

## REFERENCES

- [1] X. Ren, Z. Xing, X. Xia, D. Lo, X. Wang, and J. Grundy, "Neural network-based detection of self-admitted technical debt: From performance to explainability," *ACM transactions on software engineering and methodology (TOSEM)*, vol. 28, no. 3, pp. 1–45, 2019.
- [2] Z. T. Sworna, C. Islam, and M. A. Babar, "Apiro: A framework for automated security tools api recommendation," *ACM Transactions on Software Engineering and Methodology*, vol. 32, no. 1, pp. 1–42, 2023.
- [3] Y. Zhao, T. Su, Y. Liu, W. Zheng, X. Wu, R. Kavuluru, W. G. Halfond, and T. Yu, "Recdroid+: Automated end-to-end crash reproduction from bug reports for android apps," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 31, no. 3, pp. 1–33, 2022.
- [4] F. Ebrahimi, M. Tushev, and A. Mahmoud, "Classifying mobile applications using word embeddings," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 31, no. 2, pp. 1–30, 2021.
- [5] Y. Zhou, J. K. Siow, C. Wang, S. Liu, and Y. Liu, "Spi: Automated identification of security patches via commits," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 31, no. 1, pp. 1–27, 2021.
- [6] Z. Chen, Y. Cao, H. Yao, X. Lu, X. Peng, H. Mei, and X. Liu, "Emoji-powered sentiment and emotion detection from software developers' communication data," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 30, no. 2, pp. 1–48, 2021.
- [7] H. Guo, S. Chen, Z. Xing, X. Li, Y. Bai, and J. Sun, "Detecting and augmenting missing key aspects in vulnerability descriptions," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 31, no. 3, pp. 1–27, 2022.
- [8] D. Wu, X.-Y. Jing, H. Zhang, Y. Feng, H. Chen, Y. Zhou, and B. Xu, "Retrieving api knowledge from tutorials and stack overflow based on natural language queries," *ACM Transactions on Software Engineering and Methodology*, vol. 32, no. 5, pp. 1–36, 2023.
- [9] J. Sarker, A. K. Turzo, M. Dong, and A. Bosu, "Automated identification of toxic code reviews using toxicr," *ACM Transactions on Software Engineering and Methodology*, vol. 32, no. 5, pp. 1–32, 2023.
- [10] X. Chen, C. Chen, D. Zhang, and Z. Xing, "Sethesaurus: Wordnet in software engineering," *IEEE Transactions on Software Engineering*, vol. 47, no. 9, pp. 1960–1979, 2019.
- [11] C. Chen, Z. Xing, Y. Liu, and K. O. L. Xiong, "Mining likely analogical apis across third-party libraries via large-scale unsupervised api semantics embedding," *IEEE Transactions on Software Engineering*, vol. 47, no. 3, pp. 432–447, 2019.
- [12] X. Li, H. Jiang, Y. Kamei, and X. Chen, "Bridging semantic gaps between natural languages and apis with word embedding," *IEEE Transactions on Software Engineering*, vol. 46, no. 10, pp. 1081–1097, 2018.
- [13] S. Ma, Z. Xing, C. Chen, C. Chen, L. Qu, and G. Li, "Easy-to-deploy api extraction by multi-level feature embedding and transfer learning," *IEEE Transactions on Software Engineering*, vol. 47, no. 10, pp. 2296–2311, 2019.
- [14] Z. Liu, X. Xia, D. Lo, Z. Xing, A. E. Hassan, and S. Li, "Which variables should i log?," *IEEE Transactions on Software Engineering*, vol. 47, no. 9, pp. 2012–2031, 2019.
- [15] N. Zhang, Q. Huang, X. Xia, Y. Zou, D. Lo, and Z. Xing, "Chatbot4qr: Interactive query refinement for technical question retrieval," *IEEE Transactions on Software Engineering*, vol. 48, no. 4, pp. 1185–1211, 2020.
- [16] Y. Huang, N. Jia, X. Chen, K. Hong, and Z. Zheng, "Code review knowledge perception: Fusing multi-features for salient-class location," *IEEE Transactions on Software Engineering*, vol. 48, no. 5, pp. 1463–1479, 2020.
- [17] D. Sondhi, M. Jobanputra, D. Rani, S. Purandare, S. Sharma, and R. Purandare, "Mining similar methods for test adaptation," *IEEE Transactions on Software Engineering*, vol. 48, no. 7, pp. 2262–2276, 2021.
- [18] D. Liu, H. Jiang, X. Li, Z. Ren, L. Qiao, and Z. Ding, "Dpword2vec: better representation of design patterns in semantics," *IEEE Transactions on Software Engineering*, vol. 48, no. 4, pp. 1228–1248, 2020.
- [19] W. Ou, S. H. Ding, Y. Tian, and L. Song, "Scs-gan: Learning functionality-agnostic stylistic representations for source code authorship verification," *IEEE Transactions on Software Engineering*, vol. 49, no. 4, pp. 1426–1442, 2022.
- [20] H. Khalajzadeh, M. Shahin, H. O. Obie, P. Agrawal, and J. Grundy, "Supporting developers in addressing human-centric issues in mobile apps," *IEEE Transactions on Software Engineering*, vol. 49, no. 4, pp. 2149–2168, 2022.
- [21] H. Wang, C. Chen, Z. Xing, and J. Grundy, "Difftech: Differencing similar technologies from crowd-scale comparison discussions," *IEEE Transactions on Software Engineering*, vol. 48, no. 7, pp. 2399–2416, 2021.
