Information Sciences Letters
An International Journal

http://dx.doi.org/10.18576/isl/090105

Needs and Importance of Reliability Prediction: An Industrial Perspective

Kavita Sahu[1,*] *and R. K. Srivastava*[2]

Department of Computer Science, Dr. Shakuntala Misra National Rehabilitation University, Lucknow, Uttar Pradesh, India - 226017

Received: 2 Sep. 2019, Revised: 22 Nov. 2019, Accepted: 23 Dec. 2019
Published online: 1 Jan. 2020

Abstract: Reliability plays a really important role for getting quality software. Already developed models of reliability prediction are well-recognised resources to support the management of software quality. Though many practitioners proposed several reliability prediction models but still, these prediction models have various problems during the use in the industry because there is a gap between proper implementation of software reliability prediction model development and their industrial use. There is a need to fill the gap between these two bridges of problems. Also the literature review of previous work and best practices, discloses the noticeable needs and importance of software reliability prediction. With this, author also gives some suggestions to practitioners for increasing the usability of the reliability prediction models. This article may help to developers for reducing the failure rate and enhancing the software reliability.

Keywords: Software Reliability, Software Failure, Reliability Prediction, Reliability Growth Model, Usable Prediction Models.

1 Introduction

The reliability decides the acceptance or failure of the software product. Due to high costs of development as well as rising economic competition, developers have increased the pressure to enhance the software reliability for business continuity[1]. Further, practitioners are trying to measure, control and predict the software reliability. Reliability is the most noteworthy characteristic of quality software and prediction of reliability is most measurable process related to customer sake. Also, reliability prediction is capable of delivering the outcomes perfectly every time to the practitioners [3]. Also, it measures the functionality of software to meet the operational needs. In other words, it may be helpful to the acceptance of the software product and ensures the software capability to rectify its failure. Software applications are multifaceted knowledgeable product, some errors are unavoidable during software development process [7]. The development process should include some measures to find out these errors and failures as soon as possible. Further, management process have many steps including identifying the errors, classifying the errors on the basis of error severity and mitigating the issues [11]. The developers along with error criticality and severity management process can use reliability prediction methods for assessing and minimizing the faults to improve the reliability of the software.

The main aim of this article is to identify the requirement of reliability prediction models and their importance for industries out there in the market of software [14]. Software reliability prediction models, which provides sufficient information to the practitioners for estimate the actual use of the prediction. There are many growth models of software reliability developed by different authors [20]. Unfortunately, use of these growth models is never realized, every model is validated only on a particular data set [21]. The range of datasets for prediction models are validated and limited. Hence,there is need to investigate the appropriate model of software reliability prediction in industrial perspective. Therefore,analysing the applicability of different popular growth models of software reliability is an important concern. To analyse the importance of popular reliability prediction models, there is need to identify the features of reliability, concerning the issues and suggestions for improvement [25].

Rest of the paper is structures as follows: Definition of software reliability, needs and importance of reliability during software development process is discussed in

* Corresponding author e-mail: kavi9839@gmail.com

section 2. Section 3 is discussing about prediction models for software reliability. Section 4 is discussing about needs of prediction models for software reliability. Section 5 is discussing about importance of prediction models for software reliability. Section 6 is given the suggestion for developers to increase the software reliability through prediction and finally conclusion is in section 7.

2 Software Reliability

Software reliability is defined as the possibility that software will not face the failure for a specified period of time [4] and also it performs as expected in a pre-defined condition without being faulty for a specified time under specified conditions [8]. Measuring reliability of software is significant step for developing a quality software. Almost all work of an organization depends totally on software system thats why evaluating and improving software reliability is necessary. Are reliability model states the failure process on the key factors that affect reliability of software including fault identification, removal, and the operational environment. Further, the failure rate of software applications commonly reduces due to fault identification and removal [9]. Reliability of hardware continues to change with change in time even after the distribution of system, while the software reliability is continuously improved throughout its development time [7]. The main reasons for software failure are complexity of software design, poor quality control,marketplace, capability profitable targets and engineering design estimation. Software complexity is enlarged through number of lines increased in a software application for lot of features. Lack of awareness is also a main accountable purpose of software failure. Due to the inherent complexity of the code and design of the software, producing error-free and 100 percent quality software is difficult [9].But there is a chance to improve the reliability. And this is why importance of reliability and reliability growth models is discussed in this article.

