**Selected Papers**

—-----------------------------

Format:

Keyword:(original paper number(not including snowballing) -> number after deleting the duplicated ones)

1. Paper1
2. Paper2

* snowballing

1. Paper3 (duplicated)

—------------------------------

Artificial Intelligence Technical Debt:(7->6)

1. Sculley, David, et al. "Hidden technical debt in machine learning systems." *Advances in neural information processing systems* 28 (2015): 2503-2511.
2. Sculley, David, et al. "Machine learning: The high interest credit card of technical debt." (2014).
3. Breck, Eric, et al. "The ML test score: A rubric for ML production readiness and technical debt reduction." *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017.

* Haakman, M. P. A. "Studying the Machine Learning Lifecycle and Improving Code Quality of Machine Learning Applications." (2020).

1. Tang, Yiming, et al. "An Empirical Study of Refactorings and Technical Debt in Machine Learning Systems." *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*. IEEE, 2021.

* Dilhara, Malinda, Ameya Ketkar, and Danny Dig. "Understanding Software-2.0: a study of machine learning library usage and evolution." *ACM Transactions on Software Engineering and Methodology (TOSEM)* 30.4 (2021): 1-42.

1. Foidl, Harald, Michael Felderer, and Stefan Biffl. "Technical debt in data-intensive software systems." *2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE, 2019.
2. Bogner, Justus, Roberto Verdecchia, and Ilias Gerostathopoulos. "Characterizing Technical Debt and Antipatterns in AI-Based Systems: A Systematic Mapping Study." *arXiv preprint arXiv:2103.09783* (2021).

* M. Felderer and R. Ramler, “Quality Assurance for AI-Based Systems: Overview and Challenges,” in Software Quality: Future Perspectives on Software Engineering Quality. SWQD 2021. Lecture Notes in Business Information Processing, vol. 404, Springer International Publishing, 2021.
* J. Siebert, L. Joeckel, J. Heidrich, K. Nakamichi, K. Ohashi, I. Namba, R. Yamamoto, and M. Aoyama, “Towards Guidelines for Assessing Qualities of Machine Learning Systems,” in Quality of Information and Communications Technology, Springer, 2020.
* Matthews, Jeanna. "Patterns and Anti-Patterns, Principles and Pitfalls: Accountability and Transparency in AI." *AI Magazine* 41.1 (2020): 82-89.

1. Alahdab, Mohannad, and Gül Çalıklı. "Empirical Analysis of Hidden Technical Debt Patterns in Machine Learning Software." *International Conference on Product-Focused Software Process Improvement*. Springer, Cham, 2019.

Machine Learning Technical Debt:(6->0)

1. Sculley, David, et al. "Hidden technical debt in machine learning systems." *Advances in neural information processing systems* 28 (2015): 2503-2511.
2. Sculley, David, et al. "Machine learning: The high interest credit card of technical debt." (2014).
3. Tang, Yiming, et al. "An Empirical Study of Refactorings and Technical Debt in Machine Learning Systems." *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*. IEEE, 2021.
4. Breck, Eric, et al. "The ML test score: A rubric for ML production readiness and technical debt reduction." *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017.
5. Alahdab, Mohannad, and Gül Çalıklı. "Empirical Analysis of Hidden Technical Debt Patterns in Machine Learning Software." *International Conference on Product-Focused Software Process Improvement*. Springer, Cham, 2019.
6. Foidl, Harald, Michael Felderer, and Stefan Biffl. "Technical debt in data-intensive software systems." *2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE, 2019.

Deep Learning Technical Debt:(8->2)

1. Liu, Jiakun, et al. "Is using deep learning frameworks free? characterizing technical debt in deep learning frameworks." *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Software Engineering in Society*. 2020.
2. Liu, Jiakun, et al. "An exploratory study on the introduction and removal of different types of technical debt in deep learning frameworks." *Empirical Software Engineering* 26.2 (2021): 1-36.
3. Sculley, David, et al. "Hidden technical debt in machine learning systems." *Advances in neural information processing systems* 28 (2015): 2503-2511.
4. Breck, Eric, et al. "The ML test score: A rubric for ML production readiness and technical debt reduction." *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017.
5. Tang, Yiming, et al. "An Empirical Study of Refactorings and Technical Debt in Machine Learning Systems." *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*. IEEE, 2021.
6. Sculley, David, et al. "Machine learning: The high interest credit card of technical debt." (2014).
7. Alahdab, Mohannad, and Gül Çalıklı. "Empirical Analysis of Hidden Technical Debt Patterns in Machine Learning Software." *International Conference on Product-Focused Software Process Improvement*. Springer, Cham, 2019.
8. Bogner, Justus, Roberto Verdecchia, and Ilias Gerostathopoulos. "Characterizing Technical Debt and Antipatterns in AI-Based Systems: A Systematic Mapping Study." *arXiv preprint arXiv:2103.09783* (2021).

