

## Reference List

- [1] Marco Brambilla, Jordi Cabot, Manuel Wimmer. (2012) Model-Driven Software Engineering in Practise. Morgan & Claypool Publishers.
- [2] Robert France, Bernard Rumpe, Model Driven Development of Complex Software: A Research RoadMap. Future of Software Engineering 2007.
- [3] A brief introduction to model-driven engineering. May 24th 2020. [http://www.scielo.org.co/scielo.php?script=sci\\_arttext&pid=S0123-921X2014000200011](http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0123-921X2014000200011).
- [4] Manifesto for Agile Software Development. Retrieved May 11<sup>th</sup> 2020. <http://agileManifesto.org>
- [5] Qing He, Beijun Shen, Yuting Chen, Software Defect Prediction Using Semi – Supervised Learning with Change Burst Information. IEEE Computer Society 2016.
- [6] Wenjing Han, Chung-Horng Lung, Samuel A. Ajila, Empirical Investigation of Code and Process Metrics for Defect Prediction, IEEE Computer Society 2016.
- [7] Kanika Chandra, Gagan Kapoor, Rashi Kohli, Archana Gupta, IMPROVING SOFTWARE QUALITY USING MACHINE LEARNING, 1st International Conference on Innovation and Challenges in Cyber Security 2016.
- [8] Ankita Bansal, Ruchika Malhotra, Kimaya Raje, Analyzing and Assessing the Security-Related Defects, 1st International Conference on Innovation and Challenges in Cyber Security. 2016.
- [9] Puntitra Sawadpong, Toward a Defect Prediction Model of Exception Handling Method Call Structures, ACM SE '14, March 28 - 29 2014.
- [10] Feng Zhang, Ahmed E.Hassan, Shane McIntosh, Ying Zou, The use of Summation to Aggregate Software Metrics Hinders the Performance of Defect Prediction Models. IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, 2016.
- [11] Xiaobing Sun, Xiangyue Liu, Bin Li, Yucong Duan, Hui Yang, Jianjun Hu, Exploring Topic Models in Software Engineering Data Analysis: A Survey, IEEE 2016.
- [12] Chun Shan, Hongjin Zhu, Changzhen Hu, Jin Cui, Jinefeng Xue, Software Defect Prediction Model Based on Improved LLE-SVM. 4th International Conference on Computer Science and Network Technology, 2015.

- [13] Komukai-Toshiba-cho, Saiwai-ku, Kawasaki, Superposed Naïve Bayes for Accurate and Interpretable Prediction, IEEE 14th International Conference on Machine Learning and Applications, 2015.
- [14] Carol Woody, Robert Ellison, William Nichols, Predicting Cybersecurity Using Quality Data, IEEE, 2015.
- [15] Wutthichai Chansuwath, Twittie Senivongse, A Model-Driven Development of Web Applications Using AngularJS Framework. IEEE, 2016.
- [16] OMG, UML Profile. [Online]. Available: <http://www.omg.org/mda/specs.htm>. Last Accessed: 30 Mar 2016.
- [17] Eric Freeman, Elisabeth Freeman, Kathy Sierra and Bert Bates. Head First Design Patterns, 2004.
- [18] MxGraph. Retrieved May 11<sup>th</sup>, 2020. <https://github.com/jgraph/mxgraph>
- [19] Asadullah Shaikh, Overview of Slicing and Feedback Techniques for Efficient Verification of UML/OCL Class Diagrams.
- [20] J.cabot, R Clariso, and D.Riera. Verification of uml/ocl class diagrams using constraint programming. In ASE 2007, pages 547-548. ACM 2007.
- [21] The ECLiPSe Constraint Programming System. <http://www.ecipseclp.org/>, Mar 2007, version 5.10. Last checked on January 11, 2018.
- [22] Object Management Group (OMG). Object Constraint Language. Retrieved May 11th, 2020. <http://www.omg.org/spec/ocl>.
- [23] Object management Group (OMG). Unified Modelling Lanuguage homepage 2018. Retrieved May 11<sup>th</sup> 2020. <http://www.omg.org>.
- [24] Shamsul Huda, Sultan Alyahya, MD Mohsin Ali, Shafiq Ahmed, Jemal Abawajy, Hmood Al Dossari, John Yearwood. A Framework for Software Defect Prediction and Metric selection.
- [25] L. Pelayo and S.Dick, “Applying novel resampling strategies to software defect prediction” in Proc. Annu. Meeting North Amer. Fuzzy Inf. Process. Soc (NAFIPS), Jan. 2007, pp. 69-72.
- [26] H.B Yadav and D. K Yadav, “A Fussy logic based approach for phase wise software defects prediction using software metrics.” Inf. Softw. Tech-nol., vol 63, pp. 44-57. Jul. 2015.

