

Empirical Software Engineering

Report 1: Searching and Classifying Empirical Studies (year 2018)

学 院: 软件学院
创建时间: 2021年5月10日
学生姓名: 李镔达、蒋祚竑、刘育麟、李泳劭
学 号: 181250060、181250066、181250090、181870108
组 号: 15组

2021 年 5 月 14 日

1. Search Process

1.1 Search Description

1.1.1 Plan description

Our plan is to complete the literature search of empirical studies in software engineering in 2018. The task of manually searching is delivered averagly to the group members and each group member should finish the search work and submit the result for the final completion of our presentation and report. The result should contain some basic informaiton about the studies we have chosen and the classification of them.

1.1.2 Search venues and database

According to the advice mentioned in the teaching lecture slides, three venues are chosen as our search venues, which are Empirical Software Engineering Journal(EMSE), International Symposium on Empirical Software Engineering and Measurement(ESEM) and International Conference on Evaluation and Assessment in Software Engineering(EASE), and the related databases are presented in the following table.

VENUE/JOURNAL	DATABASE
Empirical Software Engineering Journal(EMSE)	Springer https://www.springer.com/journal/10664
International Symposium on Empirical Software Engineering and Measurement(ESEM)	ACM digital library https://dl.acm.org
International Conference on Evaluation and Assessment in Software Engineering(EASE)	ACM digital library https://dl.acm.org

1.1.3 Selection and Classification criteria

Given that so many papers are published in one year, some of which do not meet our demand, we've made some inclusion criteria in the first place.

INCLUSION CRITERIA

- C1 Published in 2018
- C2 Written in English
- C3 Must use empirical research methods
- C4 No less than 6 pages
- C5 Must be full research paper

Any paper not conform to the requirements above would not be included.

We classified those papers according to the research methods applied in their studies. Since each of these method has some specific features, we would refer to these features as the criteria when classifying. And the features are as follows.

RESEARCH METHOD	FEATURES
Experiments	variable, comparison, repeat work
Case studies	certain phenomena, context, real
Survey	cluster, interview
Ethnographies	community culture, observation
Action Research	put thoughts into practice
Systematic reviews	summary pre work
Expert opinions	Interview domain expert

1.1.4 Execution process

step1 choosing database

After our discussion and consultation about suitable database, we've decided to use Springer and ACM digital library. The former is for journals, and the latter is for conferences. According to the venues we chose, Springer is used on EMSE searching, while ACM digital library is prepared for ESEM and EASE.

step2 screening & getting papers list and download by sci-hub

Referring to the inclusion criteria, some searching result were filtered down, and as a result, the works in ESEM have been reduced to 41 from original 59. After screening, we attempted to get the papers list and started downloading.

For ESEM, we set the search string and conditions to get the target result. The string is like: [[All: empirical] OR [All: evidence]] AND [[All: research] OR [All: method] OR [All: study]] AND [[All: "software engineering"] OR [All: engineer]] AND [All: 2018] AND [Publication Date:(01/01/2018 TO 12/31/2018)]. For EASE and ESME, we would get all paper's doi from proceeding page and then download.

step3 Categorizing

As the second table in 1.1.3 has presented, seven research methods were reserved for the classification, including Experiments, Case studies, Survey, Ethnographies, Action Research, Systematic reviews and Expert opinions. And they are regarded as important reference when categorizing.

We would check the key words and sentences in the abstract at first. to make the judgement more precise, the next step is to scan the whole article and classified them according to the features mentioned above. If there exists some conditions when the paper is hard to classify, we would mark them and discuss together with other group members.

1.2 Problems and experience

Problems:

problem1: Some papers use multiple research methods that it's hard to define the specific research method. For example, some papers involves both survey and case study, in which case it is not easy to make a conclusion.

solution: We would check the research detail and find the main research method after discussion.

problem2: some papers focus on introducing an algorithm or a concrete tech, like a test framework

solution: These kind of papers usually use experiment to validate their algorithm/tech, so classify them into experiment

Experience:

Our experience is mainly about survey. Survey usually plays a role as an additional research method for case study, because survey can offer perception from practitioners. And practitioners' insight help researcher switch their perspective or fullfill their evidence.