- [22] F. Zhang, H. Niu, I. Keivanloo, and Y. Zou, "Expanding queries for code search using semantically related api class-names," *IEEE Transactions on Software Engineering*, vol. 44, no. 11, pp. 1070–1082, 2017.
- [23] P. Salza, C. Schwizer, J. Gu, and H. C. Gall, "On the effectiveness of transfer learning for code search," *IEEE Transactions on Software Engineering*, 2022.
- [24] S. Fatima, T. A. Ghaleb, and L. Briand, "Flakify: A black-box, language model-based predictor for flaky tests," *IEEE Transactions on Software Engineering*, 2022.
- [25] T. Ahmed, P. Devanbu, and A. A. Sawant, "Learning to find usages of library functions in optimized binaries," *IEEE Transactions on Software Engineering*, vol. 48, no. 10, pp. 3862–3876, 2021.
- [26] T. Ahmed, N. R. Ledesma, and P. Devanbu, "Synshine: improved fixing of syntax errors," *IEEE Transactions on Software Engineering*, vol. 49, no. 4, pp. 2169–2181, 2022.
- [27] J. Zhang, S. Liu, L. Gong, H. Zhang, Z. Huang, and H. Jiang, "Beqain: An effective and efficient identifier normalization approach with bert and the question answering system," *IEEE Transactions on Software Engineering*, 2022.
- [28] C. Pornprasit and C. K. Tantithamthavorn, "Deeplinedp: Towards a deep learning approach for line-level defect prediction," *IEEE Transactions on Software Engineering*, vol. 49, no. 1, pp. 84–98, 2022.
- [29] S. Yu, C. Fang, Q. Zhang, Z. Cao, Y. Yun, Z. Cao, K. Mei, and Z. Chen, "Mobile app crowdsourced test report consistency detection via deep image-and-text fusion understanding," *IEEE Transactions on Software Engineering*, vol. 49, no. 8, pp. 4115–4134, 2023.
- [30] C. Shi, B. Cai, Y. Zhao, L. Gao, K. Sood, and Y. Xiang, "Coss: Leveraging statement semantics for code summarization," *IEEE Transactions on Software Engineering*, vol. 49, no. 6, pp. 3472–3486, 2023.
- [31] M. Viggiano, D. Paas, C. Buzon, and C.-P. Bezemer, "Identifying similar test cases that are specified in natural language," *IEEE Transactions on Software Engineering*, vol. 49, no. 3, pp. 1027–1043, 2022.
- [32] C. Wang, X. Peng, Z. Xing, and X. Meng, "Beyond literal meaning: Uncover and explain implicit knowledge in code through wikipedia-based concept linking," *IEEE Transactions on Software Engineering*, vol. 49, no. 5, pp. 3226–3240, 2023.
- [33] B. Pathik and M. Sharma, "Source code change analysis with deep learning based programming model," *Automated Software Engineering*, vol. 29, pp. 1–25, 2022.
- [34] B. Li, Y. Wei, X. Sun, L. Bo, D. Chen, and C. Tao, "Towards the identification of bug entities and relations in bug reports," *Automated Software Engineering*, vol. 29, no. 1, p. 24, 2022.
- [35] Z. Liu, X. Xia, D. Lo, and J. Grundy, "Automatic, highly accurate app permission recommendation," *Automated Software Engineering*, vol. 26, pp. 241–274, 2019.
- [36] A. Ferrari and A. Esuli, "An nlp approach for cross-domain ambiguity detection in requirements engineering," *Automated Software Engineering*, vol. 26, no. 3, pp. 559–598, 2019.
- [37] P. Sangaroonsilp, M. Choetkiertikul, H. K. Dam, and A. Ghose, "An empirical study of automated privacy requirements classification in issue reports," *Automated Software Engineering*, vol. 30, no. 2, p. 20, 2023.
- [38] K. Kolthoff, C. Bartelt, and S. P. Ponzetto, "Data-driven prototyping via natural-language-based gui retrieval," *Automated software engineering*, vol. 30, no. 1, p. 13, 2023.
- [39] N. Borovits, I. Kumara, D. Di Nucci, P. Krishnan, S. D. Palma, F. Palomba, D. A. Tamburri, and W.-J. v. d. Heuvel, "Findici: Using machine learning to detect linguistic inconsistencies between code and natural language descriptions in infrastructure-as-code," *Empirical Software Engineering*, vol. 27, no. 7, p. 178, 2022.
- [40] A. Mahadi, N. A. Ernst, and K. Tongay, "Conclusion stability for natural language based mining of design discussions," *Empirical Software Engineering*, vol. 27, pp. 1–42, 2022.

- [41] C. Chen, Z. Xing, and Y. Liu, "What's spain's paris? mining analogical libraries from q&a discussions," *Empirical Software Engineering*, vol. 24, pp. 1155–1194, 2019.
- [42] Y. Li, M. Soliman, and P. Avgeriou, "Identifying self-admitted technical debt in issue tracking systems using machine learning," *Empirical Software Engineering*, vol. 27, no. 6, p. 131, 2022.
- [43] F. Calefato, F. Lanubile, F. Maiorano, and N. Novielli, "Sentiment polarity detection for software development," *Empirical Software Engineering*, vol. 23, pp. 1352–1382, 2018.
- [44] B. Xu, Z. Xing, X. Xia, D. Lo, and S. Li, "Domain-specific cross-language relevant question retrieval," *Empirical Software Engineering*, vol. 23, pp. 1084–1122, 2018.
