

Dr. Krishna Narasimhan - Publications (Conferences, Books, Workshops, Journals)

- [1] Simon Binder et al. “jGuard: Programming Misuse-Resilient APIs”. In: *Proceedings of the 15th ACM SIGPLAN International Conference on Software Language Engineering, SLE 2022, Auckland, New Zealand, December 6-7, 2022*. Ed. by Bernd Fischer, Lola Burgueño, and Walter Cazzola. ACM, 2022, pp. 161–174. DOI: 10.1145/3567512.3567526. URL: <https://doi.org/10.1145/3567512.3567526>.
- [2] Rodrigo Bonifácio et al. “Dealing with Variability in API Misuse Specification”. In: *35th European Conference on Object-Oriented Programming, ECOOP 2021, July 11-17, 2021, Aarhus, Denmark (Virtual Conference)*. Ed. by Anders Møller and Manu Sridharan. Vol. 194. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, 19:1–19:27. DOI: 10.4230/LIPIcs.ECOOP.2021.19. URL: <https://doi.org/10.4230/LIPIcs.ECOOP.2021.19>.
- [3] Leon Chemnitz et al. “Towards Code Generation from BDD Test Case Specifications: A Vision”. In: *CoRR* abs/2305.11619 (2023). DOI: 10.48550/arXiv.2305.11619. arXiv: 2305.11619. URL: <https://doi.org/10.48550/arXiv.2305.11619>.
- [4] Francisco Handrick da Costa et al. “Exploring the use of static and dynamic analysis to improve the performance of the mining sandbox approach for android malware identification”. In: *J. Syst. Softw.* 183 (2022), p. 111092. DOI: 10.1016/j.jss.2021.111092. URL: <https://doi.org/10.1016/j.jss.2021.111092>.
- [5] Foad Jafarinejad, Krishna Narasimhan, and Mira Mezini. “NerdBug: automated bug detection in neural networks”. In: *AISTA 2021: Proceedings of the 1st ACM International Workshop on AI and Software Testing/Analysis, Virtual Event, Denmark, 12 July 2021*. Ed. by Shuai Wang, Xiaofei Xie, and Lei Ma. ACM, 2021, pp. 13–16. DOI: 10.1145/3464968.3468409. URL: <https://doi.org/10.1145/3464968.3468409>.
- [6] Patrick Müller, Krishna Narasimhan, and Mira Mezini. “Fex: Assisted Identification of Domain Features from C Programs”. In: *21st IEEE International Working Conference on Source Code Analysis and Manipulation, SCAM 2021, Luxembourg, September 27-28, 2021*. IEEE, 2021, pp. 170–

