

Sebastiano Panichella - Curriculum vitae

BIOGRAPHICAL SKETCH

Sebastiano Panichella is a passionate Computer Science Researcher (permanent position) at Zurich University of Applied Science (ZHAW), leading research in Software Engineering (SE) and cloud computing (CC) research fields.

Sebastiano Panichella received (cum laude) the Laurea in Computer Science from the University of Salerno (Italy) in December 2010 defending a thesis on IR-based Traceability Recovery. He received the PhD in Computer Science from the University of Sannio (Department of Engineering) in July 18th 2014 defending the thesis entitled “*Supporting Newcomers in Open Source Software Development Projects*”.

His research interests are in the domain of Software Engineering (SE) and cloud computing (CC): DevOps (e.g., Continuous Delivery, Continuous integration), Machine learning applied to SE, Software maintenance and evolution (with particular focus on Cloud, mobile, and Cyber-physical applications), Mobile Computing. Moreover, he is promoting research on *Summarization Techniques for Code, Changes, and Testing*. His research was/is funded by Swiss National Science Foundation Grants. He authored or co-authored several papers appeared in International Conferences and Journals. These research works involved studies with industrial and open projects and received best paper awards or best paper nominations (see his main collaborators at the following link: <https://spanichella.github.io/#bio>). He supervised (or co-supervised) 9 undergrad students, 7 MSc students and currently/recently 7 PhD students. He serves and has served as a program committee member of various international conference (e.g., ICSE, ASE, FSE, ICSME, etc.). Dr. Panichella was selected as one of the top-20 (second in Switzerland) Most Active Early Stage Researchers (results reported by the JSS journal) Worldwide in SE. He is Editorial Board Member of Journal of Software: evolution and process (JSEP). He is also Review Board member of the EMSE and TOSEM journals.

He was **chair of the First International Workshop *Cloud-Native Applications Design and Experience*** (co-located with UCC and BDCAT):
cnax.servicelaboratory.ch

See the [blogpost dedicated to the event](https://bit.ly/2tNtEhk): <https://bit.ly/2tNtEhk>

His research on mobile computing was funded by one Swiss National Science Foundation Grant:

www.ifi.uzh.ch/en/seal/research/projects/SURF-MobileData.html

Recent Achievements of Sebastiano Panichella:

- According to the [Results reported by the JSS journal] Sebastiano Panichella was selected, according to the results reported by the JSS journal ¹, as one of the **top-20 (second in Switzerland) Most Active Early Stage Researchers in Software Engineering (SE)**. We take this opportunity to thank the SNF for supporting the research in SE and mobile computing with the project SURF-MobileAppsData SNF project.
- The research proposal submitted to the **H2020 grant** called *COSMOS: DevOps for Complex Cyber-physical Systems* was recently selected for funding.

¹<https://bit.ly/2DbpQgJ>

- The paper [Sebastiano Panichella, Andrea Di Sorbo, Emitza Guzman, Corrado Aaron Visaggio, Gerardo Canfora, Harald C. Gall: How can I improve my app? Classifying user reviews for software maintenance and evolution. ICSME 2015: 281-290], which originated the idea behind this SNF project, is one of the **most cited papers of ICMSE 2015** (as reported in Google scholar), with over 250 citations in around 4-5 years.
- The paper ICPC wrote during the bachelor studies of Dr. Panichella-[Giovanni Capobianco, Andrea De Lucia, Rocco Oliveto, Annibale Panichella, Sebastiano Panichella: On the role of the nouns in IR-based traceability recovery. ICPC 2009: 148-157] is **among the most influential papers of ICPC in the last decade [period 2009-2019]**.

Previously he was postdoc at University of Zurich (**01-11-2014 - 19-08-2018**) working in the *SEAL Lab* of **Prof. Harald Gall**. During the postdoctoral experience, Sebastiano wrote 100% of a proposal that was awarded (Sebastiano figured as co-applicant with Prof. Gall) by the Swiss National Science Foundation, i.e., the project SURF-MobileAppsData SNF – No. 200021–166275– (current results of the projects are available on-line²), which is funding his research collaboration with the UZH (since 2016), on mobile computing and mobile testing, and two PhD Students. During the experience as postdoc in the SEAL group he investigated further SE research fields such as Mobile Computing, Continuous Delivery and Continuous integration, and the new line of research related to the use of Summarization Techniques for Code, Changes and Testing.

He is author or co-author of around **sixty** (considering also demos, datasets and poster) papers appeared in International Conferences and Journals (**26** of them published during the experience at the SEAL lab). In summary he published in high-ranked, peer-reviewed (according to the <http://www.core.edu.au/conference-portal>), and international venues (where he also received best and distinguished paper awards³).

These research works involved studies with industrial and open projects and received best paper awards or best paper nominations. For instance, these research works involved relevant industrial companies (e.g., ING NEDERLAND, Sony Mobile Communication) and their extensions will involve further (e.g., Siemens, GMV, etc.) and research partners (as reported in the proposal) and open source projects⁴.

He serves and has served as program committee member of various international conference (e.g., ICSE, SBST, ASE, ICPC, ICSME, SANER, MSR, SEAA) and as reviewer for various international journals (e.g., TSE, TOSEM, EMSE, JSS, IST, JSEP) in the fields of software engineering and evolutionary computation. He is currently Editorial Board Member of *Journal of Software: evolution and process* and (Leading) Editor of special Issues at International Journals such as EMSE, IST, and SCP.

CONTACT INFORMATION

Address: Zurich University of Applied Science (ZHAW)

Obere Kirchgasse 2, 8400 Winterthur, Switzerland.

