# Investigating the Use of Evaluation Measures in the Defect Prediction Literature

Jan 28 2022

Below we provide the list of defect prediction papers examined in our study. We report the list of evaluation measures used for binary classification (Measures); the year (Year) and the venue (Venue) a study was published in; whether rationale is given for using certain measures (Justified); whether it is discussed to what extend different evaluation measures can yield different results (Ack); and the proposed possible mitigation (Mitigation).

| Study | Measures | Year | Venue | Justified | Ack | Mitigation |
|---|---|---|---|---|---|---|
| [67] | Accuracy, Precision, Recall, AUC | 2010 | ICSE | | | |
| [12] | Accuracy, Specificity, Recall | 2010 | ICSE | | | |
| [26] | Accuracy, Recall, Precision, F-measure | 2010 | EMSE | | | |
| [19] | Accuracy, Precision, Recall, F-measure, AUC | 2010 | JSS | | | Using multiple measures |
| [5] | Accuracy, Precision, Recall, F-measure, AUC | 2010 | JSS | | | Using multiple measures |
| [33] | Precision, Accuracy | 2010 | IST | | | |
| [111] | AUC | 2010 | JSS | | | |
| [93] | Recall, FPR, Balance, AUC | 2010 | IST | | | Choosing measures is based on the goal investigated |
| [86] | Balance, AUC | 2011 | TSE | | | |
| [42] | Precision, Recall, F-measures | 2011 | ICSE | | | |
| [48] | F-measure | 2011 | EMSE | | | |
| [52] | F-measure | 2011 | FSE | | | |
| [78] | Precision, Recall, F-measure | 2011 | ESEM | | | |
| [77] | AUC | 2011 | ASE | | | Using a threshold independent measure |
| [15] | Precision, Recall, AUC | 2011 | IST | | | Using a threshold independent measure |
| [18] | AUC | 2012 | EMSE | | | |
| [73] | F-measure | 2012 | ICSE | | | |
| [21] | AUC | 2012 | EMSE | | | |
| [58] | AUC, FPR | 2012 | ASE | | | |
| [82] | Recall, FPR, Accuracy, Precision | 2012 | ASE | | | |
| [16] | Precision, Recall, AUC | 2012 | IST | | | Using a threshold independent measure |
| [17] | Precision, Recall, AUC | 2012 | JSS | | | Using a threshold independent measure |
| [59] | Recall, FPR, F-measure, AUC | 2012 | IST | | | Using a threshold independent measure |
| [39] | Accuracy, Precision, Recall, F-measure, AUC | 2013 | TSE | | | Using multiple measures |
| [74] | Recall, FPR, G-measure | 2013 | TSE | | | Using G-measure when the data is imbalanced |
| [80] | F-measure, AUC | 2013 | ICSE | | | |
| [20] | AUC, H-measure | 2013 | TSE | | | |
| [64] | F-measure | 2013 | ICSE | | | |
| [28] | Recall, FPR, G-measure | 2013 | ESEM | | | |
| [34] | Precision, Recall, F-measure | 2013 | ASE | | | |
| [13] | Precision, Recall, F-measure | 2013 | ICST | | | |
| [94] | FPR, Recall, Balance | 2013 | IST | | | |
| [37] | FPR, Recall, F-measure | 2014 | ICSE | | | |
| [69] | AUC | 2014 | EMSE | | | |
| [35] | Recall rate, FPR, F-measure | 2015 | FSE | | | |
| [75] | G-measure | 2015 | ICSE | | | Using G-measure when the data is imbalanced |
| [87] | Precision, Recall, F-measure | 2015 | ICSE | | | |
| [41] | Precision, Recall, F-measure | 2015 | FSE | | | |
| [25] | AUC | 2015 | ICSE | | | |
| [88] | Precision, Recall, F-measure | 2015 | ICSE | | | |
| [27] | Precision, Recall, F-measure | 2015 | IST | | | |
| [14] | Recall, FPR, G-measure, MCC | 2015 | IST | | | |
| [51] | G-mean | 2015 | IST | | | Using an unbiased measure |
| [89] | AUC | 2016 | ICSE | | | Future work should investigate the use of other measures |
| [95] | Precision, Recall, F-measure | 2016 | ICSE | | | |
| [109] | AUC | 2016 | ICSE | | | |
| [46] | AUC, Recall, Precision | 2016 | ICSE | | | |
| [53] | F-measure | 2016 | TSE | | | |
| [96] | F-measure | 2016 | TSE | | | |
| [38] | AUC, Precision, Recall, F-measure | 2016 | EMSE | | | Using a threshold independent measure |
| [31] | Precision, Recall, F-measure | 2016 | EMSE | | | |
| [108] | Precision, Recall, FPR, F-measure, G-measure, MCC, AUC | 2016 | EMSE | | | |
| [81] | Recall, FPR, AUC, H-measure | 2016 | EMSE | | | |
| [84] | AUC, Recall, FPR, F-measure, Balance | 2016 | ESEM | | | |
| [76] | MCC | 2016 | ESEM | | | Using an unbiased measure which describes entire confusion matrix |
| [10] | MCC | 2016 | ISSTA | | | Using an unbiased measure which describes entire confusion matrix |
