International Journal of Information Processing, 8(4), 35-44, 2014
ISSN : 0973-8215
IK International Publishing House Pvt. Ltd., New Delhi, India

Assessing the Fault Proneness Degree (DFP) of Change Request Artifacts by Estimating the Impact of Change Request Artifacts Correlation

Rudra Kumar M[a], A Ananda Rao[b]

[a]Department of Computer Science and Engineering, Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh, India, Contact: rk.madapuri@yahoo.com

[b]Professor, Director of IRP, SCDE JNTU Anantapur, Anantapur, India.

Exploring the impact of change requests applied on a software maintenance project helps to assess the fault-proneness of the change request to be handled further, which is perhaps a bug fix or even a new feature demand. In practice the major development community stores change requests and related data using bug tracking systems such as bugzilla. These data, together with the data stored in a versioning system, such as Concurrent Versioning Systems, are a valuable source of information to create descriptions and also can perform useful analyses. In our earlier work we proposed a novel statistical bipartite weighted graph based approach to assess the degree of fault proneness of the change request and Change Request artifacts. With the motivation gained from this model, here we propose a novel strategy that estimates the degree of fault proneness of a change request by assessing the impact of a change request artifact towards fault proneness that considers the correlation between change requests artifact as another factor, which is in addition to our earlier strategy. The proposed model can be titled as Assessing the Fault Proneness Degree of Change Request Artifacts by estimating the impact of Change Requests Correlation (DFP-CRC). As stated in our earlier model, the method DFP-CRC also makes use of information retrieval methods to identify the change request artifacts of the devised change request. And further evaluates the degree of fault proneness of the Change Requests by estimating the correlation between change requests. The proposed method is evaluated by applying on concurrent versioning and Change request logs of the production level maintenance project.

Keywords : Artifacts, Change Request, Concurrent Versioning System, Defect Forecasting, Fault Proneness, Product Metrics, Risk Prediction, SDLC.

1. INTRODUCTION

The present project development scenarios are letting to access the version histories due to the usage of tools such as concurrent versioning systems (CVS) [1]. These version histories are volume wise very high. This version history helps to extract the information regarding the progress of stages and strategies of that project development scenario, also provides information of the time and resource related to a change acquired. In recent literature related to software engineering and development, we can observe the extended role of this version history. Few of such developments are, using to access the change proliferation [2]; examining the impact of the bugs [3], accessing complexities of software [4], and also can use to access the reusability[5][6].

The said issues [2][3][4][5][6] issues usually raised due to analyzing the outcome of the development instead of process of the development. In related to this, the research work devised in [3] concluded that fault proneness is proportional to the count of code changes applied. The research article [2] devised a strategy that extracts patterns from changes registered in version history and the same used to recognize the tuples of te code need to be modified in related to a modification required. In this regard in our earlier effort we defined chain

Software Maintenance ICSM, pages 190–198, 1998.
17. Saxena P and Saini M. Empirical Studies to Predict Fault Proneness: A Review, *In International Journal Computer Applications*, 22(8):41-45, 2011.
18. Malhotra R, Kaur A and Singh Y. Empirical Validation of Object-Oriented Metrics for Predicting Fault Proneness at Different Severity Levels using Support Vector Machines, *In International Journal System Assurance Engineering Management*, 1(3):269-281, 2010.
19. Handa, A and Wayal, G Sofware Quality Enhancement using Fuzzy Logic with Object Oriented Metrics in Design, *In International Journal Computer Engineering Technology (IJCET)*, 3(1):169-179, 2012.
20. Chandra E and Linda, P E. Assesment of Sofware Quality Through Object Oriented Metrics, *In CIIT International Journal on Software Engineering*, 2:2, 2010.
21. *http://www.innatesolutions.net/.*
22. Bohner, S A and Arnold, R S (Eds.). Software Change Impact Analysis, An Introduction to Software Change Impact Analysis, *In IEEE Computer Society Press*, pages 1-26, 1996.
23. Rudra Kumar M and Ananda Rao A. Assessing the Fault Proneness Degree(DFP) of Change Requests and Change Request Artifacts: A Statistical Bipartite Graph Strategy, *In* 11^{th} *International Multi Conference on Information Processing*, 2014.
24. Malhotra R. A Defect Prediction Model for Open Source Software. *In Proceedings of the world Congress on Engineering, Vol.II.,London(UK)*, pages 4-6, 2012.
25. Zimmermann T and Nagappan N. Predicting Defects Using Network Analysis on Dependency Graphs, *In International Conference on Software Engineering ICSE*, pages 531-540,2008.
26. Kobayashi K, Matsuo A, Inoue K, Hayase. Y Kamimura, M Yoshino, T Impact Scale. Quantifying Change Impact To Predict Faults in Large Software Systems, *In IEEE International Conference on Software Maintenance (ICSM)*, pages 43-52, 25-30, 2011.

Ananda Rao Akepogu received B.Sc. (M.P.C) Degree from Silver Jubilee Government College, SV University, Andhra Pradesh, India. He received BTech. degree in Computer Science and Engineering and M.Tech degree in AI & Robotics from University of Hyderabad, Andhra Pradesh, India. He received Ph.D from Indian Institute of Technology, Madras, India. He is Professor of Computer Science and Engineering and Director of IR& P, SCDE of JNTU, Anantapur, India. Ananda Rao published more than fifty research papers in international journals, conferences and authored three books. His main research interest includes software engineering and data mining.

Rudra Kumar M is pursuing Ph.D in Computer Science & Engineering from JNTUA, Anantapur, Andhra Pradesh. He received his MTech in Computer Science and Engineering from the same university. He received BTech degree in Computer Science and Engineering from Sri Venkateswara University, India. He is associate professor of computer science and Engineering, AITS, JNTUA, Anantapur. He is a member of IEEE.