2. Search Result

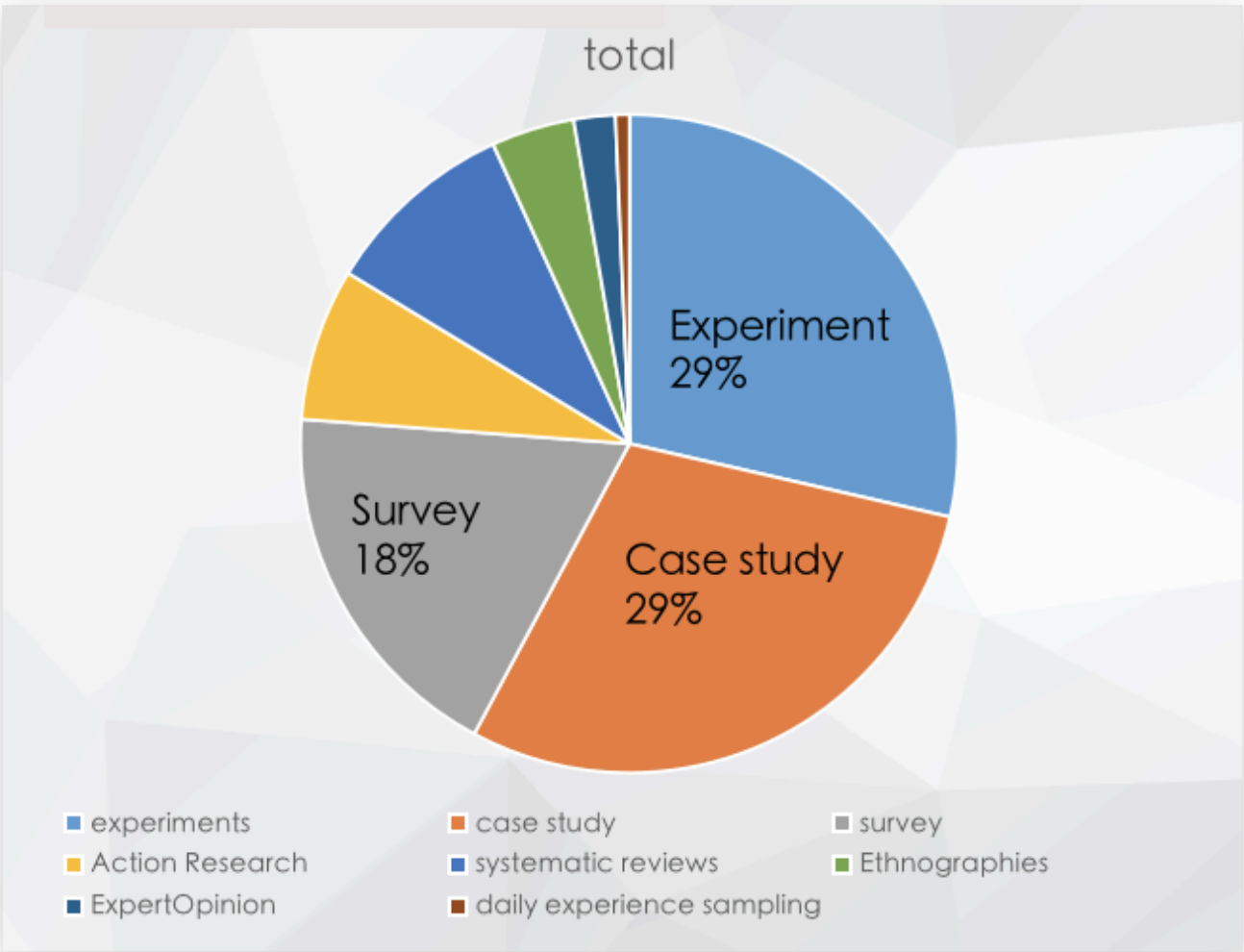
2.1 Statistic Summary

VENUES	COUNT
EASE	26
EMSE	80
ESEM	41
Total	147

Table1: sum of venues

	EASE	EMSE	ESEM	TOTAL
Experiments	6	18	13	37
Case studies	6	25	12	43
Survey	7	13	7	27
Action research	2	6	3	11
Systematic literature review	2	11	1	14
Ethnographies	2	3	1	6
Expert opinion	0	3	0	3
other/can not judge	1	1	4	6
total	26	80	41	147

Table2: Methd statistics



2.2 Study list

2.2.1 ESEM

Table 1 Method categorizing studies

EMPIRICAL METHOD	EMPIRICAL STUDIES	COUNT
Experiments	[1],[8],[10],[14],[15],[17],[19],[26],[27],[30],[31],[32],[38]	13
Case studies	[2],[3],[4],[5],[6],[9],[18],[28],[34],[35],[37],[41]	12
Survey	[12],[13],[16],[20],[22],[24],[29]	7
Action research	[21],[23],[33]	3
Systematic literature review	[40]	1
Ethnographies	[7]	1
Expert opinion		0
other/can not judge	[11],[25],[36],[39]	4

Table 2 Paper citation and their applied methods

	PAPER CITATION	METHOD
1	Comparing techniques for aggregating interrelated replications in software engineering Adrian Santos, Natalia Juristo 10.1145/3239235.3239239	Experiments
2	An empirical investigation of transferring research to software technology innovation: a case of data-driven national security software Mansoor Zahedi, Muhammad Ali Babar, Brenton Cooper 10.1145/3239235.3239241	Case studies
3	An empirical study of WIP in kanban teams Dag I. K. Sjøberg 10.1145/3239235.3239238	Case studies
4	An empirical study of design discussions in code review Farida El Zanaty, Toshiki Hirao, Shane McIntosh, Akinori Ihara, Kenichi Matsumoto 10.1145/3239235.3239525	Case studies
5	Needs and challenges for a platform to support large-scale requirements engineering: a multiple-case study Davide Fucci, Cristina Palomares, Xavier Franch, Dolors Costal et al.	Case studies

10.1145/3239235.3240498

- | | | |
|----|--|---------------------------|
| 6 | Understanding what industry wants from requirements engineers: an exploration of RE jobs in Canada
Chong Wang, Pengwei Cui, Maya Daneva, Mohamad Kassab
10.1145/3239235.3268916 | Case studies |
| 7 | An exploratory study of software sustainability dimensions and characteristics: end user perspectives in the kingdom of Saudi Arabia (KSA)
Sulaiman Aljarallah, Russell Lock
10.1145/3239235.3239240 | Ethnography |
| 8 | An empirical study of inadequate and adequate test suite reduction approaches
Carmen Coviello, Simone Romano, Giuseppe Scanniello
10.1145/3239235.3240497 | Experiments |
| 9 | An empirical study of process policies and metrics to manage productivity and quality for maintenance of critical software systems at the jet propulsion laboratory
Dan Port, Bill Taber
10.1145/3239235.3268925 | Case studies |
| 10 | The effect of noise on software engineers' performance
Simone Romano, Giuseppe Scanniello, Davide Fucci, Natalia Juristo, Burak Turhan
10.1145/3239235.3240496 | Experiments |
| 11 | Using experience sampling to link software repositories with emotions and work well-being
Miikka Kuutila, Mika V. Mäntylä, Maëlick Claes, Marko Elovainio, Bram Adams
10.1145/3239235.3239245 | Daily experience sampling |
| 12 | Understanding the Software Development Practices of Blockchain Projects: A Survey
Partha Chakraborty, Rifat Shahriyar, Anindya Iqbal, Amiangshu Bosu
10.1145/3239235.3240298 | Survey |
| 13 | Software analytics in continuous delivery: a case study on success factors
Hennie Huijgens, Davide Spadini, Dick Stevens, Niels Visser, Arie van Deursen | Survey |

