Information and Software Technology 69 (2016) 1–15

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

Risk-based test case prioritization using a fuzzy expert system

Charitha Hettiarachchi [a,*], Hyunsook Do [b], Byoungju Choi [c]

[a] North Dakota State University, United States
[b] University of North Texas, United States
[c] Ewha Womans University, South Korea

ARTICLE INFO

Article history:
Received 20 December 2014
Revised 25 August 2015
Accepted 31 August 2015
Available online 18 September 2015

Keywords:
Regression testing
Requirements risks-based testing
Test case prioritization
Fuzzy expert system
Empirical study

ABSTRACT

Context: The use of system requirements and their risks enables software testers to identify more important test cases that can reveal the faults associated with system components.

Objective: The goal of this research is to make the requirements risk estimation process more systematic and precise by reducing subjectivity using a fuzzy expert system. Further, we provide empirical results that show that our proposed approach can improve the effectiveness of test case prioritization.

Method: In this research, we used requirements modification status, complexity, security, and size of the software requirements as risk indicators and employed a fuzzy expert system to estimate the requirements risks. Further, we employed a semi-automated process to gather the required data for our approach and to make the risk estimation process less subjective.

Results: The results of our study indicated that the prioritized tests based on our new approach can detect faults early, and also the approach can be effective at finding more faults earlier in the high-risk system components compared to the control techniques.

Conclusion: We proposed an enhanced risk-based test case prioritization approach that estimates requirements risks systematically with a fuzzy expert system. With the proposed approach, testers can detect more faults earlier than with other control techniques. Further, the proposed semi-automated, systematic approach can easily be applied to industrial applications and can help improve regression testing effectiveness.

1. Introduction

Software products change over time due to feature updates or user demand changes. When software functionalities change, software engineers need to retest the software to ensure that the changes did not affect software quality. Regression testing is one of the important maintenance activities, but it requires a great deal of time and effort. Often, software companies have pressures with time and budget, so expensive and time-consuming regression testing could be a major burden for them.

To overcome these schedule and cost-related concerns with regression testing, many researchers have proposed various cost-effective regression testing techniques [20,33,35]; in particular, test case prioritization techniques have been actively studied because they provide appealing benefits, such as flexibility for testers who need to adjust their testing efforts for the limited time and budget [10,13,42,44]. While the majority of test case prioritization approaches utilize source code information, some researchers have investigated using other software artifacts, such as system requirements and design documents, produced during early development phases [7,25,41]. For instance, Krishnamoorthi and Mary [25] proposed a system-level test case prioritization approach using the information obtained from the requirements specification, such as requirements completeness and implementation complexity. Srikanth et al. [41] also introduced a system-level test case prioritization technique that analyzes and evaluates the requirements in terms of requirement volatility, complexity, customer priority, and fault proneness.

In addition to utilizing requirements information for test case prioritization, some researchers used risk information that can help identify more important test cases that are likely to detect defects associated with the system's risks (e.g., safety or security risks) [43,48]. The results of previous research work empirically showed that the effectiveness of test case prioritization could be improved by using requirements risks. However, these risk-based test case prioritization techniques did not consider the direct relationship between requirements risks and test cases [48] or only used one type of risk, such as fault information obtained from preceding versions [43]. Further, these studies evaluated the approaches by measuring how fast the

* Corresponding author. Tel.: +001 7014910184.
E-mail addresses: charitha.hettiarachc@ndsu.edu (C. Hettiarachchi), hyunsook.do@unt.edu (H. Do), bjchoi@ewha.ac.kr (B. Choi).

http://dx.doi.org/10.1016/j.infsof.2015.08.008
0950-5849/

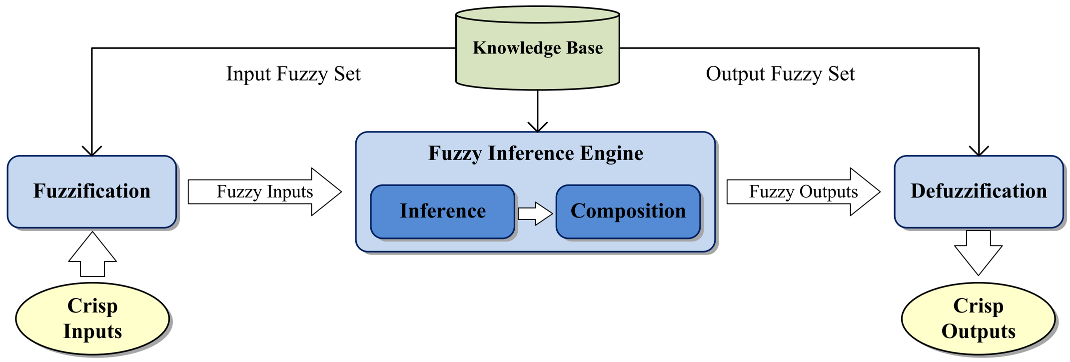

Fig. 1. Architecture of a fuzzy expert system.