Tel.: +41 (0) 58 934 41 56

E-mail: panc@zhaw.ch (or alternatively spanichella@gmail.com)

Home Page: <https://spanichella.github.io/>

Google Scholar Ref:

<https://scholar.google.it/citations?user=HiNuBFgAAAAJ&hl=en&oi=ao>

Short CV: <https://spanichella.github.io/img/CV-short.pdf>

Detailed CV: <https://spanichella.github.io/img/CV.pdf>

²<http://www.ifi.uzh.ch/en/seal/people/panichella/SNF-Projects.html>

³ <http://www.ifi.uzh.ch/en/seal/people/panichella/Awards-Best-Paper-Nominations.html>

⁴<https://spanichella.github.io/#bio>

Machine Learning and Genetic Algorithms

Machine learning (ML) and Genetic Algorithms (GA) deals with the issue of how to build computer programs that improve their performance at some tasks through experience. ML and Genetic algorithms have proven to be of great practical value in a variety of application domains. Not surprisingly, the field of software engineering turns out to be a fertile ground where many software development and maintenance tasks could be formulated as learning problems and approached in terms of learning algorithms. **Work in progress.** Panichella was also very fascinated by the potential of ML and Genetic Algorithms for solving SE problems. He started to study them during the PhD studies. Examples of the successful application of ML and genetic algorithms to SE problems by Panichella are bug prediction, code (and code change) prediction [37] [9] [11] [52], prioritization or clustering of user reviews (in the context of mobile apps) [30][31][33][35] [36][42], test case generation [38], automated prediction of issue labels in Github [17], etc.. Current research interest are toward experimenting customized solutions based on ML and Genetic Algorithms for enhancing traditional testing approaches and GUI testing processes 1.

Continuous Delivery

Continuous delivery (CD) is a software engineering approach in which teams produce software in short cycles, ensuring that the software can be reliably released at any time. It aims at building, testing, and releasing software faster and more frequently. The approach helps reduce the cost, time, and risk of delivering changes by allowing for more incremental updates to applications in production. A straightforward and repeatable deployment process is important for continuous delivery. Continuous Integration (CI) consists in a specific stage of CD process where team members integrate their work in an automatic manner, which allows a fast building, testing, and releasing of software, leading to multiple integrations per day. Researchers in this field have as main focus the development of recommender systems able to provide suggestions to developers and testers during Continuous Integration activities.

Work in progress. Panichella is very interested in investigate and overcome contemporary limitations of DevOps (e.g., continuous delivery and continuous integration) practices and tools for complex systems (e.g., Sloud and Cyber-physical systems). In the context of CI Panichella is currently conducting empirical work to understand the problems that developers face when integrating new changes in the code base [28] 1. The main focus is the development of recommender systems able to provide suggestions to developers and testers during Continuous Integration activities. In recent work he also investigated strategies to optimize test case generation in CI pipelines[3], contemporary bad practices affecting CI adoption [2], and how developers engage with static analysis tools in different development contexts (i.e., Code Review, CI, local development) [4]

Empirical Software Engineering

Empirical software engineering is a sub-domain of software engineering focusing on experiments on software systems (software products, processes, and resources). It is interested in devising experiments on software, in collecting data from these experiments, and in devising laws and theories from this data. Proponents of experimental software engineering advocate that the nature of software is such that we can advance the knowledge on software through experiments only. The scientific method suggests a cycle of observations, laws, and theories to advance science. Empirical software engineering applies this method to software.

Work in progress. In past work Panichella performed empirical studies to under-

stand (i) how OSS communities upgrades dependencies [10][49]; (ii) to what extent static analysis tools help developers with code reviews [44]; (iii) how developers' collaborations identified from different sources vary when they are mined from different sources [45]; (iv) how the evolution of emerging collaborations relates to code changes [48]; or (v) to study the behaviour of developers during maintenance tasks (e.g., while they modify existing features or fix a bug) by analyzing their navigation patterns [50]. Currently Panichella is focusing his attention in performing empirical work to understand possible ways to measure and foster developer productivity during testing [38], maintenance [50] and code reviewing tasks [44].

Mining Software Repositories

Software repositories such as source control systems, archived communications between project personnel, and defect tracking systems are used to help manage the progress of software projects. Software practitioners and researchers are recognizing the benefits of mining this information to support the maintenance of software systems, improve software design/reuse, and empirically validate novel ideas and techniques. Research is now proceeding to uncover the ways in which mining these repositories can help to understand software development and software evolution, to support predictions about software development, and to exploit this knowledge concretely in planning future development. The Mining Software Repositories (MSR) field analyzes the rich data available in software repositories to uncover interesting and actionable information about software systems and projects.

Work in progress. In past work Panichella focused his attention in mining software repository to build recommender systems for supporting developers during maintenance and program comprehension tasks. For instance, he conceived tools for (i) enabling the automatic re-documentation of existing systems [47] [55]; (ii) summarizing software artifacts [54] [12]; (iii) or profiling developers or experts in OSS projects [40][45][46][48][51][53]. Currently Panichella is focusing his attention in mining App Store data and data from traditional repositories for designing and developing tools to help developers digest the huge amount of feedback they receive from users on a daily basis, transforming user reviews into maintenance tasks (fixing issues or building features) [30][31][33][35]. More in general, he is interested to conceive tools to support developers in evolving modern software applications [36][42].

Code Review

Peer code review, a manual inspection of source code by developers other than the author, is recognized as a valuable tool for reducing software defects and improving the quality of software projects. In 1976, Fagan formalized a highly structured process for code reviewing, based on line-by-line group reviews, done in extended meetings-code inspections. Over the years, researchers provided evidence on code inspection benefits, especially in terms of defect finding, but the cumbersome, time-consuming, and synchronous nature of this approach hinders its universal adoption in practice. Nowadays, many organizations are adopting more lightweight code review practices to limit the inefficiencies of inspections. In particular, there is a clear trend toward the usage of tools specifically developed to support code review. Modern code reviews are (1) informal (in contrast to Fagan-style), (2) tool-based, and (3) occurs regularly in practice nowadays, for example at companies such as Microsoft, Google, Facebook, and in other companies and OSS projects.