10.1145/3239235.3240505

- | | | |
|----|--|-----------------|
| 14 | A longitudinal cohort study on the retainment of test-driven development
Davide Fucci, Simone Romano, Maria Teresa Baldassarre, Danilo Caivano et al.
10.1145/3239235.3240502 | Experiments |
| 15 | Applying pattern-driven maintenance: a method to prevent latent unhandled exceptions in web applications
Diogo S. Mendonça, Tarcila G. da Silva, Daniel Ferreira de Oliveira et al.
10.1145/3239235.3268924 | Experiments |
| 16 | Defining, measuring and monitoring IT service goals and strategies: preliminary results and pitfalls from a qualitative study with IT service managers
Bianca Trinkenreich, Tayana Conte, Monalessa Perini Barcellos, Gleison Santos
10.1145/3239235.3268919 | Survey |
| 17 | Simultaneous measurement of program comprehension with fMRI and eye tracking: a case study
Norman Peitek, Janet Siegmund, Chris Parnin et al.
10.1145/3239235.3240495 | Experiments |
| 18 | Decision making and visualizations based on test results
Per Erik Strandberg, Wasif Afzal, Daniel Sundmark
10.1145/3239235.3268921 | Case studies |
| 19 | Improving problem identification via automated log clustering using dimensionality reduction
Carl Martin Rosenberg, Leon Moonen
10.1145/3239235.3239248 | Experiments |
| 20 | Implementing agile practices: the experience of TSol
María Cecilia Bastarrica, Germán Espinoza, Jacqueline Marín
10.1145/3239235.3268918 | Survey |
| 21 | What do concurrency developers ask about?: a large-scale study using stack overflow
Syed Ahmed, Mehdi Bagherzadeh
10.1145/3239235.3239524 | Action research |

22	Development processes and practices in a small but growing software industry: a practitioner survey in New Zealand Di Wang, Matthias Galster 10.1145/3239235.3268926	Survey
23	Architecture, technologies and challenges for cyber-physical systems in industry 4.0: a systematic mapping study Florian Hofer 10.1145/3239235.3239242	Action research
24	Building a collaborative culture: a grounded theory of well succeeded devops adoption in practice Welder Pinheiro Luz, Gustavo Pinto, Rodrigo Bonifácio 10.1145/3239235.3240299	Survey
25	Revisiting the size effect in software fault prediction models Amjed Tahir, Kwabena E. Bennin, Stephen G. MacDonell, Stephen Marsland 10.1145/3239235.3239243	can not judge
26	Prediction of relatedness in stack overflow: deep learning vs. SVM: a reproducibility study Bowen Xu, Amirreza Shirani, David Lo, Mohammad Amin Alipour 10.1145/3239235.3240503	Experiments
27	Are mutants really natural?: a study on how "naturalness" helps mutant selection Matthieu Jimenez, Thiery Titchou Checkam, Maxime Cordy, Mike Papadakis et al. 10.1145/3239235.3240500	Experiments
28	Computer games are serious business and so is their quality: particularities of software testing in game development from the perspective of practitioners Ronnie E. S. Santos, Cleyton V. C. Magalhães, Luiz Fernando Capretz et al. 10.1145/3239235.3268923	Case studies
29	Relationship between geographical location and evaluation of developer contributions in github Ayushi Rastogi, Nachiappan Nagappan, Georgios Gousios, André van der Hoek 10.1145/3239235.3240504	Survey

30	Calibrating use case points using bayesian analysis Kan Qi, Anandi Hira, Elaine Venson, Barry W. Boehm 10.1145/3239235.3239236	Experiments
31	No search allowed: what risk modeling notation to choose? Katsiaryna Labunets 10.1145/3239235.3239247	Experiments
32	identifying unmaintained projects in github Jailton Coelho, Marco Tulio Valente, Luciana L. Silva, Emad Shihab 10.1145/3239235.3240501	Experiments
33	The most common causes and effects of technical debt: first results from a global family of industrial surveys Nicolli Rios, Rodrigo Oliveira Spínola, Manoel Mendonça, Carolyn Seaman 10.1145/3239235.3268917	Action research
34	Vulnerable open source dependencies: counting those that matter Ivan Pashchenko, Henrik Plate, Serena Elisa Ponta, Antonino Sabetta, Fabio Massacci 10.1145/3239235.3268920	Case studies
35	Are 20% of files responsible for 80% of defects? Neil Walkinshaw, Leandro Minku 10.1145/3239235.3239244	Case studies
36	Automatic topic classification of test cases using text mining at an Android smartphone vendor Junji Shimagaki, Yasutaka Kamei, Naoyasu Ubayashi, Abram Hindle 10.1145/3239235.3268927	can not judge
37	A scalable and efficient approach for compiling and analyzing commit history Pooyan Behnamghader, Patavee Meemeng, Iordanis Fostiropoulos, Di Huang et al. 10.1145/3239235.3239237	Case studies
38	Assessing the effect of data transformations on test suite compilation Panagiotis Stratis, Vanya Yaneva, Ajitha Rajan 10.1145/3239235.3240499	Experiments
39	The birth, growth, death and rejuvenation of software maintenance communities	can not judge

Qiong Feng, Yuanfang Cai, Rick Kazman, Ran Mo
10.1145/3239235.3239246

40	Is there a "golden" feature set for static warning identification?: an experimental evaluation Junjie Wang, Song Wang, Qing Wang 10.1145/3239235.3239523	Systematic literature review
41	Speeding up mutation testing via the cloud: lessons learned for further optimisations Sten Vercammen, Serge Demeyer, Markus Borg, Sigrid Eldh 10.1145/3239235.3240506	Case studies