| [68] | Precision, Recall, F-measure | 2016 | JSS | | | |
| [24] | Recall, Precision, FPR, F-measure | 2016 | IST | | | Choosing measures based on the goal investigated |
| [90] | Precision, Recall, AUC | 2017 | TSE | | | Using both threshold dependent and independent measures |
| [36] | Skewed F-measure, Recall, FPR, AUC | 2017 | TSE | | | Using multiple measures |
| [107] | Precision, Recall, F-measure, Balance, AUC | 2017 | EMSE | | | |
| [29] | Recall, Precision, F-measure, Accuracy, AUC | 2017 | EMSE | | | Using other measures to mitigate the bias coming from using error |
| [106] | AUC | 2017 | TSE | | | |
| [8] | AUC, Recall, FPR, G-mean | 2017 | ESEM | | | Other measures have been shown to be biased |
| [57] | Precision, Recall, F-measure | 2017 | ESEM | | | |
| [9] | AUC | 2017 | ESEM | | | |
| [104] | AUC | 2017 | JSS | | | Using AUC as its variance was the lowest among the metrics examined |
| [62] | Accuracy, Balanced Accuracy | 2017 | JSS | | | |
| [101] | F-measure | 2017 | IST | | | |
| [70] | AUC, G-mean | 2017 | IST | | | |
| [30] | AUC, Recall,Precision, Accuracy, F-measure, G-measure, MCC | 2018 | TSE | | | Using multiple measures |
| [103] | F-measure | 2018 | ICSE | | | |
| [60] | AUC, Brier | 2018 | TSE | | | |
| [22] | Recall, FPR, Precision, F-measure | 2018 | ICSE | | | |
| [65] | Precision, Recall, F-measure | 2018 | ICSE | | | |
| [43] | F-measure | 2018 | ICSE | | | |
| [98] | F-measure, G-mean, Balance | 2018 | ICSE | | | |
| [63] | AUC | 2018 | TSE | | | |
| [3] | Recall, Precision, FPR, AUC | 2018 | ICSE | | | Using more than one measures (but not on all as there are too many), by selecting those that are widely used in the literature |
| [7] | Recall, FPR | 2018 | TSE | | | Other measures have been shown to be biased |
| [79] | AUC, F-measure | 2018 | ASE | | | |
| [47] | Precision, Recall, F-measure, AUC | 2018 | ASE | | | |
| [32] | Precision, Recall, F-measure, G-mean | 2018 | IST | | | |
| [55] | Balance, G-measure, F-measure | 2018 | IST | | | Future work should investigate the use of other measures |
| [50] | Accuracy, F-measure | 2018 | JSS | | | |
| [92] | F-measure, AUC, MCC | 2018 | IST | | | Using MCC when date is imbalanced |
| [85] | MCC | 2019 | TSE | | | Other measures have been shown to be biased |
| [61] | AUC | 2019 | EMSE | | | |
| [49] | G-measure | 2019 | TSE | | | |
| [11] | Recall, G-mean | 2019 | ICSE | | | |
| [105] | Precision, Recall, F-measure, AUC | 2019 | TSE | | | |
| [2] | Balance | 2019 | TSE | | | Using the most prominent measures in the literature |
| [102] | AUC, Precision, Recall, F-measure | 2019 | ICSE | | | Using a threshold independent measure |
| [54] | G-measure, AUC | 2019 | TSE | | | Using G-measure when the data is imbalanced |
| [6] | AUC, Balance, G-mean | 2019 | EMSE | | | |
| [91] | Precision, Recall, F-measure, FPR, MCC, AUC, G-mean, G-measure, Balance | 2019 | TSE | | | Using multiple measures |
| [23] | AUC, F-measure, G-mean | 2019 | TSE | | | |
| [44] | AUC | 2019 | EMSE | | | Future work should investigate the use of other measures |
| [56] | F-measure | 2019 | IST | | | |
| [66] | AUC | 2019 | JSS | | | Using a threshold independent measure |
| [83] | Precision, Recall, F-measure, mcc | 2019 | IST | | | |
| [71] | Precision, Recall, F-measure, AUC | 2019 | JSS | | | Using a threshold independent measure |
| [110] | AUC, Precision, Recall, F-measure | 2019 | IST | | | Using a threshold independent measure among multiple measures |
| [99] | F-measure, AUC | 2019 | JSS | | | Using a threshold independent measure |
| [100] | F-measure, G-measure, MCC, AUC | 2019 | IST | | | |
| [97] | F-measure, MCC, AUC | 2019 | JSS | | | Future work should investigate the use of other measures |
| [40] | F-measure, AUC | 2020 | TOSEM | | | Using the average of two measures as the final accuracy measure |
| [4] | AUC, F-measure, G-measure, MCC | 2020 | EMSE | | | |
| [45] | AUC, MCC | 2020 | EMSE | | | Other measures have been shown to be biased |
| [72] | Precision, Recall, F-measure, AUC | 2020 | JSS | | | Using a threshold independent measure |
| [1] | F-measure, Precision, Recall | 2020 | JSS | | | |

