

REFERENCES

- [1]. J. Martins, C. Bezerra, A. Uchoa, and A. Garcia, "Are code smell co-occurrences harmful to internal quality attributes? a mixed-method study," in 34th SBES, 2020, pp. 52–61.
- [2]. L. Sousa, A. Oliveira, W. Oizumi, S. Barbosa, A. Garcia, J. Lee, M. Kalinowski, R. de Mello, B. Fonseca, R. Oliveira et al., "Identifying design problems in the source code: A grounded theory," in 40th ICSE, 2018, pp. 921–931.
- [3]. C. Barbosa, A. Uchoa, F. Falcao, D. Coutinho, H. Brito, G. Amaral, A. Garcia, B. Fonseca, M. Ribeiro, V. Soares, and L. Sousa, "Revealing the social aspects of design decay: A retrospective study of pull requests," in 34th SBES, 2020, pp. 364–373.
- [4]. M. Beller, A. Bacchelli, A. Zaidman, and E. Juergens, "Modern code reviews in open-source projects: Which problems do they fix?" in 11th MSR, 2014, pp. 202–211.
- [5]. E. Fernandes, A. Uchoa, A. C. Bibiano, and A. Garcia, "On the alternatives for composing batch refactoring," in 3rd IWoR. IEEE, 2019, pp. 9–12.
- [6]. L. Pascarella, D. Spadini, F. Palomba, and A. Bacchelli, "On the effect of code review on code smells," in 27th SANER, 2020.
- [7]. M. Paixão, A. Uchoa, A. C. Bibiano, D. Oliveira, A. Garcia, J. Krinke, and E. Arvonio, "Behind the intents: An in-depth empirical study on software refactoring in modern code review," in 17th MSR, 2020, pp. 125–136.
- [8]. A. Uchoa, C. Barbosa, W. Oizumi, P. Blenilio, R. Lima, A. Garcia, and C. Bezerra, "How does modern code review impact software design degradation? an in-depth empirical study," in 36th ICSME. IEEE, 2020, pp. 511–522.
- [9]. D. Feitelson, E. Frachtenberg, and K. Beck, "Development and deployment at facebook," *IEEE Internet Comput.*, vol. 17, no. 4, pp. 8–17, 2013.
- [10]. M. Paixao, J. Krinke, D. Han, C. Ragkhitwetsagul, and M. Harman, "The impact of code review on architectural changes," *IEEE Trans. Softw. Eng. (TSE)*, pp. 1–19, 2019.
- [11]. R. Morales, S. McIntosh, and F. Khomh, "Do code review practices impact design quality? a case study of the Qt, VTK, and ITK projects," in 22nd SANER. IEEE, 2015, pp. 171–180.
- [12]. M. C. O. Silva, M. T. Valente, and R. Terra, "Does technical debt lead to the rejection of pull requests?" in 12th SBSI. ACM, 2016, pp. 248–254.
- [13]. R. de Mello, A. Uchoa, R. Oliveira, W. Oizumi, J. Souza, K. Mendes, D. Oliveira, B. Fonseca, and A. Garcia, "Do research and practice of code smell identification walk together? a social representations analysis," in 13th ESEM. IEEE, 2019, pp. 1–6.
- [14]. T. Hirao, A. Ihara, Y. Ueda, P. Phannachitta, and K.-i. Matsumoto, "The impact of a low level of agreement among reviewers in a code review process," in 12th OSS, 2016, pp. 97–110.

- [15]. Z. Li, P. Avgeriou, and P. Liang, "A systematic mapping study on technical debt and its management," *J. Syst. Softw. (JSS)*, vol. 101, pp. 193–220, 2015.
- [16]. G. Suryanarayana, G. Samarthyam, and T. Sharma, *Refactoring for software design smells: managing technical debt*. Morgan Kaufmann, 2014.
- [17]. A. Bacchelli and C. Bird, "Expectations, outcomes, and challenges of modern code review," in *35th ICSE*, 2013, pp. 712–721.
- [18]. W. Oizumi, L. Sousa, A. Oliveira, L. Carvalho, A. Garcia, T. Colanzi, and R. Oliveira, "On the density and diversity of degradation symptoms in refactored classes: A multi-case study," in *30th ISSRE*, 2019, pp. 346–357.