Work in progress. The research focus of Panichella is to develop recommender systems able to (better) support developers during the code review process [44].

IR-based Traceability Recovery

Traceability has been defined as "the ability to describe and follow the life of an artefact (requirements, code, tests, models, reports, plans, etc.), in both a forwards and backwards direction". Thus, traceability links help software engineers to understand the relationships and dependencies among various software artefacts (requirements, code, tests, models, etc.) developed during the software lifecycle. The two main research topics related to the traceability management are event-based systems for traceability management and information retrieval based methods and tools supporting the software engineer in the traceability link recovery.

Work in progress. In past work Panichella explored several enhancing strategies for improving IR-based Traceability Recovery approaches, most of them are based on (i) smoothing filters [11] [13] and (ii) NLP approaches [56] [57] [58]. Recently Panichella is focusing his effort in tracing link between data and software artifacts stored in modern software repositories [30] [31] [33].

Textual analysis

Textual analysis can be described as the examination of a text in which an educated guess is formed as to the most likely interpretations that might be made of that text. It is where the researcher must decentre the text to reconstruct it, working back through the narrative mediations of form, appearance, rhetoric, and style to uncover the underlying social and historical processes, the metalanguage that guided the production. It is suggested that textual analysis can cover four main underlying constructs: language and meaning, ideology, ideology and myth, and historicity. In this sense, textual analysis is a methodology: a way of gathering and analysing information in academic research (McKee, A 2001).

Work in progress. Panichella studied text analysis approaches since his bachelor and master studies and was always fascinated by the great usability of Natural Language Processing (NLP) and Information Retrieval (IR) tools and techniques for solving several practical problems. He adopted such techniques in several work during his PhD and also during the postdoctoral experience. He is currently learning new techniques and tools based on Textual Analysis (e.g. WORD2VEC) and neural networks techniques [29].

ACADEMIC APPOINTMENTS

Currently he is a (Permanent) Senior Research Associate at Zurich University of Applied Science (from **20-08-2018**), leading research in Software Engineering (SE) and cloud computing (CC) research fields. Previously he was postdoc at University of Zurich (**01-11-2014 - 19-08-2018**) working in the *SEAL Lab* of **Prof. Harald Gall**. He is a **member of IEEE** and **ACM**. During the experience as postdoc in the SEAL group he investigated further SE research fields such as Mobile Computing, Continuous Delivery and Continuous integration, and the new line of research related to the use of Summarization Techniques for Code, Changes and Testing. Currently His research interests include Mobile/Cloud Computing, IR-based Traceability Recovery, Textual Analysis, Machine Learning and Genetic Algorithms applied to SE problems, Continuous Delivery (with special attention to Continuous Integration Problems), Software maintenance and evolution and Empirical Software Engineering (with particular focus on Cloud Applications). Another topic that is also of his interest is Code Review, indeed, he is currently working and advising students on research ideas aimed at automating the process of code inspection. His research is funded by a Swiss National Science Foundations project. He is author or co-author of **57** (considering also demos, datasets and poster) papers appeared in International Conferences and Journals (**26** of them published during the experience at the SEAL lab). In summary he published in high-ranked, peer-reviewed (according to the [5 of 24](http://www.core.edu.au/conference-</p>
</div>
<div data-bbox=)

portal), and international venues (where he also received best and distinguished paper awards⁵).

ACADEMIC
EXPERIENCE AND
HISTORY

University of Zurich, Switzerland

Postdoc at University of Zurich working in the SEAL Lab of **Prof. Harald Gall**.
Period **01-11-2014 - 19-08-2018**.

University of Sannio, Italy

PhD., Computer Engineering, July 2014

- Thesis Title: *“Supporting Newcomers in Open Source Software Development Projects”*
- Thesis Topics: Supporting Developers, Mining of Software Repositories (Mailing lists, Issue trackers, Versioning Systems etc.)

University of Salerno, Italy

M.S., Computer Science, December 2010

- *Magna cum Laude*
- Thesis Title: *Improving IR-based Traceability Recovery Using Smoothing Filters*
- Adviser: Prof. Andrea De Lucia
- Thesis Topics: Software Engineering, Traceability Recovery, Textual Analysis

University of Molise, Italy

B.S., Computer Science, October 2008

- *Magna cum Laude*
- Thesis Title: *Improving IR-based traceability recovery via noun-based indexing of software artifacts*
- Advisers: Prof. Giovanni Capobianco, Dr Rocco Oliveto
- Thesis Topics: Software Engineering, Traceability Management, Natural Language Processing (NLP)

⁵ <http://https://spanichella.github.io/#awards>

In papers marked with (*) the authors are listed in alphabetic order.

Journal Publications after the P.h.D.

