SOFTWARE CLASSIFICATION USING STRUCTURE-BASED DESCRIPTORS

A thesis submitted to the College of Arts and Science

In partial fulfillment of the requirement for the degree

Master of Science (Information Technology)

Universiti Utara Malaysia

By

Qusai Hussein Ramadan

KOLEJ SASTERA DAN SAINS
(College of Arts and Sciences)
Universiti Utara Malaysia

PERAKUAN KERJA KERTAS PROJEK
(Certificate of Project Paper)

Saya, yang bertandatangan, memperakukan bahawa
(I, the undersigned, certify that)

QUSAI HUSSEIN RAMADAN
(802376)

calon untuk Ijazah
(candidate for the degree of) **MSc. (Information Technology)**

telah mengemukakan kertas projek yang bertajuk
(has presented his/her project paper of the following title)

SOFTWARE CLASSIFICATION USING STRUCTURE-BASED DESCRIPTORS

seperti yang tercatat di muka surat tajuk dan kulit kertas projek
(as it appears on the title page and front cover of project paper)

bahawa kertas projek tersebut boleh diterima dari segi bentuk serta kandungan dan meliputi bidang ilmu dengan memuaskan.
(that the project paper acceptable in form and content, and that a satisfactory knowledge of the field is covered by the project paper).

Nama Penyelia Utama
(Name of Main Supervisor): **DR. YUHANIS YUSOF**

Tandatangan
(Signature) : _____

Tarikh
(Date) : 9/11/2009

PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a postgraduate degree from Universiti Utara Malaysia, I agree that the University Library may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purpose may be granted by my supervisor(s) or, in their absence by the Dean of the Graduate School. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

UUM College of Arts and Science

Universiti Utara Malaysia

06010 UUM Sintok

Kedah Darul Aman.

ABSTRACT

With the huge increase of software functionalities, sizes and application domain, the difficulty of categorizing and classifying software packages for reuse and maintenance purposes is on demand. Building automatic classification mechanism will help to save the budget, time, and the efforts of the organizations, especially the administrators of software repositories.

This work includes the use of structure information contained in source code programs to automate program classification. Three software metrics namely; LOC, MVG and WMC1 have been extracted from programs of category board and puzzle obtained from *SourceForge.net*. A total of 2800 programs have been used during the training process while two different datasets of size (28) were used for testing. Based on the undertaken experiment, the IBK algorithm is noted to generate the highest classification accuracy (74.8%) compared to several other algorithms provided in the Weka tool. The study also shows that board programs are written in different structure compared to the puzzle programs. Hence, showing that structure information can be used to classify programs into application domain.

ACKNOWLEDGEMENTS

Firstly, I would like to express my appreciation to my supervisor, Dr. Yuhanis Yusof for her patience, constructive suggestions and guidance, consistence during this study. I appreciate the time she spent to discuss about the progress of the study.

It is also a pleasure to thank and present this study to my first teacher who helped me step by step to inspire me through the studying terms, so I dedicate all the brilliant moments of this study to my father, Mr. Hussein Saleem Ramadan.

Finally, I would like to offer my thanks to all persons who had helped and encouraged me during my study of MSc in IT; my mother for being the inspiration source, my dear family who supported me in all of my life for their patience and understanding especially my dear brother Mohammad, my friends for their ideas, moral supports and concern during the study development process. I am grateful to my faculty members and who taught me all basics of my specialization. I am deeply indebted to all of you for your sacrifices during my study. Thank you for everything.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

1.0 Introduction

This chapter aims to provide description on the undertaken study. This chapter contains background about the study area to provide useful information about software classification using structure-based descriptors. The problem statement, research questions, objectives, scope and significance of study are discussed in this chapter.

