

Week 6 Journal

Rodger Byrd

I. OVERVIEW

For my area I'm looking at Anti-Patterns in code, these are also known as code smells. I've also seen them referred to as Atoms of Confusion and nano patterns.

II. JOURNAL ENTRY

This week I continued finding recent papers to include in my survey. I decided to focus on recent (2018 and 2019) papers that have been published since the last survey paper of 2018 because it was a good survey. One very interesting thing that was previously missing from the other papers was machine learning. This week I spent most of the time looking for papers related to using machine learning to correct or detect anti-patterns and code smells in code. I found quite a few different papers so I think this will probably be the primary focus of my survey paper that I've started drafting.

The files for this latex document are in the github repository located at <https://github.com/rodger79/CS6000>

Relevant papers are referenced in the bibliography below.

REFERENCES

- [1] M. Mantyla, J. Vanhanen, and C. Lassenius, "A taxonomy and an initial empirical study of bad smells in code," in *International Conference on Software Maintenance, 2003. ICSM 2003. Proceedings.*, Sep. 2003, pp. 381–384.
- [2] R. Arcoverde, A. Garcia, and E. Figueiredo, "Understanding the Longevity of Code Smells: Preliminary Results of an Explanatory Survey," in *Proceedings of the 4th Workshop on Refactoring Tools*, ser. WRT '11. New York, NY, USA: ACM, 2011, pp. 33–36, event-place: Waikiki, Honolulu, HI, USA. [Online]. Available: <http://doi.acm.org/10.1145/1984732.1984740>
- [3] A. Yamashita and L. Moonen, "Do developers care about code smells? An exploratory survey," Oct., pp. 242–251.
- [4] V. Garousi and B. Kk, "Smells in software test code: A survey of knowledge in industry and academia," *Journal of Systems and Software*, vol. 138, pp. 52 – 81, 2018. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0164121217303060>
- [5] N. Yoshioka, H. Washizaki, and K. Maruyama, "A survey on security patterns," *Progress in Informatics*, no. 5, p. 35, Mar. 2008. [Online]. Available: http://www.nii.ac.jp/pi/n5/5_35.html
- [6] T. Sharma and D. Spinellis, "A survey on software smells," *Journal of Systems and Software*, vol. 138, pp. 158 – 173, 2018. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0164121217303114>
- [7] D. Gopstein, J. Iannaccone, Y. Yan, L. DeLong, Y. Zhuang, M. K.-C. Yeh, and J. Cappos, "Understanding Misunderstandings in Source Code," in *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering*, ser. ESEC/FSE 2017. New York, NY, USA: ACM, 2017, pp. 129–139, event-place: Paderborn, Germany. [Online]. Available: <http://doi.acm.org/10.1145/3106237.3106264>
- [8] S. Singh and S. Kaur, "A systematic literature review: Refactoring for disclosing code smells in object oriented software," *Ain Shams Engineering Journal*, vol. 9, no. 4, pp. 2129 – 2151, 2018. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S2090447917300412>
- [9] Z. Li, T.-H. P. Chen, J. Yang, and W. Shang, "Dlfinder: Characterizing and Detecting Duplicate Logging Code Smells," in *Proceedings of the 41st International Conference on Software Engineering*, ser. ICSE '19. Piscataway, NJ, USA: IEEE Press, 2019, pp. 152–163, event-place: Montreal, Quebec, Canada. [Online]. Available: <https://doi.org/10.1109/ICSE.2019.00032>
- [10] M. S. Haque, J. Carver, and T. Atkison, "Causes, Impacts, and Detection Approaches of Code Smell: A Survey," in *Proceedings of the ACMSE 2018 Conference*, ser. ACMSE '18. New York, NY, USA: ACM, 2018, pp. 25:1–25:8, event-place: Richmond, Kentucky. [Online]. Available: <http://doi.acm.org/10.1145/3190645.3190697>
- [11] F. A. Fontana, V. Lenarduzzi, R. Roveda, and D. Taibi, "Are architectural smells independent from code smells? An empirical study," *Journal of Systems and Software*, vol. 154, pp. 139 – 156, 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0164121219301013>
- [12] B. Walter, F. A. Fontana, and V. Ferme, "Code smells and their collocations: A large-scale experiment on open-source systems," *Journal of Systems and Software*, vol. 144, pp. 1 – 21, 2018. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0164121218301109>
- [13] S. S. Afjehei, T.-H. P. Chen, and N. Tsantalis, "iPerfDetector: Characterizing and detecting performance anti-patterns in iOS applications," *Empirical Software Engineering*, Apr. 2019. [Online]. Available: <https://doi.org/10.1007/s10664-019-09703-y>
- [14] F. Tian, P. Liang, and M. A. Babar, "How Developers Discuss Architecture Smells? An Exploratory Study on Stack Overflow," in *2019 IEEE International Conference on Software Architecture (ICSA)*, Mar. 2019, pp. 91–100.