- [1] Yu Zhou, Yanqi Su, Taolue Chen, Zhiqiu Huang, Harald Gall, Sebastiano Panichella: User Review-Based Change File Localization for Mobile Applications . Transactions on Software Engineering (TSE) Journal.
doi:<https://doi.org/10.1109/TSE.2020.2967383>
- [2] Fiorella Zampetti, Carmine Vassallo, Sebastiano Panichella, Gerardo Canfora, Harald Gall, Massimiliano Di Penta: **An Empirical Characterization of Bad Practices in Continuous Integration**. *Empirical Software Engineering (EMSE)*.
doi:<https://doi.org/10.1109/TSE.2019.2946773>
- [3] Giovanni Grano, Christoph Laaber, Annibale Panichella, and Sebastiano Panichella: **Testing with Fewer Resources: An Adaptive Approach to Performance-Aware Test Case Generation**. *Transactions on Software Engineering (TSE)*.
doi:<https://doi.org/10.1109/TSE.2019.2946773>
- [4] A. Sorbo, S. Panichella, Aaron Visaggio, Di Massimiliano Di Penta, Canfora Gerardo, and Harald Gall. **Exploiting Natural Language Structures in Software Informal Documentation**. *Transactions on Software Engineering (TSE)* 2019.
doi:<https://doi.org/10.1109/TSE.2019.2930519>
- [5] C. Vassallo, S. Panichella, F. Palomba, S. Proksch, Harald Gall, Andy Zaidman. **How Developers Engage with Static Analysis Tools in Different Contexts**. *Empirical Software Engineering (EMSE)* 2019.
- [6] G.Grano, T. Titov, S. Panichella, H. Gall: **Branch Coverage Prediction in Automated Testing**. *Journal of Software: Evolution and Process (JSEP)* 2019.
doi:<https://doi.org/10.1016/j.infsof.2019.05.005>
- [7] C. Alexandru, S. Panichella, S. Proksch, Harald Gall. ***Redundancy-free Analysis of Multi-revision Software Artifacts**. *Empirical Software Engineering (EMSE)* 2019.
doi:<http://doi.acm.org/10.1145/3276954.3276960>
- [8] Y. Zhou and C. Wang and Y. Xin and T. Chen and S. Panichella and H. Gall . **Automatic Detection and Repair Recommendation of Directive Defects in Java API Documentation**. *Transaction on Software Engineering 2018* -
<https://ieeexplore.ieee.org/document/8478004>.
- [9] G. Canfora, A. De Lucia, M. Di Penta, R. Oliveto, A. Panichella, S. Panichella. ***Defect Prediction as a Multi-Objective Optimization Problem**. *Software Testing, Verification and Reliability (STVR)* 2015.
doi:[10.1002/stvr.1570](https://doi.org/10.1002/stvr.1570)

Journal Publications during the PhD study:

- [10] G. Bavota, G. Canfora, M. Di Penta, R. Oliveto, S. Panichella. ***How the Apache Community Upgrades Dependencies**. *Empirical Software Engineering (EMSE)* 2014.
doi:10.1007/s10664-014-9325-9
- [11] A. De Lucia, M. Di Penta, R. Oliveto, A. Panichella, S. Panichella. ***Applying a Smoothing Filter to Improve IR-based Traceability Recovery Processes: An Empirical Investigation**. *Information and Software Technology (INFOSOF)* 2012.
doi:10.1016/j.infsof.2012.08.002
- [12] A. De Lucia, M. Di Penta, R. Oliveto, A. Panichella, S. Panichella. ***Labeling Source Code with Information Retrieval Methods: An Empirical Study**. *Empirical Software Engineering (EMSE)* 2013. doi:doi:10.1007/s10664-013-9285-5

Journal Publications during the master study:

- [13] G. Capobianco, A. De Lucia, R. Oliveto, A. Panichella, S. Panichella. ***Improving IR-based traceability recovery via noun-based indexing of software artifacts**. *Journal of Software: Evolution and Process (JSE)* 2012.
doi:10.1002/smr.1564

CONFERENCE
PUBLICATIONS

In papers marked with (*) the authors are listed in alphabetic order.

Conference Publications after the Ph.D.:

- [14] Sebastiano Panichella and Marcela Ruiz: **Requirements-Collector: Automating Requirements Specification from Elicitation Sessions and User Feedback**. *IEEE International Requirements Engineering Conference (RE'20)*. -
to Appear.
- [15] Usman Ashraf, Christoph Mayr-Dorn, Alexander Egyed, and Sebastiano Panichella: **A Mixed Graph-Relational Dataset of Socio-technical interactions in Open Source Systems**. *Mining Software Repositories (MSR 2020)*. -
to Appear.
- [16] Muhammad Ilyas Azeem, Sebastiano Panichella, Andrea Di Sorbo, Alexander Serebrenik, and Qing Wang: **Action-based Recommendation in Pull-request Development**. *International Conference on Software and System Processes (ICSSP2020)* -
to Appear.
- [17] Rafael Kallis, Andrea Di Sorbo, Gerardo Canfora and Sebastiano Panichella: **ICSE 2019 - To Appear. Ticket Tagger: Machine Learning Driven Issue Classification**. *35th IEEE International Conference on Software Maintenance and Evolution (ICSME 2019)* -
to Appear.
- [18] Y. Zhou, C. Wang, Y. Xin, T. Chen, Sebastiano Panichella, and H. Gall.: **ICSE 2019 - To Appear. DRONE: A Tool to Detect and Repair Directive Defects in Java APIs Documentation**. *International Conference on Software Engineering, ICSE 2019* -
to Appear.