1.1 Study Background

Automatic software classification became one of the most important topics in software engineering area (Kawaguchi, Garg, Makoto, & Inoue, 2002). This is because of the new problems occurred upon constructing of software archives. For instance in 2002 the SourceForge.net had over seventy thousand registered software (Kawaguchi, Garg, Makoto, & Inoue, 2004). As this repository receives input (i.e. software files) from various developers whom have various backgrounds, categorizing the packages relies on the text input provided and/or contained in them. One issue which arises from such situation is to find a way to enhance the search process in the software's archive. So there is a need for alternative method in software classification (Kawaguchi, et al., 2002).

Existing approaches that adopts manual classification require more time and high level of software understanding and classification polices (Kawaguchi, et al., 2002). This is because of the large size code embedded in software and the ambiguous code

The contents of the thesis is for internal user only

REFERENCES

SourceForge Open Source. Retrieved 7 July 2009, from http://www.sourceForge.net

Alghamdi, J. S., Rufai, R. A., & Khan, S. M. (2005). OOMeter: A Software Quality Assurance Tool [Electronic Version]. *Ninth European Conference on Software Maintenance and Reengineering*, 190-191.

Ali, S. (2006). AutoAbstract: Problem Statement and Hypothetical Solutions [Electronic Version]. *Proceedings of the Testing: Academic & Industrial Conference – Practice And Research Techniques*, 75-80.

Ceylan, E., Kutlubay, F. O., & Bener, A. B. (2006). Software Defect Identification Using Machine Learning Techniques [Electronic Version]. *EUROMICRO Conference on Software Engineering and Advanced Applications*, 240 – 247.

Chan, V. K. Y., & Wong, W. E. (2005). Optimizing and simplifying software metric models constructed using maximum likelihood methods [Electronic Version]. *29th Annual International Computer Software and Applications Conference*, 1, 65-70.

Ciaramita, M., Murdock, V., & Plachouras, V. (2008). Online Learning from Click Data for Sponsored Search [Electronic Version]. *Proceedings of the 17th International Conference on World Wide Web*, 227-236.

Cufoglu, A., Lohi, M., & Madani, K. (2009). A Comparative Study of Selected Classifiers with Classification Accuracy in User Profiling [Electronic Version]. *WRI World Congress on Computer Science and Information Engineering* 3, 708-712.

Dracke, J. M. (1997). Social issues in the collection and use of software metric data position [Electronic Version]. *The Twenty-First Annual International Computer Software and Application Conference*, 586-587.

Eixelsberger, W., & Gall, H. (1998). Describing Software Architectures by System Structure and Properties [Electronic Version]. *The Twenty-Second Annual International Computer Software and Applications Conference*, 106 – 111.

Fang, Z. Z., Yu, L. P., & Ran, L. (2008). Research of text classification technology based on genetic annealing algorithm [Electronic Version]. *International Symposium on Computational Intelligence and Design*, 1, 256-269.

Fuchs, N. E. (1992). Specifications are (preferably) executable. *Software Engineering Journal, 7(5)*, 323-334.

Ghwanmeh, S., Kanaan, G., & Al-Shalabi, R. (2008). Enhance Arabic Information Retrieval System based on Arabic Text Classification [Electronic Version]. *4th International Conference on Innovations in Information Technology*, 461-465.

Gray, A., & Donell, S. M. (1997). Applications of fuzzy logic to software metric models for development effort estimation [Electronic Version]. *Annual Meeting of the North American Fuzzy Information Processing Society*, 394-399.

Guo, Y., Shao, Z., & Nan, H. (2008). Content-oriented automatic text categorization with the cognitive situation models [Electronic Version]. *International Symposium on Computer Science and Computational Technology*, 1, 512-516.

Hajji, M. S., Bas, J. M., Browne, k. R., Scbroder, P., Cmll, P. R., & Hemin, P. J. (1996). The Development Framework: work in progress towards a real-time control system design environment [Electronic Version]. *IEE Colloquium on Advances in Computer-Aided Control System Design*.

Hao, L., & Hao, L. (2008). Automatic Identification of Stop Words in Chinese Text Classification [Electronic Version]. *International Conference on Computer Science and Software Engineering*, 1, 718-722.