180. DOI: 10.1109/SCAM52516.2021.00029. URL: <https://doi.org/10.1109/SCAM52516.2021.00029>.
- [7] Krishna Narasimhan. “Clone Merge - An Eclipse Plugin to Abstract Near-Clone C++ Methods”. In: *30th IEEE/ACM International Conference on Automated Software Engineering, ASE 2015, Lincoln, NE, USA, November 9-13, 2015*. Ed. by Myra B. Cohen, Lars Grunske, and Michael Whalen. IEEE Computer Society, 2015, pp. 819–823. DOI: 10.1109/ASE.2015.103. URL: <https://doi.org/10.1109/ASE.2015.103>.
 - [8] Krishna Narasimhan. “Combining user-interaction and automation to evolve source code”. PhD thesis. Goethe University Frankfurt, Frankfurt am Main, Germany, 2017. URL: <http://publikationen.ub.uni-frankfurt.de/frontdoor/index/index/docId/42783>.
 - [9] Krishna Narasimhan and Christoph Reichenbach. “Copy and paste redeemed”. In: *Software Engineering 2016, Fachtagung des GI-Fachbereichs Softwaretechnik, 23.-26. Februar 2016, Wien, Österreich*. Ed. by Jens Knoop and Uwe Zdun. Vol. P-252. LNI. GI, 2016, pp. 61–62. URL: <https://dl.gi.de/20.500.12116/722>.
 - [10] Krishna Narasimhan and Christoph Reichenbach. “Copy and Paste Redeemed (T)”. In: *30th IEEE/ACM International Conference on Automated Software Engineering, ASE 2015, Lincoln, NE, USA, November 9-13, 2015*. Ed. by Myra B. Cohen, Lars Grunske, and Michael Whalen. IEEE Computer Society, 2015, pp. 630–640. DOI: 10.1109/ASE.2015.39. URL: <https://doi.org/10.1109/ASE.2015.39>.
 - [11] Krishna Narasimhan, Christoph Reichenbach, and Julia Lawall. “Cleaning up copy-paste clones with interactive merging”. In: *Autom. Softw. Eng.* 25.3 (2018), pp. 627–673. DOI: 10.1007/s10515-018-0238-5. URL: <https://doi.org/10.1007/s10515-018-0238-5>.
 - [12] Krishna Narasimhan, Christoph Reichenbach, and Julia Lawall. “Interactive data representation migration: exploiting program dependence to aid program transformation”. In: *Proceedings of the 2017 ACM SIGPLAN Workshop on Partial Evaluation and Program Manipulation, PEPM 2017, Paris, France, January 18-20, 2017*. Ed. by Ulrik Pagh Schultz and Jeremy Yallop. ACM, 2017, pp. 47–58. DOI: 10.1145/3018882.3018890. URL: <https://doi.org/10.1145/3018882.3018890>.
 - [13] Marcel Ochs, Krishna Narasimhan, and Mira Mezini. “Evaluating and improving transformers pre-trained on ASTs for Code Completion”. In: *IEEE International Conference on Software Analysis, Evolution and Reengineering, SANER 2023, Taipa, Macao, March 21-24, 2023*. Ed. by Tao Zhang, Xin Xia, and Nicole Novielli. IEEE, 2023, pp. 834–844. DOI: 10.1109/SANER56733.2023.00096. URL: <https://doi.org/10.1109/SANER56733.2023.00096>.

- [14] Michael Schlichtig et al. “FUM - A Framework for API Usage constraint and Misuse Classification”. In: *IEEE International Conference on Software Analysis, Evolution and Reengineering, SANER 2022, Honolulu, HI, USA, March 15-18, 2022*. IEEE, 2022, pp. 673–684. DOI: 10.1109/SANER53432.2022.00085. URL: <https://doi.org/10.1109/SANER53432.2022.00085>.
- [15] Michael Schlichtig et al. “Introducing FUM: A Framework for API Usage Constraint and Misuse Classification”. In: *Software Engineering 2023, Fachtagung des GI-Fachbereichs Softwaretechnik, 20.-24. Februar 2023, Paderborn*. Ed. by Gregor Engels, Regina Hebig, and Matthias Tichy. Vol. P-332. LNI. Gesellschaft für Informatik e.V., 2023, pp. 105–106. URL: <https://dl.gi.de/20.500.12116/40106>.
- [16] Sebastian Sztwierz et al. “Impact of programming languages on machine learning bugs”. In: *AISTA 2021: Proceedings of the 1st ACM International Workshop on AI and Software Testing/Analysis, Virtual Event, Denmark, 12 July 2021*. Ed. by Shuai Wang, Xiaofei Xie, and Lei Ma. ACM, 2021, pp. 9–12. DOI: 10.1145/3464968.3468408. URL: <https://doi.org/10.1145/3464968.3468408>.
- [17] Anna-Katharina Wickert et al. “To Fix or Not to Fix: A Critical Study of Crypto-misuses in the Wild”. In: *IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2022, Wuhan, China, December 9-11, 2022*. IEEE, 2022, pp. 315–322. DOI: 10.1109/TrustCom56396.2022.00051. URL: <https://doi.org/10.1109/TrustCom56396.2022.00051>.
- [18] Yu Zhou et al. “BRAID: an API recommender supporting implicit user feedback”. In: *ESEC/FSE ’21: 29th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering, Athens, Greece, August 23-28, 2021*. Ed. by Diomidis Spinellis et al. ACM, 2021, pp. 1510–1514. DOI: 10.1145/3468264.3473111. URL: <https://doi.org/10.1145/3468264.3473111>.