2.2.2 EASE

Table 1 Method categorizing studies

EMPIRICAL METHOD	EMPIRICAL STUDIES	COUNT
Experiments	[1],[5],[10],[14],[22],[23]	6
Case studies	[2],[6],[9],[15],[24],[25]	6
Survey	[11],[12],[16],[17],[20],[21],[26]	7
Action research	[3],[13]	2
Systematic literature review	[4],[18]	2
Ethnographies	[7],[8]	2
Expert opinion		0
other/can not judge	[19]	1

Table 2 Paper citation and their applied methods

	PAPER CITATION	COUNT
1	Interrelating Use Cases and Associated Requirements by Links: An Eye Tracking Study on the Impact of Different Linking Variants on the Reading Behavior Oliver Karras, Alexandra Risch, Kurt Schneider 10.1145/3210459.3210460	Experiments
2	Construct Validity in Software Engineering Research and Software	Case study

Metrics

Paul Ralph, Ewan Tempero

10.1145/3210459.3210461

- | | | |
|----|---|-------------------------------|
| 3 | The Role of Rapid Reviews in Supporting Decision-Making in Software Engineering Practice
Bruno Cartaxo, Gustavo Pinto, Sergio Soares
10.1145/3210459.3210462 | Action Research |
| 4 | On Different Search Methods for Systematic Literature Reviews and Maps: Experiences from a Literature Search on Validation and Verification of Emergent Behavior
Jennifer Brings, Marian Daun, Markus Kempe, Thorsten Weyer
10.1145/3210459.3210463 | Systematic literature reviews |
| 5 | Using reasoning markers to select the more rigorous software practitioners' online content when searching for grey literature
Ashley Williams | Experiments |
| 6 | DevOps Capabilities, Practices, and Challenges: Insights from a Case Study
Mali Senapathi, Jim Buchan, Hady Osman
10.1145/3210459.3210465 | Case study |
| 7 | Stack Overflow
Amjed Tahir, Aiko Yamashita, Sherlock Licorish, Jens Dietrich, Steve Counsell
10.1145/3210459.3210466 | Ethnographies |
| 8 | Linking User Requests, Developer Responses and Code Changes: Android OS Case Study
Sherlock A. Licorish, Elijah Zolduoarrati, Nigel Stanger
10.1145/3210459.3210467 | Ethnographies |
| 9 | Using Bayesian Network to estimate the value of decisions within the context of Value-Based Software Engineering
Emilia Mendes, Mirko Perkusich, Vitor Freitas, João Nunes
10.1145/3210459.3210468 | Case study |
| 10 | Bug Localization with Semantic and Structural Features using Convolutional Neural Network and Cascade Forest
Yan Xiao, Jacky Keung, Qing Mi, Kwabena E. Bennin
10.1145/3210459.3210469 | Experiments |

11	Satisfaction, Practices, and Influences in Agile Software Development Martin Kropp, Andreas Meier, Craig Anslow, Robert Biddle 10.1145/3210459.3210470	Survey
12	Task Interruption in Software Development Projects: What Makes some Interruptions More Disruptive than Others? Zahra Shakeri Hossein Abad, Oliver Karras, Kurt Schneider, Ken Barker, Mike Bauer 10.1145/3210459.3210471	Survey
13	Evolving Project based Learning to Suit Diverse Student Cohorts Charles Thevathayan 10.1145/3210459.3210472	Action Research
14	An Inception Architecture-Based Model for Improving Code Readability Classification Qing Mi, Jacky Keung, Yan Xiao, Solomon Mensah, Xiupei Mei 10.1145/3210459.3210473	Experiments
15	How do Secondary Studies in Software Engineering report Automated Searches?: A Preliminary Analysis Paramvir Singh, Matthias Galster, Karanpreet Singh 10.1145/3210459.3210474	Case study
16	Do software engineering practitioners cite research on software testing in their online articles?: A preliminary survey. Ashley Williams 10.1145/3210459.3210475	Survey
17	Power and Politics of User Involvement in Software Development Muneera Bano, Didar Zowghi, Francesca da Rimini 10.1145/3210459.3210477	Survey
18	A Systematic Mapping Study On Software Startups Education Rafael Chanin, Afonso Sales, Leandro Pompermaier, Rafael Prikladnicki 10.1145/3210459.3210478	Systematic literature reviews
19	A Process for Evidence-Based Engineering of Domain-Specific Languages Michael Felderer, Fabian Jeschko 10.1145/3210459.3210479	can not judge

20	Two Sides of the Same Coin: Software Developers' Perceptions of Task Switching and Task Interruption Zahra Shakeri Hossein Abad, Mohammad Noaeen, Didar Zowghi, Behrouz H. Far, Ken Barker 10.1145/3210459.3214170	Survey
21	Toward Understanding IoT Developers in Chinese Startups Rui Zhang, Genying Xie 10.1145/3210459.3216093	Survey
22	Usability Evaluation Framework for Mobile Apps using Code Analysis Neeraj Mathur, Sai Anirudh Karre, Y. Raghu Reddy 10.1145/3210459.3210480	Experiments
23	Assessing the Effect of Device-Based Test Scheduling on Heterogeneous Test Suite Execution Panagiotis Stratis, Gordon Brown 10.1145/3210459.3210481	Experiments
24	Leveraging the Mob Mentality: An Experience Report on Mob Programming Jim Buchan, Mark Pearl 10.1145/3210459.3210482	Case study
25	Why Johnny Can't Store Passwords Securely?: A Usability Evaluation of Bouncycastle Password Hashing Chamila Wijayarathna, Nalin A. G. Arachchilage 10.1145/3210459.3210483	Case study
26	Why developers cannot embed privacy into software systems?: An empirical investigation Awanthika Senarath, Nalin A. G. Arachchilage 10.1145/3210459.3210484	Survey

