International Journal of Research and Development in Applied Science and Engineering (IJRDASE)
ISSN: 2454-6844

Analysing Software Defect Prediction Techniques - A Review of Literature

Preetika Verma
Computer Science and Engg
B.I.E.T., Lucknow
preetikaverma.smsit@gmail.com

Sameer Awasthi
Computer Science and Engg
B.I.E.T., Lucknow
sameer.awasthi@gmail.com

***Abstract*--Faults in software systems continue to be a major problem [1]. Fault is a flaw that results in failure. These software defects may lead to degradation of the quality which might be the underlying cause of failure. High quality of software is ensured by Software reliability and Software quality assurance. Both these concepts are drawn in throughout the software development and maintenance process. The activities like the performance analysis, functional tests, quantifying time and budget along with measurement of metrics are used to ensure quality. A software bug is an error, flaw, mistake, failure, or fault in a computer program that prevents it from behaving as intended (e.g., producing an incorrect result) [2]. Further, Software fault prediction facilitates to software engineers to pay attention to development activities on defect less code which enhance the software quality and minimize the cost and time to develop software system in today's era. There are many prediction models which are used to filter the software defects. This paper surveys literature review of articles for the past many years in order to explore how various prediction methodologies have been developed during this period in order to take care of the issues related to software defect.**

Keywords: Software Defects, software quality, assurance

1. Introduction

A software defect is an error, flaw, bug, mistake, failure, or fault in a computer program or system that may generate an inaccurate or unexpected outcome, or precludes the software from behaving as intended. A project team always aspires to procreate a quality software product with zero or little defects. High risk components within the software project should be caught as soon as possible, in order to enhance software quality. Software defects always incur cost in terms of quality and time. Moreover, identifying and rectifying defects is one of the most time consuming and expensive software processes. It is not practically possible to eliminate each and every defect but reducing the magnitude of defects and their adverse effect on the projects is achievable.

Software defects are expensive in terms of quality and cost. Moreover, the cost of capturing and correcting defects is one of the most expensive software development activities. It will not be possible to eliminate all defects but it is possible to minimize the number of defects and their severe impact on the projects. Software defect prediction is the process of locating defective modules in software. To produce high quality software, the final product should have as few defects as possible. Early detection of software defects could lead to reduced development costs and rework effort and more reliable software. So, the study of the defect prediction is important to achieve software quality.

Software quality prediction models seek to predict quality aspect such as whether a component is fault prone or not. Methods for identifying fault prone software modules support helps to improve resource planning and scheduling as facilitating cost avoidance by efficient verification. Such models can be used to predict the response variable which can either be the class of a module (e.g. fault-prone or not fault-prone) or a quality factor (e.g. number of faults) for a module. Statistical methods, machine learning method, and soft computing techniques are widely used in literature to predict software faults. This paper provides a critical review of the various work carried out in this field with the purpose of identifying future avenues of research.

2. Review of literature

This systematic literature review aims to identify and analyse the models used to predict faults in source code in papers published over 10 years (between January 2000 and December
2009). The output from the review is a summary and evaluation of how studies have developed, used and validated models for fault prediction in software engineering. Many important aspects of model construction are analysed and a rigorous starting point in the area is provided for future researchers.

Vipul Vashisht, Manohar Lal and G. S. Sureshchandar (2016), provided a framework which is expected to be more user-friendly, effective and acceptable for predicting the defects in multiple phases across software enhancement projects. A series of empirical experiments were carried out based on input and output measures extracted from 50 'real world' project subsystems. In order to increase the adoption and make the prediction framework easily accessible to project managers, a graphical user interface (GUI) based tool was designed and implemented that allowed input data to be fed easily.

P. Gupta, P. Sahai (2016) presented a literature review of the use of various techniques for artificial intelligence techniques in the field of software defect prediction. They showed that various techniques like Support Vector Logic Regression, neural network, Boosting etc. techniques could not give accuracy higher than 88 percent. Hence there is a large scope in the field of defect prediction.

Pradeep Kumar Singh, et. al. (2015) showed that product metric, process metrics and object oriented metrics are widely used in fault prediction techniques. Still fault prediction result is dependent on human expertise. So measuring human expertise in software fault prediction techniques is expected for future work. It is evident that fault prediction is dependent on skewed data but there is no evidence of fault prediction techniques for big data with real time.

Xiaoxing Yang , Ke Tang, (2015) introduced a learning-to-rank approach to construct software defect prediction models by directly optimizing the ranking performance. The work includes a novel application of the learning-to-rank approach to real-world data sets for software defect prediction, and a comprehensive evaluation and comparison of the learning-to-rank method against other algorithms that have been used for predicting the order of software modules according to the predicted number of defects.

Pooja Paramshetti , et. al. (2014) studied various machine learning techniques for software defect predication. They showed that software defect is indeed a major issue in software engineering. Software Defect Prediction using different machine learning techniques is to improve the quality of software development process. By using this technique, software manager effectively allocate resources. Analyzed the merits and demerits of artificial neural network, Support vector machine, Decision tree, association rule and Clustering machine learning techniques.

Mrs.Agasta Adline, Ramachandran. M(2014) They attempted to predict the fault–proneness of a program modules when fault labels for modules are not present. Supervised techniques like Genetic algorithm based software fault prediction approach for classification has been proposed.

