An automated framework for software test oracle

Seyed Reza Shahamiri *, Wan Mohd Nasir Wan Kadir, Suhaimi Ibrahim, Siti Zaiton Mohd Hashim

Department of Software Engineering, Faculty of Computer Science and Information Systems, Universiti Teknologi Malaysia, 81310 UTM Skudai, Johor, Malaysia

Abstract

Context: One of the important issues of software testing is to provide an automated test oracle. Test oracles are reliable sources of how the software under test must operate. In particular, they are used to evaluate the actual results that produced by the software. However, in order to generate an automated test oracle, oracle challenges need to be addressed. These challenges are output-domain generation, input domain to output domain mapping, and a comparator to decide on the accuracy of the actual outputs.

Objective: This paper proposes an automated test oracle framework to address all of these challenges.

Method: I/O Relationship Analysis is used to generate the output domain automatically and Multi-Networks Oracles based on artificial neural networks are introduced to handle the second challenge. The last challenge is addressed using an automated comparator that adjusts the oracle precision by defining the comparison tolerance. The proposed approach was evaluated using an industry strength case study, which was injected with some faults. The quality of the proposed oracle was measured by assessing its accuracy, precision, misclassification error and practicality. Mutation testing was considered to provide the evaluation framework by implementing two different versions of the case study: a Golden Version and a Mutated Version. Furthermore, a comparative study between the existing automated oracles and the proposed one is provided based on which challenges they can automate.

Results: Results indicate that the proposed approach automated the software generation process 97% in this experiment. Accuracy of the proposed oracle was up to 98.26%, and the oracle detected up to 97.7% of the injected faults.

Conclusion: Consequently, the results of the study highlight the practicality of the proposed oracle in addition to the automation it offers.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

Software testing is the process of evaluating the software behavior to check whether it operates as expected in order to improve its quality and reliability. Since the testing process is highly time and resource consuming, complete testing is almost impossible; thus, testers use automatic approaches to facilitate the process and decrease its costs [1].

Test oracle is a mechanism to determine whether an application is executed correctly. It is a reliable source of how the SUT must operate [2]. It is also expected to provide correct results for any inputs that are specified by the software specifications, and a comparator to verify the actual behavior [3]. Automated test oracles are helpful in providing an adequate automated testing framework.

After test cases are executed and results of the testing are generated, it is necessary to decide whether the results are valid in order to determine the correctness of the software behavior. To verify the behavior of the SUT, correct results are compared with the results generated by the software. The results produced by the SUT that need to be verified are called actual outputs, and the correct results that are used to evaluate actual outputs are called expected outputs [2]. Test oracles are used as a complete and reliable source of expected outputs and a tool to verify the correctness of actual outputs. Usually, the verifier makes a comparison between actual and expected outputs. The process of finding correct and reliable expected outputs is called oracle problem [4].

Whitaker [2] explained software testing is divided into four phases: modeling the software’s environment, selecting test scenario, running and evaluation test scenario, and measuring testing process. Test oracle is what testers usually use in the third phase when they want to evaluate the scenarios. In addition, according to [5], oracle information and oracle procedure are building blocks of each test oracle. The former is a source of expected results, and the last is the comparator. Thus, we concluded that the activities to provide an oracle and verify test cases could be as follows [6]:

- **Oracle Generation**: This phase involves creating an automated test oracle. It is done by analyzing the software’s behavior and generating expected outputs. An oracle is a mechanism to determine whether an application is executed correctly. It is a reliable source of how the SUT must operate.

- **Oracle Verification**: This phase involves verifying the correctness of the actual outputs against the expected outputs. It is done by comparing the actual outputs with the expected outputs. An oracle is a tool to verify the correctness of the actual outputs.

- **Oracle Improvement**: This phase involves improving the oracle’s accuracy and precision. It is done by adjusting the oracle’s comparison tolerance. An oracle is a mechanism to determine whether an application is executed correctly.

- **Oracle Deployment**: This phase involves deploying the oracle in the software testing process. It is done by integrating the oracle into the testing framework. An oracle is a reliable source of how the software under test must operate.

Keywords: Software test oracle, I/O Relationship Analysis, Artificial neural networks, Mutation testing.

* Corresponding author. Address: E1B-18-8, The Tamarind Condo, Jalan Sentul Indah, Sentul East, 51100 Kuala Lumpur, Malaysia. Tel.: +60 177136662 (H/P); +60 340411490 (home).

E-mail addresses: admin@rezanet.com (S.R. Shahamiri), wnasir@cs.utm.my (W.M.N. Wan-Kadir), suhaimibrahim@utm.my (S. Ibrahim), sitiaizaton@utm.my (S.Z. Mohd Hashim).

1 Software Under Test.