- [19] Diego Martin, Sebastiano Panichella. **The Cloudification Perspectives of Search-based Software Testing**. *The 12th Int. Workshop on Search-Based Software Testing, 2019* - *to Appear*.
- [20] Carol V. Alexandru; Jose J. Merchante; Sebastiano Panichella; Sebastian Proksch; Harald C. Gall; Gregorio Robles. **On the Usage of Pythonic Idioms..** *Onward! 2018* - *to Appear*.
- [21] S. Panichella. **Summarization Techniques for Code, Change, Testing and User Feedback..** *Proceedings of the IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER 2018)* - <https://doi.org/10.1109/VST.2018.8327148>.
- [22] G. Grano, T. Titov, S. Panichella, H. Gall. **How High Will It Be? Using Machine Learning Models to Predict Branch Coverage in Automated Testing**. *MaLTesQuE (collocated with SANER 2018)* - <https://doi.org/10.1109/MALTESQUE.2018.8368454>.
- [23] L. Pelloni, G. Grano, A. Ciurumelea, S. Panichella, F. Palomba, H. Gall. **BE-CLoMA: Augmenting Stack Traces with User Review Information..** *Proceedings of the IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER 2018)* - <https://doi.org/10.1109/SANER.2018.8330252>.
- [24] A. Ciurumelea, S. Panichella, H. Gall. **Automated User Reviews Analyser..** *Proceedings of the 40th International Conference on Software Engineering (ICSE 2018)* - <http://doi.acm.org/10.1145/3183440.3194988>.
- [25] G. Grano, A. Ciurumelea, S. Panichella, F. Palomba, H. Gall. **Exploring the Integration of User Feedback in Automated Testing of Android Applications..** *Proceedings of the IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER 2018)* - <https://doi.org/10.1109/SANER.2018.8330198>.
- [26] C. Vassallo, S. Panichella, F. Palomba, S. Proksch, A. Zaidman and H. Gall. **Context is King: The Developer Perspective on the Usage of Static Analysis Tools..** *Proceedings of the IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER 2018)* - <https://doi.org/10.1109/SANER.2018.8330195>.
- [27] G. Grano, A. Di Sorbo, F. Mercaldo, C. Visaggio, G. Canfora, S. Panichella. **Android Apps and User Feedback: a Dataset for Software Evolution and Quality Improvement..** *Proceedings of the International Workshop on App Market Analytics (WAMA 2017)*. <http://doi.acm.org/10.1145/3121264.3121266>
- [28] C. Vassallo, G. Schermann, F. Zampetti, D. Romano, P. Leitner, A. Zaidman, M. di Penta, S. Panichella. **A Tale of CI Build Failures: an Open Source and a Financial Organization Perspective..** *Proceedings of the 33rd International Conference on Software Maintenance and Evolution (ICSME 2017)*. *Core RANK: A*. <https://doi.org/10.1109/ICSME.2017.67>

- [29] C. V. Alexandru, S. Panichella, Harald Gall. **Replicating Parser Behavior using Neural Machine Translation**. *Proceedings of the 25th International Conference on Program Comprehension (ICPC 2017)*. Core RANK: C.
<https://doi.org/10.1109/ICPC.2017.11>
- [30] A. Di Sorbo, S. Panichella, C. V. Alexandru, C. A. Visaggio, G. Canfora. **SURF: Summarizer of User Reviews Feedback**. Demonstrations Track of the *39th International Conference on Software Engineering (ICSE 2017)*. Core RANK: A*.
<https://doi.org/10.1109/ICSE-C.2017.5>
- [31] F. Palomba, P. Salza, A. Ciurumelea, S. Panichella, H. Gall, F. Ferrucci, A. De Lucia **Recommending and Localizing Change Requests for Mobile Apps based on User Reviews**. In: *39th International Conference on Software Engineering (ICSE 2017)*. Core RANK: A*.
<https://doi.org/10.1109/ICSE.2017.18>
- [32] Y. Zhou, R. Gu, T. Chen, Z. Huang, S. Panichella, H. Gall. **Analyzing APIs Documentation and Code to Detect Directive Defects**. In: *39th International Conference on Software Engineering (ICSE 2017)*. Core RANK: A*.
<https://doi.org/10.1109/ICSE.2017.11>
- [33] A. Ciurumelea, A. Schaufelbuhl, S. Panichella, Harald Gall. **Analyzing Reviews and Code of Mobile Apps for better Release Planning**. In: *Proceedings of the 24th IEEE International Conference on Software Analysis, Evolution, and Reengineering (SANER) 2017*. Klagenfurt, Austria.
<https://doi.org/10.1109/SANER.2017.7884612>
- [34] C. Alexandru, S. Panichella, Harald Gall. **Reducing Redundancies in Multi-Revision Code Analysis**. In: *24th IEEE International Conference on Software Analysis, Evolution, and Reengineering (SANER) 2017*. Klagenfurt, Austria.
<https://doi.org/10.1109/SANER.2017.7884617>
- [35] S. Panichella, A. Di Sorbo, E. Guzman, C. Visaggio, G. Canfora, H. Gall. **ARdoc: App Reviews Development Oriented Classifier**. In: *24th ACM SIGSOFT International Symposium on the Foundations of Software Engineering* will be held in Seattle, WA, USA. Core RANK: A*.
<http://doi.acm.org/10.1145/2950290.2983938>
- [36] A. Di Sorbo, S. Panichella, C. V. Alexandru, J. Shimagaki, C. A. Visaggio, G. Canfora, H. Gall. **What Would Users Change in My App? Summarizing App Reviews for Recommending Software Changes**. In: *24th ACM SIGSOFT International Symposium on the Foundations of Software Engineering* will be held in Seattle, WA, USA. Core RANK: A.
<http://doi.acm.org/10.1145/2950290.2950299>
- [37] A. Panichella, C. Alexandru, S. Panichella, A. Bacchelli, H. Gall. **A Search-based Training Algorithm for Cost-aware Defect Prediction**. *25th International Conference on Genetic Algorithms (ICGA) and the 21st Annual Genetic Programming Conference (GP) (GECCO 2016)*. Denver, Colorado, USA. Core RANK: A.
<http://doi.acm.org/10.1145/2908812.2908938>
- [38] S. Panichella, A. Panichella, M. Bella, A. Zaidman, H. Gall. **The impact of test case summaries on bug fixing performance: An empirical investigation**. In: *Proceedings of the 38th International Conference on Software Engineering (ICSE 2016)*, Austin, TX. Core RANK: A*.
<http://doi.acm.org/10.1145/2884781.2884847>

