}$: The set of all projects. $\mathcal{P} = \{p_1, ..., p_{55}\}$, each p_i being one of the projects presented in the dataset presented in [40].
- $\mathcal{N} \sim (\mu, \sigma^2)$: The Normal distribution, defined the parameters mean μ and variance σ^2.
- Inv-Gamma(α, β) / Γ^{-1}: The Inverse Gamma distribution, defined by parameters α and β. Usually, the Inverse Gamma is used as prior for the variance in BDA.

Hierarchical Models formula for Bayesian Data Analysis

$$f_1(a, b) : a - b,$$

$$E[f_1(a, b)] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_1(a, b) p(a) p(b) \, \mathrm{d}a \, \mathrm{d}b,$$

$$f_2(a, b) : 1 \text{ if } a > b, \ 0 \text{ otherwise},$$

$$E[f_2(a, b)] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_2(a, b) p(a) p(b) \, \mathrm{d}a \, \mathrm{d}b.$$

"Bayesian Analysis of Bug-Fixing Time using Report Data"

Result : RQ1

How does the existence of links in bug reports impacts the BFT ?

No Links (a)

CI_L	CI_U	$\mu_{0_{MAP}}$	E[F1(a,b)]	E[F2(a,b)]
5.63	8.41	6.78	-9.76	0.00

W/ Links (b)

CI_L	CI_U	$\mu_{0_{MAP}}$	E[F1(a,b)]	E[F2(a,b)]
13.20	20.59	16.25	9.76	1.00

"Bayesian Analysis of Bug-Fixing Time using Report Data"

Result : RQ2

How does the priority level of a bug report impacts the BFT ?

Low Priority (a)

CI_L	CI_U	μ_{0MAP}	E[F1(a,b)]	E[F1(a,c)]	E[F2(a,b)]	E[F2(a,c)]
6.53	10.04	8.03	-0.37	-0.12	0.39	0.47

Medium Priority (b)

CI_L	CI_U	μ_{0MAP}	E[F1(b,a)]	E[F1(b,c)]	E[F2(b,a)]	E[F2(b,c)]
6.93	10.50	8.31	0.37	0.25	0.61	0.58

High Priority (c)

CI_L	CI_U	μ_{0MAP}	E[F1(c,a)]	E[F1(c,b)]	E[F2(c,a)]	E[F2(c,b)]
6.41	10.57	7.92	0.12	-0.25	0.53	0.42

“Bayesian Analysis of Bug-Fixing Time using Report Data”

Result : RQ3

How does the code-churn size of fixing commits relates to the BFT ?

Lower Code Churn

CI_L	CI_U	$\mu 0_{MAP}$	E[F1(a,b)]	E[F2(a,b)]
4.20	6.31	5.09	-4.78	0.00

Higher Code Churn

CI_L	CI_U	$\mu 0_{MAP}$	E[F1(a,b)]	E[F1(a,c)]
7.89	12.42	9.83	4.78	1.00

Future Work

As a future work, a “**Regression analysis”** can provide a more thoughtful view of all features presented in the used dataset, not only the ones selected in this paper, and how they relate to the BFT.

Conclusion

- ❖ They use of Bayesian workflow to analyze bug report data and assess the influence of three features.
- ❖ Evaluate the relation between the **BFT** and
  - ◆ Bug report priority.
  - ◆ Links between reports.
  - ◆ Code-churn size of bug-fix commits.
- ❖ And showed evidence that priority plays no role in BFT.
- ❖ In contrast, bug reports with higher values of code churn
- ❖ Or bugs reports related to other bugs (with links) need at least double the time to be fixed compared to their counterparts.


THANK
YOU


WHAT IF I TOLD YOU
VIA 9GAG.COM
THAT IT'S NOT A BUG, BUT AN
UNDOCUMENTED FEATURE
MEMEFUL.COM