## References


[1] Petar Afric, Lucija Sikic, Adrian Satja Kurdija, and Marin Silic. 2020. REPD: Source code defect prediction as anomaly detection. *JSS* (2020), 110641.

[2] A. Agrawal, W. Fu, D. Chen, X. Shen, and T. Menzies. 2019. How to "DODGE" Complex Software Analytics. *IEEE TSE* (2019).

[3] A. Agrawal and T. Menzies. 2018. Is "better data" better than "better data miners"?: On the benefits of tuning SMOTE for defect prediction. *Procs. of ICSE* (2018), 1050–1061.

[4] S. Amasaki. 2020. Cross-version defect prediction: use historical data, cross-project data, or both? *ESE* 25, 2 (2020), 1573–1595.

[5] Erik Arisholm, Lionel C. Briand, and Eivind B. Johannessen. 2010. A systematic and comprehensive investigation of methods to build and evaluate fault prediction models. *JSS* 83 (1 2010), 2–17. Issue 1.

[6] K.E. Bennin, J.W. Keung, and A. Monden. 2019. On the relative value of data resampling approaches for software defect prediction. *ESE* 24, 2 (2019), 602–636.

[7] K.E. Bennin, J. Keung, P. Phannachitta, A. Monden, and S. Mensah. 2018. MAHAKIL: Diversity Based Oversampling Approach to Alleviate the Class Imbalance Issue in Software Defect Prediction. *IEEE TSE* 44, 6 (2018), 534–550.

[8] Kwabena Ebo Bennin, Jacky Keung, Akito Monden, Passakorn Phannachitta, and Solomon Mensah. 2017. The significant effects of data sampling approaches on software defect prioritization and classification. In *Procs. of ESEM*. 364–373.

[9] Yi Bin, Kai Zhou, Hongmin Lu, Yuming Zhou, and Baowen Xu. 2017. Training data selection for cross-project defection prediction: which approach is better?. In *Procs. of ESEM*. 354–363.

[10] David Bowes, Tracy Hall, Mark Harman, Yue Jia, Federica Sarro, and Fan Wu. 2016. Mutation-aware fault prediction. In *Procs. ISSTA*. 330–341.

[11] G.G. Cabral, L.L. Minku, E. Shihab, and S. Mujahid. 2019. Class Imbalance Evolution and Verification Latency in Just-in-Time Software Defect Prediction. *Procs. of ICSE* (2019), 666–676.

