



Performance-Aware Approach for Software Risk Management Using Random Forest Algorithm

Alankrita Aggarwal, IKG Punjab Technical University, India

 <https://orcid.org/0000-0002-0931-1118>

Kanwalvir Singh Dhindsa, Baba Banda Singh Bahadur Engineering College, India

 <https://orcid.org/0000-0002-7911-9244>

P. K. Suri, Kurukshetra University, India

ABSTRACT

Software quality assurance and related methodologies are quite prominent before actual launching the application so that any type of issues can be resolved at prior notifications. The process of software evaluation is one of the key tasks that are addressed by the quality assurance teams so that the risks in the software suite can be identified and can be removed with prior notifications. Different types of metrics can be used in defect prediction model and widely used metrics are source code and process metrics. The focus of this research manuscript is to develop a narrative architecture and design for software risk management using soft computing in integration with the proposed approach of random forest approach is expected to have the effectual results on multiple parameters with the flavor of multiple decision trees. The proposed approach is integrated with the framework of meta-heuristics with random forest in different substances and elements to produce a new substance.

KEYWORDS

Software Defects, Software Defects Avoidance, Software Risk Modeling

1. INTRODUCTION

The procedure of software bugs detection or programming absconds based designing with related expectation is one of the key undertakings while building up the strong programming items. (T.A. Shaikh, A. Chhabra, 2014)(Lyytinen, Kalle, Lars Mathiassen, and Janne Ropponen, 2016). Software Defect Prediction in development process is used to foresee the deformation in the product module (Kehan Gao, Taghi M. Khoshgoftaar, Naeem Seliya, 2011). Quantities of imperfection are available amid the improvement or after the conveyance of programming module. (Turhan, Burak, and Ayse Basar Bener, 2007)(Dorigo, Marco, and Luca Maria Gambardella, 1997). To get brilliant programming the expectation procedure is pursued to anticipate to the imperfections. The general execution and deformity free programming make immense steadfastness and confidence of the customer. Hardly

DOI: 10.4018/IJSI.2021010102

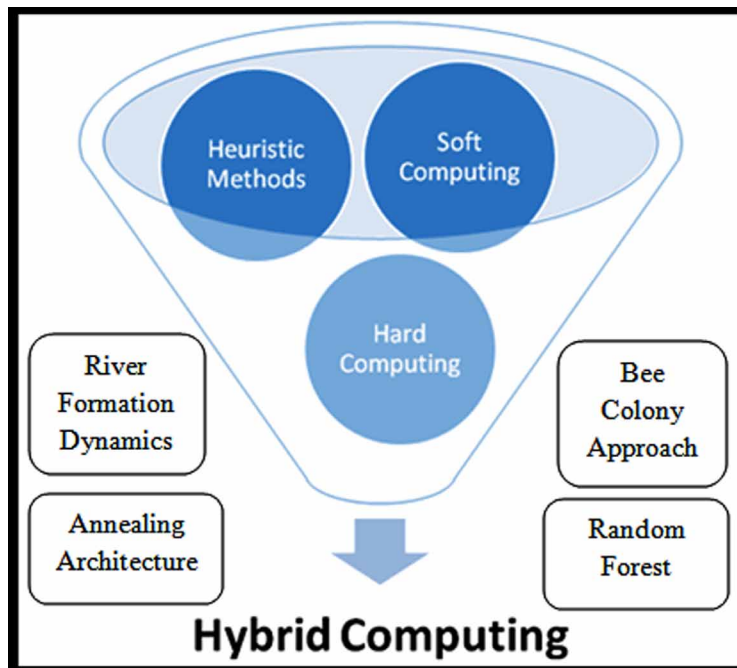
Copyright © 2021, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.

any enormous associations are utilizing this forecast procedure as they discharge their product and programming renditions much of the time and they have less time so rather than physically anticipating the deformities they use programming investigation process. (Fister Jr I, Yang XS, Fister I, Brest J, Fister D, 2013)(Wu, Shelly Xiaonan, and Wolfgang Banzhaf, 2010) (Abraham, Ajith, 2002) (Zadeh, Lotfi A, 1994).

2. KEY DEMARCATION BETWEEN HARD AND SOFT COMPUTING FOR BUGS PREDICTION

In case of hard computing based approach, the software bugs prediction is based on the static mathematical formulations while the soft computing based architecture with efficiency and performance aware with the usage of fuzzy based formulations. In these the river formation dynamics, simulated annealing, honey bee approach, random forest or any similar can be used.(Özcan, Ender, Burak Bilgin and Emin Erkan Korkmaz,2008)(Burke E, Kendall G, Newall J, Hart E, Ross P,2003) (Özcan E, Bilgin B, Korkmaz EE,2008)

Figure 1. Hard computing vs. soft computing



3. PROPOSED EFFECTIVE DESIGN AND FRAMEWORK FOR SOFTWARE RISK MANAGEMENT

The present work is having key focuses on the development of a novel architecture and implementation using Random Forest Approach so that the higher degree of efficiency and accuracy can be achieved. (Karaboga D, Akay B, 2009) (Karaboga D, Basturk B, 2007) Random Forest algorithmic program may be a supervised classification algorithmic program whereby it creates a forest by a way and makes it random towards the answer or optimization perspectives. There is a direct relationship between

6 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/article/performance-aware-approach-for-software-risk-management-using-random-forest-algorithm/266279

Related Content

An Introduction to Remote Installation Vulnerability in Content Management Systems

Mehdi Dadkhahand Shahaboddin Shamshirband (2015). *International Journal of Secure Software Engineering* (pp. 52-63).

www.irma-international.org/article/an-introduction-to-remote-installation-vulnerability-in-content-management-systems/142040

Decoupling Computation and Result Write-Back for Thread-Level Parallelization

Hiroaki Hirataand Atsushi Nunome (2020). *International Journal of Software Innovation* (pp. 19-34).

www.irma-international.org/article/decoupling-computation-and-result-write-back-for-thread-level-parallelization/256234

Towards a Multi-Formalism Multi-Solution Framework for Model-Driven Performance Engineering

Catalina M. Lladó, Pere Bonetand Connie U. Smith (2014). *Theory and Application of Multi-Formalism Modeling* (pp. 34-55).

www.irma-international.org/chapter/towards-a-multi-formalism-multi-solution-framework-for-model-driven-performance-engineering/91940

Admission Control in the Cloud: Algorithms for SLA-Based Service Model

Jose Luis Vazquez-Poletti, Rafael Moreno-Vozmedianoand Ignacio M. Llorente (2014). *Handbook of Research on Architectural Trends in Service-Driven Computing* (pp. 701-717).

www.irma-international.org/chapter/admission-control-in-the-cloud/115450

Constructivist Learning During Software Development

Václav Rajlichand Shaochun Xu (2009). *Software Applications: Concepts, Methodologies, Tools, and Applications* (pp. 910-922).

www.irma-international.org/chapter/constructivist-learning-during-software-development/29427