

Kenichi Matsumoto

List of Publications by Year in descending order

Source: <https://exaly.com/author-pdf/5945804/publications.pdf>

Version: 2024-02-01

99
papers

2,038
citations

623188

14
h-index

377514

34
g-index

104
all docs

104
docs citations

104
times ranked

1033
citing authors

#	ARTICLE	IF	CITATIONS
1	Predicting Defective Lines Using a Model-Agnostic Technique. IEEE Transactions on Software Engineering, 2022, 48, 1480-1496.	4.3	38
2	Code Reviews With Divergent Review Scores: An Empirical Study of the OpenStack and Qt Communities. IEEE Transactions on Software Engineering, 2022, 48, 69-81.	4.3	8
3	GitHub repositories with links to academic papers: Public access, traceability, and evolution. Journal of Systems and Software, 2022, 183, 111117.	3.3	11
4	GitHub sponsors. , 2022, , .		1
5	Analysis of Work Efficiency and Quality of Software Maintenance Using Cross-Company Dataset. IEICE Transactions on Information and Systems, 2021, E104.D, 76-90.	0.4	2
6	Anti-patterns in Modern Code Review: Symptoms and Prevalence. , 2021, , .		17
7	Understanding shared links and their intentions to meet information needs in modern code review:. Empirical Software Engineering, 2021, 26, 1.	3.0	13
8	SARS-CoV-2 infection initiates interleukin-17-enriched transcriptional response in different cells from multiple organs. Scientific Reports, 2021, 11, 16814.	1.6	43
9	How are project-specific forums utilized? A study of participation, content, and sentiment in the Eclipse ecosystem. Empirical Software Engineering, 2021, 26, 1.	3.0	5
10	Can we benchmark Code Review studies? A systematic mapping study of methodology, dataset, and metric. Journal of Systems and Software, 2021, 180, 111009.	3.3	9
11	Meta-analysis of single-cell RNA-seq data reveals phenotypic switching of immune cells in severe COVID-19 patients. Computers in Biology and Medicine, 2021, 137, 104792.	3.9	25
12	Automatic patch linkage detection in code review using textual content and file location features. Information and Software Technology, 2021, 139, 106637.	3.0	7
13	Expert Programmers Have Fine-Tuned Cortical Representations of Source Code. ENeuro, 2021, 8, ENEURO.0405-20.2020.	0.9	11
14	Conceptual Framework for Next-Generation Software Ecosystems. , 2021, , .		0
15	The Impact of Class Rebalancing Techniques on the Performance and Interpretation of Defect Prediction Models. IEEE Transactions on Software Engineering, 2020, 46, 1200-1219.	4.3	160
16	Wait for it: identifying "On-Hold" self-admitted technical debt. Empirical Software Engineering, 2020, 25, 3770-3798.	3.0	22
17	Automated Identification of On-hold Self-admitted Technical Debt. , 2020, , .		11
18	An Empirical Study of README contents for JavaScript Packages. IEICE Transactions on Information and Systems, 2019, E102.D, 280-288.	0.4	4

#	ARTICLE	IF	CITATIONS
19	Are Donation Badges Appealing?: A Case Study of Developer Responses to Eclipse Bug Reports. IEEE Software, 2019, 36, 22-27.	2.1	13
20	Sentiment Classification Using N-Gram Inverse Document Frequency and Automated Machine Learning. IEEE Software, 2019, 36, 65-70.	2.1	19
21	The review linkage graph for code review analytics: a recovery approach and empirical study. , 2019, , .		22
22	Mining Source Code Improvement Patterns from Similar Code Review Works. , 2019, , .		2
23	Towards Generation of Visual Attention Map for Source Code. , 2019, , .		3
24	Automatic Classifying Self-Admitted Technical Debt Using N-Gram IDF. , 2019, , .		6
25	The impact of human factors on the participation decision of reviewers in modern code review. Empirical Software Engineering, 2019, 24, 973-1016.	3.0	22
26	The Impact of Automated Parameter Optimization on Defect Prediction Models. IEEE Transactions on Software Engineering, 2019, 45, 683-711.	4.3	211
27	Cross project defect prediction using class distribution estimation and oversampling. Information and Software Technology, 2018, 100, 87-102.	3.0	40
28	Identifying Design and Requirement Self-Admitted Technical Debt Using N-gram IDF. , 2018, , .		17
29	How Do Gamification Rules and Personal Preferences Affect Coding?. , 2018, , .		4
30	How are IF-Conditional Statements Fixed Through Peer CodeReview?. IEICE Transactions on Information and Systems, 2018, E101.D, 2720-2729.	0.4	2
31	An empirical study of design discussions in code review. , 2018, , .		30
32	Do Review Feedbacks Influence to a Contributor's Time Spent on OSS Projects?. , 2018, , .		4
33	An Exploratory Study to Identify Similar Patches: A Case Study in Modern Code Review. , 2018, , .		1
34	Towards Smoother Library Migrations: A Look at Vulnerable Dependency Migrations at Function Level for npm JavaScript Packages. , 2018, , .		19
35	Catalogen: Generating Catalogs of Code Examples Collected from OSS. , 2018, , .		0
36	Analyzing Software Maintenance Cost Based on Work Efficiency and Unit Cost. , 2018, , .		0