2.2.3 EMSE

Table 1 Method categorizing studies

EMPIRICAL METHOD	EMPIRICAL STUDIES	COUNT
Experiments	[1],[12],[18],[24],[25],[26],[27],[29],[38],[40],[43],[50],[51],[52],[57],[67],[78]	18
Case studies	[3],[5],[6],[8],[10],[13],[15],[17],[23],[35],[44],[46],[54],[58],[59],[63],[64],[65],[66],[70],[71],[72],[75],[77],[80]	25
Survey	[9],[11],[16],[19],[33],[37],[39],[41],[68],[69],[73],[74],[79]	13
Action research	[14],[30],[36],[49],[60],[61]	6
Systematic literature review	[7],[21],[28],[32],[34],[42],[45],[47],[53],[55],[56]	11
Ethnographies	[4],[22],[62]	3
Expert opinion	[2],[20],[31]	3
other/can not judge	[76]	1

Table 2 Paper citation and their applied methods

	PAPER CITATION	METHOD
1	A comprehensive study of pseudo-tested methods Oscar Luis Vera-Perez, Benjamin Danglot, Martin Monperrus, Benoit Baudry 10.1007/s10664-018-9653-2	Experiments
2	A systematic approach for performance assessment using process mining Simona Bernardi, Juan L. Dominguez, Abel Gomez, Christophe Joubert, Jose Merseguer, Diego Perez-Palacin, Jose I. Requeno, Alberto Romeu 10.1007/s10664-018-9606-9	Expert Opinions
3	Alleviating patch overfitting with automatic test generation: a study of feasibility and effectiveness for the Nopol repair system Zhongxing Yu, Matias Martinez, Benjamin Danglot, Thomas Durieux, Martin Monperrus 10.1007/s10664-018-9619-4	Case study
4	An empirical analysis of the transition from Python 2 to Python 3	Ethnography

	Brian A. Malloy, James F. Power 10.1007/s10664-018-9637-2	
5	An empirical assessment of best-answer prediction models in technical Q&A sites Fabio Calefato, Filippo Lanubile, Nicole Novielli 10.1007/s10664-018-9642-5	Case study
6	An empirical comparison of dependency network evolution in seven software packaging ecosystems Alexandre Decan, Tom Mens, Philippe Grosjean 10.1007/s10664-017-9589-y	Case study
7	An empirical study of Android Wear user complaints Suhaib Mujahid, Giancarlo Sierra, Rabe Abdalkareem, Emad Shihab, Weiyi Shang 10.1007/s10664-018-9615-8	Systematic literature reviews
8	An empirical study of architecting for continuous delivery and deployment Mojtaba Shahin & Mansooreh Zahedi & Muhammad Ali Babar & Liming Zhu 10.1007/s10664-018-9651-4	Case study
9	An empirical study of game reviews on the Steam platform Dayi Lin, Cor-Paul Bezemer, Ying Zou, Ahmed E. Hassan 10.1007/s10664-018-9627-4	Survey
10	An empirical study of patch uplift in rapid release development pipelines Marco Castelluccio, Le An, Foutse Khomh 10.1007/s10664-018-9665-y	Case Study
11	An empirical study on the issue reports with questions raised during the issue resolving process Yonghui Huang, Daniel Alencar da Costa, Feng Zhang, Ying Zou 10.1007/s10664-018-9636-3	Survey
12	An ensemble-based model for predicting agile software development effort Onkar Malgonde, Kaushal Chari 10.1007/s10664-018-9647-0	Experiments
13	An industrial case study on the use of UML in software maintenance	Case study

and its perceived benefits and hurdles

Ana M. Fernández-Sáez & Michel R. V. Chaudron & Marcela Genero

10.1007/s10664-018-9599-4

14	An investigation of misunderstanding code patterns in C open-source software projects Flavio Medeiros, Gabriel Lima, Guilherme Amaral, Sven Apel, Christian Kastner, Marcio Ribeiro, Rohit Gheyi 10.1007/s10664-018-9666-x	Action Research
15	APIReal: an API recognition and linking approach for online developer forums Deheng Ye, Lingfeng Bao, Zhenchang Xing, Shang-Wei Lin 10.1007/s10664-018-9608-7	Case study
16	App store mining is not enough for app improvement Maleknaz Nayebi, Henry Cho, Guenther Ruhe 10.1007/s10664-018-9601-1	Survey
17	Augmenting and structuring user queries to support efficient free-form code search Raphael Sirres, Tegawende F. Bissyande, Dongsun Kim, David Lo, Jacques Klein, Kisub Kim, Yves Le Traon 10.1007/s10664-017-9544-y	Case study
18	Balancing the trade-off between accuracy and interpretability in software defect prediction Toshiki Mori & Naoshi Uchihira 10.1007/s10664-018-9638-1	Experiments
19	Categorizing the Content of GitHub README Files Gede Artha Azriadi Prana, Christoph Treude, Ferdian Thung, Thushari Atapattu, David Lo 10.1007/s10664-018-9660-3	Survey
20	Commenting source code: is it worth it for small programming tasks? Sebastian Nielebock, Dariusz Krolkowski, Jacob Kruger, Thomas Leich, Frank Ortmeier 10.1007/s10664-018-9664-z	Expert Opinions
21	Comparing the influence of using feature-oriented programming and conditional compilation on comprehending feature-oriented software Alcemir Rodrigues Santos, Ivan do Carmo Machado, Eduardo Santana	Systematic literature reviews