[12] A.E. Camargo Cruz. 2010. Exploratory study of a UML metric for fault prediction. *Procs. of ICSE* 2 (2010), 361–364.

[13] Gabriella Carrozza, Domenico Cotroneo, Roberto Natella, Roberto Pietrantuono, and Stefano Russo. 2013. Analysis and prediction of mandelbugs in an industrial software system. In *Procs. of ICST*. 262–271.

[14] Lin Chen, Bin Fang, Zhaowei Shang, and Yuanyan Tang. 2015. Negative samples reduction in cross-company software defects prediction. *IST* 62 (2015), 67–77. Issue 1.

[15] Jehad Al Dallal. 2011. Improving the applicability of object-oriented class cohesion metrics. *IST* 53 (9 2011), 914–928. Issue 9.

[16] Jehad Al Dallal. 2012. Fault prediction and the discriminative powers of connectivity-based object-oriented class cohesion metrics. *IST* 54 (4 2012), 396–416. Issue 4.

[17] Jehad Al Dallal. 2012. The impact of accounting for special methods in the measurement of object-oriented class cohesion on refactoring and fault prediction activities. *JSS* 85 (2012), 1042–1057.

[18] M. D'Ambros, M. Lanza, and R. Robbes. 2012. Evaluating defect prediction approaches: A benchmark and an extensive comparison. *ESE* 17, 4-5 (2012), 531–577.

[19] André B. de Carvalho, Aurora Pozo, and Silvia Regina Vergilio. 2010. A symbolic fault-prediction model based on multiobjective particle swarm optimization. *JSS* 83 (5 2010), 868–882. Issue 5.

[20] K. Dejaeger, T. Verbraken, and B. Baesens. 2013. Toward comprehensible software fault prediction models using bayesian network classifiers. *IEEE TSE* 39, 2 (2013), 237–257.

[21] J. Ekanayake, J. Tappolet, H.C. Gall, and A. Bernstein. 2012. Time variance and defect prediction in software projects: Towards an exploitation of periods of stability and change as well as a notion of concept drift in software projects. *ESE* 17, 4-5 (2012), 348–389.

[22] B. Eken. 2018. Assessing personalized software defect predictors. *Procs. of ICSE* (2018), 488–491.

[23] Y. Fan, X. Xia, D. Alencar da Costa, D. Lo, A.E. Hassan, and S. Li. 2019. The Impact of Changes Mislabeled by SZZ on Just-in-Time Defect Prediction. *IEEE TSE* (2019).

[24] Wei Fu, Tim Menzies, and Xipeng Shen. 2016. Tuning for software analytics: Is it really necessary? *IST* 76 (8 2016), 135–146.

[25] B. Ghotra, S. McIntosh, and A.E. Hassan. 2015. Revisiting the impact of classification techniques on the performance of defect prediction models. *Procs. of ICSE* (2015), 789–800.

[26] H. Hata, O. Mizuno, and T. Kikuno. 2010. Fault-prone module detection using large-scale text features based on spam filtering. *ESE* 15, 2 (2010), 147–165.

[27] Peng He, Bing Li, Xiao Liu, Jun Chen, and Yutao Ma. 2015. An empirical study on software defect prediction with a simplified metric set. *IST* 59 (3 2015), 170–190.

[28] Zhimin He, Fayola Peters, Tim Menzies, and Ye Yang. 2013. Learning from open-source projects: An empirical study on defect prediction. In *Procs. of ESEM*. 45–54.

[29] S. Herbold, A. Trautsch, and J. Grabowski. 2017. Global vs. local models for cross-project defect prediction: A replication study. *ESE* 22, 4 (2017), 1866–1902.

[30] S. Herbold, A. Trautsch, and J. Grabowski. 2018. A Comparative Study to Benchmark Cross-Project Defect Prediction Approaches. *IEEE TSE* 44, 9 (2018), 811–833.

[31] K. Herzig, S. Just, and A. Zeller. 2016. The impact of tangled code changes on defect prediction models. *ESE* 21, 2 (2016), 303–336.

[32] Seyedrebvar Hosseini, Burak Turhan, and Mika Mäntylä. 2018. A benchmark study on the effectiveness of search-based data selection and feature selection for cross project defect prediction. *IST* 95 (3 2018), 296–312.

