C02029: Doctor of Philosophy
33874: PhD Thesis: Software Engineering
Sep 2022

Multi-triage: A Multi-Task Learning Approach to Bug Triaging

Thazin Win Win Aung

School of Computer Science
Faculty of Eng. & IT
University of Technology Sydney
NSW - 2022, Australia

Multi-triage : A Multi-Task Learning approach to Bug Triaging

A thesis submitted in fulfillment of the requirements
for the degree of

Doctor of Philosophy
in
Software Engineering

by

Thazin Win Win Aung

to

School of Computer Science
Faculty of Engineering and Information Technology
University of Technology Sydney
NSW, Australia

Sep 2022

CERTIFICATE OF ORIGINAL AUTHORSHIP

I, *Thazin Win Win Aung*, declare that this thesis is submitted in fulfilment of the requirements for the award of the award of Doctor of Philosophy, in the *Faculty of Engineering and IT* at the University of Technology Sydney, Australia.

This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

SIGNATURE: _____

[Thazin Win Win Aung]

DATE: 30th Sep, 2022

PLACE: Sydney, Australia

Dedication

To my family who brings out the best in me.
To my supervisors for their inspiration and endless support.
To my late partner, my best friend, and my little buddy Albert, for their unconditional love and encouragement.

ABSTRACT

Bug triage plays a significant role in software maintenance activities, including optimization, error correction, and feature enhancement. Triage is the procedure of assigning the severity, issue type, and developer to resolve the issue in the most effective order. Performing triage is time-consuming and challenging, depending on the system's complexity. Thus, it is time-consuming and hinders the effectiveness of linkages between two triage tasks. An automated approach to assisting the issue allocation process to relevant category and developer benefits bug triages. A large body of previous work aims to address the allocation problem by conjecturing the extensive list of approaches ranging from the heuristics-based approach, text retrieval approach, and machine learning approach. However, these studies treated the issue of categorization and assignment tasks as a single task learning model and developed a multiple recommendation system.

This dissertation aims at leveraging the bug triage process by adopting the multi-task learning approach. We developed a multi-triage model, a system for predicting developers and issue kinds for a brand-new issue report. In our approach, we split issue reports, the text description, and code snippets into two separate tokens to conjecture the contributions of each context in the learning model. Furthermore, we addressed the class-unbalanced challenge of data sets by generating synthetic issue reports using the contextual data-augmentation approach.

We conducted four studies in this thesis. The first was an empirical study of the automatic traceability link recovery approach to analyze how previous studies addressed the linkages between software artifacts (e.g., requirements, issue reports, test cases, and source code). The second was the experimental studies about visualizing the linkages of software artifacts using the hierarchical trace map. The first two studies were mainly focused on understanding the broad concepts of how software artifacts can be linked together and presented to stakeholders effectively. Based on this accumulated knowledge, we designed the multi-triage model to identify the linkages between developers, issue types, and issue reports to leverage the bug triage process. Lastly, we conducted a case study of the issue tracking system used by the software consulting company to conjure the process of introducing the automatic developer assignment and labeling recommendation model in the bug triage process.

Our study led to several key findings. We found that the multi-triage model training time and performance are better than single-task learning models. We also uncovered that including the contextual data augmentation-based synthetic bug reports in training data sets can improve the learning model's performance noticeably. Lastly, we presented

the ability to learn issue descriptions and code snippets in two separate tokens and their effect on the learning model.

ACKNOWLEDGMENTS

First and foremost, thank you to my principal supervisor, Professor Yulei Sui, for his mentorship, guidance, and support throughout this long Ph.D. journey and his positive attitude toward life and study encouragement when it was most needed.

Thank you to my kind advisor, Dr. Huan (Angela) Huo and Dr. Yao Wan, for their invaluable lessons, guidance, and encouragement during my studies. Also, I am sincerely grateful to Dr. Christy Jie Liang, who directed me to the brighter side of the research world while I lost my way.

I am extremely grateful to them for their unconditional support, advice, and patience over the years. They opened my eyes to what research in software engineering is all about and taught me how exciting and challenging academia truly is. This achievement would not have been possible without their incredible feedback and mentorship.

I must thank the Australian Postgraduate Research Intern (APR. Intern) industrial internship grant for providing me with funding for the research work. I am also thankful to Dialog IT company for allowing me to conduct a field study in real-world software projects for my thesis writing. I would also like to thank all the software consultants involved in the field study for this research project. The field study could not have been successfully conducted without their participation and input.

Special thanks are due to my beloved late partner Kyi Thar Hain for his continuous support and understanding until his last moment. My thanks are extended to my family members and my friends for their constant source of inspiration. Last but not least, I would like to thank my little energizer Albert (chocolate labrador).

PUBLICATIONS

RELATED TO THE THESIS :

1. T. W. W. AUNG, H. HUO, AND Y. SUI, Interactive traceability links visualization using hierarchical trace map, in 2019 IEEE International Conference on Software Maintenance and Evolution (ICSME), 2019, pp. 367-369.

2. T. W. W. AUNG, H. HUO, AND Y. SUI, A Literature Review of Automatic Traceability Link Recovery for Software Change Impact Analysis, Association for Computing Machinery, New York, NY,USA, 2020, p. 14-24.

3. T. W. W. AUNG, Y. WAN, H. HUO, AND Y. SUI, Multi-triage: A multi-task learning framework for bug triage, Journal of Systems and Software, 184 (2022),p. 111133.

TABLE OF CONTENTS

List of Figures

List of Tables