de Almeida, Janet Siegmund, Sven Apel

10.1007/s10664-018-9658-x

- | | | |
|----|---|-------------------------------|
| 22 | Comparison of release engineering practices in a large mature company and a startup
Eero Laukkanen, Maria Paasivaara, Juha Itkonen, Casper Lassenius ¹
10.1007/s10664-018-9616-7 | Ethnography |
| 23 | Cross-project code clones in GitHub
Mohammad Gharehyazie, Baishakhi Ray, Mehdi Keshani, Masoumeh Soleimani Zavosht, Abbas Heydarnoori, Vladimir Filkov
10.1007/s10664-018-9648-z | Case study |
| 24 | Detecting requirements defects with NLP patterns: an industrial experience in the railway domain
Alessio Ferrari, Gloria Gori, Benedetta Rosadini, Iacopo Trotta ³ · Stefano Bacherini, Alessandro Fantechi, Stefania Gnesi
10.1007/s10664-018-9596-7 | Experiments |
| 25 | Do software models based on the UML aid in source-code comprehensibility? Aggregating evidence from 12 controlled experiments
Giuseppe Scanniello, Carmine Gravino, Marcela Genero ³ · Jose A. Cruz-Lemus, Genoveffa Tortora, Michele Risi, Gabriella Dodero
10.1007/s10664-017-9591-4 | Experiments |
| 26 | Does syntax highlighting help programming novices?
Christoph Hannebauer, Marc Hesenius, Volker Gruhn
10.1007/s10664-017-9579-0 | Experiments |
| 27 | Early prediction of merged code changes to prioritize reviewing tasks
Yuanrui Fan, Xin Xia, David Lo, Shanping Li
10.1007/s10664-018-9602-0 | Experiments |
| 28 | Effective fault localization of automotive Simulink models: achieving the trade-off between test oracle effort and fault localization accuracy
Bing Liu, Shiva Nejati, Lucia, Lionel C. Briand
10.1007/s10664-018-9611-z | Systematic literature reviews |
| 29 | Experimenting with information retrieval methods in the recovery of feature-code SPL traces
Tassio Vale, Eduardo Santana de Almeida
10.1007/s10664-018-9652-3 | Experiments |

30	Factors and actors leading to the adoption of a JavaScript framework Amantia Pano, Daniel Graziotin, Pekka Abrahamsson 10.1007/s10664-018-9613-x	Action Research
31	Four commentaries on the use of students and professionals in empirical software engineering experiments Robert Feldt & Thomas Zimmermann & Gunnar R. Bergersen & Davide Falessi & Andreas Jedlitschka & Natalia Juristo & Jürgen Münch & Markku Oivo & Per Runeson & Martin Shepperd & Dag I. K. Sjøberg & Burak Turhan 10.1007/s10664-018-9655-0	Expert Opinions
32	Finding better active learners for faster literature reviews Zhe Yu, Nicholas A. Kraft, Tim Menzies 10.1007/s10664-017-9587-0	Systematic literature reviews
33	Getting the most from map data structures in Android Ruben Saborido, Rodrigo Morales, Foutse Khomh, Yann-Gael Gueheneuc, Giuliano Antoniol 10.1007/s10664-018-9607-8	Survey
34	GreenScaler: training software energy models with automatic test generation Shaiful Chowdhury, Stephanie Borle, Stephen Romansky, Abram Hindle 10.1007/s10664-018-9640-7	Systematic literature reviews
35	Test them all, is it worth it? Assessing configuration sampling on the JHipster Web development stack Axel Halin, Alexandre Nuttinck, Mathieu Acher, Xavier Devroey, Gilles Perrouin, Benoit Baudry 10.1007/s10664-018-9635-4	Case study
36	High-level software requirements and iteration changes: a predictive model Kelly Blincoe · Ali Dehghan · Abdoul-Djawadou Salaou · Adam Neal, Johan Linaker, Daniela Damian 10.1007/s10664-018-9656-z	Action Research
37	How do developers utilize source code from stack overflow? Yuhao Wu · Shaowei Wang · Cor-Paul Bezemer · Katsuro Inoue 10.1007/s10664-018-9634-5	Survey
38	Using human error information for error prevention	Experiments

Wenhua Hu · Jeffrey C. Carver · Vaibhav Anu · Gursimran S. Walia · Gary L. Bradshaw
10.1007/s10664-018-9623-8

- 39 Identification and prioritization of SLR search tool requirements: an SLR and a survey
Ahmed Al-Zubidy · Jeffrey C. Carver
10.1007/s10664-018-9626-5

- 40 Improved representation and genetic operators for linear genetic programming for automated program repair
Vinicius Paulo L. Oliveira · Eduardo Faria de Souza · Claire Le Goues · Celso G. Camilo-Junior
10.1007/s10664-017-9562-9

- 41 Investigating whether and how software developers understand open source software licensing
Daniel A. Almeida · Gail C. Murphy · Greg Wilson · Michael Hoyer
10.1007/s10664-018-9614-9

- 42 Software engineering in start-up companies: An analysis of 88 experience reports
Eriks Klotins · Michael Unterkalmsteiner · Tony Gorschek
10.1007/s10664-018-9620-y

- 43 On the correctness of electronic documents: studying, finding, and localizing inconsistency bugs in PDF readers and files
Tomasz Kuchta · Thibaud Lutellier · Edmund Wong · Lin Tan · Cristian Cadar
10.1007/s10664-018-9600-2

- 44 Eye tracking analysis of computer program comprehension in programmers with dyslexia
Ian McChesney & Raymond Bond
10.1007/s10664-018-9649-y

- 45 Empowering OCL research: a large-scale corpus of open-source data from GitHub
Josh G. M. Mengerink · Jeroen Noten · Alexander Serebrenik
10.1007/s10664-018-9641-6

- 46 On the challenges of open-sourcing proprietary software projects
Gustavo Pinto · Igor Steinmacher · Luiz Felipe Dias · Marco Gerosa