- [45] D. Ye, L. Bao, Z. Xing, and S.-W. Lin, "Apreal: an api recognition and linking approach for online developer forums," *Empirical Software Engineering*, vol. 23, pp. 3129–3160, 2018.
- [46] C. Zhou, B. Li, X. Sun, and L. Bo, "Why and what happened? aiding bug comprehension with automated category and causal link identification," *Empirical Software Engineering*, vol. 26, no. 6, p. 118, 2021.
- [47] R. F. G. da Silva, C. K. Roy, M. M. Rahman, K. A. Schneider, K. Paixão, C. E. d. C. Dantas, and M. d. A. Maia, "Crokage: effective solution recommendation for programming tasks by leveraging crowd knowledge," *Empirical Software Engineering*, vol. 25, pp. 4707–4758, 2020.
- [48] Y. Li, S. Wang, W. Wang, T. N. Nguyen, Y. Wang, and X. Ye, "Rap4dq: Learning to recommend relevant api documentation for developer questions," *Empirical Software Engineering*, vol. 27, no. 1, p. 23, 2022.
- [49] H. Liang, D. Hang, and X. Li, "Modeling function-level interactions for file-level bug localization," *Empirical Software Engineering*, vol. 27, no. 7, p. 186, 2022.
- [50] Y. Li, M. Soliman, and P. Avgeriou, "Automatic identification of self-admitted technical debt from four different sources," *Empirical Software Engineering*, vol. 28, no. 3, p. 65, 2023.
- [51] X. Chen, P. Liao, Q. Kong, Y. Huang, and X. Zhou, "Studying differentiated code to support smart contract update," *Empirical Software Engineering*, vol. 28, no. 5, p. 119, 2023.
- [52] J. Lan, L. Gong, J. Zhang, and H. Zhang, "Btlink: automatic link recovery between issues and commits based on pre-trained bert model," *Empirical Software Engineering*, vol. 28, no. 4, p. 103, 2023.
- [53] R. Sepahvand, R. Akbari, and S. Hashemi, "Predicting the bug fixing time using word embedding and deep long short term memories," *IET Software*, vol. 14, no. 3, pp. 203–212, 2020.
- [54] J. Sun, X. Luo, W. Wang, Y. Gao, and W. Zhao, "Robust malware identification via deep temporal convolutional network with symmetric cross entropy learning," *IET Software*, vol. 17, no. 4, pp. 392–404, 2023.
- [55] H. Cho, S. Lee, and S. Kang, "Classifying issue reports according to feature descriptions in a user manual based on a deep learning model," *Information and Software Technology*, vol. 142, p. 106743, 2022.
- [56] M. Ochodek, S. Kopczyńska, and M. Staron, "Deep learning model for end-to-end approximation of cosmic functional size based on use-case names," *Information and Software Technology*, vol. 123, p. 106310, 2020.
- [57] W. Zhang, Z. Li, Q. Wang, and J. Li, "Finelocator: A novel approach to method-level fine-grained bug localization by query expansion," *Information and Software Technology*, vol. 110, pp. 121–135, 2019.
- [58] E. M. D. B. Fávero, D. Casanova, and A. R. Pimentel, "Se3m: A model for software effort estimation using pre-trained embedding models," *Information and Software Technology*, vol. 147, p. 106886, 2022.
- [59] X.-F. Qi, Y.-L. Hua, P. Wang, and Z.-Y. Wang, "Leveraging keyword-guided exploration to build test models for web applications," *Information and Software Technology*, vol. 111, pp. 110–119, 2019.
- [60] L. Ghadhab, I. Jenhani, M. W. Mkaouer, and M. B. Messaoud, "Augmenting commit classification by using fine-grained source code changes and a pre-trained deep neural language model," *Information and Software Technology*, vol. 135, p. 106566, 2021.
- [61] L. Wartschinski, Y. Noller, T. Vogel, T. Kehler, and L. Grunske, "Vudenc: vulnerability detection with deep learning on a natural codebase for python," *Information and Software Technology*, vol. 144, p. 106809, 2022.
- [62] L. Gomes, R. da Silva Torres, and M. L. Côrtes, "Bert-and tf-idf-based feature extraction for long-lived bug prediction in floss: a comparative study," *Information and Software Technology*, vol. 160, p. 107217, 2023.
- [63] X. Li, L. Liu, Z. Chen, Y. Liu, and H. Liu, "Describing the apis comprehensively: Obtaining the holistic representations from multiple modalities data for different tasks," *Information and Software Technology*, vol. 158, p. 107188, 2023.
- [64] K. Kaur and P. Kaur, "Sabdm: A self-attention based bidirectional-rnn deep model for requirements classification," *Journal of Software: Evolution and Process*, p. e2430, 2022.
- [65] X. Yang, X. Zhang, and Y. Tong, "Simplified abstract syntax tree based semantic features learning for software change prediction," *Journal of Software: Evolution and Process*, vol. 34, no. 4, p. e2445, 2022.
- [66] L. Liu, X. Li, Y. Liu, and H. Liu, "Application programming interface recommendation according to the knowledge indexed by app feature mined from app stores," *Journal of Software: Evolution and Process*, vol. 33, no. 11, p. e2380, 2021.
- [67] S. Majumdar, A. Bansal, P. P. Das, P. D. Clough, K. Datta, and S. K. Ghosh, "Automated evaluation of comments to aid software maintenance," *Journal of Software: Evolution and Process*, vol. 34, no. 7, p. e2463, 2022.
- [68] A. O. A. Semasaba, W. Zheng, X. Wu, S. A. Agyemang, T. Liu, and Y. Ge, "An empirical evaluation of deep learning-based source code vulnerability detection: Representation versus models," *Journal of software: evolution and process*, vol. 35, no. 11, p. e2422, 2023.