- [15] C. Vassallo, S. Proksch, H. C. Gall, and M. Di Penta, "Automated Reporting of Anti-patterns and Decay in Continuous Integration," in *Proceedings of the 41st International Conference on Software Engineering*, ser. ICSE '19. Piscataway, NJ, USA: IEEE Press, 2019, pp. 105–115, event-place: Montreal, Quebec, Canada. [Online]. Available: <https://doi.org/10.1109/ICSE.2019.00028>
- [16] D. Taibi, V. Lenarduzzi, and C. Pahl, *Microservices Anti Patterns: A Taxonomy*, 2019.
- [17] A. Tahir, A. Yamashita, S. Licorish, J. Dietrich, and S. Counsell, "Can You Tell Me if It Smells?: A Study on How Developers Discuss Code Smells and Anti-patterns in Stack Overflow," in *Proceedings of the 22Nd International Conference on Evaluation and Assessment in Software Engineering 2018*, ser. EASE'18. New York, NY, USA: ACM, 2018, pp. 68–78, event-place: Christchurch, New Zealand. [Online]. Available: <http://doi.acm.org/10.1145/3210459.3210466>
- [18] R. Ibrahim, M. Ahmed, R. Nayak, and S. Jamel, "Reducing redundancy of test cases generation using code smell detection and refactoring," *Journal of King Saud University - Computer and Information Sciences*, 2018. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1319157818300296>
- [19] H. Brabra, A. Mtibaa, F. Petrillo, P. Merle, L. Sliman, N. Moha, W. Gaaloul, Y.-G. Guhneuc, B. Benatallah, and F. Gargouri, "On semantic detection of cloud API (anti)patterns," *Information and Software Technology*, vol. 107, pp. 65 – 82, 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S095058491830226X>
- [20] S. Hussain, J. Keung, M. K. Sohail, A. A. Khan, G. Ahmad, M. R. Mufti, and H. A. Khatak, "Methodology for the quantification of the effect of patterns and anti-patterns association on the software quality," *IET Software*, vol. 13, no. 5, pp. 414–422, 2019.
- [21] Y. Lyu, D. Li, and W. G. J. Halfond, "Remove RATs from Your Code: Automated Optimization of Resource Inefficient Database Writes for Mobile Applications," in *Proceedings of the 27th ACM SIGSOFT International Symposium on Software Testing and Analysis*, ser. ISSTA 2018. New York, NY, USA: ACM, 2018, pp. 310–321, event-place: Amsterdam, Netherlands. [Online]. Available: <http://doi.acm.org/10.1145/3213846.3213865>
- [22] A. Abadi, M. Abadi, and I. Ben-Harrush, "Fixing anti-patterns in javascript," US Patent US983975B2, May, 2018. [Online]. Available: <https://patents.google.com/patent/US983975B2/en>
- [23] M. Kessentini, R. Mahaouachi, and K. Ghedira, "What you like in design use to correct bad-smells," *Software Quality Journal*, vol. 21, no. 4, pp. 551–571, Dec. 2013. [Online]. Available: <https://doi.org/10.1007/s11219-012-9187-6>
- [24] A. Kaur, K. Kaur, and S. Jain, "Predicting software change-proneness with code smells and class imbalance learning," in *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Sep. 2016, pp. 746–754.
- [25] N. Pritam, M. Khari, L. H. Son, R. Kumar, S. Jha, I. Priyadarshini, M. Abdel-Basset, and H. V. Long, "Assessment of Code Smell for Predicting Class Change Proneness Using Machine Learning," *IEEE Access*, vol. 7, pp. 37414–37425, 2019.
- [26] F. Arcelli Fontana, M. V. Mntyl, M. Zanoni, and A. Marino, "Comparing and experimenting machine learning techniques for code smell detection," *Empirical Software Engineering*, vol. 21, no. 3, pp. 1143–1191, Jun. 2016. [Online]. Available: <https://doi.org/10.1007/s10664-015-9378-4>
- [27] A. Kaur, S. Jain, and S. Goel, "SP-J48: a novel optimization and machine-learning-based approach for solving complex problems: special application in software engineering for detecting code smells," *Neural Computing and Applications*, Apr. 2019. [Online]. Available: <https://doi.org/10.1007/s00521-019-04175-z>
- [28] A. Maiga, N. Ali, N. Bhattacharya, A. Saban, Y. Guhneuc, and E. Aimeur, "SMURF: A SVM-based Incremental Anti-pattern Detection Approach," in *2012 19th Working Conference on Reverse Engineering*, Oct. 2012, pp. 466–475.
- [29] L. Kumar and A. Sureka, "An Empirical Analysis on Web Service Anti-pattern Detection Using a Machine Learning Framework," in *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, vol. 01, Jul. 2018, pp. 2–11.
- [30] D. D. Nucci, F. Palomba, D. A. Tamburri, A. Serebrenik, and A. D. Lucia, "Detecting code smells using machine learning techniques: Are we there yet?" in *2018 IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, Mar. 2018, pp. 612–621.
- [31] R. Malhotra, "A systematic review of machine learning techniques for software fault prediction," *Applied Soft Computing*, vol. 27, pp. 504 – 518, 2015. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1568494614005857>
- [32] F. A. Fontana and M. Zanoni, "Code smell severity classification using machine learning techniques," *Knowledge-Based Systems*, vol. 128, pp. 43 – 58, 2017. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0950705117301880>
- [33] F. Arcelli Fontana, M. V. Mntyl, M. Zanoni, and A. Marino, "Comparing and experimenting machine learning techniques for code smell detection," *Empirical Software Engineering*, vol. 21,

