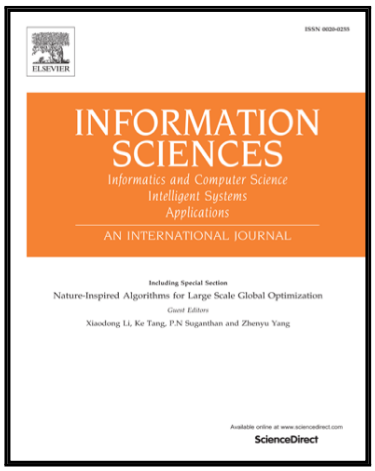
Accepted Manuscript



A novel approach for software defect prediction through hybridizing gradual relational association rules with artificial neural networks

Diana-Lucia Miholca, Gabriela Czibula, Istvan Gergely Czibula

PII: S0020-0255(18)30115-4
DOI: 10.1016/j.ins.2018.02.027
Reference: INS 13432

To appear in: *Information Sciences*

Received date: 20 September 2017
Revised date: 11 February 2018
Accepted date: 14 February 2018

Please cite this article as: Diana-Lucia Miholca, Gabriela Czibula, Istvan Gergely Czibula, A novel approach for software defect prediction through hybridizing gradual relational association rules with artificial neural networks, *Information Sciences* (2018), doi: 10.1016/j.ins.2018.02.027

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

ACCEPTED MANUSCRIPT

A novel approach for software defect prediction through hybridizing gradual relational association rules with artificial neural networks

Diana-Lucia Miholca, Gabriela Czibula*, Istvan Gergely Czibula

Department of Computer Science
Babeş-Bolyai University
1, M. Kogalniceanu Street
Tel: +40-264-405.327, Fax: +40-264-591.906
400084, Cluj-Napoca, Romania

Abstract

The growing complexity of software projects requires increasing consideration of their analysis and testing. Identifying defective software entities is essential for software quality assurance and it also improves activities related to software testing. In this study, we developed a novel supervised classification method called *HyGRAR* for software defect prediction. *HyGRAR* is a non-linear hybrid model that combines *gradual relational association rule mining* and *artificial neural networks* to discriminate between *defective* and *non-defective* software entities. Experiments performed based on 10 open-source data sets demonstrated the excellent performance of the *HYGRAR* classifier. *HyGRAR* performed better than most of the previously proposed approaches for software defect prediction in performance evaluations using the same data sets.

Keywords: Artificial Neural Network, Gradual Relational Association Rule, Machine Learning, Software Defect Prediction

1. Introduction

Data mining [23] techniques are now employed widely to extract meaningful knowledge from large volumes of data in different domains. Data mining applications include various real world scenarios such as assessing cybersecurity awareness [21], detecting communities in social networks based on user frequency pattern mining [31], and software process improvement to produce high-quality software [22]. Data mining-based solutions for predicting software defects have also been investigated [29] [17] [12].

Software defect prediction involves identifying defective software modules in new versions of a software system, which makes the testing process more efficient by focusing on testing and analyzing the modules identified as defective [10]. Identifying software defects is also useful for guiding code review, which is often employed as a quality assurance activity in agile methodologies for software development.

Despite its importance and extensive applicability, the detection of defective software modules is a complex task, particularly in complex and large-scale software systems. In particular, *cross-project* prediction is one of the main challenges in the field of defect prediction. The *cross-project* prediction strategy involves training the prediction model based on past data from different software projects and then using it to predict the defective software entities in a new project [34]. From the perspective of supervised learning, software defect prediction remains a difficult problem, mainly because the data used for training are highly imbalanced, where there is a very small number of *defective* instances compared with the number of *non-defective* instances. Numerous approaches have been proposed for software defect prediction and detection, but there is still a need to develop accurate and robust defect de-

*Corresponding author
Email addresses: mdir1308@scs.ubbcluj.ro (Diana-Lucia Miholca), gabis@cs.ubbcluj.ro (Gabriela Czibula), istvanc@cs.ubbcluj.ro (Istvan Gergely Czibula)
URL: http://www.cs.ubbcluj.ro/~gabis (Gabriela Czibula), http://www.cs.ubbcluj.ro/~istvanc (Istvan Gergely Czibula)

Download English Version:

https://daneshyari.com/en/article/6856561

Download Persian Version:

https://daneshyari.com/article/6856561

Daneshyari.com