- [69] K. Abhinav, G. Kaur Bhatia, A. Dubey, S. Jain, and N. Bhardwaj, "Crowdassist: A multidimensional decision support system for crowd workers," *Journal of Software: Evolution and Process*, vol. 35, no. 6, p. e2404, 2023.
- [70] R. F. Silva, M. M. Rahman, C. E. Dantas, C. Roy, F. Khomh, and M. A. Maia, "Improved retrieval of programming solutions with code examples using a multi-featured score," *Journal of Systems and Software*, vol. 181, p. 111063, 2021.
- [71] H. Ruan, B. Chen, X. Peng, and W. Zhao, "Deeplink: Recovering issue-commit links based on deep learning," *Journal of Systems and Software*, vol. 158, p. 110406, 2019.
- [72] Q. Mi, Y. Hao, L. Ou, and W. Ma, "Towards using visual, semantic and structural features to improve code readability classification," *Journal of Systems and Software*, vol. 193, p. 111454, 2022.
- [73] C. Zhou, B. Li, and X. Sun, "Improving software bug-specific named entity recognition with deep neural network," *Journal of Systems and Software*, vol. 165, p. 110572, 2020.
- [74] N. Nazar, A. Aleti, and Y. Zheng, "Feature-based software design pattern detection," *Journal of Systems and Software*, vol. 185, p. 111179, 2022.
- [75] H. Chen, Y. Huang, Z. Liu, X. Chen, F. Zhou, and X. Luo, "Automatically detecting the scopes of source code comments," *Journal of Systems and Software*, vol. 153, pp. 45–63, 2019.
- [76] C. Li, L. Xu, M. Yan, and Y. Lei, "Tagdc: A tag recommendation method for software information sites with a combination of deep learning and collaborative filtering," *Journal of Systems and Software*, vol. 170, p. 110783, 2020.
- [77] D. Rajpathak, P. M. Peranandam, and S. Ramesh, "Automatic development of requirement linking matrix based on semantic similarity for robust software development," *Journal of Systems and Software*, vol. 186, p. 111211, 2022.
- [78] G. Yang, Y. Zhou, X. Chen, X. Zhang, T. Han, and T. Chen, "Exploitgen: Template-augmented exploit code generation based on codebert," *Journal of Systems and Software*, vol. 197, p. 111577, 2023.
- [79] C. Zhou, B. Li, X. Sun, and S. Yu, "Leveraging multi-level embeddings for knowledge-aware bug report reformulation," *Journal of Systems and Software*, vol. 198, p. 111617, 2023.
- [80] Y. Wang, L. Shi, M. Li, Q. Wang, and Y. Yang, "Detecting coreferent entities in natural language requirements," *Requirements Engineering*, vol. 27, no. 3, pp. 351–373, 2022.
- [81] H. Barzamini, M. Shahzad, H. Alhoori, and M. Rahimi, "A multi-level semantic web for hard-to-specify domain concept, pedestrian, in ml-based software," *Requirements Engineering*, pp. 1–22, 2022.
- [82] C. Velázquez-Rodríguez, D. Di Nucci, and C. De Roover, "A text classification approach to api type resolution for incomplete code snippets," *Science of Computer Programming*, vol. 227, p. 102941, 2023.
- [83] X. Li, L. Liu, Y. Liu, and H. Liu, "A lightweight api recommendation method for app development based on multi-objective evolutionary algorithm," *Science of Computer Programming*, vol. 226, p. 102927, 2023.

- [84] V. Geist, M. Moser, J. Pichler, R. Santos, and V. Wieser, "Leveraging machine learning for software redocumentation—a comprehensive comparison of methods in practice," *Software: Practice and Experience*, vol. 51, no. 4, pp. 798–823, 2021.
- [85] H. Sankar, V. Subramaniyaswamy, V. Vijayakumar, S. Arun Kumar, R. Logesh, and A. Umamakeswari, "Intelligent sentiment analysis approach using edge computing-based deep learning technique," *Software: Practice and Experience*, vol. 50, no. 5, pp. 645–657, 2020.
- [86] T. Kim, S. Kim, S. Park, and Y. Park, "Automatic recommendation to appropriate log levels," *Software: Practice and Experience*, vol. 50, no. 3, pp. 189–209, 2020.
- [87] H. Liu, M. Zhang, L. Liu, and Z. Liu, "A method to acquire cross-domain requirements based on syntax direct technique," *Software: Practice and Experience*, vol. 52, no. 1, pp. 236–253, 2022.
- [88] P. Bian, B. Liang, J. Huang, W. Shi, X. Wang, and J. Zhang, "Sink-finder: harvesting hundreds of unknown interesting function pairs with just one seed," in *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pp. 1101–1113, 2020.
- [89] N. D. Bui, Y. Yu, and L. Jiang, "Sar: learning cross-language api mappings with little knowledge," in *Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pp. 796–806, 2019.
- [90] S.-R. Lee, M.-J. Heo, C.-G. Lee, M. Kim, and G. Jeong, "Applying deep learning based automatic bug triager to industrial projects," in *Proceedings of the 2017 11th Joint Meeting on foundations of software engineering*, pp. 926–931, 2017.
- [91] Y. Chen, X. Yang, H. Dong, X. He, H. Zhang, Q. Lin, J. Chen, P. Zhao, Y. Kang, F. Gao, et al., "Identifying linked incidents in large-scale online service systems," in *Proceedings of the 28th ACM joint meeting on European software engineering conference and symposium on the foundations of software engineering*, pp. 304–314, 2020.
