Identifying Patterns of Inaccurate Time Estimations for Issue Resolution in Software Development

Adam Broniewski*, Himanshu Choudhary†, Tejaswini Dhupad‡ and Luiz Fonseca§
Mathematics and Computer Science, Eindhoven University of Technology
Eindhoven, The Netherlands
Email: *a.broniewski@student.tue.nl, †h.choudhary@student.tue.nl, ‡t.p.dhupad@student.tue.nl, §l.a.fonseca@student.tue.nl

***Abstract*—This paper analyzes the estimation time of issue resolution for software engineering projects. The public Jira dataset is used to (1) determine the accuracy of time estimation for different project types and (2) identify if there are characteristics of issues that result in more or less accurate time estimations. Statistical analysis is done using an analysis Of variance (ANOVA).**

***Index Terms*—Programming, Software engineering, Support, Agile, Time Estimation**

I. Introduction

Estimating efforts is an important step in managing software engineering projects. Estimates are an expectation of how much effort a task will take to be completed. This effort is usually taken as the time spent on the task, which can be converted into some scale, such as story points or use case points, or it can also be used as the time itself. In the latter case the estimation could be represented in some time measure like hours or days.

The effort estimation is crucial to the management of a software project because it helps split the tasks between the delivery cycles, assign an appropriate amount of work between different members of the project, predict deadlines, define milestones, and identify overall project costs and launch timelines. Underestimating efforts could lead to schedule pressure on developers or increased project overtime cost. On the other hand, overestimating efforts can decrease overall productivity by increasing idle resource time. Therefore, it is essential that the estimations are as close to reality as possible.

Jira is a popular issue tracking platform that supports agile project management when developing features or fixing bugs. In 2022, an open-source Jira dataset [3] was released that incorporates data from 16 different Jira repositories with 1822 projects. Although many studies on issue tracking have been conducted using data from GitHub or GitLab, Jira data has not received the same attention.

In this paper we consider the research problem of *(1) determining the typical accuracy of time estimates to resolve issues and (2) identifying if there are specific characteristics of issues that are correlated with (2a) inaccurate time estimates or (2b) accurate time estimates.*

In the following, we give an overview of what other related studies exist in this area in section II. In section III we enlighten the design decisions behind the methodology. Sections IV and VI present the results and their analysis, and in sections V and VII we highlight the threats to the validity of this study and what we conclude from it.

II. Related Work

Some previous studies [5] have aimed to identify how accurate the effort estimations are for different estimation approaches. Other studies aimed at identifying the reasons for errors in software effort estimation [4] using a combination of interviews with professionals responsible for estimates and statistical analysis of project's characteristics in the context of one specific company.

An empirical comparative study was completed between using effort-size (e.g. story-points) or effort-time (e.g. actual time spent) to estimate effort using open-source industrial projects in Jira found through a snowball search in Google. [1]. A Bayesian analysis was conducted to study the amount of time it will take to fix a reported bug using a dataset consisting of bug report information of 55 open-source projects from the Apache ecosystem [2]. However, we have not found a study that analyzes the Jira dataset [3] and identifies correlations between the accuracy of an estimation to resolve an issue and other information available in issue the issue metadata.

III. Methodology

To answer the research questions, we apply an approach of data extraction, data filtering, and statistical analysis.

We will need to determine if there is a particular subset of data we want to filter down to. Each research question will then use the same filtered dataset for analysis. To begin, the following will be explored for potentially relevant subsets:

1) explore to see what different types of issues exist and if all of them are relevant to the study
2) explore statistics for each project (number of issues, average number of sub-tasks per issue, issues types, length of time of project, overlap between creators and assignee's)

We extract the data using PyMongo [1], convert it to a Pandas DataFrame, and explore the data using a Jupyter notebook.

An appropriate subset of data will be one that remains relevant to software engineering stakeholders where estimated issue resolution time has an impact on project cost, quality

[1]PyMongo is a Python distribution containing tools for working with MongoDB

or schedule. We will also be able to describe the subset of data to ensure the reader can determine whether the paper's conclusions apply to their own work.