- [39] A. Di Sorbo, S. Panichella, C. Visaggio, M. Di Penta, G. Canfora, H. Gall. . **DECA: Development Emails Content Analyzer**. In: *Proceedings of the 38th International Conference on Software Engineering (ICSE 2016)*, Austin, TX. *Core RANK: A**.
<http://doi.acm.org/10.1145/2889160.2889170>
- [40] S. Panichella. **Supporting Newcomers in Software Development Projects**. In: *Proceedings of the 31st International Conference on Software Maintenance and Evolution (ICSME 2015)*. Bremen, Germany. *Core RANK: A*.
<https://doi.org/10.1109/ICSM.2015.7332519>
- [41] A. Di Sorbo, S. Panichella, C. Visaggio, M. Di Penta, G. Canfora, H. Gall. **Development Emails Content Analyzer: Intention Mining in Developer Discussions**. In: *30th international conference on Automated Software Engineering (ASE 2015)*. Lincoln, Nebraska. *Core RANK: A*.
<https://doi.org/10.1109/ASE.2015.12>
- [42] S. Panichella, A. Di Sorbo, E. Guzman, C. Visaggio, G. Canfora, H. Gall. **How Can I Improve My App? Classifying User Reviews for Software Maintenance and Evolution**. In: *Proceedings of the 31st International Conference on Software Maintenance and Evolution (ICSME 2015)*. Bremen, Germany. *Core RANK: A*.
<https://doi.org/10.1109/ICSM.2015.7332474>
- [43] G. Schermann, M. Brandtner, S. Panichella, P. Leitner, H. Gall. **Discovering Loners and Phantoms in Commit and Issue Data**. In: *Proceedings of the 37th International Conference on Program Comprehension (ICPC 2015)*. Firenze, Italy. *Core RANK: C*.
<https://doi.org/10.1109/ICPC.2015.10>
- [44] S. Panichella, V. Arnaoudova, M. Di Penta, G. Antoniol. **Would Static Analysis Tools Help Developers with Code Reviews?**. In: *Proceedings of the 22nd International Conference on Software Analysis, Evolution and Reengineering (SANER 2015)*. Montreal, Canada.
<https://doi.org/10.1109/SANER.2015.7081826>

Conference Publications during the PhD experience:

- [45] S. Panichella, G. Bavota, M. Di Penta, G. Canfora, G. Antoniol. **How Developers' Collaborations Identified from Different Sources Tell us About Code Changes**. In: *Proceedings of the 30th International Conference on Software Maintenance and Evolution (ICSME 2014)*. Victoria, Canada. *Core RANK: A*.
<https://doi.org/10.1109/ICSME.2014.47>
- [46] G. Bavota, S. Panichella, N. Tsantalis, M. Di Penta, R. Oliveto, G. Canfora. **Recommending Refactorings based on Team Co-Maintenance Patterns..** In: *29th international conference on Automated Software Engineering (ASE 2014)*. Vasteras, Sweden. *Core RANK: A*.
<https://doi.org/10.1109/ICSE.2017.18>
- [47] C. Vassallo, S. Panichella, G. Canfora, M. Di Penta. **CODES: mining source Code Descriptions from developer discussions**. In: *Proceedings of the 36th International Conference on Program Comprehension (ICPC 2014)*. Hyderabad, India. *Core RANK: C*.
<http://doi.acm.org/10.1145/2597008.2597799>

- [48] S. Panichella, G. Canfora, M. Di Penta, R. Oliveto. **How the Evolution of Emerging Collaborations Relates to Code Changes: an Empirical Study**. In: *Proceedings of the 36th International Conference on Program Comprehension (ICPC 2014)*. Hyderabad, India. *Core RANK: C*.
<http://doi.acm.org/10.1145/2597008.2597145>
- [49] G. Bavota, G. Canfora, M. Di Penta, R. Oliveto, S. Panichella. ***The Evolution of Project Inter-Dependencies in a Software Ecosystem: the Case of Apache**. In: *Proceedings of the 29th International Conference on Software Maintenance (ICSM 2013)*. Eindhoven, Netherlands. *Core RANK: A*.
<https://doi.org/10.1109/ICSM.2013.39>
- [50] G. Bavota, G. Canfora, M. Di Penta, R. Oliveto, S. Panichella. ***An Empirical Investigation on Documentation Usage Patterns in Maintenance Tasks**. In: *Proceedings of the 29th International Conference on Software Maintenance (ICSM 2013)*. Eindhoven, Netherlands. *Core RANK: A*.
<https://doi.org/10.1109/ICSM.2013.32>
- [51] G. Canfora, M. Di Penta, S. Giannantonio, R. Oliveto, S. Panichella. ***YODA: Young and newOmer Developer Assistant**. In: *Proceedings of the 35th International Conference on Software Engineering (ICSE 2013)*. San Francisco, CA, USA. *Core RANK: A**.
<https://doi.org/10.1109/ICSE.2013.6606710>
- [52] G. Canfora, A. De Lucia, M. Di Penta, R. Oliveto, A. Panichella, S. Panichella. ***Multi-Objective Cross-Project Defect Prediction**. In: *Proceedings of the 7th International Conference on Software Testing, Verification and Validation (ICST 2013)*. Luxembourg. *Core RANK: A*.
<https://doi.org/10.1109/ICST.2013.38>
- [53] G. Canfora, M. Di Penta, R. Oliveto, S. Panichella. ***Who is going to Mentor Newcomers in Open Source Projects?**. In: *Proceedings of the 29th ACM SIGSOFT International Symposium on Foundations of Software Engineering (FSE 2012)*. Cary, North Carolina, USA. *Core RANK: A**.
<http://doi.acm.org/10.1145/2393596.2393647>
- [54] A. De Lucia, M. Di Penta, R. Oliveto, A. Panichella, S. Panichella. ***Using IR Methods for Labeling Source Code Artifacts: Is It Worthwhile?**. In: *Proceedings of the 20th IEEE International Conference on Program Comprehension (ICPC)*, 2012. Passau, Germany. *Core RANK: C*.
<https://doi.org/10.1109/ICPC.2012.6240488>
- [55] S. Panichella, J. Aponte, M. Di Penta, A. Marcus, G. Canfora. **Mining source code descriptions from developer communications**. In: *Proceedings of the 20th IEEE International Conference on Program Comprehension (ICPC)*, 2012. Passau, Germany. *Core RANK: C*.
<https://doi.org/10.1109/ICPC.2012.6240510>
- [56] A. De Lucia, M. Di Penta, R. Oliveto, A. Panichella, S. Panichella. ***Improving IR-based Traceability Recovery Using Smoothing Filters**. In: *Proceedings of the 19th IEEE International Conference on Program Comprehension (ICPC)* 2011. Kingston, ON, Canada. *Core RANK: C*.
<https://doi.org/10.1109/ICPC.2011.34>