- [92] M. Pradel, G. Gousios, J. Liu, and S. Chandra, "Typewriter: Neural type prediction with search-based validation," in *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pp. 209–220, 2020.
- [93] J. Patra and M. Pradel, "Semantic bug seeding: a learning-based approach for creating realistic bugs," in *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pp. 906–918, 2021.
- [94] Z. Gao, X. Xia, D. Lo, J. Grundy, and T. Zimmermann, "Automating the removal of obsolete todo comments," in *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pp. 218–229, 2021.
- [95] X. Wang, J. Liu, L. Li, X. Chen, X. Liu, and H. Wu, "Detecting and explaining self-admitted technical debts with attention-based neural networks," in *Proceedings of the 35th IEEE/ACM international conference on automated software engineering*, pp. 871–882, 2020.
- [96] Q. Huang, X. Xia, Z. Xing, D. Lo, and X. Wang, "Api method recommendation without worrying about the task-api knowledge gap," in *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering*, pp. 293–304, 2018.
- [97] B. Xu, D. Ye, Z. Xing, X. Xia, G. Chen, and S. Li, "Predicting semantically linkable knowledge in developer online forums via convolutional neural network," in *Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering*, pp. 51–62, 2016.
- [98] J.-W. Lin, R. Jabbarvand, and S. Malek, "Test transfer across mobile apps through semantic mapping," in *2019 34th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pp. 42–53, IEEE, 2019.
- [99] V.-H. Le and H. Zhang, "Log-based anomaly detection without log parsing," in *2021 36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pp. 492–504, IEEE, 2021.
- [100] X. Luo, Y. Xue, Z. Xing, and J. Sun, "Prcbert: Prompt learning for requirement classification using bert-based pretrained language models," in *Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering*, pp. 1–13, 2022.
- [101] Z. Liu, X. Xia, M. Yan, and S. Li, "Automating just-in-time comment updating," in *Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering*, pp. 585–597, 2020.
- [102] Z. Li, T.-H. Chen, and W. Shang, "Where shall we log? studying and suggesting logging locations in code blocks," in *Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering*, pp. 361–372, 2020.
- [103] D. Clerissi, G. Denaro, M. Mobilio, and L. Mariani, "Plug the database & play with automatic testing: Improving system testing by exploiting persistent data," in *Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering*, pp. 66–77, 2020.
- [104] M. V. Phong, T. T. Nguyen, H. V. Pham, and T. T. Nguyen, "Mining user opinions in mobile app reviews: A keyword-based approach (t)," in *2015 30th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pp. 749–759, IEEE, 2015.
- [105] Y. Liu, M. Liu, X. Peng, C. Treude, Z. Xing, and X. Zhang, "Generating concept based api element comparison using a knowledge graph," in *Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering*, pp. 834–845, 2020.
- [106] F. Behrang and A. Orso, "Test migration between mobile apps with similar functionality," in *2019 34th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pp. 54–65, IEEE, 2019.
- [107] K. W. Nafi, T. S. Kar, B. Roy, C. K. Roy, and K. A. Schneider, "Clcdsa: cross language code clone detection using syntactical features and api documentation," in *2019 34th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pp. 1026–1037, IEEE, 2019.
- [108] J. Zhou, M. Pacheco, Z. Wan, X. Xia, D. Lo, Y. Wang, and A. E. Hassan, "Finding a needle in a haystack: Automated mining of silent vulnerability fixes," in *2021 36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pp. 705–716, IEEE, 2021.
- [109] E. Guglielmi, G. Rosa, S. Scalabrino, G. Bavota, and R. Oliveto, "Sorry, i don't understand: Improving voice user interface testing," in *Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering*, pp. 1–12, 2022.
- [110] Y. Ling, K. Wang, G. Bai, H. Wang, and J. S. Dong, "Are they toeing the line? diagnosing privacy compliance violations among browser extensions," in *Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering*, pp. 1–12, 2022.
- [111] Q. Huang, Z. Yuan, Z. Xing, X. Xu, L. Zhu, and Q. Lu, "Prompt-tuned code language model as a neural knowledge base for type inference in statically-typed partial code," in *Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering*, pp. 1–13, 2022.
- [112] H. Tian, X. Tang, A. Habib, S. Wang, K. Liu, X. Xia, J. Klein, and T. F. Bissyandé, "Is this change the answer to that problem? correlating descriptions of bug and code changes for evaluating patch correctness," in *Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering*, pp. 1–13, 2022.
- [113] J. Guo, J. Cheng, and J. Cleland-Huang, "Semantically enhanced software traceability using deep learning techniques," in *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, pp. 3–14, IEEE, 2017.
- [114] X. Ye, H. Shen, X. Ma, R. Bunescu, and C. Liu, "From word embeddings to document similarities for improved information retrieval in software engineering," in *Proceedings of the 38th international conference on software engineering*, pp. 404–415, 2016.
- [115] N. Zhang, C. Liu, X. Xia, C. Treude, Y. Zou, D. Lo, and Z. Zheng, "Shellfusion: answer generation for shell programming tasks via knowledge fusion," in *Proceedings of the 44th International Conference on Software Engineering*, pp. 1970–1981, 2022.
- [116] P. Chatterjee, K. Damevski, and L. Pollock, "Automatic extraction of opinion-based q&a from online developer chats," in *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*, pp. 1260–1272, IEEE, 2021.