[33] Timea Illes-Seifert and Barbara Paech. 2010. Exploring the relationship of a file's history and its fault-proneness: An empirical method and its application to open source programs. *IST* 52 (5 2010), 539–558. Issue 5.

[34] Tian Jiang, Lin Tan, and Sunghun Kim. 2013. Personalized defect prediction. In *Procs. of ASE*. 279–289.

[35] X. Jing, F. Wu, X. Dong, F. Qi, and B. Xu. 2015. Heterogeneous cross-company defect prediction by unified metric representation and CCA-based transfer learning. *Procs. of ESEC/FSE* (2015), 496–507.

[36] X.-Y. Jing, F. Wu, X. Dong, and B. Xu. 2017. An Improved SDA Based Defect Prediction Framework for Both Within-Project and Cross-Project Class-Imbalance Problems. *IEEE TSE* 43, 4 (2017), 321–339.

[37] X.-Y. Jing, S. Ying, Z.-W. Zhang, S.-S. Wu, and J. Liu. 2014. Dictionary learning based software defect prediction. *Procs. of ICSE* (2014), 414–423.

[38] Y. Kamei, T. Fukushima, S. McIntosh, K. Yamashita, N. Ubayashi, and A.E. Hassan. 2016. Studying just-in-time defect prediction using cross-project models. *ESE* 21, 5 (2016), 2072–2106.

[39] Y. Kamei, E. Shihab, B. Adams, A.E. Hassan, A. Mockus, A. Sinha, and N. Ubayashi. 2013. A large-scale empirical study of just-in-time quality assurance. *IEEE TSE* 39, 6 (2013), 757–773.

[40] R. Kapur and B. Sodhi. 2020. A Defect Estimator for Source Code. *ACM TOSEM* 29, 2 (2020).

[41] M. Kim, J. Nam, J. Yeon, S. Choi, and S. Kim. 2015. REMI: Defect prediction for efficient API testing. *Procs. of ESEC/FSE* (2015), 990–993.

[42] S. Kim, H. Zhang, R. Wu, and L. Gong. 2011. Dealing with noise in defect prediction. *Procs. of ICSE* (2011), 481–490.

[43] P. Koch, K. Schekotihin, D. Jannach, B. Hofer, F. Wotawa, and T. Schmitz. 2018. Combining spreadsheet smells for improved fault prediction. *Procs. of ICSE* (2018), 25–28.

[44] M. Kondo, C.-P. Bezemer, Y. Kamei, A.E. Hassan, and O. Mizuno. 2019. The impact of feature reduction techniques on defect prediction models. *ESE* 24, 4 (2019), 1925–1963.

[45] M. Kondo, D.M. German, O. Mizuno, and E.-H. Choi. 2020. The impact of context metrics on just-in-time defect prediction. *ESE* 25, 1 (2020), 890–939.

[46] Y. Koroglu, A. Sen, D. Kutluay, A. Bayraktar, Y. Tosun, M. Cinar, and H. Kaya. 2016. Defect prediction on a legacy industrial software: A case study on software with few defects. *Procs. of ICSE* (2016), 14–20.

[47] Vladimir Kovalenko, Fabio Palomba, and Alberto Bacchelli. [n.d.]. Mining file histories: Should we consider branches?. In *Procs. of ASE, pages=202–213, year=2018.*

[48] S. Kpodjedo, F. Ricca, P. Galinier, Y.-G. Guéhéneuc, and G. Antoniol. 2011. Design evolution metrics for defect prediction in object oriented systems. *ESE* 16, 1 (2011), 141–175.

[49] R. Krishna and T. Menzies. 2019. Bellwethers: A Baseline Method for Transfer Learning. *IEEE TSE* 45, 11 (2019), 1081–1105.

[50] Lov Kumar, Sai Krishna Sripada, Ashish Sureka, and Santanu Ku Rath. 2018. Effective fault prediction model developed using Least Square Support Vector Machine (LSSVM). *JSS* 137 (3 2018), 686–712.

[51] Issam H Laradji, Mohammad Alshayeb, and Lahouari Ghouti. 2015. Software defect prediction using ensemble learning on selected features. *IST* 58 (2015), 388–402.