Neural Network Technical Debt:(3->0)

1. Breck, Eric, et al. "The ML test score: A rubric for ML production readiness and technical debt reduction." *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017.
2. Liu, Jiakun, et al. "Is using deep learning frameworks free? characterizing technical debt in deep learning frameworks." *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Software Engineering in Society*. 2020.
3. Liu, Jiakun, et al. "An exploratory study on the introduction and removal of different types of technical debt in deep learning frameworks." *Empirical Software Engineering* 26.2 (2021): 1-36.

Data Science Technical Debt:(2->0)

1. Foidl, Harald, Michael Felderer, and Stefan Biffl. "Technical debt in data-intensive software systems." *2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE, 2019.
2. Bogner, Justus, Roberto Verdecchia, and Ilias Gerostathopoulos. "Characterizing Technical Debt and Antipatterns in AI-Based Systems: A Systematic Mapping Study." *arXiv preprint arXiv:2103.09783* (2021).

Artificial Intelligence Refactoring:(1->1)

1. Ross, Andrew Slavin, and Jessica Zosa Forde. "Refactoring Machine Learning." *Workshop on Critiquing and Correcting Trends in Machine Learning at NeuRIPS*. 2018.

Machine Learning Refactoring:(1->0)

1. Tang, Yiming, et al. "An Empirical Study of Refactorings and Technical Debt in Machine Learning Systems." *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*. IEEE, 2021.

Deep Learning Refactoring:(3->1)

1. Ross, Andrew Slavin, and Jessica Zosa Forde. "Refactoring Machine Learning." *Workshop on Critiquing and Correcting Trends in Machine Learning at NeuRIPS*. 2018.
2. Jebnoun, Hadhemi, et al. "The Scent of Deep Learning Code: An Empirical Study." *Proceedings of the 17th International Conference on Mining Software Repositories*. 2020.
3. Tang, Yiming, et al. "An Empirical Study of Refactorings and Technical Debt in Machine Learning Systems." *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*. IEEE, 2021.

Neural Network Refactoring:(0)

-

Data Science Refactoring:(0)

-

Artificial Intelligence Code Smell: (0)

-

Machine Learning Code Smell:(0)

-

Deep Learning Code Smell:(0)

-

Neural Network Code Smell:(0)

-

Data Science Code Smell:(0)

-

Artificial Intelligence Code Quality:(2->2)

1. Gonzalez, Danielle, Thomas Zimmermann, and Nachiappan Nagappan. "The State of the ML-universe: 10 Years of Artificial Intelligence & Machine Learning Software Development on GitHub." *Proceedings of the 17th International Conference on Mining Software Repositories*. 2020.

* Sun, Xiaobing, et al. "An empirical study on real bugs for machine learning programs." *2017 24th Asia-Pacific Software Engineering Conference (APSEC)*. IEEE, 2017.
* Thung, Ferdian, et al. "An empirical study of bugs in machine learning systems." *2012 IEEE 23rd International Symposium on Software Reliability Engineering*. IEEE, 2012.

1. Vinayagasundaram, B., and S. K. Srivatsa. "Software quality in artificial intelligence system." *Information Technology Journal* 6.6 (2007): 835-842.

Machine Learning Code Quality:(2->2)

1. Masuda, Satoshi, et al. "A survey of software quality for machine learning applications." *2018 IEEE International conference on software testing, verification and validation workshops (ICSTW)*. IEEE, 2018.
2. Santhanam, P. "Quality Management of Machine Learning Systems." *International Workshop on Engineering Dependable and Secure Machine Learning Systems*. Springer, Cham, 2020.

Deep Learning Code Quality:(4->2)

1. Jebnoun, Hadhemi, et al. "The Scent of Deep Learning Code: An Empirical Study." *Proceedings of the 17th International Conference on Mining Software Repositories*. 2020.
2. Liu, Jiakun, et al. "Is using deep learning frameworks free? characterizing technical debt in deep learning frameworks." *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Software Engineering in Society*. 2020.
3. Zhang, Tianyi, et al. "An empirical study of common challenges in developing deep learning applications." *2019 IEEE 30th International Symposium on Software Reliability Engineering (ISSRE)*. IEEE, 2019.
4. Arpteg, Anders, et al. "Software engineering challenges of deep learning." *2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE, 2018.

Neural Network Code Quality:(0)

-

Data Science Code Quality:(0)

-

Artificial Intelligence Coding Best Practice:(0)

-

Machine Learning Coding Best Practice:(7->6)

1. Allamanis, Miltiadis. "The adverse effects of code duplication in machine learning models of code." *Proceedings of the 2019 ACM SIGPLAN International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software*. 2019.
2. Braiek, Houssem Ben, and Foutse Khomh. "On testing machine learning programs." *Journal of Systems and Software* 164 (2020): 110542.
3. Weber, Thomas, Christina Winiker, and Heinrich Hussmann. "A Closer Look at Machine Learning Code." *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*. 2021.
4. Beam, Andrew L., Arjun K. Manrai, and Marzyeh Ghassemi. "Challenges to the reproducibility of machine learning models in health care." *Jama* 323.4 (2020): 305-306.
5. Sculley, David, et al. "Hidden technical debt in machine learning systems." *Advances in neural information processing systems* 28 (2015): 2503-2511.
6. Serban, Alex, et al. "Adoption and effects of software engineering best practices in machine learning." *Proceedings of the 14th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. 2020.
7. van Oort, Bart, et al. "The Prevalence of Code Smells in Machine Learning projects." *arXiv preprint arXiv:2103.04146* (2021).

