followed by association rules mining inside each cluster. We have validated our results on 1,695 bug reports of AddOnSDK, Thunderbird and Bugzilla products of Mozilla open source project. We have presented top five association rules for 20% minimum confidence and 3% and 7% minimum support. We observe that, if we apply association mining after clustering, we get different association rules. As we are partitioning the datasets into clusters, we get association rules with decreased support count i.e. 2%. Results show that, the confidence count lies in the range of 21 to 100%.

By using these rules we can predict the bug-fix time for a newly coming bug. We also observe that our approach for bug-fix time prediction will be helpful in bug triaging by assigning a bug to the most potential and experienced assignee that will solve the bug in minimum time period. Prediction of bug-fix time will help the managers in measuring software quality and in software development process. From results, we can observe the number of association rules having high confidence and support with higher severity and priority as antecedents and short bug-fix time as consequent. A large number for such rules show that more important bugs are fixed with out any delay. This information is useful in determining software quality during software evolution process. Further, for bugs with long predicted fix time we need to pay more attention to the related source files to make sure that the files remain stable during fixing process. This will again help in determining software quality. We will extend our work with other association mining algorithms to empirically validate the results.

References

[1] S. Kim and J. E. Whitehead, "How long did it take to fix bugs?," Int. Workshop Mining Software Repositories. New York, NY, USA, ACM, pp. 173–174, 2006

[2] P. Hooimeijer and W. Weimer, "Modeling bug report quality," ASE 2007.

[3] P. Anbalagan and M. Vouk, "On predicting the time taken to correct bug reports in open source projects," Int. Conf. Software Management (Edmonton, AB). IEEE, pp. 523-526, September 20-26, 2009, DOI= http://ieeexplore.ieee.org/10.1109/ICSM.2009.5306337.

[4] P. Bhattacharya and I. Neamtiu, "Bug-fix Time Prediction Models: Can We Do Better?," 8th Working Conf. Mining Software Repositories (New York, NY, USA). ACM, pp. 207-210, 2012, DOI= http://dl.acm.org/10.1145/1985441.1985472.

[5] E. Giger, M. Pinzger and H. Gall, "Predicting the fix time of bugs," Int. Workshop Recommendation Systems on Software Enginnering (New York, NY, USA), ACM, pp. 52-56, 2010.

[6] M. Sharma, M. Kumari and V.B. Singh, "Understanding the Meaning of Bug Attributes and Prediction Models," 5th IBM Collaborative Academia Research Exchange Workshop, I-CARE, Article No. 15, ACM, 2013.

[7] R. Agrawal, T. Imielinski and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," SIGMOD Conf. Management of Data, ACM, May 1993.

[8] Q. Song, M. Shepperd, M. Cartwright and C. Mair, "Software defect association mining and defect correction effort prediction," IEEE Transactions on Software Engineering, Vol. 32(2) pp. 69 – 82, 2006.

[9] K. Ali, S. Manganaris and R. Srikant, "Partial Classification Using Association Rules," Int. Conf. Knowledge Discovery and Data Mining., pp. 115-118, 1997

[10] G. Dong, X. Zhang, L. Wong, and J. Li, "CAEP: Classification by Aggregating Emerging Patterns," Int. Conf. Discovery Science, pp. 30-42, 1999.

[11] B. Liu, W. Hsu, and Y. Ma, "Integrating Classification and Association Rule Mining," Int. Conf. Knowledge Discovery and Data Mining, pp. 80-86, 1998.

[12] R. She, F. Chen, K. Wang, M. Ester, J.L. Gardy and F.L. Brinkman, "Frequent-Subsequence-Based Prediction of Outer Membrane Proteins," ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, 2003.

[13] K. Wang, S.Q. Zhou and S.C. Liew, "Building Hierarchical Classifiers Using Class Proximity," Int. Conf. Very Large Data Bases, pp. 363-374, 1999.

[14] K. Wang, S. Zhou and Y. He, "Growing Decision Tree on Support-Less Association Rules." Int. Conf. Knowledge Discovery and Data Mining, 2000.

[15] Q. Yang, H.H. Zhang and T. Li, "Mining Web Logs for Prediction Models in WWW Caching and Prefetching," ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, 2001.

[16] X. Yin and J. Han, "CPAR: Classification Based on Predictive Association Rules," SIAM Int. Conf. Data Mining, 2003.

[17] A.T.T. Ying, C.G. Murphy, R. Ng and M.C. Chu-Carroll,. "Predicting Source Code Changes by Mining Revision History," Int. Workshop Mining Software Repositories, 2004.

[18] T. Zimmermann, P. Weigerber, S. Diehl and A. Zeller, "Mining Version Histories to Guide Software Changes," Int. Conf. Software Engineering, 2004.

[19] I. Mierswa, M. Wurst, R. Klinkenberg, M. Scholz and T. Euler, "YALE: Rapid Prototyping for Complex Data Mining Tasks," ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD-06), 2006 (http://www.rapid-i.com).

[20] M. Porter, "An algorithm for suffix stripping," Program.Vol. 14 (3), pp. 130–137, 2008.

[21] "http://in.mathworks.com/.../3016-armada-data-mining-tool-version-1-4", 2015, URL: http://in.mathworks.com/[accessed:2015-07-24].

[22] A. Mockus, R. T. Fielding and J. D. Herbsleb, "Two case studies of open source software development: Apache and Mozilla," ACM Trans. on Software Eng. Vol. (11)3, 2002.

[23] M. Plassea, N. Nianga, G. Saportaa, A. Villeminotb and L. Leblondb, "Combined use of association rules mining and clustering methods to find relevant links between binary rare attributes in a large data set," Computational Statistics & Data Analysis, ELSEVIER, 2007.

[24] M. Sharma, M. Kumari and V.B. Singh, "Bug Assignee Prediction Using Association Rule Mining," ICCSA 2015, Part IV, LNCS 9158, pp.444–457, 2015.

[25] F. Zhang , F. Khomh , Y. Zou and A. E. Hassan, "An Empirical Study on Factors Impacting Bug Fixing Time," 19th Working Conference on Reverse Engineering (WCRE), pp. 225-234, 15-18 Oct 2012.

[26] W. AbdelMoez, M. Kholief and F. M. Elsalmy, "Improving bug fix-time prediction model by filtering out outliers," International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE), 2013 , pp.359-364, 9-11 May 2013.