[52] T. Lee, D.G. Han, S. Kim, and H.P. In. 2011. Micro interaction metrics for defect prediction. *Procs. of FSE* (2011), 311–321.

[53] T. Lee, J. Nam, D. Han, S. Kim, and H. Peter In. 2016. Developer Micro Interaction Metrics for Software Defect Prediction. *IEEE TSE* 42, 11 (2016), 1015–1035.

[54] Z. Li, X.-Y. Jing, X. Zhu, H. Zhang, B. Xu, and S. Ying. 2019. On the Multiple Sources and Privacy Preservation Issues for Heterogeneous Defect Prediction. *IEEE TSE* 45, 4 (2019), 391–411.

[55] Nachai Limsettho, Kwabena Ebo Bennin, Jacky W. Keung, Hideaki Hata, and Kenichi Matsumoto. 2018. Cross project defect prediction using class distribution estimation and oversampling. *IST* 100 (8 2018), 87–102.

[56] Chao Liu, Dan Yang, Xin Xia, Meng Yan, and Xiaohong Zhang. 2019. A two-phase transfer learning model for cross-project defect prediction. *IST* 107 (3 2019), 125–136.

[57] Jinping Liu, Yuming Zhou, Yibiao Yang, Hongmin Lu, and Baowen Xu. 2017. Code churn: A neglected metric in effort-aware just-in-time defect prediction. In *Procs. of ESEM*. 11–19.

[58] Huihua Lu, Bojan Cukic, and Mark Culp. 2012. Software defect prediction using semi-supervised learning with dimension reduction. In *Procs. of ASE*. 314–317.

[59] Ying Ma, Guangchun Luo, Xue Zeng, and Aiguo Chen. 2012. Transfer learning for cross-company software defect prediction. *IST* 54 (3 2012), 248–256. Issue 3.

[60] S. McIntosh and Y. Kamei. 2018. Are Fix-Inducing Changes a Moving Target? A Longitudinal Case Study of Just-In-Time Defect Prediction. *IEEE TSE* 44, 5 (2018), 412–428.

[61] T. Mori and N. Uchihira. 2019. Balancing the trade-off between accuracy and interpretability in software defect prediction. *ESE* 24, 2 (2019), 779–825.

[62] Rebecca Moussa and Danielle Azar. 2017. A PSO-GA approach targeting fault-prone software modules. *JSS* 132 (10 2017), 41–49.

[63] J. Nam, W. Fu, S. Kim, T. Menzies, and L. Tan. 2018. Heterogeneous Defect Prediction. *IEEE TSE* 44, 9 (2018), 874–896.

[64] J. Nam, S.J. Pan, and S. Kim. 2013. Transfer defect learning. *Procs. of ICSE* (2013), 382–391.

[65] M. Nayrolles and A. Hamou-Lhadj. 2018. CLEVER: Combining code metrics with clone detection for just-in-time fault prevention and resolution in large industrial projects. *Procs. of ICSE* (2018), 153–164.

[66] Chao Ni, Xiang Chen, Fangfang Wu, Yuxiang Shen, and Qing Gu. 2019. An empirical study on pareto based multi-objective feature selection for software defect prediction. *JSS* 152 (6 2019), 215–238.

[67] A. Nugroho, M.R.V. Chaudron, and E. Arisholm. 2010. Assessing UML design metrics for predicting fault-prone classes in a Java system. *Procs. of ICSE* (2010), 21–30.

[68] Ahmet Okutan and Olcay Taner Yildiz. 2016. A novel kernel to predict software defectiveness. *JSS* 119 (9 2016), 109–121.

[69] A. Okutan and O.T. Yıldız. 2014. Software defect prediction using Bayesian networks. *ESE* 19, 1 (2014), 154–181.

[70] Muhammed Maruf Öztürk. 2017. Which type of metrics are useful to deal with class imbalance in software defect prediction? *IST* 92 (2017), 17–29.

[71] Luca Pascarella, Fabio Palomba, and Alberto Bacchelli. 2019. Fine-grained just-in-time defect prediction. *JSS* 150 (4 2019), 22–36.

[72] Luca Pascarella, Fabio Palomba, and Alberto Bacchelli. 2020. On the performance of method-level bug prediction: A negative result. *JSS* 161 (3 2020).