3 Reliability Prediction Models

A reliability growth model of software is designed to develop suggestions by predicting the failure in advance by taking lessons from the past failures to make quality software which is reliable too [28]. If the evaluated growth reduces through planned development, the designers will have appropriate time to improve new designs including new plans or resources to get the recognized difficulties. Measuring and predicting software reliability and its prediction needs the use of a suitable reliability model that defines the difference of reliability including time with its proper usage to software industry as well [29]. The parameters for establishing the model can be achieved either from the results of forecasted data during the period of model application, or, from the literature review [30]. In the broad category growth models of software reliability can be distributed into two categories including statistical models and soft computing based models. Both of these categories of models have its own pros and cons which will be discussed later.

4 Needs and Importance of Software Reliability Prediction

Increasing need of software systems is influencing the each and every field of todays life [6]. This importance of software in every days life makes it more important to develop a failure free software [28]. The growth activities of developing software applications are principally completed in a labour-intensive technique [5]. Also, it is essential to design more reliable software and predict its reliability, accurately. The impact of software failures may have critical consequences for economy and other important factors. To reduce the failure rates of software, there are different activities to prevent errors including error prevention, fault detection. Different software metrics are also available to measure the software project and thus its reliability. Unfortunately, software reliability prediction models are not good as it could be [5]. Software reliability is still not at its highest after having plenty of SRGMs available [6]. This is because of incorrect application of these models. Software products used in these days demand high quality and reliability in software because these software are highly used in different reliable sectors such as aerospace, aviation, defence, high level data warehouses etc. For example: NASA projects including space shuttle launching consumes approximately 500,000 lines of software code that is on board, and 3.5 million lines of code for ground control [27,28]. These types of software demands high accuracy and reliability with zero tolerance to faults. Further, software reliability is one of important attributes for software developers when developers are evaluating the quality of the software on the basis of its performance. Unfortunately, software failures are increasing in these days [28]. Different types of software have its different reliability severity also these software have adopted different reliability growth models. Some of reliability models that have focused on different types of factors shown in table 1.

Table 1 is describing the research that has been done in the field of reliability prediction on different phases of software development. Plenty of prediction models has been developed to work with execution or operation profiling phase, but there are very less research in phase of post-delivery defects. This strengthens the fact that there is need to focus on this phase. Post-delivery defects are those which cannot be distinguished while developing

Table 1: Reliability Models Focussing on Different Factors

Model v/s Attributes	Design/Com -plexity	Reuse/COTS	Code Complexity /Tech	UAT/Post Delivery Defects	Test Coverage Testing /Inspection	Execution Operation Profiling	Process and Product Metrics /SDLC	Total Input Areas Addressed
Gaffney[7]	X						X	2
GoelOkumoto[8]			X		X			2
Gokhale[9]					X	X		2
JM[10]				X	X			2
Markov Reward[11]						X		2
McCall[12]	X					X	X	2
Musa Basic[13]					X			1
Musa Okumoto[14]					X			1
Nelson[15]					X		X	2
Ning Huang[16]	X					X		2
Rayleigh[17]							X	1
Singh's Bayesian[18]	X							1
Shooman[19]				X		X		2
Tomek& Trivedi[20]					X			1
VivekGoswami[21]						X		1
Yacoub Scenario Based[22]	X			X		X		3
Chin Yu Huang[23]	X				X			2
P K Kapur[24]				X				1
Michael R. Lyu[25]				X				1
Pratik Roy et al.[26]				X		X		2

software. Generally, these kinds of defects are hard to remove and debug. Hence these bugs and defects effects maximum on reliability of software.

5 Suggestions for Practitioners

Reliability plays an important role to get quality software. With the help of practitioners, failures of software are identified and rectified during software development process. Further, software testers are tested to detect and remove the all errors with respect to failures. Measuring the detected and removing the errors with respect to software failures is called reliability of the software.For many important software applications, growth model of reliability plays a significance role to critically examine the software quality on the bases of previous failures [1]. The growth model is based on quantitative and practical based model. Also, this type model is helpful to estimate and predict the reliability of the software to improving the quality.

Developers are using such models to detect failures and decide whether and when to release the software in market according to its predictability of faults. Predictability of the growth model of reliability aims at predicting the number of faults and measurement of reliability, which is further useful in maintaining the trust of user towards software. In this work, some pertinent reliability prediction models have been examined to improve the usability of growth model of reliability in industrial perspective. Existing growth models of reliability prediction do not address during whole process of software development. Because, it is tough to calculate the software reliability of the end product.