- [19]. O. Baysal, O. Kononenko, R. Holmes, and M. W. Godfrey, "The influence of non-technical factors on code review," in *20th WCRE. IEEE*, 2013, pp. 122–131.
- [20]. M. S. Zanetti, I. Scholtes, C. J. Tessone, and F. Schweitzer, "Categorizing bugs with social networks: a case study on four open source software communities," in *35th ICSE. IEEE*, 2013, pp. 1032–1041.
- [21]. T. L. Graves, A. F. Karr, J. S. Marron, and H. Siy, "Predicting fault incidence using software change history," *IEEE Trans. Softw. Eng. (TSE)*, vol. 26, no. 7, pp. 653–661, 2000.
- [22]. Y. Jiang, B. Adams, and D. M. German, "Will my patch make it? and how fast? case study on the linux kernel," in *10th MSR. IEEE*, 2013, pp. 101–110.
- [23]. A. Mockus and D. M. Weiss, "Predicting risk of software changes," *Bell Labs Technical Journal*, vol. 5, no. 2, pp. 169–180, 2000.
- [24]. K. Herzig, S. Just, and A. Zeller, "It's not a bug, it's a feature: how misclassification impacts bug prediction," in *35th ICSE. IEEE*, 2013, pp. 392–401.
- [25]. Y. Fan, X. Xia, D. Lo, and S. Li, "Early prediction of merged code changes to prioritize reviewing tasks," *Emp. Softw. Eng. (ESE)*, vol. 23, no. 6, pp. 3346–3393, 2018.
- [26]. V. Soares, A. Oliveira, P. Farah, A. Bibiano, D. Coutinho, A. Garcia, S. Vergilio, M. Schots, D. Oliveira, and A. Uch^oao, "On the relation between complexity, explicitness, effectiveness of refactorings and nonfunctional concerns," in *34th SBES*, 2020, pp. 788–797.
- [27]. A. Hindle, D. M. German, and R. Holt, "What do large commits tell us? a taxonomical study of large commits," in *5th MSR*, 2008, pp. 99–108.
- [28]. A. Uch^oao, C. Barbosa, D. Coutinho, W. Oizumi, W. K. G. Assunc[~]ao, S. Regina Vergilio, J. Alves Pereira, A. Oliveira, and A. Garcia. (2021) Replication package for the paper: "predicting design impactful changes in modern code review: A large-scale empirical study". Accessed: 2024-03-14. [Online]. Available: <http://doi.org/10.5281/zenodo.4563214>
- [29]. W. Oizumi, A. Garcia, L. Sousa, B. Cafeo, and Y. Zhao, "Code anomalies flock together: Exploring code anomaly agglomerations for locating design problems," in *38th ICSE*, 2016.
- [30]. R. C. Martin and M. Martin, *Agile Principles, Patterns, and Practices in C#* (Robert C. Martin). Upper Saddle River, NJ, USA: Prentice Hall PTR, 2006.

- [31]. M. Paixao, J. Krinke, D. Han, and M. Harman, "Crop: Linking code reviews to source code changes," in 15th MSR, 2018, pp. 46–49.
- [32]. S. B. Kotsiantis, I. D. Zaharakis, and P. E. Pintelas, "Machine learning: a review of classification and combining techniques," *Artificial Intelligence Review*, vol. 26, no. 3, pp. 159–190, 2006.
- [33]. <https://git.eclipse.org/r/#/c/825/>, 2020, accessed in: August 2023.
- [34]. J. A. Pereira, H. Martin, M. Acher, J.-M. J'éz'equel, G. Botterweck, and A. Ventresque, "Learning software configuration spaces: A systematic literature review," *arXiv preprint arXiv:1906.03018*, 2019.
- [35]. S. Krishnan, C. Strasburg, R. R. Lutz, and K. Gořseva-Popstojanova, "Are change metrics good predictors for an evolving software product line?" in 7th PROMISE, 2011, pp. 1–10.
- [36]. M. D'Ambros, M. Lanza, and R. Robbes, "An extensive comparison of bug prediction approaches," in 7th MSR. IEEE, 2010, pp. 31–41.