[73] F. Peters and T. Menzies. 2012. Privacy and utility for defect prediction: Experiments with MORPH. *Procs. of ICSE* (2012), 189–199.

[74] F. Peters, T. Menzies, L. Gong, and H. Zhang. 2013. Balancing privacy and utility in cross-company defect prediction. *IEEE TSE* 39, 8 (2013), 1054–1068.

[75] F. Peters, T. Menzies, and L. Layman. 2015. LACE2: Better privacy-preserving data sharing for cross project defect prediction. *Procs. of ICSE* (2015), 801–811.

[76] Jean Petrić, David Bowes, Tracy Hall, Bruce Christianson, and Nathan Baddoo. 2016. Building an ensemble for software defect prediction based on diversity selection. In *Procs. of ESEM*. 1–10.

[77] Daryl Posnett, Vladimir Filkov, and Premkumar Devanbu. 2011. Ecological inference in empirical software engineering. In *Procs. of ASE*. 362–371.

[78] Rahul Premraj and Kim Herzig. 2011. Network versus code metrics to predict defects: A replication study. In *Procs. of ESEM*. 215–224.

[79] Yu Qu, Ting Liu, Jianlei Chi, Yangxu Jin, Di Cui, Ancheng He, and Qinghua Zheng. 2018. node2defect: using network embedding to improve software defect prediction. In *Procs. of ASE*. 844–849.

[80] F. Rahman and P. Devanbu. 2013. How, and why, process metrics are better. *Procs. of ICSE* (2013), 432–441.

[81] D. Ryu, O. Choi, and J. Baik. 2016. Value-cognitive boosting with a support vector machine for cross-project defect prediction. *ESE* 21, 1 (2016), 43–71.

[82] Lwin Khin Shar and Hee Beng Kuan Tan. 2012. Predicting common web application vulnerabilities from input validation and sanitization code patterns. In *Procs. of ASE*. 310–313.

[83] Thomas Shippey, David Bowes, and Tracy Hall. 2019. Automatically identifying code features for software defect prediction: Using AST N-grams. *IST* 106 (2 2019), 142–160.

[84] Behjat Soltanifar, Atakan Erdem, and Ayse Bener. 2016. Predicting defectiveness of software patches. In *Procs. of ESEM*. 1–10.

[85] Q. Song, Y. Guo, and M. Shepperd. 2019. A Comprehensive Investigation of the Role of Imbalanced Learning for Software Defect Prediction. *IEEE TSE* 45, 12 (2019), 1253–1269.

[86] Q. Song, Z. Jia, M. Shepperd, S. Ying, and J. Liu. 2011. A general software defect-proneness prediction framework. *IEEE TSE* 37, 3 (2011), 356–370.

[87] M. Tan, L. Tan, S. Dara, and C. Mayeux. 2015. Online Defect Prediction for Imbalanced Data. *Procs. of ICSE* (2015), 99–108.

[88] C. Tantithamthavorn, S. McIntosh, A.E. Hassan, A. Ihara, and K. Matsumoto. 2015. The impact of mislabelling on the performance and interpretation of defect prediction models. *Procs. of ICSE* (2015), 812–823.

[89] C. Tantithamthavorn, S. McIntosh, A.E. Hassan, and K. Matsumoto. 2016. Automated parameter optimization of classification techniques for defect prediction models. *Procs. of ICSE* (2016), 321–332.

[90] C. Tantithamthavorn, S. McIntosh, A.E. Hassan, and K. Matsumoto. 2017. An Empirical Comparison of Model Validation Techniques for Defect Prediction Models. *IEEE TSE* 43, 1 (2017), 1–18.

[91] C. Tantithamthavorn, S. McIntosh, A.E. Hassan, and K. Matsumoto. 2019. The Impact of Automated Parameter Optimization on Defect Prediction Models. *IEEE TSE* 45, 7 (2019), 683–711.

[92] Haonan Tong, Bin Liu, and Shihai Wang. 2018. Software defect prediction using stacked denoising autoencoders and two-stage ensemble learning. *IST* 96 (4 2018), 94–111.