This article is trying to identify some pertinent methods according to its applicability and provide suggestions for the practitioners to incorporate these models while development. The suggestions are prepared after thoroughly studying the current scenario of industry usage of growth models of software reliability. The suggestions are summarized in the following points:

- It is found that no thorough literature has been done to get the details of using reliability modelling researches into the real time software systems during development
- Only a few tools of reliability modelling are merely used in software industry.
- Almost all the models and methodologies have been implemented only on small and limited data sets.
- There is a need to conduct more experiments over large datasets and real world scenarios to extract concrete conclusion about the implication of software reliability growth models.
- After critically examining famous Software Reliability Models, it can be concluded that number of models are available in the market or industry but none is able to use most of reliability attributes hence delivers unreliable systems to stakeholders.
- According to this study, every reliability prediction model focussed on maximum three of attributes which may nowhere provide a reliable software.
- Based on assumption of using more than three phases for designing a reliability prediction model, it can be a wide area for research.
- This study may motivate developers to develop reliability prediction model which can produce a reliable as well as usable software with the perspective of industrial usage.

6 Conclusion

The reliability decides the acceptance or failure of the software product. Due to high costs of development as well as rising economic competition, developers have increased the pressures to enhance the reliability for increasing the users satisfaction [31]. For increasing the reliability, practitioners are trying to measure, control and predict the software reliability.This paper studied a

number of already developed reliability growth models (RGM) and its use in different phases of development respectively. This was found in the study that there is no generalised reliability prediction model that can be used during software development process. Authors come up with the suggestions to developers to develop and prescribe a generalised reliability prediction model which might use with every phase of development. This generalisation will also reduce the time and cost spent on application of different reliability prediction tools at different phases.

References

[1] Kaswan KS, Choudhary S, Sharma K., Software Reliability Modeling using Soft Computing Techniques: Critical Review. J Inform Tech SoftwEng 5:144. 2015

[2] Sultan H. Aljahdali and Khalid A. Buragga, Employing four ANNs Paradigms for Software Reliability Prediction: An Analytical Study, in ICGST-AIML Journal, ISSN: 1687-4846, Vol. 8, Issue II, 2008.

[3] C.Y. Huang, M.R. Lyu, Estimation and Analysis of Some Generalized Multiple Change-Point Software Reliability Models, IEEE Transaction on Reliability, Vol. 60, no. 2, pp. 498-514, 2011.

[4] M. Bisi and N. K. Goyal. Software Reliability Prediction using Neural Network with Encoded Input.International Journal of Computer Applications, 47(22),46-52, 2012.

[5] K.Khatatneh and T. Mustafa. Software Reliability Modeling Using Soft Computing Technique. European Journal of Scientific Research, 26(1), 147-152, 2009.

[6] [Online]. Available at:https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en

[7] J. E. Gaffney and J. Pietrolewiez. An Automated Model for Software Early Error Prediction. In Proc. of 13th Minnow Brook Workshop on Software Reliability, Blue Mountain Lake, NY,45-57, 1990.

[8] A. L. Goeland K. Okumoto. Time-Dependent Error-Detection Rate Model for Software Reliability and Other Performance Measures. IEEE Transactions on Reliability, 28(3), 206-211, 1979.

[9] S. S. Gokhale. Architecture-Based Software Reliability Analysis: Overview and Limitations. IEEE Transactions on Dependable and Secure Computing, 4(1), 32-40, 2007.

[10] J. D.Musa and K.Okumoto. A Logarithmic Poisson Execution Time Model for Software Reliability Measurement. In Proc. of the 7th International Conference on Software Engineering, IEEE Press,230-238,1984.

[11] K. S. Trivedi, M. Malhotra and R. M. Fricks. Markov Reward Approach toPerformabilityand Reliability Analysis. In Proc. of International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, IEEE,7-11, 1994.

[12] J. A. McCall. Quality Factors. In Encyclopedia of Software Engineering, J.J. Marciniak (Ed.). doi:10.1002/0471028959.sof265, John Wiley & Sons, 8-16, 2002.

[13] R. Kumar, M. Zarour, M. Alenezi, A. Agrawal and R. A. Khan. Measuring Security Durability of Software through Fuzzy-Based Decision-Making Process. International Journal of Computational Intelligence Systems, 12(2), 627-642, 2019.

[14] J. D. Musa and K.Okumoto. Application of Basic and Logarithmic Poisson Execution Time Models in Software Reliability Measurement. Software System Design Methods, Springer, Berlin, Heidelberg, 275-298, 1986.

[15] W. Nelson. Weibull Analysis of Reliability Data with Few or No Failures. Journal of Quality Technology, 17(3), 140-146, 1985.

