

Looking Over the Research Literature on Software Engineering from 2016 to 2018

Ruben Heradio, David Fernandez-Amoros,
Daniel Galan,
F. Javier Cabrerizo, and Enrique Herrera-Viedma

Universidad Nacional de Educacion a Distancia (Spain)

Universidad Politecnica de Madrid (Spain)

Universidad de Granada (Spain)

rheradio@issi.uned.es

https://www.researchgate.net/profile/Ruben_Heradio

Motivation

Looking Over the **Research Literature** on **Software Engineering** from **2016 to 2018**

Research Questions:

- RQ1: What articles are the most impacting?
- RQ2: Where have those articles been published?
- RQ3: What are the upward research trends?
- RQ4: Which countries and institutions are playing the principal role?

Previous literature reviews

- In 2007, **Kitchenham and Charters** published an influential technical report proposing some **guidelines** to perform systematic software engineering reviews (*surveys, systematic literature reviews, systematic mapping studies, and meta-analyses*).
- Those review strategies **work** for **small-to-medium** literature samples of a few hundreds of articles
- **This paper analyzes 6,365 documents gathered from Elsevier's Scopus using bibliometric techniques**

Document sample

To obtain a representative **sample of the software engineering literature **population****

15 Journals:

- Automated Software Engineering
- IEEE Software
- IEEE Transactions on Software Engineering
- IEICE Transactions on Information and Systems
- IET Software
- Information and Software Technology
- Innovations in Systems and Software Engineering
- ...
- Empirical Software Engineering

Document sample

17 Conferences:

- ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE)
- Asia Pacific Software Engineering Conference (APSEC)
- EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA)
- IEEE ACM International Conference on Automated Software Engineering (ASE)
- IEEE International Conference on Software Testing Verification and Validation (ICST)
- ...
- International Conference on Software Engineering (ICSE)

Document sample

6,365 documents

The corpus is available at:

<https://github.com/rheradio/SwEngScopus2016-18>

Analysis tools

The document sample was analyzed using the following **R packages**:

- **bib2df** to parse the Scopus data (in Bibtex format) to an R data frame
- **tidyverse** for processing the corpus
- **wordcloud** for identifying the most common article keywords
- **ggplot2** for obtaining a visual representation of the countries whose research institutions are the most prolific

RQ1: Hot papers

Typical measures for **citation classics (e.g., H-core)** **do not work** for recent papers as **articles from 2018 do not have enough time to accumulate citations to compete against papers from 2016**

We have used another criterium: the **99th citation percentile** of their corresponding year

Hot papers from 2016

Article	Year	#Citations
<i>Mechtaev et al.</i> : Angelix: Scalable multiline program patch synthesis via symbolic analysis	2016	111
<i>Gu et al.</i> : Deep API Learning	2016	87
<i>Wang et al.</i> : Automatically learning semantic features for defect prediction	2016	84
<i>Tantithamthavorn et al.</i> : Automated parameter optimization of classification techniques for defect prediction models	2016	80
<i>Sajnani et al.</i> : SourcererCC: Scaling code clone detection to big-code	2016	74
<i>Stol et al.</i> : Grounded theory in software engineering research: A critical review and guidelines	2016	72
<i>White et al.</i> : Deep learning code fragments for code clone detection	2016	71
<i>Ye et al.</i> : From word embeddings to document similarities for improved information retrieval in software engineering	2016	71
<i>Zhang et al.</i> : Cross-project defect prediction using a connectivity-based unsupervised classifier	2016	71
<i>Xia et al.</i> : HYDRA: Massively compositional model for cross-project defect prediction	2016	64
<i>Villarroel et al.</i> : Release planning of mobile apps based on user reviews	2016	64
<i>Sorbo et al.</i> : What would users change in my App? Summarizing app reviews for recommending software changes	2016	63
<i>Segura et al.</i> : A Survey on Metamorphic Testing	2016	61
<i>Kosar et al.</i> : Domain-Specific Languages: A Systematic Mapping Study	2016	59
...