reordered test cases detected faults; the approaches utilized risk information to prioritize tests, so they should be evaluated by measuring whether the detected faults are, indeed, from the locations where risks reside in the product. To address these limitations, our previous work [21] proposed a new requirements risk-based test case prioritization approach by considering the direct relationship between requirements risks and the test case. We also introduced a new evaluation method to measure how effective test case prioritization approaches were for detecting defects in risky components of software systems.

While our previous research was shown to be promising, the approach required human experts' involvement during the risk estimation process. Human involvement with the risk estimation process is important, but it makes the estimation process subjective and imprecise. To avoid the possible imprecision introduced by human judgment, a more systematic approach should be considered. Often, fuzzy expert systems have been utilized to address such problems because they can provide a mechanism to simulate the judgment and reasoning of experts in a particular field. To date, many researchers have used fuzzy expert systems in different application areas to help with complex decision-making problems, such as the diagnosis of disease [4] and risk estimation in aviation [18]. These studies have shown that fuzzy expert systems can be used to systematically represent human expertise in a particular domain and to deal with imprecision and subjectivity-related issues of the decision-making process while making the decision-making process more effective.

In this research, we propose a systematic risk estimation approach using a fuzzy expert system to address the limitations of our previous approach. We also reduced the number of risk items used for the risk estimation and simplified the prioritization approach so that we can perform test case prioritization with less effort. Further, from the results of our previous requirements risk-based approach, we learned that incorporating code information with the requirements could improve the rate of fault detection. Therefore, in this study, we used code information to extract requirements risks with respect to a few risk indicators in addition to the information obtained from requirements specifications written in natural language. Because we use code information, the proposed approach is applied during the testing phase after coding is done. To evaluate our approach, we used one open source application and one industrial application developed in Java.

The results of our study indicate that the systematic, risk-based test case prioritization approach has the capability to find faults earlier compared with other test case prioritization techniques, including our previous requirements risk-based approach. Moreover, the new approach is also better at finding more faults earlier in high-risk components than other techniques.

The rest of the paper is organized as follows. Section 2 describes the fuzzy expert system used in this research and the related work. Section 3 describes our new prioritization technique in detail. Section 4 describes our experiment including the research questions. Section 5 presents the results and analysis. Section 6 discusses our results and their implications. Section 7 presents the conclusions and discusses future work.

2. Fuzzy expert systems and related work

In this section, we provide background information on the fuzzy expert system and the existing work related to test case prioritization techniques, mainly focusing on techniques that use requirements, risks, and fuzzy expert systems which are most closely related to our work.

2.1. Fuzzy expert systems

In this research, we use a fuzzy expert system to derive requirements modification status (RMS) and potential security threats (PST) values. The fuzzy expert system used in this work simulates human expert's reasoning to derive the RMS and PST values for each requirement in a similar way that a human expert would estimate these values using the same input values. Existing empirical studies [19,46] indicate that fuzzy expert systems can improve the effectiveness of decision making process in many different application areas including regression testing. Moreover, fuzzy expert systems can handle ambiguity, which in turn produce output values much closer to realistic values.

To provide a better understanding about the process of acquiring the RMS and PST values, we summarize the mechanism for a fuzzy expert system used with our approach. A fuzzy expert system is comprised of fuzzy membership functions and rules. It contains four main parts: fuzzification, inference, composition, and defuzzification. Fig. 1 shows the typical architecture of a fuzzy expert system. The fuzzification process transforms the crisp input into a fuzzy input set. The inference process uses the fuzzy input set to determine the fuzzy output set using rules formulated in the knowledge base and the membership functions. The composition process aggregates all output fuzzy sets into a single fuzzy set. Finally, the defuzzification process calculates a crisp output using the fuzzy set produced by the composition process.

The knowledge base shown in Fig. 1 contains the selected fuzzy rule set. In a fuzzy expert system, fuzzy rules play a vital role because they are formulated based upon the experts' knowledge about the domain of interest. The fuzzy rule's antecedent defines the fuzzy region of the input space, and the consequent defines the fuzzy region of the output space. Fuzzy rules can not only support multiple input variables, but also multiple output variables. The following equation shows an example of a fuzzy rule:

if x is A and y is B then z is$C$