- [27] T.Bages, “An essay towards solving a problem in the doctrine of changes.” Philos, Trans, Roy. Soc. London, no 53, pp. 370-418, 1763. [Online]. Available: <http://www.sta.ucls.edu/history/essay.pdf>
- [28] N. Cristianini and J.Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods. New York, NY, USA: Cambridge Univ. Press 2000
- [29] J.R. Quinlan, “Induction of decision trees,” Mach. Learn., vol. 1, no0 1, pp. 81-106, Mar. 1986.
- [30] B. Krose and P.V.D Smagt, An Introduction toNeural Networks. Amsterdam, The Netherlands: Univ. Amsterdam, 1993.
- [31] J. A. Rodger, “Toward reducing failure risk in an integrated vehicle health maintenance system: A fuzzy multi-sensor data fusion Kalman filter approach for IVHMS,” Expert Syst. Appl., vol. 39, no. 10, pp. 9821-9836, Aug. 2012.
- [32] F. Aparisi and J.Sanz, “Interpreting the out-of-control signals of multi-variate control charts employing neural networks,”. Int. J. Comput., Electl, Autom., Control Inf. Eng., vol. 4, no. 1, pp. 24-28, 2010.
- [33] A. A Asad and I. Alsmadi, “Evaluating the impact of software metrices on defects prediction. Part 2,” Comput. Sci. J. Moldova, vol. 22, no. 1, pp. 127-144, 2014.
- [34] R. Malhotra, “A systematic review of machine learning techniques for software fault prediction,” Appl. Soft Comput., vol. 27, pp. 504-518, Feb. 2015.
- [35] H. Zhang, “An investigation of the relationships between lines of code and the defects.” In Proc. IEEE Int. Conf. Softw. Maintenance (ICSM), Sep. 2009, pp. 274-283.
- [36] L. Yu and A. Mishra, “Experience predicting fault-prone software modules using complexity metrices” Quality Technolol. Quantitative Manage., vol. 9, no. 4, pp. 421-433, May 2012. [http://web.it.nctu.edu.tw/~qtqm/qtqmpaers/2012V9N4/2012V9N4/2012\\_F7.pdf](http://web.it.nctu.edu.tw/~qtqm/qtqmpaers/2012V9N4/2012V9N4/2012_F7.pdf)
- [37] Feng Zhang and Audris Mockus and Iman Keivanloo and Ying Zou, Towards Building a Universal Defect Prediction Model, 2014.
- [38] J. Nam, S. J. Pan, and S. Kim. Transfer defect learning. In Proceedings of the 35th International Conference on Software Engineering, pages 382–391, 2013
- [39] F. Zhang, A. Mockus, Y. Zou, F. Khomh, and A. E.Hassan. How does context affect the distribution of software maintainability metrics? In Proceedings of the 29th IEEE International Conference on Software Maintainability, pages 350 – 359, 2013.

- [40] N. Ohlsson and H. Alberg, “Predicting fault-prone software modules in telephone switches,” *IEEE Trans. Softw. Eng.*, vol. 22, no. 12, pp. 886–894, Dec. 1996
- [41] R. Malhotra, “A systematic review of machine learning techniques for software fault prediction,” *Appl. Soft Comput.*, vol. 27, pp. 504–518, Feb. 2015.
- [42] S. R. Chidamber and C. F. Kemerer, “A metrics suite for object oriented design,” *IEEE Trans. Softw. Eng.*, vol. 20, no. 6, pp. 476–493, Jun. 1994.
- [43] T. T. Nguyen, T. N. Nguyen, and T. M. Phuong. Topic-based defect prediction (nier track). In *Proceedings of the 33rd International Conference on Software Engineering, ICSE ’11*, pages 932–935, New York, NY, USA, 2011. ACM.
- [44] D. Andrzejewski, A. Mulhern, B. Liblit, and X. Zhu. Statistical debugging using latent topic models. In *Proceedings of the 18th European Conference on Machine Learning*, pages 6–17, 2007
- [45] T.-H. Chen, S. W. Thomas, M. Nagappan, and A. E. Hassan. Explaining software defects using topic models. In *Proceedings of the 9th IEEE Working Conference on Mining Software Repositories*, pages 189–198, 2012.
- [46] W. Hu and K. Wong. Using citation influence to predict software defects. In *Proceedings of the 10th Working Conference on Mining Software Repositories, MSR ’13*, pages 419–428, Piscataway, NJ, USA, 2013. IEEE Press.
- [47] P. Sawadpong, E. B. Allen, and B. J. Williams. Exception handling defects: An empirical study. In *IEEE 14th International Symposium on High-Assurance Systems Engineering*, pages 90–97, 2012
- [48] B. G. Ryder. Constructing the call graph of a program. *IEEE Transactions on Software Engineering*, SE-5(3):216–226, May 1979.
- [49] C. Marinescu. Are the classes that use exceptions defect prone? In *Proceedings of the 12th International Workshop on Principles of Software Evolution and the 7th annual ERCIM Workshop on Software Evolution*, pages 56–60. ACM, 2011.
- [50] L. Briand, S. Morasca, and V. Basili. Property-based software engineering measurement. *IEEE Transactions on Software Engineering*, 22(1):68–86, 1996
- [51] T. M. Khoshgoftaar and E. B. Allen. Logistic regression modeling of software quality. *International Journal of Reliability, Quality & Safety Engineering*, 6(4):303–317, 1999
- [52] Yucong Duan, Qiang Duan, Ruisheng Shi, Honghao Gao, Toward an Automated View Abstraction for Distributed Model-Driven Service Development. *IEEE International Conference on Services Computing*, 2016