Ahmet Okutan, et.al.(2012), proposed a novel method using Bayesian networks to explore the relationships among software metrics and defect proneness. In addition to the metrics used in Promise data repository, two more metrics, i.e. NOD for the number of developers and LOCQ for the source code quality has been proposed. The usefulness of the marginal defect proneness probability of the whole software system, the set of most effective metrics, and the influential relationships among metrics and defectiveness has been derived.

Mohamad Mahdi Askari and Vahid Khatibi Bardsiri (2014) used multilayer neural network method in order to improve and increase generalization capability of learning algorithm in predicting software defects. Proposed a new model by developing new learning methods based on support vector machine principles and using evolutionary algorithms. Results reveal that the proposed algorithm provides higher accuracy and precision compared to the other models.

Baljinder Ghotra, et. al. (2014), showed that the impact to the classification technique is minimum using NASA datasets. Further, they applied this to two new datasets i.e. the cleaned version of the NASA dataset and PROMISE dataset, which contains open source software developed in a variety of settings. The results in these new datasets show a clear, statistically distinct separation of groups of techniques, i.e., the choice of classification technique has an impact on the performance of defect prediction models.

Ahmet Okutan and Olcay Taner Yıldız, (2013) in order to predict the number of defects in the software modules proposed a new kernel method. It is based on a pre-computed kernel matrix which is based on the similarities among the modules of the software system. This method has been compared and show that it achieves comparable results. Furthermore, the proposed defect prediction method is also comparable with some existing defect prediction methods like linear regression and IBK. It was seen that prior to test phase or maintenance, developers can use the proposed method to easily predict the most defective modules in the software system.

Ms. Puneet Jai Kaur, Ms. Pallavi, (2013) discussed data mining techniques like association mining, classification and clustering for software defect prediction. It assisted the developers to detect software defects and correct them. Unsupervised techniques may be used for defect prediction in software modules, more so in those cases where defect labels are not available.

Karpagavadivu. K, et.al. (2012) Here the performance of various techniques used in software fault prediction were analysed. They also described some algorithms and its uses. They found that the aim of the fault prone module prediction using data mining is to improve the quality of software development process. By using this technique, software manager effectively allocate resources. The overall error rates of all techniques are compared and the advantages of all methods were analyzed.

Yajnaseni Dash, Sanjay Kumar Dubey, (2012) aimed to survey various research methodologies proposed to predict quality of OO metrics by using neural network approach. The application of artificial neural networks is an efficient method to estimate maintainability in object oriented system. It was seen that among the different soft computing techniques ANN possesses advantages of predicting the software maintenance effort by minimal computation. It can be used as a predictive model because of its incredible representation techniques and ability to perform complicated functions.

3. Conclusion:

This paper presents a survey of various machine learning techniques for software defect predication. From the survey, it can be observed that software defect is indeed a major issue in software engineering. Software defect module prediction using different machine learning techniques is to improve the quality of software development process. The main aim is to examine the performance of different techniques in software fault prediction. The fault prediction in software is significant because it can help in directing test effort, reducing cost, and increasing quality of software and its reliability. By using this technique, software manager effectively allocate resources. The results of software defect prediction show that the performance of the models depends upon the quality of data, its nature and the accuracy of the predictor and classifier variables. It was seen that most of the work has been carried out using NASA data. From the the above analysis it is seen that this topic attracts a great deal of interest by researchers. Also for better model development a lot depends on the quality, type and volume of data available.

REFERENCES

[1]. Vipul Vashisht, Manohar Lal and G. S. Sureshchandar (2016), "Defect Prediction Framework Using Neural Networks for Software Enhancement Projects", British Journal of Mathematics & Computer Science 16(5): 1-12.

[2]. P. Gupta, P. Sahai (2016), "A Review on Artificial intelligence Approach on Prediction of Software Defects", IJRDASE, Volume 9, Issue 1, pp. 1-4.

[3]. Pradeep Kumar Singh, Ranjan Kumar Panda and Om Prakash Sangwan, (2015), A Critical Analysis on Software Fault Prediction Techniques World Applied Sciences Journal 33 (3): 371-379, ISSN 1818-4952

[4]. Xiaoxing Yang , Ke Tang , Senior Member,(2015) IEEE, and Xin Yao , Fellow, IEEE A Learning-to-Rank Approach to Software Defect Prediction, This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

[5]. Pooja Paramshetti , D. A. Phalke, (2014), "Survey on Software Defect Prediction Using Machine Learning Techniques", International Journal of Science and Research (IJSR)), Volume 3 Issue 12.

[6]. Mrs.Agasta Adline, Ramachandran. M(2014), "Predicting the Software Fault Using the Method of Genetic Algorithm", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 3, Special Issue 2,, pp 390-398.

[7]. Ahmet Okutan, Olcay Taner Yıldız,(2013) "Software defect prediction using Bayesian networks", Empir Software Eng (2014) 19:154–181 © Springer Science+Business Media, LLC.

[8]. Ms. Puneet Jai Kaur, Ms. Pallavi, (2013), " Data Mining Techniques for Software Defect Prediction", International Journal of Software and Web Sciences (IJSWS), International Journal of Software and Web Sciences 3(1), pp. 54-57.

[9]. Karpagavadivu.K, et.al. (2012), "A Survey of Different Software Fault Prediction Using Data Mining Techniques Methods", International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 1, Issue 8, pp 1-3.

[10]. Yajnaseni Dash, Sanjay Kumar Dubey, (2012), " Quality Prediction in Object Oriented System by Using ANN: A Brief Survey", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 2,, pp.1-6.