Hot papers from 2017

Article	Year	#Citations
<i>Tantithamthavorn et al.</i> : An Empirical Comparison of Model Validation Techniques for Defect Prediction Models	2017	67
<i>Bröring et al.</i> : Enabling IoT Ecosystems through Platform Interoperability	2017	61
<i>Xuan et al.</i> : Nopol: Automatic Repair of Conditional Statement Bugs in Java Programs	2017	57
<i>Beller et al.</i> : TravisTorrent: Synthesizing Travis CI and GitHub for Full-Stack Research on Continuous Integration	2017	55
<i>Kitchenham et al.</i> : Robust Statistical Methods for Empirical Software Engineering	2017	50
<i>Taivalsaari et al.</i> : A Roadmap to the Programmable World: Software Challenges in the IoT Era	2017	44
<i>Groen et al.</i> : The Crowd in Requirements Engineering: The Landscape and Challenges	2017	40
<i>Munaiah et al.</i> : Curating GitHub for engineered software projects	2017	35
<i>Jongeling et al.</i> : On negative results when using sentiment analysis tools for software engineering research	2017	35
<i>Lu et al.</i> : Adaptable Blockchain-Based Systems: A Case Study for Product Traceability	2017	34
<i>Le et al.</i> : S3: Syntax- and semantic-guided repair synthesis via programming by examples	2017	34
<i>Jing et al.</i> : An Improved SDA Based Defect Prediction Framework for Both Within-Project and Cross-Project Class-Imbalance Problems	2017	33
...

Hot papers from 2018

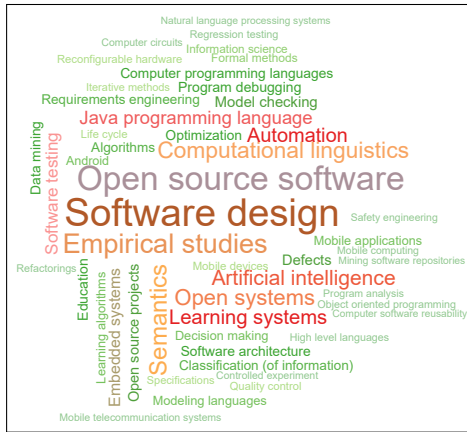
Article	Year	#Citations
<i>Morschheuser et al.</i> : How to design gamification? A method for engineering gamified software	2018	24
<i>Dingsoyr et al.</i> : Exploring software development at the very large-scale: a revelatory case study and research agenda for agile method adaptation	2018	24
<i>Panichella et al.</i> : Automated Test Case Generation as a Many-Objective Optimisation Problem with Dynamic Selection of the Targets	2018	21
<i>Falessi et al.</i> : Empirical software engineering experts on the use of students and professionals in experiments	2018	21
<i>Leemans et al.</i> : Scalable process discovery and conformance checking	2018	16
<i>Taibi et al.</i> : On the Definition of Microservice Bad Smells	2018	15
<i>García et al.</i> [?]: Complete and Interpretable Conformance Checking of Business Processes	2018	14
<i>Kula et al.</i> : Do developers update their library dependencies?: An empirical study on the impact of security advisories on library migration	2018	14
<i>Chen et al.</i> : MULTI: Multi-objective effort-aware just-in-time software defect prediction	2018	14
<i>Bennin et al.</i> : MAHAKIL: Diversity Based Oversampling Approach to Alleviate the Class Imbalance Issue in Software Defect Prediction	2018	13
<i>Palomba et al.</i> : On the diffuseness and the impact on maintainability of code smells: a large scale empirical investigation	2018	13
...

RQ2: Where were published the hot papers?

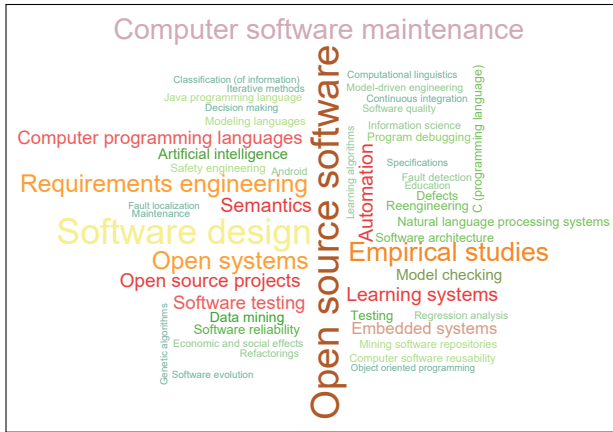
More than **75 % of all hot papers** have been published in **a few sources**:

- International Conference on Software Engineering - **ICSE** (16.67 % of the hot papers)
- **Empirical Software Engineering** (16.67 %)
- **Information and Software Technology** (16.67 %)
- **IEEE Transactions on Software Engineering** (15.15 %)
- **IEEE Software** (10.61 %)