- [117] J. Zhai, X. Xu, Y. Shi, G. Tao, M. Pan, S. Ma, L. Xu, W. Zhang, L. Tan, and X. Zhang, "Cpc: Automatically classifying and propagating natural language comments via program analysis," in *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering*, pp. 1359–1371, 2020.
- [118] L. Yang, J. Chen, Z. Wang, W. Wang, J. Jiang, X. Dong, and W. Zhang, "Semi-supervised log-based anomaly detection via probabilistic label estimation," in *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*, pp. 1448–1460, IEEE, 2021.
- [119] Y. Huo, Y. Su, H. Zhang, and M. R. Lyu, "Arclin: automated api mention resolution for unformatted texts," in *Proceedings of the 44th International Conference on Software Engineering*, pp. 138–149, 2022.

- [120] A. Ciborowska and K. Damevski, "Fast changeset-based bug localization with bert," in *Proceedings of the 44th International Conference on Software Engineering*, pp. 946–957, 2022.
- [121] A. M. Mir, E. Latoškinas, S. Proksch, and G. Gousios, "Type4py: Practical deep similarity learning-based type inference for python," in *Proceedings of the 44th International Conference on Software Engineering*, pp. 2241–2252, 2022.
- [122] J. Patra and M. Pradel, "Nalin: learning from runtime behavior to find name-value inconsistencies in jupyter notebooks," in *Proceedings of the 44th International Conference on Software Engineering*, pp. 1469–1481, 2022.
- [123] P. Liu, X. Zhang, M. Pistoia, Y. Zheng, M. Marques, and L. Zeng, "Automatic text input generation for mobile testing," in *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, pp. 643–653, IEEE, 2017.
- [124] T. D. Nguyen, A. T. Nguyen, H. D. Phan, and T. N. Nguyen, "Exploring api embedding for api usages and applications," in *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, pp. 438–449, IEEE, 2017.
- [125] H. Xia, Y. Zhang, Y. Zhou, X. Chen, Y. Wang, X. Zhang, S. Cui, G. Hong, X. Zhang, M. Yang, et al., "How android developers handle evolution-induced api compatibility issues: a large-scale study," in *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering*, pp. 886–898, 2020.
- [126] J. Lin, Y. Liu, Q. Zeng, M. Jiang, and J. Cleland-Huang, "Traceability transformed: Generating more accurate links with pre-trained bert models," in *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*, pp. 324–335, IEEE, 2021.
- [127] H. Wu, W. Deng, X. Niu, and C. Nie, "Identifying key features from app user reviews," in *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*, pp. 922–932, IEEE, 2021.
- [128] E. Dinella, G. Ryan, T. Mytkowicz, and S. K. Lahiri, "Toga: A neural method for test oracle generation," in *Proceedings of the 44th International Conference on Software Engineering*, pp. 2130–2141, 2022.
- [129] Z. Li, C. Luo, T.-H. Chen, W. Shang, S. He, Q. Lin, and D. Zhang, "Did we miss something important? studying and exploring variable-aware log abstraction," in *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*, pp. 830–842, IEEE, 2023.
- [130] Y. Huo, Y. Su, C. Lee, and M. R. Lyu, "Semparser: A semantic parser for log analytics," in *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*, pp. 881–893, IEEE, 2023.
- [131] C. Fang, Z. Liu, Y. Shi, J. Huang, and Q. Shi, "Functional code clone detection with syntax and semantics fusion learning," in *Proceedings of the 29th ACM SIGSOFT international symposium on software testing and analysis*, pp. 516–527, 2020.
- [132] A. Blasi, A. Goffi, K. Kuznetsov, A. Gorla, M. D. Ernst, M. Pezzè, and S. D. Castellanos, "Translating code comments to procedure specifications," in *Proceedings of the 27th ACM SIGSOFT international symposium on software testing and analysis*, pp. 242–253, 2018.
- [133] H. Wang, W. Qu, G. Katz, W. Zhu, Z. Gao, H. Qiu, J. Zhuge, and C. Zhang, "Jtrans: Jump-aware transformer for binary code similarity detection," in *Proceedings of the 31st ACM SIGSOFT International Symposium on Software Testing and Analysis*, pp. 1–13, 2022.
- [134] Y. Huang, Q. Kong, N. Jia, X. Chen, and Z. Zheng, "Recommending differentiated code to support smart contract update," in *2019 IEEE/ACM 27th International Conference on Program Comprehension (ICPC)*, pp. 260–270, IEEE, 2019.
- [135] Y. Guo, Y. Li, Z. Yang, and X. Chen, "What's inside my app?: Understanding feature redundancy in mobile apps," in *2018 IEEE/ACM 26th International Conference on Program Comprehension (ICPC)*, pp. 266–26610, IEEE, 2018.
- [136] C. Tao, Q. Zhan, X. Hu, and X. Xia, "C4: Contrastive cross-language code clone detection," in *2022 IEEE/ACM 30th International Conference on Program Comprehension (ICPC)*, pp. 413–424, IEEE, 2022.
- [137] K. Luong, M. Hadi, F. Thung, F. Fard, and D. Lo, "Arseek: Identifying api resource using code and discussion on stack overflow," in *2022 IEEE/ACM 30th International Conference on Program Comprehension (ICPC)*, pp. 331–342, IEEE, 2022.
- [138] I. N. B. Yusuf, L. Jiang, and D. Lo, "Accurate generation of trigger-action programs with domain-adapted sequence-to-sequence learning," in *2022 IEEE/ACM 30th International Conference on Program Comprehension (ICPC)*, pp. 99–110, IEEE, 2022.