#	ARTICLE	IF	CITATIONS
37	The impact of IR-based classifier configuration on the performance and the effort of method-level bug localization. Information and Software Technology, 2018, 102, 160-174.	3.0	21
38	Maintaining third-party libraries through domain-specific category recommendations. , 2018, , .		4
39	Extraction of Library Update History Using Source Code Reuse Detection. IEICE Transactions on Information and Systems, 2018, E101.D, 799-802.	0.4	0
40	Towards understanding an open-source bounty: Analysis of Bountysource. , 2017, , .		5
41	Examining Software Engineering Beliefs about System Testing Defects. IT Professional, 2017, 19, 58-64.	1.4	5
42	Analysis of Donations in the Eclipse Project. , 2017, , .		8
43	Which review feedback did long-term contributors get on OSS projects?. , 2017, , .		3
44	Understanding Key Features of High-Impact Bug Reports. , 2017, , .		11
45	Benchmarking IT operations cost based on working time and unit cost. Science of Computer Programming, 2017, 135, 75-87.	1.5	3
46	An Empirical Comparison of Model Validation Techniques for Defect Prediction Models. IEEE Transactions on Software Engineering, 2017, 43, 1-18.	4.3	351
47	A stability assessment of solution adaptation techniques for analogy-based software effort estimation. Empirical Software Engineering, 2017, 22, 474-504.	3.0	18
48	Bug or Not? Bug Report Classification Using N-Gram IDF. , 2017, , .		40
49	An Analysis of Library Rollbacks: A Case Study of Java Libraries. , 2017, , .		2
50	Using High-Rising Cities to Visualize Performance in Real-Time. , 2017, , .		10
51	Extracting Insights from the Topology of the JavaScript Package Ecosystem. , 2017, , .		5
52	How is IF Statement Fixed Through Code Review? A Case Study of Qt Project. , 2017, , .		3
53	Understanding When to Adopt a Library: A Case Study on ASF Projects. IFIP Advances in Information and Communication Technology, 2017, , 128-138.	0.5	5
54	EyeNav. , 2016, , .		4

#	ARTICLE	IF	CITATIONS
55	A Study of Redundant Metrics in Defect Prediction Datasets. , 2016, , .		24
56	Code review participation. , 2016, , .		7
57	Automated parameter optimization of classification techniques for defect prediction models. , 2016, , .		219
58	Comments on "Researcher Bias: The Use of Machine Learning in Software Defect Prediction". IEEE Transactions on Software Engineering, 2016, 42, 1092-1094.	4.3	58
59	Project IS^3: Incentive-Based Intelligent Intervention for Smart and Sustainable Society. , 2016, , .		3
60	Analysis of information system operation cost based on working time and unit cost. , 2016, , .		0
61	Influence of outliers on analogy based software development effort estimation. , 2016, , .		3
62	Unsupervised Bug Report Categorization Using Clustering and Labeling Algorithm. International Journal of Software Engineering and Knowledge Engineering, 2016, 26, 1027-1053.	0.6	21
63	Investigating and Projecting Population Structures in Open Source Software Projects: A Case Study of Projects in GitHub. IEICE Transactions on Information and Systems, 2016, E99.D, 1304-1315.	0.4	9
64	ROCAT on KATARIBE: Code Visualization for Communities. , 2016, , .		2
65	A hosting service of multi-language historage repositories. , 2016, , .		1
66	LSA-X: Exploiting Productivity Factors in Linear Size Adaptation for Analogy-Based Software Effort Estimation. IEICE Transactions on Information and Systems, 2016, E99.D, 151-162.	0.4	0
67	Towards Building API Usage Example Metrics. , 2016, , .		8
68	Case consistency. , 2015, , .		6
69	Characteristics of Sustainable OSS Projects: A Theoretical and Empirical Study. , 2015, , .		18
70	A Dataset of High Impact Bugs: Manually-Classified Issue Reports. , 2015, , .		43
71	The Impact of Mislabelling on the Performance and Interpretation of Defect Prediction Models. , 2015, , .		53
72	How we resolve conflict: an empirical study of method-level conflict resolution. , 2015, , .		10