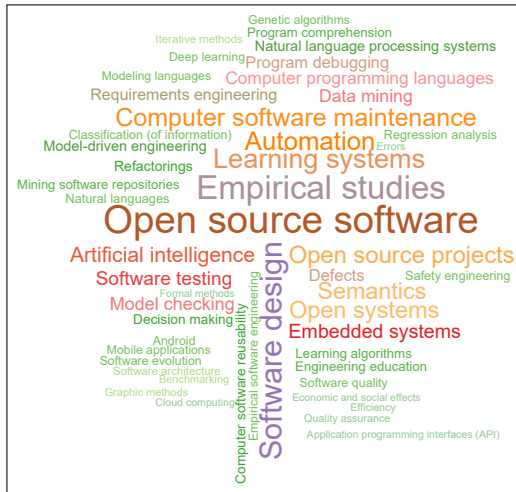
Hot Topics: 2016



Hot Topics: 2017



Hot Topics: 2018

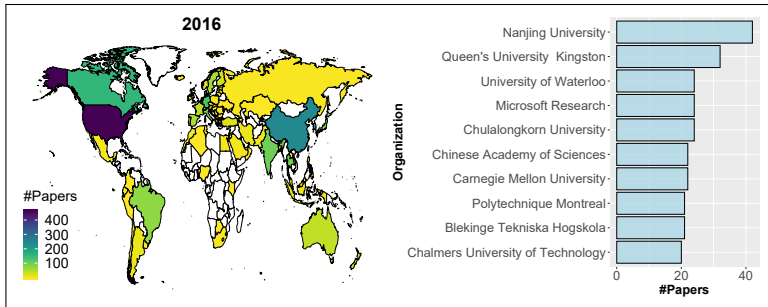


Hot Topics

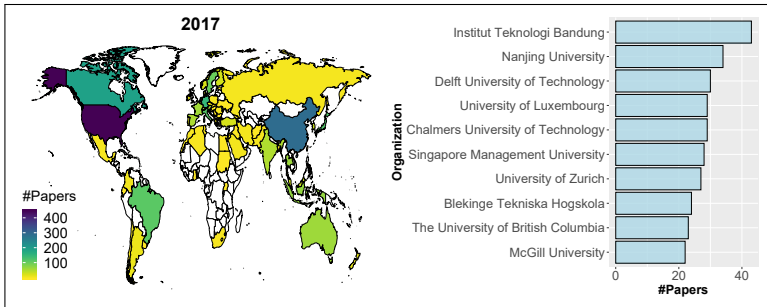
3 upward trends can be identified:

- 1 Researching on **open-source software**: *open-source software, open systems, and open-source projects.*
- 2 The **application of artificial intelligence techniques** to software engineering: *deep learning, artificial intelligence, learning systems, learning algorithms, genetic algorithms, natural language processing, and classification of information.*
- 3 Undertaking **empirical software engineering**: *empirical studies, empirical software engineering, benchmarking, and mining software repositories.*

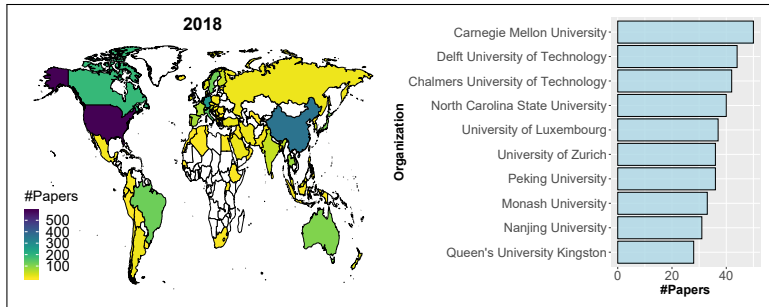
Most prolific institutions: 2016



Most prolific institutions: 2017



Most prolific institutions: 2018



Most prolific countries

The countries whose institutions published most papers were:

- In **2016: USA** (23.96 %), **China** (12.71 %) and **South Korea** (11.25 %).
- In **2017: USA** (22.37 %), **China** (14.61 %) and **Canada** (9.58 %).
- In **2018: USA** (23.16 %), **China** (14.38 %) and **Germany** (11.03 %).

Conclusions

- Most literature on software engineering is produced in **two countries: USA** (23.16 %) **and China** (13.9 %)
- The **publication sources** of the most influential papers **are rather concentrated**: one conference and four journals have published 75.77 % of the documents in the top 0.1 % citation ranking
- **3 hot topics: open-source software**, the **application of artificial intelligence** to deal with software engineering problems, and the importance of **performing empirical studies about software development**

That's all

Thanks for your attention!

Any questions?