Deep Learning Coding Best Practice:(4->1)

1. Jebnoun, Hadhemi, et al. "The Scent of Deep Learning Code: An Empirical Study." *Proceedings of the 17th International Conference on Mining Software Repositories*. 2020.
2. Arpteg, Anders, et al. "Software engineering challenges of deep learning." *2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE, 2018.
3. Zhang, Tianyi, et al. "An empirical study of common challenges in developing deep learning applications." *2019 IEEE 30th International Symposium on Software Reliability Engineering (ISSRE)*. IEEE, 2019.
4. Santhanam, P., Eitan Farchi, and Victor Pankratius. "Engineering reliable deep learning systems." *arXiv preprint arXiv:1910.12582* (2019).

* L. Floridi, "Establishing the rules for building trustworthy AI", Nature Machine Intelligence, v.1, pp. 261–262 (2019)

Neural Network Coding Best Practice:(0)

-

Data Science Coding Best Practice:(2->2)

1. Simmons, Andrew J., et al. "A large-scale comparative analysis of Coding Standard conformance in Open-Source Data Science projects." *Proceedings of the 14th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. 2020.
2. Brown, Sarah M. "Participatory Live Coding and Learning-Centered Assessment in Programming for Data Science." *ECMLPKDD 2021 Workshop TeachML*. 2021.

* Buitinck, Lars, et al. "API design for machine learning software: experiences from the scikit-learn project." *arXiv preprint arXiv:1309.0238* (2013).

Artificial Intelligence Coding Anti-pattern:(2->1)

1. Washizaki, Hironori, et al. "Machine Learning Architecture and Design Patterns." (2020).
2. Washizaki, Hironori, et al. "Studying software engineering patterns for designing machine learning systems." *2019 10th International Workshop on Empirical Software Engineering in Practice (IWESEP)*. IEEE, 2019.

Machine Learning Coding Anti-pattern:(3->0)

1. Washizaki, Hironori, et al. "Studying software engineering patterns for designing machine learning systems." *2019 10th International Workshop on Empirical Software Engineering in Practice (IWESEP)*. IEEE, 2019.
2. Washizaki, Hironori, et al. "Machine Learning Architecture and Design Patterns." (2020).
3. Tang, Yiming, et al. "An Empirical Study of Refactorings and Technical Debt in Machine Learning Systems." *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*. IEEE, 2021.

Deep Learning Coding Anti-pattern:(1->0)

1. Tang, Yiming, et al. "An Empirical Study of Refactorings and Technical Debt in Machine Learning Systems." *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*. IEEE, 2021.

Neural Network Coding Anti-pattern:(0)

Data Science Coding Anti-pattern:(1->1)

1. Rajbahadur, Gopi Krishnan, et al. "Pitfalls Analyzer: Quality Control for Model-Driven Data Science Pipelines." *2019 ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems (MODELS)*. IEEE, 2019.

Artificial Intelligence Common Coding Mistakes:(0)

-

Machine Learning Common Coding Mistakes:(3->1)

1. Sculley, David, et al. "Hidden technical debt in machine learning systems." *Advances in neural information processing systems* 28 (2015): 2503-2511.
2. Zhang, Jie M., et al. "Machine learning testing: Survey, landscapes and horizons." *IEEE Transactions on Software Engineering* (2020).
3. Zhang, Tianyi, et al. "An empirical study of common challenges in developing deep learning applications." *2019 IEEE 30th International Symposium on Software Reliability Engineering (ISSRE)*. IEEE, 2019.

Deep Learning Common Coding Mistakes:(5->4)

1. Humbatova, Nargiz, et al. "Taxonomy of real faults in deep learning systems." *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering*. 2020.
2. Zhang, Tianyi, et al. "An empirical study of common challenges in developing deep learning applications." *2019 IEEE 30th International Symposium on Software Reliability Engineering (ISSRE)*. IEEE, 2019.
3. Islam, Md Johirul, et al. "A comprehensive study on deep learning bug characteristics." *Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 2019.
4. Zhang, Yuhao, et al. "An empirical study on TensorFlow program bugs." *Proceedings of the 27th ACM SIGSOFT International Symposium on Software Testing and Analysis*. 2018.
5. Nikanjam, Amin, et al. "Faults in deep reinforcement learning programs: a taxonomy and a detection approach." *Automated Software Engineering* 29.1 (2022): 1-32.

Neural Network Common Coding Mistakes:(2->1)

1. Braiek, Houssem Ben, and Foutse Khomh. "TFCheck: A TensorFlow Library for Detecting Training Issues in Neural Network Programs." *2019 IEEE 19th International Conference on Software Quality, Reliability and Security (QRS)*. IEEE, 2019.
2. Zhang, Tianyi, et al. "An empirical study of common challenges in developing deep learning applications." *2019 IEEE 30th International Symposium on Software Reliability Engineering (ISSRE)*. IEEE, 2019.

Data Science Common Coding Mistakes:(0)

-