- no. 3, pp. 1143–1191, Jun. 2016. [Online]. Available: <https://doi.org/10.1007/s10664-015-9378-4>
- [34] *Findings from FUMEC University Provides New Data on Machine Learning (Machine Learning Techniques for Code Smells Detection: a Systematic Mapping Study)*, 2019.
 - [35] U. Azadi, F. A. Fontana, and M. Zaroni, “Poster: machine learning based code smell detection through WekaNose,” in *2018 IEEE/ACM 40th International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*. IEEE, 2018, pp. 288–289.
 - [36] M. Kessentini, “Understanding the Correlation between Code Smells And Software Bugs,” 2019.
 - [37] A. Barbez, F. Khomh, and Y.-G. Guhneuc, “A Machine-learning Based Ensemble Method For Anti-patterns Detection,” *CoRR*, vol. abs/1903.01899, 2019. [Online]. Available: <http://arxiv.org/abs/1903.01899>
 - [38] S. Saluja and U. Batra, “Assessing Quality by Anti-pattern Detection in Web Services,” Social Science Research Network, Rochester, NY, SSRN Scholarly Paper ID 3350876, Mar. 2019. [Online]. Available: <https://papers.ssrn.com/abstract=3350876>
 - [39] W. Song, C. Zhang, and H. Jacobsen, “An Empirical Study on Data Flow Bugs in Business Processes,” *IEEE Transactions on Cloud Computing*, pp. 1–1, 2018.
 - [40] M. I. Azeem, F. Palomba, L. Shi, and Q. Wang, “Machine learning techniques for code smell detection: A systematic literature review and meta-analysis,” *Information and Software Technology*, vol. 108, pp. 115 – 138, 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0950584918302623>
 - [41] V. Garousi, B. Kucuk, and M. Felderer, “What We Know About Smells in Software Test Code,” *IEEE Software*, vol. 36, no. 3, pp. 61–73, May 2019.
 - [42] D. Arcelli, V. Cortellessa, and D. D. Pompeo, “Performance-driven software model refactoring,” *Information and Software Technology*, vol. 95, pp. 366 – 397, 2018. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0950584917301787>