Conference Publications during the bachelor and master studies:

- [57] G. Capobianco, A. De Lucia, R. Oliveto, A. Panichella, S. Panichella. ***On the role of the nouns in IR-based traceability recovery**. In: *Proceedings of the 19th IEEE International Conference on Program Comprehension (ICPC)* 2009. Vancouver, British Columbia, Canada. *Core RANK: C*.
<https://doi.org/10.1109/ICPC.2009.5090038>
- [58] G. Capobianco, A. De Lucia, R. Oliveto, A. Panichella, S. Panichella. ***Traceability Recovery Using Numerical Analysis**. In: *Proceedings of the 16th IEEE Working Conference on Reverse Engineering (WCRE)* 2009. Lille, France. *Core RANK: B*.
<https://doi.org/10.1109/WCRE.2009.14>

BOOK CHAPTERS:

1. Harald C. Gall, Carol V. Alexandru, Adelina Ciurumelea, Giovanni Grano, Christoph Laaber, Sebastiano Panichella, Sebastian Proksch, Gerald Schermann, Carmine Vassallo, Jitong Zhao: **Data-Driven Decisions and Actions in Today's Software Development**. *The Essence of Software Engineering 2018*: 137-168

AWARDS

Awards as Reviewer:

1. **Distinguished Reviewer Award SANER 2018**
2. **Distinguished Reviewer Award SATToSE 2017**

Best Paper Awards⁶:

1. G. Grano, T. Titov, S. Panichella, H. Gall. **How High Will It Be? Using Machine Learning Models to Predict Branch Coverage in Automated Testing**. *MaLTeSQuE* (collocated with SANER 2018)
2. **Best paper award**
A. De Lucia, M. Di Penta, R. Oliveto, A. Panichella, S. Panichella. ***Improving IR-based Traceability Recovery Using Smoothing Filters**. In: *Proceedings of the 19th IEEE International Conference on Program Comprehension (ICPC)* 2011. Kingston, ON, Canada. *Core RANK: B*.
3. **Best tool award**
C. Vassallo, S. Panichella, G. Canfora, M. Di Penta. **CODES: mining source cOde Descriptions from developErs diScussions**. In: *Proceedings of the 36th International Conference on Program Comprehension (ICPC 2014)*. Hyderabad, India. *Core RANK: B*.
4. **Best tool award**
L. Pelloni, G. Grano, A. Ciurumelea, S. Panichella, F. Palomba, H. Gall. **BE-CLOMA: Augmenting Stack Traces with User Review Information..** *Proceedings of the IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER 2018)*

⁶ In papers marked with (*) the authors are listed in alphabetic order

NOMINATED AS
BEST PAPER

In papers marked with (*) the authors are listed in alphabetic order.

1. G. Grano, A. Ciurumelea, S. Panichella, F. Palomba, H. Gall. **Exploring the Integration of User Feedback in Automated Testing of Android Applications..** *Proceedings of the IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER 2018)*
2. C. Vassallo, S. Panichella, F. Palomba, S. Proksch, A. Zaidman and H. Gall. **Context is King: The Developer Perspective on the Usage of Static Analysis Tools..** *Proceedings of the IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER 2018)*
3. C. Alexandru, S. Panichella, Harald Gall. **Reducing Redundancies in Multi-Revision Code Analysis.** In: *24th IEEE International Conference on Software Analysis, Evolution, and Reengineering (SANER) 2017. Klagenfurt, Austria. Core RANK: B.*
4. S. Panichella, G. Bavota, M. Di Penta, G. Canfora, G. Antoniol. **How Developers' Collaborations Identified from Different Sources Tell us About Code Changes.** In: *Proceedings of the 30th International Conference on Software Maintenance and Evolution (ICSME 2014).* Victoria, Canada. *Core RANK: A.*
5. S. Panichella, G. Canfora, M. Di Penta, R. Oliveto. **How the Evolution of Emerging Collaborations Relates to Code Changes: an Empirical Study.** In: *Proceedings of the 36th International Conference on Program Comprehension (ICPC 2014).* Hyderabad, India. *Core RANK: C.*
6. G. Bavota, G. Canfora, M. Di Penta, R. Oliveto, S. Panichella. ***The Evolution of Project Inter-Dependencies in a Software Ecosystem: the Case of Apache.** In: *Proceedings of the 29th International Conference on Software Maintenance (ICSM 2013).* Eindhoven, Netherlands. *Core RANK: A.*
7. G. Canfora, A. De Lucia, M. Di Penta, R. Oliveto, A. Panichella, S. Panichella. ***Multi-Objective Cross-Project Defect Prediction.** In: *Proceedings of the 7th International Conference on Software Testing, Verification and Validation (ICST 2013).* Luxembourg. *Core RANK: C.*
8. A. De Lucia, M. Di Penta, R. Oliveto, A. Panichella, S. Panichella. ***Using IR Methods for Labeling Source Code Artifacts: Is It Worthwhile?.** In: *Proceedings of the 20th IEEE International Conference on Program Comprehension (ICPC), 2012.* Passau, Germany. *Core RANK: C.*
9. A. De Lucia, M. Di Penta, R. Oliveto, A. Panichella, S. Panichella. ***Improving IR-based Traceability Recovery Using Smoothing Filters.** In: *Proceedings of the 19th IEEE International Conference on Program Comprehension (ICPC) 2011.* Kingston, ON, Canada. *Core RANK: C.*