10.1007/s10664-018-9609-6		
47	On the relative value of data resampling approaches for software defect prediction Kwabena Ebo Bennin · JackyW. Keung · Akito Monden 10.1007/s10664-018-9633-6	Systematic literature reviews
48	Overfitting in semantics-based automated program repair Xuan Bach D. Le · Ferdian Thung · David Lo · Claire Le Goues 10.1007/s10664-017-9577-2	Experiments
49	Large-scale agile transformation at Ericsson: a case study Maria Paasivaara · Benjamin Behm · Casper Lassenius · Minna Hallikainen 10.1007/s10664-017-9555-8	Action Research
50	Preventing duplicate bug reports by continuously querying bug reports Abram Hindle · Curtis Onuczko 10.1007/s10664-018-9643-4	Experiments
51	Program comprehension of domain-specific and general-purpose languages: replication of a family of experiments using integrated development environments Tomaz Kosar · Saso Gaberc · Jeffrey C. Carver · Marjan Mernik 10.1007/s10664-017-9593-2	Experiments
52	Programmers do not favor lambda expressions for concurrent object-oriented code Sebastian Nielebock · Robert Heumuller · Frank Ortmeier 10.1007/s10664-018-9622-9	Experiments
53	Querying distilled code changes to extract executable transformations Reinout Stevens · Tim Molderez · Coen De Roover 10.1007/s10664-018-9644-3	Systematic literature reviews
54	Redundancy-free analysis of multi-revision software artifacts Carol V. Alexandru · Sebastiano Panichella · Sebastian Proksch · Harald C. Gall 10.1007/s10664-018-9630-9	Case study
55	Revisiting supervised and unsupervised models for effort-aware just-in-time defect prediction Qiao Huang · Xin Xia · David Lo 10.1007/s10664-018-9661-2	Systematic literature reviews

56	Revisiting the performance of automated approaches for the retrieval of duplicate reports in issue tracking systems that perform just-in-time duplicate retrieval Mohamed Sami Rakha · Cor-Paul Bezemer · Ahmed E. Hassan 10.1007/s10664-017-9590-5	Systematic literature reviews
57	A controlled experiment on time pressure and confirmation bias in functional software testing Iflaah Salman · Burak Turhan · Sira Vegas 10.1007/s10664-018-9668-8	Experiments
58	Security code smells in Android ICC Pascal Gadiet · Mohammad Ghafari · Patrick Frischknecht · Oscar Nierstrasz 10.1007/s10664-018-9673-y	Case study
59	Semi-automatic rule-based domain terminology and software feature-relevant information extraction from natural language user manuals An approach and evaluation at Roche Diagnostics GmbH Thomas Quirchmayr · Barbara Paech · Roland Kohl2 · Hannes Karey · Gunar Kasdepke 10.1007/s10664-018-9597-6	Case study
60	Shorter identifier names take longer to comprehend Johannes C. Hofmeister · Janet Siegmund · Daniel V. Holt 10.1007/s10664-018-9621-x	Action Research
61	Mock objects for testing java systems Why and how developers use them, and how they evolve Davide Spadini · Maurício Aniche · Magiel Bruntink · Alberto Bacchelli 10.1007/s10664-018-9663-0	Action Research
62	Studying and detecting log-related issues Mehran Hassani · Weiyi Shang · Emad Shihab · Nikolaos Tsantalis 10.1007/s10664-018-9603-z	Ethnography
63	Studying software logging using topic models Heng Li · Tse-Hsun (Peter) Chen · Weiyi Shang · Ahmed E. Hassan 10.1007/s10664-018-9595-8	Case study
64	Studying the consistency of star ratings and reviews of popular free hybrid Android and iOS apps	Case study

Hanyang Hu · Shaowei Wang · Cor-Paul Bezemer · Ahmed E. Hassan
10.1007/s10664-018-9617-6

- 65 Studying the consistency of star ratings and the complaints in 1 & 2- star user reviews for top free cross-platform Android and iOS apps Case study
Hanyang Hu · Cor-Paul Bezemer · Ahmed E. Hassan
10.1007/s10664-018-9604-y

- 66 Successes, challenges, and rethinking – an industrial investigation on crowdsourced mobile application testing Case study
Ruizhi Gao & Yabin Wang & Yang Feng & Zhenyu Chen & W. Eric Wong
10.1007/s10664-018-9618-5

- 67 Supporting the analyzability of architectural component models - empirical findings and tool support Experiments
Srdjan Stevanetic · Uwe Zdun
10.1007/s10664-017-9583-4

- 68 Syntax, predicates, idioms — what really affects code complexity? Surveys
Shulamyt Ajami · Yonatan Woodbridge · Dror G. Feitelson
10.1007/s10664-018-9628-3

- 69 System requirements-OSS components: matching and mismatch resolution practices – an empirical study Surveys
Claudia Ayala & Anh Nguyen-Duc & Xavier Franch et al.
10.1007/s10664-017-9594-1

- 70 Discovering community patterns in open-source: a systematic approach and its evaluation Case study
Damian A. Tamburri · Fabio Palomba · Alexander Serebrenik · Andy Zaidman
10.1007/s10664-018-9659-9

- 71 The impact of human factors on the participation decision of reviewers in modern code review Case study
Shade Ruangwan · Patanamon Thongtanunam · Akinori Ihara · Kenichi Matsumoto
10.1007/s10664-018-9646-1

- 72 The modular and feature toggle architectures of Google Chrome Case study
Md Tajmilur Rahman · Peter C. Rigby · Emad Shihab
10.1007/s10664-018-9639-0

73	Understanding the behaviour of hackers while performing attack tasks in a professional setting and in a public challenge Mariano Ceccato · Paolo Tonella · Cataldo Basile et al. 10.1007/s10664-018-9625-6	Survey
74	Usage and attribution of Stack Overflow code snippets in GitHub projects Sebastian Baltes · Stephan Diehl 10.1007/s10664-018-9650-5	Survey
75	Using frame semantics for classifying and summarizing application store reviews Nishant Jha · Anas Mahmoud 10.1007/s10664-018-9605-x	Case study
76	ProMeTA: a taxonomy for program metamodels in program reverse engineering Hironori Washizaki · Yann-Gaël Guéhéneuc · Foutse Khomh 10.1007/s10664-017-9592-3	can not judge
77	What are the effects of history length and age on mining software change impact? Leon Moonen · Thomas Rølfesnes · Dave Binkley · Stefano Di Alesio 10.1007/s10664-017-9588-z	Case study
78	What can Android mobile app developers do about the energy consumption of machine learning? Andrea McIntosh · Safwat Hassan · Abram Hindle 10.1007/s10664-018-9629-2	Experiments
79	What's Spain's Paris? Mining analogical libraries from Q&A discussions Chunyang Chen · Zhenchang Xing · Yang Liu 10.1007/s10664-018-9657-y	Survey
80	Will this clone be short-lived? Towards a better understanding of the characteristics of short-lived clones Patanamon Thongtanunam · Weiyi Shang · Ahmed E. Hassan 10.1007/s10664-018-9645-2	Case study