- [37]. S. Matsumoto, Y. Kamei, A. Monden, K.-i. Matsumoto, and M. Nakamura, "An analysis of developer metrics for fault prediction," in 6th PROMISE, 2010, pp. 1–9.
- [38]. Y. Kamei, E. Shihab, B. Adams, A. E. Hassan, A. Mockus, A. Sinha, and N. Ubayashi, "A large-scale empirical study of just-in-time quality assurance," *IEEE Trans. Softw. Eng. (TSE)*, vol. 39, no. 6, pp. 757–773, 2012.
- [39]. H. S. Yazdi, M. Mirbolouki, P. Pietsch, T. Kehrler, and U. Kelter, "Analysis and prediction of design model evolution using time series," in 26th CAiSE. Springer, 2014, pp. 1–15.
- [40]. J. Alves Pereira, M. Acher, H. Martin, and J.-M. J'éz'equel, "Sampling effect on performance prediction of configurable systems: A case study," in 11th ICPE, 2020, pp. 277–288.
- [41]. Eclipse, <https://git.eclipse.org/r/#/c/3345/>, 2020, accessed in: August 2023.
- [42]. T. Sharma, P. Mishra, and R. Tiwari, "Designite: a software design quality assessment tool," in 1st BRIDGE. ACM, 2016, pp. 1–4.
- [43]. T. Menzies and M. Shepperd, "'bad smells' in software analytics papers," *Inf. Softw. Technol. (IST)*, vol. 112, pp. 35–47, 2019.
- [44]. T. Sharma and D. Spinellis, "A survey on software smells," *J. Syst. Softw. (JSS)*, vol. 138, pp. 158–173, 2018.
- [45]. F. Palomba, G. Bavota, M. Di Penta, R. Oliveto, and A. De Lucia, "Do they really smell bad? a study on developers' perception of bad code smells," in 2014 ICSME. IEEE, 2014, pp. 101–110.
- [46]. A. Yamashita, M. Zanoni, F. A. Fontana, and B. Walter, "Inter-smell relations in industrial and open source systems: A replication and comparative analysis," in 31st ICSME. IEEE, 2015, pp. 121–130.
- [47]. M. Paixao and P. H. Maia, "Rebasing in code review considered harmful: A large-scale empirical investigation," in 19th SCAM. IEEE, 2019, pp. 45–55.
- [48]. C. Rosen, B. Grawi, and E. Shihab, "Commit guru: analytics and risk prediction of software commits," in 10th FSE, 2015, pp. 966–969.

- [49]. P. Weißgerber, D. Neu, and S. Diehl, "Small patches get in!" in 5th MSR, 2008, pp. 67–76.
- [50]. P. Thongtanunam, S. McIntosh, A. E. Hassan, and H. Iida, "Review participation in modern code review," *Emp. Softw. Eng. (ESE)*, vol. 22, no. 2, pp. 768–817, 2017.
- [51]. G. Bavota and B. Russo, "Four eyes are better than two: On the impact of code reviews on software quality," in 31st ICSME. IEEE, Sep. 2015, pp. 81–90.
- [52]. "Can automated impact analysis techniques help predict decaying modules?" in 35th ICSME, 2019, pp. 541–545.
- [53]. M. I. Azeem, F. Palomba, L. Shi, and Q. Wang, "Machine learning techniques for code smell detection: A systematic literature review and meta-analysis," *Inf. Softw. Technol.*, vol. 108, pp. 115 – 138, 2019.
- [54]. D. Di Nucci, F. Palomba, D. A. Tamburri, A. Serebrenik, and A. De Lucia, "Detecting code smells using machine learning techniques: Are we there yet?" in 25th SANER. IEEE, 2018, pp. 612–621.
- [55]. G. Gousios, M. Pinzger, and A. v. Deursen, "An exploratory study of the pull-based software development model," in 36th ICSE, 2014, pp. 345–355.
- [56]. M. Alenezi and M. Zarour, "An empirical study of bad smells during software evolution using designite tool," *i-Manager's Journal on Software Engineering*, vol. 12, no. 4, p. 12, 2018.
- [57]. M. Caulo, B. Lin, G. Bavota, G. Scanniello, and M. Lanza, "Knowledge transfer in modern code review," in 28th ICPC, 2020, pp. 230–240.