- [139] T. Hey, J. Keim, A. Koziolok, and W. F. Tichy, "Norbert: Transfer learning for requirements classification," in *2020 IEEE 28th International Requirements Engineering Conference (RE)*, pp. 169–179, IEEE, 2020.
- [140] M. B. Hosseini, J. Heaps, R. Slavin, J. Niu, and T. Breaux, "Ambiguity and generality in natural language privacy policies," in *2021 IEEE 29th International Requirements Engineering Conference (RE)*, pp. 70–81, IEEE, 2021.
- [141] M. Ajagbe and L. Zhao, "Retraining a bert model for transfer learning in requirements engineering: A preliminary study," in *2022 IEEE 30th International Requirements Engineering Conference (RE)*, pp. 309–315, IEEE, 2022.
- [142] D. Martens and W. Maalej, "Extracting and analyzing context information in user-support conversations on twitter," in *2019 IEEE 27th International Requirements Engineering Conference (RE)*, pp. 131–141, IEEE, 2019.
- [143] S. Abualhaija, C. Arora, A. Sleimi, and L. C. Briand, "Automated question answering for improved understanding of compliance requirements: A multi-document study," in *2022 IEEE 30th International Requirements Engineering Conference (RE)*, pp. 39–50, IEEE, 2022.
- [144] L. Ackermann, J. Neuberger, and S. Jablonski, "Data-driven annotation of textual process descriptions based on formal meaning representations," in *International Conference on Advanced Information Systems Engineering*, pp. 75–90, Springer, 2021.
- [145] L. Burgueño, R. Clarisó, S. Gérard, S. Li, and J. Cabot, "An nlp-based architecture for the autocompletion of partial domain models," in *International Conference on Advanced Information Systems Engineering*, pp. 91–106, Springer, 2021.
- [146] X. Wang, P. Zhou, Y. Wang, X. Liu, J. Liu, and H. Wu, "Servicebert: A pre-trained model for web service tagging and recommendation," in *International Conference on Service-Oriented Computing*, pp. 464–478, Springer, 2021.
- [147] M. Liu, Y. Zhu, H. Xu, Z. Tu, and Z. Wang, "T2I2: A tiny three linear layers model for service mashup creation," in *Service-Oriented Computing: 19th International Conference, ICSOC 2021, Virtual Event, November 22–25, 2021, Proceedings 19*, pp. 317–331, Springer, 2021.
- [148] P. Mohapatra, Y. Deng, A. Gupta, G. Dasgupta, A. Paradkar, R. Mahindru, D. Rosu, S. Tao, and P. Aggarwal, "Domain knowledge driven key term extraction for it services," in *Service-Oriented Computing: 16th International Conference, ICSOC 2018, Hangzhou, China, November 12–15, 2018, Proceedings 16*, pp. 489–504, Springer, 2018.
- [149] F. Zampetti, A. Serebrenik, and M. Di Penta, "Automatically learning patterns for self-admitted technical debt removal," in *2020 IEEE 27th International conference on software analysis, evolution and reengineering (SANER)*, pp. 355–366, IEEE, 2020.
- [150] J. K. Siow, C. Gao, L. Fan, S. Chen, and Y. Liu, "Core: Automating review recommendation for code changes," in *2020 IEEE 27th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pp. 284–295, IEEE, 2020.
- [151] D. Wang, Y. Yu, S. Li, W. Dong, J. Wang, and L. Qing, "Mulcode: A multi-task learning approach for source code understanding," in *2021 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pp. 48–59, IEEE, 2021.
- [152] C. K. Saifullah, M. Asaduzzaman, and C. K. Roy, "Exploring type inference techniques of dynamically typed languages," in *2020 IEEE 27th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pp. 70–80, IEEE, 2020.
- [153] H. Kirinuki, S. Matsumoto, Y. Higo, and S. Kusumoto, "Web element identification by combining nlp and heuristic search for web testing," in *2022 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pp. 1055–1065, IEEE, 2022.
- [154] K. W. Nafi, M. Asaduzzaman, B. Roy, C. K. Roy, and K. A. Schneider, "Mining software information sites to recommend cross-language analogical libraries," in *2022 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pp. 913–924, IEEE, 2022.
- [155] H. Jiang, J. Zhang, X. Li, Z. Ren, and D. Lo, "A more accurate model for finding tutorial segments explaining apis," in *2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, vol. 1, pp. 157–167, IEEE, 2016.
- [156] J. Gu, P. Salza, and H. C. Gall, "Assemble foundation models for automatic code summarization," in *2022 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pp. 935–946, IEEE, 2022.

- [157] S. Wang, Y. Zhang, L. Bao, X. Xia, and M. Wu, "Vcmatch: A ranking-based approach for automatic security patches localization for oss vulnerabilities," in *2022 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pp. 589–600, IEEE, 2022.
- [158] I. C. Irsan, T. Zhang, F. Thung, K. Kim, and D. Lo, "Multi-modal api recommendation," in *2023 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pp. 272–283, IEEE, 2023.
- [159] S. Jahan and M. M. Rahman, "Towards understanding the impacts of textual dissimilarity on duplicate bug report detection," in *2023 IEEE international conference on software analysis, evolution and reengineering (SANER)*, pp. 25–36, IEEE, 2023.
- [160] Z. Han, X. Li, Z. Xing, H. Liu, and Z. Feng, "Learning to predict severity of software vulnerability using only vulnerability description," in *2017 IEEE International conference on software maintenance and evolution (ICSME)*, pp. 125–136, IEEE, 2017.