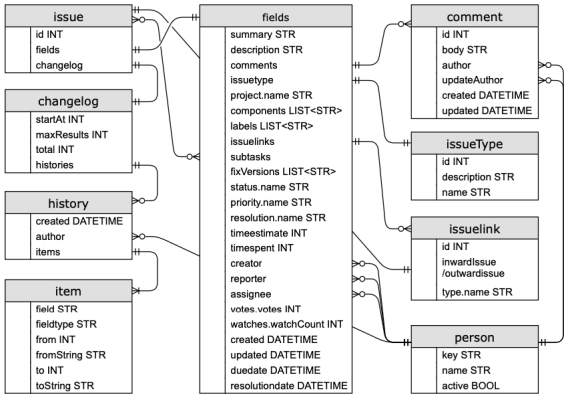

Fig. 1. Jira database schema [3]

(1) Determining the typical accuracy of time estimates to resolve issues

For each issue, the Jira dataset (see schema Figure 1) contains information regarding the estimated amount of time required to resolve an issue (`timeestimate`) and the actual amount of time spent resolving the issue (`timespent`). We will analyze the difference between theses two values using

A **t-test** is used to determine the sample mean of the difference between the estimated and actual time required for issue resolution, as well as the statistical distribution of the time estimates. This statistical test assumes that the underlying data is normally distributed and has definite variance. We use a generally applied significance level of 95%. Based on the analysis, we provide an interpretation and description of the distribution of values, and look at values for different types of projects, where types of projects are defined based on the initial data exploration.

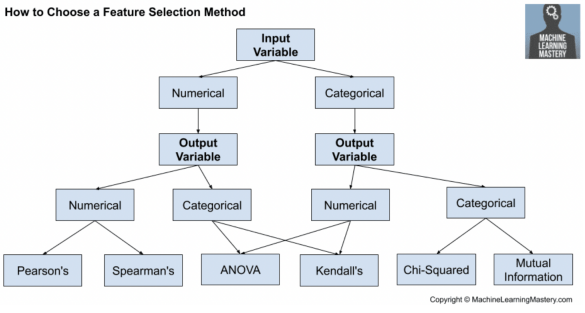

Fig. 2. Choosing feature selection methods

(2) Identifying if there are specific characteristics of issues that are correlated with (2a) inaccurate time estimates or (2b) accurate time estimates.

In order to analyze the characteristics impacting accurate or inaccurate time estimates, we are interested in identifying the individual as well as combined effects of different independent variables, which can be both categorical as well as numerical. An analysis of variance (**ANOVA**) methodology was chosen among other methods as it can be used to analyze both categorical as well as numeric variables [6] (Figure 2). The method uses an F-test for hypothesis testing which, compares variance within a group to the overall variance. It also assumes the independence between independent variables and normal distribution for the dependent variable.

We performed feature engineering to extract potentially insightful features from the Jira dataset that might be correlated to accurate or inaccurate time estimates.

- Project name - The parent project to which the issue belongs
- Number of associated sub-tasks - Sub-issues to this issue
- Priority - The issue importance in relation to other issues
- Votes - Number of people who want this issue addressed
- Watches - Number of people watching this issue
- Assignee - The person responsible to resolve the issue
- Creator - The person who created the issue
- Reporter - The person who found/reported the issue
- Number of comments - Community discussion on each issue

The difference between actual and estimated time is used as a dependent (target) variable to analyze the characteristics. Accurate estimates are those that are within one (1) standard deviation of the mean. Inaccurate estimates are more than one (1) standard deviation from the mean. Underestimated time is analyzed separately from overestimated time, and the analysis is completed on different project types as well.

IV. Results

Insert test here.

V. Threats to Validity

Insert test here.

VI. Discussion

Insert test here.

VII. Conclusion

Insert test here.

References

[1] H. H. Arifin, J. Daengdej and N. T. Khanh, "An Empirical Study of Effort-Size and Effort-Time in Expert-Based Estimations," 2017 8th International Workshop on Empirical Software Engineering in Practice (IWESEP), 2017, pp. 35-40, doi: 10.1109/IWESEP.2017.21.

[2] R. Vieira, D. Mesquita, C. L. Mattos, R. Britto, L. Rocha, and J. Gomes, 'Bayesian Analysis of Bug-Fixing Time using Report Data', in International Symposium on Empirical Software Engineering and Measurement, 2022, pp. 57–68.

[3] Lloyd Montgomery, Clara Lüders, and Walid Maalej. 2022. An alternative issue tracking dataset of public Jira repositories. In Proceedings of the 19th International Conference on Mining Software Repositories (MSR '22). Association for Computing Machinery, New York, NY, USA, 73–77. https://doi.org/10.1145/3524842.3528486

[4] Jorgensen, M., Molokken-Ostvold, K. (2004, December). Reasons for software effort estimation error: impact of respondent role, information collection approach, and data analysis method. https://ieeexplore.ieee.org/document/1377193. Retrieved December 8, 2022, from https://ieeexplore.ieee.org/document/1377193/

[5] Basha, S., Ponnurangam, D. (2010, April 8). Analysis of empirical software effort estimation models. arXiv.org. Retrieved December 8, 2022, from https://arxiv.org/abs/1004.1239v1
[6] Brownlee, J. (2019) How to choose a feature selection method for machine learning, MachineLearningMastery.com. Available at: https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/ (Accessed: December 8, 2022).