[16] S. K. Chen, T.Kao, C. T.Chan, C. N.Huang, C. Y.Chiang, C. Y.Lai, and P. C. Wang. A Reliable Transmission Protocol for Zigbee-Based Wireless Patient Monitoring. IEEE Transactions on Information Technology in Biomedicine, 16(1), 6-16, 2012.

[17] D.Roy. Discrete Rayleigh Distribution. IEEE Transactions on Reliability, 53(2), 255-260, 2004.

[18] H.Singh, V.Cortellessa, B.Cukic, E.Gunel, V. Bharadwaj. A Bayesian Approach to Reliability Prediction and Assessment of Component Based Systems. In Proc. 12th International Symposium on Software Reliability Engineering, IEEE, 12-21, 2001.

[19] M. L. Shooman. Reliability of Computer Systems and Networks: Fault Tolerance, Analysis, and Design. John Wiley & Sons, ISBN: 978-0-471-46406-8, 2003.

[20] L.A. Tomek and K.S. Trivedi, Analyses Using Stochastic Reward Nets, in Software Fault Tolerance, M.R. Lyu (ed.), New York: Wiley, 1995, pp. 139-165.

[21] V.Goswami and Y. B. Acharya. Method for Reliability Estimation of COTS Components based Software Systems. In International Symposium on Software Reliability Engineering, 2009. [Online]. Available at: http://2009.issre.net/papers/issre2009 193.pdf

[22] S. Yacoub, B. Cukic and H. H. Ammar. A Scenario-Based Reliability Analysis Approach for Component-Based Software. IEEE Transactions on Reliability, 53(4), 465-480, 2004.

[23] C. K.Lee, T. C.Chang, Y. J.Huang, H. C.Fu, J. H.Huang, Z. C.Hsiao and K. S. Kao. Characterization and Reliability Assessment of Solder Micro Bumps and Assembly for 3D IC Integration. In 2011 IEEE 61st Electronic Components and Technology Conference (ECTC), IEEE, pp. 1468-1474, 2011.

[24] P. K.Kapur, H. Pham, A. Gupta, and P. C. Jha. Software Reliability Assessment with OR Applications, London: Springer, p. 364, 2011. [Online]. Available at: https://www.springer.com/gp/book/9780857292032

[25] M. R. Lyu. Software Reliability Engineering: A Roadmap. In Future of Software Engineering (FOSE'07), IEEE, pp. 153-170, 2007.

[26] P. Roy, G. S. Mahapatra, P. Rani, S. K. Pandey and K. N. Dey. Robust Feed Forward and Recurrent Neural Network Based Dynamic Weighted Combination Models for Software Reliability Prediction.Applied Soft Computing22, 629-637, 2014.

[27] NASA Article Case Study: NASA Space Shuttle Flight Control Software. [Online]. Available at: https://www.nap.edu/read/5018/chapter/4

[28] K.Sahu and R. K. Srivastava. Revisiting Software Reliability, Data Management, Analytics and Innovation.Advances in Intelligent Systems and Computing, Springer, 221-235, 2019.

[29] K.Sahu and R. K. Srivastava. Soft Computing Approach for Prediction of Software Reliability. ICIC Express Letters, 12(12),1213-1222, 2018.
[30] K. Sahu, Rajshree and R. Kumar. Risk Management Perspective in SDLC. International Journal of Advanced Research in Computer Science and Software Engineering, 1247-1251, 2014.
[31] R. Kumar, S. A. Khan and R. A. Khan. Revisiting Software Security Risks. British Journal of Mathematics & Computer Science, 11(6), 51-64, 2015.

Kavita Sahu is pursuing PhD from Department of Computer Science, Dr. Shakuntala Misra National Rehabilitation University, Lucknow, Uttar Pradesh, India and completed her Master's Degree (Information Technology) in 2014 from Department of Information Technology, Babasaheb Bhimrao Ambedkar University (A Central University), Lucknow, Uttar Pradesh, India. She has also published & presented papers in refereed journals and conferences. Her research interests are Artificial Intelligence, Fuzzy Logic, Neural Network and Software Reliability.

R. K. Srivastava is currently working as a Professor & Head, Department of Computer Science, Dean, Faculty of Computer Science & Information Technology and Dean, Faculty of Engineering & Technology, Dr. Shakuntala Misra National Rehabilitation University, Lucknow, Uttar Pradesh, India. Prof. Srivastava has more than 30 Years of Teaching & Research Experience. His area of interest is Artificial Intelligence, Fuzzy Logic, Neural Network and Crops Forecasting. He has published a number of National and International Research Papers, Technical Articles, Reviews and Chapters on Crops Forecasting, Artificial Intelligence, Fuzzy Logic, Neural Network and Software Reliability.