- [161] A. LeClair, Z. Eberhart, and C. McMillan, "Adapting neural text classification for improved software categorization," in *2018 IEEE international conference on software maintenance and evolution (IC-SME)*, pp. 461–472, IEEE, 2018.
- [162] W. Li, H. Qin, S. Yan, B. Shen, and Y. Chen, "Learning code-query interaction for enhancing code searches," in *2020 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pp. 115–126, IEEE, 2020.
- [163] E. Biswas, M. E. Karabulut, L. Pollock, and K. Vijay-Shanker, "Achieving reliable sentiment analysis in the software engineering domain using bert," in *2020 IEEE International conference on software maintenance and evolution (ICSME)*, pp. 162–173, IEEE, 2020.
- [164] T. Hey, F. Chen, S. Weigelt, and W. F. Tichy, "Improving traceability link recovery using fine-grained requirements-to-code relations," in *2021 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pp. 12–22, IEEE, 2021.
- [165] M. M. Rahman and C. Roy, "Effective reformulation of query for code search using crowdsourced knowledge and extra-large data analytics," in *2018 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pp. 473–484, IEEE, 2018.
- [166] T. Zhou, K. Liu, L. Li, Z. Liu, J. Klein, and T. F. Bissyandé, "Smartgift: Learning to generate practical inputs for testing smart contracts," in *2021 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pp. 23–34, IEEE, 2021.
- [167] C. Yu, G. Yang, X. Chen, K. Liu, and Y. Zhou, "Bashexplainer: Retrieval-augmented bash code comment generation based on fine-tuned codebert," in *2022 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pp. 82–93, IEEE, 2022.
- [168] O. Shuvo, P. Mahbub, and M. M. Rahman, "Recommending code reviews leveraging code changes with structured information retrieval," in *2023 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pp. 194–206, IEEE, 2023.
- [169] C. Huang, L. Yao, X. Wang, B. Benatallah, and Q. Z. Sheng, "Expert as a service: Software expert recommendation via knowledge domain embeddings in stack overflow," in *2017 IEEE International Conference on Web Services (ICWS)*, pp. 317–324, IEEE, 2017.
- [170] B. Huang, H. Dong, and A. Bouguettaya, "Conflict detection in iot-based smart homes," in *2021 IEEE International Conference on Web Services (ICWS)*, pp. 303–313, IEEE, 2021.
- [171] J. Tian, Z. Tu, Z. Wang, X. Xu, and M. Liu, "User intention recognition and requirement elicitation method for conversational ai services," in *2020 IEEE International Conference on Web Services (ICWS)*, pp. 273–280, IEEE, 2020.
- [172] X. Hu, L. Lu, and H. Wu, "A category aware non-negative matrix factorization approach for app permission recommendation," in *2020 IEEE International Conference on Web Services (ICWS)*, pp. 240–247, IEEE, 2020.
- [173] H. Yin, Y. Zheng, Y. Sun, and G. Huang, "An api learning service for inexperienced developers based on api knowledge graph," in *2021 IEEE International Conference on Web Services (ICWS)*, pp. 251–261, IEEE, 2021.
- [174] W. Yuan, Z. Feng, S. Chen, K. Huang, and J. Yao, "What biscuits to put in the basket? features prediction in release management for android system," in *2017 IEEE International Conference on Web Services (ICWS)*, pp. 73–80, IEEE, 2017.
- [175] Y. Li, X. Che, Y. Huang, J. Wang, S. Wang, Y. Wang, and Q. Wang, "A tale of two tasks: Automated issue priority prediction with deep multi-task learning," in *Proceedings of the 16th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, pp. 1–11, 2022.
- [176] R. Chen, S. Zhang, D. Li, Y. Zhang, F. Guo, W. Meng, D. Pei, Y. Zhang, X. Chen, and Y. Liu, "Logtransfer: Cross-system log anomaly detection for software systems with transfer learning," in *2020 IEEE 31st International Symposium on Software Reliability Engineering (ISSRE)*, pp. 37–47, IEEE, 2020.
- [177] W. Zheng, M. Zhang, H. Tang, Y. Cai, X. Chen, X. Wu, and A. O. A. Semasaba, "Automatically identifying bug reports with tactical vulnerabilities by deep feature learning," in *2021 IEEE 32nd International Symposium on Software Reliability Engineering (ISSRE)*, pp. 333–344, IEEE, 2021.
- [178] M. Kim, Y. Kim, and E. Lee, "A novel automatic query expansion with word embedding for ir-based bug localization," in *2021 IEEE 32nd International Symposium on Software Reliability Engineering (ISSRE)*, pp. 276–287, IEEE, 2021.
- [179] Z. Luo, L. Xu, Z. Xu, M. Yan, Y. Lei, and C. Li, "Contextual-semantic-aware linkable knowledge prediction in stack overflow via self-attention," in *2021 IEEE 32nd International Symposium on Software Reliability Engineering (ISSRE)*, pp. 115–126, IEEE, 2021.
- [180] P. Liguori, E. Al-Hossami, V. Orbinato, R. Natella, S. Shaikh, D. Cotroneo, and B. Cukic, "Evil: exploiting software via natural language," in *2021 IEEE 32nd International Symposium on Software Reliability Engineering (ISSRE)*, pp. 321–332, IEEE, 2021.
- [181] X. Xie, S. Jian, C. Huang, F. Yu, and Y. Deng, "Logrep: Log-based anomaly detection by representing both semantic and numeric information in raw messages," in *2023 IEEE 34th International Symposium on Software Reliability Engineering (ISSRE)*, pp. 194–206, IEEE, 2023.