Hitz, M., & Montazeri, B. (1996). Chidamber and Kemerer's Metrics Suite: A Measurement Theory Perspective. *IEEE Transactions onSoftware Engineering, 22*(4), 267-271.

Hmida, M. B. H., & Slimani, Y. (2009). WSRF Services for Learning Classifiers from Data Grid [Electronic Version]. *International Conference on Computer Systems and Applications*, 27 - 32.

Holmes, G., Donkin, A., & Witten, I. H. (1994). WEKA: A Machine Learning Workbench [Electronic Version]. *Australian and New Zealand Conference on Intelligent Information Systems*, 357 - 361

Jianhui, L. (2008). On malicious software classification [Electronic Version]. *International Symposium on Intelligent Information Technology Application Workshops*, 368-371.

Kawaguchi, S., Garg, P. K., Makoto, M., & Inoue, K. (2002). Automatic categorization algorithm for evolvable software archive [Electronic Version]. *Six International Workshop on principles of Software Evolution*, 195-200.

Kawaguchi, S., Garg, P. K., Makoto, M., & Inoue, K. (2004). MUDABlue: An Automatic Categorization System for Open Source Repositories [Electronic Version]. *Proceedings of the 11th Asia-Pacific Software Engineering Conference*, 184-193.

Khan, M. K. S., Al-Khatib, W. G., & Moinuddin, M. (2004). Automatic Classification of Speech and Music Using Neural Networks [Electronic Version]. *Proceedings of the 2nd ACM international workshop on Multimedia databases*, 94-99.

Kirkby, R., & Frank, E. (2006). WEKA Explorer User Guidefor Version 3-5-3 [Electronic Version]. Retrieved 23/8/2009 from http://sourceforge.net/projects/weka/files/weka-3-6-windowsjre/3.6.1/weka3-6-1jre.exe/download.

Konig, R., Johansson, U., & Niklasson, L. (2008). G-REX: A Versatile Framework for Evolutionary Data Mining [Electronic Version]. *IEEE International Conference on Neural Networks Data Mining Workshops*, 971 - 974.

Korvetz, R., Ugurel, S., & Giles, C. (2003). Classification of Source Code Archive [Electronic Version]. *26th annual international ACM SIGIR conference on Research and development in information retrieval*, 425-426.

Kothari, C. R. (1985). *Research Methodology, Methods and Technique.*: Delhi: Wiley Eastern Limited.

Kwon, J., Wellings, A., & King, S. (2003). Assessment of Java programming language for use in high integrity system [Electronic Version]. *ACM Sigplan Notice*, 38(4), 34-46.

Lai, S., & Yang, C. C. (1998). A software metric combination model for software reuse [Electronic Version]. *Asia Pacific Software Engineering Conference*, 70-77.

Lam, W., Ruiz, M., & Srinivasan, P. (1999). Automatic Text Categorization and Its Application to Text Retrieval [Electronic Version]. *IEEE Transactions on Knowledge and Data Engineering*, 11, 865 - 879

Lange, R., & Mancoridis, S. (2007). Using Code Metric Histograms and Genetic Algorithms to Perform Author Identification for Software Forensics [Electronic Version]. *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, 2082-2089.

Lee, S., & Shimoji, S. (1993). BAYESNET: Bayesian classification network based on biased random competition using Gaussian kernels [Electronic Version]. *IEEE International Conference on Neural Networks*, 3, 1354-1359.

Lerner, B., Yeshaya, J., & Koushnir, L. (2007). On the Classification of a Small Imbalanced Cytogenetic Image Database [Electronic Version]. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 4, 204-215.

Li, Q., Wang, W., Han, S., & Li, J. (2007). Evolving Classifier Ensemble with Gene Expression Programming [Electronic Version]. *Third International Conference on Natural Computation*, 3, 546 - 550.