- [58]. T. Sharma, P. Singh, and D. Spinellis, "An empirical investigation on the relationship between design and architecture smells," *Emp. Softw. Eng. (ESE)*, 2020.
- [59]. F. Arcelli Fontana and M. Zanoni, "Code smell severity classification using machine learning techniques," *Knowledge-Based Systems*, vol. 128, pp. 43 – 58, 2017.
- [60]. F. Luiz Caram, B. R. de Oliveira Rodrigues, A. Campanelli, and F. Silva Parreiras, "Machine learning techniques for code smells detection: A systematic mapping study," *Int. J. Softw. Eng. Knowl. Eng.*, vol. 29, pp. 285–316, 02 2019.
- [61]. N. Sae-Lim, S. Hayashi, and M. Saeki, "Toward proactive refactoring: An exploratory study on decaying modules," in 3rd IWoR. IEEE, 2019, pp. 39–46.
- [62]. S. McIntosh, Y. Kamei, B. Adams, and A. E. Hassan, "An empirical study of the impact of modern code review practices on software quality," *Emp. Softw. Eng. (ESE)*, vol. 21, no. 5, pp. 2146–2189, Oct. 2016.
- [63]. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

- [64]. F. Palomba, G. Bavota, M. D. Penta, R. Oliveto, D. Poshyvanyk, and A. D. Lucia, "Mining version histories for detecting code smells," *IEEE Trans. Softw. Eng. (TSE)*, vol. 41, no. 5, pp. 462–489, 2015.
- [65]. M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *Journal of the american statistical association*, vol. 32, no. 200, pp. 675–701, 1937.
- [66]. J. H. McDonald, *Handbook of biological statistics*. sparky house publishing Baltimore, MD, 2009, vol. 2.
- [67]. E. Whitley and J. Ball, "Statistics review 6: Nonparametric methods," *Critical care*, vol. 6, no. 6, p. 509, 2002.
- [68]. R. Kohavi et al., "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Ijcai*, vol. 14, no. 2. Montreal, Canada, 1995, pp. 1137–1145.
- [69]. S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *32nd ICML*, 2015.
- [70]. Q. Song, Y. Guo, and M. Shepperd, "A comprehensive investigation of the role of imbalanced learning for software defect prediction," *IEEE Trans. Softw. Eng. (TSE)*, vol. 45, no. 12, pp. 1253–1269, 2018.
- [71]. F. Hutter, L. Kotthoff, and J. Vanschoren, *Automated machine learning: methods, systems, challenges*. Springer Nature, 2019.
- [72]. T. Fawcett, "An introduction to roc analysis," *Pattern Recogn. Lett.*, vol. 27, no. 8, pp. 861–874, 2006.
- [73]. R. J. Grissom and J. J. Kim, *Effect sizes for research: A broad practical approach*. Lawrence Erlbaum Associates Publishers, 2005.
- [74]. J. Romano, J. D. Kromrey, J. Coraggio, J. Skowronek, and L. Devine, "Exploring methods for evaluating group differences on the nsse and other surveys: Are the t-test and cohen'sd indices the most appropriate choices," in *annual meeting of the Southern Association for Institutional Research*. Citeseer, 2006, pp. 1–51.
- [75]. F. Arcelli Fontana, M. V. M'antyl'a, M. Zanoni, and A. Marino, "Comparing and experimenting machine learning techniques for code smell detection," *Emp. Softw. Eng. (ESE)*, vol. 21, pp. 1143 – 1191, 2016.
- [76]. H. Lal and G. Pahwa, "Code review analysis of software system using machine learning techniques," in *11th ISCO*, 2017, pp. 8–13.
- [77]. L. Pascarella, D. Spadini, F. Palomba, and A. Bacchelli, "On the effect of code review on code smells," in *27th SANER*, 2020.
- [78]. A. Eposhi, W. Oizumi, A. Garcia, L. Sousa, R. Oliveira, and A. Oliveira, "Removal of design problems through refactorings: are we looking at the right symptoms?" in *27th ICPC*. IEEE Press, 2019, pp. 148–153.
- [79]. Caram FL, Rodrigues BRDO, Campanelli AS, Parreiras FS. Machine learning techniques for code smells detection: a systematic mapping study. *Int J Softw Eng Knowl Eng*. 2019;29(02):285-316.