[93] Aye Tosun, Aye Bener, Burak Turhan, and Tim Menzies. 2010. Practical considerations in deploying statistical methods for defect prediction: A case study within the Turkish telecommunications industry. *IST* 52, 1242–1257. Issue 11.

[94] Burak Turhan, Ayşe Tosun Misirli, and Ayşe Bener. 2013. Empirical evaluation of the effects of mixed project data on learning defect predictors. *IST* 55, 1101–1118. Issue 6.

[95] S. Wang, T. Liu, and L. Tan. 2016. Automatically learning semantic features for defect prediction. *Procs. of ICSE* (2016), 297–308.

[96] X. Xia, D. Lo, S.J. Pan, N. Nagappan, and X. Wang. 2016. HYDRA: Massively compositional model for cross-project defect prediction. *IEEE TSE* 42, 10 (2016), 977–998.

[97] Zhou Xu, Shuai Li, Xiapu Luo, Jin Liu, Tao Zhang, Yutian Tang, Jun Xu, Peipei Yuan, and Jacky Keung. 2019. TSTSS: A two-stage training subset selection framework for cross version defect prediction. *JSS* 154 (2019), 59–78.

[98] Z. Xu, S. Li, Y. Tang, X. Luo, T. Zhang, J. Liu, and J. Xu. 2018. Cross version defect prediction with representative data via sparse subset selection. *Procs. of ICSE* (2018), 132–143.

[99] Zhou Xu, Shuai Li, Jun Xu, Jin Liu, Xiapu Luo, Yifeng Zhang, Tao Zhang, Jacky Keung, and Yutian Tang. 2019. LDFR: Learning deep feature representation for software defect prediction. *JSS* 158 (12 2019).

[100] Zhou Xu, Jin Liu, Xiapu Luo, Zijiang Yang, Yifeng Zhang, Peipei Yuan, Yutian Tang, and Tao Zhang. 2019. Software defect prediction based on kernel PCA and weighted extreme learning machine. *IST* 106 (2019), 182–200.

[101] Xinli Yang, David Lo, Xin Xia, and Jianling Sun. 2017. TLEL: A two-layer ensemble learning approach for just-in-time defect prediction. *IST* 87 (7 2017), 206–220.

[102] S. Yatish, J. Jiarpakdee, P. Thongtanunam, and C. Tantithamthavorn. 2019. Mining Software Defects: Should We Consider Affected Releases? *Procs. of ICSE* 2019 (2019), 654–665.

[103] S. Young, T. Abdou, and A. Bener. 2018. A replication study: Just-in-time defect prediction with ensemble learning. *Procs. of ICSE* (2018), 42–47.

[104] Qiao Yu, Shujuan Jiang, and Yanmei Zhang. 2017. A feature matching and transfer approach for cross-company defect prediction. *JSS* 132 (10 2017), 366–378.

[105] T. Yu, W. Wen, X. Han, and J.H. Hayes. 2019. ConPredictor: Concurrency Defect Prediction in Real-World Applications. *IEEE TSE* 45, 6 (2019), 558–575.

[106] F. Zhang, A.E. Hassan, S. Mcintosh, and Y. Zou. 2017. The use of summation to aggregate software metrics hinders the performance of defect prediction models. *IEEE TSE* 43, 5 (2017), 476–491.

[107] F. Zhang, I. Keivanloo, and Y. Zou. 2017. Data Transformation in Cross-project Defect Prediction. *ESE* 22, 6 (2017), 3186–3218.

[108] F. Zhang, A. Mockus, I. Keivanloo, and Y. Zou. 2016. Towards building a universal defect prediction model with rank transformed predictors. *ESE* 21, 5 (2016), 2107–2145.

[109] F. Zhang, Q. Zheng, Y. Zou, and A.E. Hassan. 2016. Cross-project defect prediction using a connectivity-based unsupervised classifier. *Procs. of ICSE* (2016), 309–320.

[110] Tianchi Zhou, Xiaobing Sun, Xin Xia, Bin Li, and Xiang Chen. 2019. Improving defect prediction with deep forest. *IST* 114 (10 2019), 204–216.

[111] Yuming Zhou, Baowen Xu, and Hareton Leung. 2010. On the ability of complexity metrics to predict fault-prone classes in object-oriented systems. *JSS* 83 (2010), 660–674.