3. Classification Rationale

3.1 Experiments

Citation

An Inception Architecture-Based Model for Improving Code Readability Classification
Qing Mi, Jacky Keung, Yan Xiao, Solomon Mensah, Xiupei Mei
10.1145/3210459.3210473

Rationale

This article is a representative literature that introduces algorithms, tools or frameworks, that is, the whole article revolves around an algorithm, tool or framework designed by a team to solve a practical problem. This article proposes a model that uses deep learning to classify code readability-IncepCRM, and describes in detail the data processing process, model architecture, and model training process. In the second half of the article, two research questions are raised: Can annotated information improve the performance of the model? How does this model stand out compared to existing code readability classification models? Then the experiment was designed accordingly. So it is classified as experiments.

3.2 Case study

Citation:

DevOps Capabilities, Practices, and Challenges: Insights from a Case Study
Mali Senapathi, Jim Buchan, Hady Osman
10.1145/3210459.3210465

Rationale

This article on DevOps capabilities, practices, and challenges clearly mentions Case Study in the title, but the abstract mentions "The study involved interviewing six experienced software engineers who continuously monitored and reflected on the gradual implementation of DevOps principles and practices. "So it is a little confusing that

whether it is a case study, a survey, or an expert opinion. After reading the article carefully and our inner discussion, we realized that the interviews of these six engineers were actually included in this case study as a supplement to the opinions of the practitioners. So it is still classified as case study

3.3 Survey

Citation:

Do software engineering practitioners cite research on software testing in their online articles?: A preliminary survey.

Ashley Williams

10.1145/3210459.3210475

Rationale:

The purpose of this article is to investigate whether practitioners will quote some software testing research results in their online technical articles, and if so, in what way. The reason for choosing software testing is that it is already a relatively mature research field. The puzzle of this article lies in two aspects: the object of its survey is not practitioners, but practitioners' articles; the main part of this article is to introduce a semi-automated method designed by the author's team to assess the relevance of research and the index used to measure the frequency of software testing research citations. This makes us think that this article may use action research and experiments research methods. We finally decided to use the research question model of the document to decide, "Whether practitioners will cite some software testing research results in their technical articles published on the Internet, and if so, in what way?" It is an exploration of practitioners' attitudes and behaviors towards software testing research, which is in line with the typical rq model of survey research methods. And the text "When researchers survey practitioners, the practitioners' responses are based on their opinions and personal experience. Therefore, analysis of online articles can be treated as a type of survey of practitioner opinion" also explains that articles can be used as opinion of the reason.

3.4 Ethnography

Citation

An exploratory study of software sustainability dimensions and characteristics: end user perspectives in the kingdom of Saudi Arabia (KSA)

Sulaiman Aljarallah, Russell Lock

10.1145/3239235.3239240

Rationale

First, the research using this method is limited to personal participation. Second, the observation and research is about a certain area/group/cluster in order to obtain a systematic regional conclusion. In this article, the author focused on the KSA area to observe, participate and collect relevant evidence and data for analysis, which is basically in line with ethnography.

3.5 Systematic literature review

Citation

Is there a "golden" feature set for static warning identification?: an experimental evaluation

Junjie Wang, Song Wang, Qing Wang

10.1145/3239235.3239523

Rationale

The use of the SLR method requires the search for evidence, that is, the systematic collection, classification and study of relevant research in order to achieve the goal of reviewing existing research, instruction and guidance for future research directions. This article focuses on a recognition method. The realization methods of the existing researches are reviewed, and the experimental researches are classified and unified, so this method is in line with this method.

3.6 Action research

Citation:

Factors and actors leading to the adoption of a JavaScript framework
Amantia Pano, Daniel Graziotin, Pekka Abrahamsson
10.1007/s10664-018-9613-x

Rationale:

The article mainly discusses the choice of framework in JavaScript. They interviewed 18 people in the industry who were responsible for deciding the choice of framework, and built a model based on the reference factors when they were established, so that the model can help people choose the framework. This article can easily be mistaken for survey or expert opinion at the beginning, but after careful study and discussion, it is found that most of the subjects the author's team surveyed are relevant experts, and they are asking questions about a certain problem, hoping to obtain information from the experts' related personnel experience and judgment, but, unlike expert opinion, the team asks many people, and it does not directly sum up from these people's answers, but integrates and filters based on some of these people's opinions and experiences to improve the model.

3.7 Expert opinion

Citation:

A systematic approach for performance assessment using process mining
Simona Bernardi, Juan L. Dominguez, Abel Gomez, Christophe Joubert, Jose Merseguer, Diego Perez-Palacin, Jose I. Requeno, Alberto Romeu
10.1007/s10664-018-9606-9

Rationale:

Our criteria about Expert Opinions is as follows. First of all, the research using this method must limit the people who provide opinions to experts. Second, the experts must discuss specific topics, evaluate them based on their own experience and practical applications, and summarize the results of the problems they want to explore. The article

is mainly based on the authors' own experience and some methodological assessments, which can successfully predict the scalability of the system in the future, and talks about the technology and possible development of the system. The whole article mainly revolves around the author's experience, that is to say, the author, as an expert in the field, puts forward a series of opinions and scientific evidence based on some of his own applications and practical operations, and finally raises corresponding scientific opinions.