#	ARTICLE	IF	CITATIONS
73	Benchmarking Software Maintenance Based on Working Time. , 2015, , .		6
74	Bug report recommendation for code inspection. , 2015, , .		2
75	Real-Time Monitoring of Neural State in Assessing and Improving Software Developers' Productivity. , 2015, , .		9
76	Quantifying programmers' mental workload during program comprehension based on cerebral blood flow measurement: a controlled experiment. , 2014, , .		45
77	Bugarium: 3d interaction for supporting large-scale bug repositories analysis. , 2014, , .		3
78	Automatic Unsupervised Bug Report Categorization. , 2014, , .		24
79	Kataribe: a hosting service of historage repositories. , 2014, , .		11
80	Software population pyramids. , 2014, , .		9
81	Impact Analysis of Granularity Levels on Feature Location Technique. Communications in Computer and Information Science, 2014, , 135-149.	0.4	7
82	An Authentication Method with Spatiotemporal Interval and Partial Matching. , 2013, , .		0
83	Fault-Prone Module Prediction Using a Prediction Model and Manual Inspection. , 2013, , .		1
84	Assessing the Cost Effectiveness of Fault Prediction in Acceptance Testing. IEEE Transactions on Software Engineering, 2013, 39, 1345-1357.	4.3	54
85	Incorporating Expert Judgment into Regression Models of Software Effort Estimation. , 2012, , .		5
86	Analysis of Attributes Relating to Custom Software Price. , 2012, , .		3
87	Nine Years Challenge of In-process Measurement Platform for Software Development Project: Distribution of a New Generation Platform and a Collaborative Research Proposal. , 2012, , .		0
88	Evaluation of Non Functional Requirements in a Request for Proposal (RFP). , 2012, , .		9
89	Analysis of the motivation of learners in the in-house training of programming in Japanese ICT industries. , 2011, , .		3
90	Source code comprehension strategies and metrics to predict comprehension effort in software maintenance and evolution tasks - an empirical study with industry practitioners. , 2011, , .		11

#	ARTICLE	IF	CITATIONS
91	Guilty or Not Guilty: Using Clone Metrics to Determine Open Source Licensing Violations. IEEE Software, 2011, 28, 42-47.	2.1	13
92	Identifying Services in Procedural Programs for Migrating Legacy System to Service Oriented Architecture. International Journal of Information Systems in the Service Sector, 2011, 3, 54-72.	0.2	3
93	An Exploratory Study on the Impact of Usage of Screenshot in Software Inspection Recording Activity. , 2011, , .		1
94	An Authentication Method Based on Spatiotemporal Information and Actions. Journal of Japan Society for Fuzzy Theory and Intelligent Informatics, 2011, 23, 874-881.	0.0	2
95	Comparative analysis of 2D games and artwork as the motivation to learn programming. , 2009, , .		3
96	Software development productivity of Japanese enterprise applications. Information Technology and Management, 2009, 10, 193-205.	1.4	13
97	The Transition of the Motivation of the Students in the Art Faculty to Learn Programming. , 2008, , .		2
98	Analyzing the transition of learners's motivation to learn programming. , 2008, , .		5
99	Analysis of the Relation between the Teaching Materials and Motivation in Programming Education. Conference on Software Engineering Education and Training, 2007, , .	0.0	5