Lincke, R., Lundberg, J., & Lowe, W. (2008). Comparing Software Metrics Tools [Electronic Version]. *The 2008 international symposium on Software testing and analysis*, 131-141.

Markov, Z., & Russell, I. (2006,). An Introduction to the WEKA Data Mining System [Electronic Version]. *Proceedings of the 11th annual SIGCSE conference on Innovation and technology in computer science education*, 38 367-368.

Menkovski, V., Christou, I. T., & Efremidis, S. (2008). Oblique Decision Trees Using Embedded Support Vector Machines in Classifier Ensembles [Electronic Version]. *Conference on Cybernetic Intelligent Systems*, 1 - 6.

Merkl, D. (1995). Content-Based Software Classification by Self-Organization [Electronic Version]. *IEEE International Conference on Neural Networks*, 2, 1086-1091.

Nagappan, N. (2004). Toward a software testing and reliability early warning metric suite [Electronic Version]. *International Conference on Software engineering*, 60-62.

Nunamaker, J. F., & Chen, M. (1990). System Development in information system research [Electronic Version]. *Proceeding of the Twenty-Third Annual Hawaii International Conference on System Sciences*, 3, 631-640.

Nuntiyagul, A., Naruedomkul, K., Cercone, N., & Wongsawang, D. (2005). PKIP: Feature Selection in Text Categorization for Item Banks [Electronic Version]. *Conference on Tools with Artificial Intelligence*, 212-216.

O'Halloran, C., & Smith, A. (1998). Don't Verify, Abstract! [Electronic Version]. *Proceedings. 13th IEEE International Conference on Automated Software Engineering, 1998*, 53-62.

Panas, T., Quinlan, D., & Vuduc, R. (2007). Tool Support for Inspecting the Code Quality of HPC Applications [Electronic Version]. *Third International Workshop on Software Engineering for High Performance Computing Applications*, 1-5.

PANT, G., & SRINIVASAN, P. (2005). Learning to Crawl: Comparing Classification Schemes. *ACM Transactions on Information Systems (TOIS), 23*, 430–462.

Penix, J., Baraona, P., & Alexender, P. (1995). Classification and Retrieval of Reusable Component Using Semantic Feature [Electronic Version]. *10th Knowledge-Based Software Engineering Conference*, 131-138.

Phillips, N., & Black, S. (2005). Distinguish between Learning, Growth and Evolution [Electronic Version]. *IEEE International Workshop on Software Evolution*, 49-52.

Poulin, J. S., & Yglesias, K. P. (1993). Experiences with a faceted classification scheme in a large reusable software library (RLS) [Electronic Version]. *Seventeenth Annual International Computer Software and Application Conference*, 90-99.

Pumpuang, P., Srivihok, A., & Praneetpolgrang, P. (2008). Comparisons of Classifier Algorithms: Bayesian Network, C4.5, Decision Forest and NBTree for Course Registration Planning Model of Undergraduate Students [Electronic Version]. *Conference on Systems, Man and Cybernetics*, 3647 - 3651.

Sabharwal, C. L. (1998). Java, Java, Java. . *IEEE Potential Magazine, 17(3)*, 33-37.

Sebastiani, F. (2002). Machine Learning in Automated Text Categorization [Electronic Version]. *ACM Computing Surveys*, 34, 1-47.

Taylor, R. N., & Coutaz, J. (1994). Workshop on Software Engineering and Computer-Human Interaction: Joint Research Issues [Electronic Version]. *Conference on Software Engineering, 1994. Proceedings*, 356 - 357.

Vivanco, R. A., & Pizzi, N. J. (2003). Identifying Effective Software Metrics Using Genetic Algorithms [Electronic Version]. *Conference on Electrical and Computer Engineering*, 2, 1305 - 1308.

WekaDocs. Retrieved 2 September 2009, from http://wekadocs.com/node/13

Witten, I. H., & Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques* (2 ed.). San Francisco: Morgan Kaufmann.