PROFESSIONAL
SERVICES AND
EXPERIENCES

Technical Coordinator of EU grants:

- Technical coordinator of the H2020 project "COSMOS: DevOps for Complex Cyber-physical Systems" (recently selected for funding)

Reviewer/opponent of Ph.D. Dissertations:

- Reviewer/opponent of a Ph.D. Dissertation at University of Tartu, Institute of Computer Science (2019/2020)

Keynote Speaker of International Conferences and co-located events:

- Keynote speaker at VST 2018 (co-located to SANER 2018)
(<http://vst2018.scch.at/#program>)

Editor or Co-editor of special Issues at International Journals:

- Editor of a the special Issue at Science of Computer Programming Journal (Elsevier) entitled 'Software Engineering Automation: A Natural Language Perspective', 2020
- Editor of a the special Issue at EMSE entitled 'Software Engineering for Mobile Applications', July 2018.
- Editor of a the special Issue at IST entitled 'User Feedback and Software Quality in the Mobile Domain', June 2018. Link to the guest editorial:
<https://doi.org/10.1016/j.infsof.2019.05.005>

Organising Summer School:

- **1st Summer School on Software Evolution: From Monolithic to Cloud-Native**". Program available at <https://research.tuni.fi/clowee/news/inforte-cloud/>

Organising research workshops:

- **Co-organizer of the CHOOSE-forum 2017**
(<http://www.choose.s-i.ch/events/forum2017/index.html>)

Chair of International Workshop or Tool competitions:

- **Chair** of the Tool Competition at the 13th International Workshop on Search-Based Software Testing (SBST 2020)
- *First International Workshop on Cloud-Native Applications Design and Experience - CNAX 2018 Co-located with UCC 2018 and BDCAT 2018 conferences -, Zurich, Switzerland.*
- *Second International Workshop on Cloud-Native Applications Design and Experience - CNAX 2019 Co-located with UCC 2019 and BDCAT 2019 conferences -.*

Organising committee member of International Conferences:

- Program Committee member of the WAISE 2020 (Third International Workshop on Artificial Intelligence Safety Engineering)
- Program Committee member of the Symposium on Search-Based Software Engineering (SSBSE 2020)
- Program Committee member of the International Conference on Program Comprehension (ICPC 2020, 2017, 2016, 2015, 2014).
- Program Committee member of the IEEE Conference on Software Testing, Validation and Verification (ICST 2020)
- Program Committee member of the International Conference on Mining Software Repositories (MSR 2020, 2019, 2018, 2016)
- Program Committee member of the International Conference on Software Analysis, Evolution and Reengineering (SANER 2020, 2019, 2017)
- Program Committee member of the International Workshop on Search-Based Software Testing (SBST 2020, 2019, 2018)
- Program Committee member of the 3rd International Workshop on App Market Analytics (WAMA 2019)
- Program Committee member of International Conference on Software Maintenance and Evolution (ICSME 2018, 2017).
- Program Committee member of 1st International Workshop on Machine Learning and Software Engineering in Symbiosis.
- Program Committee member of ESEC/FSE 2018 - Formal Demonstration Track.
- Program Committee member of SBST 2018 (11th International Workshop on Search-Based Software Testing), Gothenburg, Sweden.

- Program Committee member of the 40th International Conference on Software Engineering - Student Research Competition (ICSE SRC 2018), Gothenburg, Sweden.
- Expert Review Panel Member of the 32nd IEEE/ACM International Conference on Automated Software Engineering (ASE 2017), Urbana-Champaign, Illinois, USA.
- Program Committee member of the Euromicro Conference on Software Engineering and Advanced Applications (SEAA 2017, 2016, 2015).
- Program Committee member of the 10th Seminar on Advanced Techniques & Tools for Software Evolution” (SATToSE 2017), Madrid, Spain.

Session Chair of International Conferences:

- *of the 24th IEEE International Conference on Software Analysis, Evolution, and Reengineering (SANER 2017 - ERA Track), Klagenfurt, Austria.*
- *at the MSR 2018 - technical session, entitled "APIs and Code", Gothenburg, Sweden.*

Web Chair

- *21st International Conference on Program Comprehension (ICPC 2013), San Francisco, California, USA.*

Editorial Board Member of International Journals:

- *Journal of Software: evolution and process*

Review Board Member of International Journals:

- *Empirical Software Engineering (EMSE).*
- *ACM TOSEM Board of Distinguished Reviewers.*

Reviewer for the following International Journals:

- *Empirical Software Engineering.*
- *Transactions on Software Engineering.*
- *Transactions on Software Engineering and Methodology.*
- *Journal of Systems and Software.*
- *Information and Software Technology.*
- *Journal of Software: Evolution and Process.*
- *Science of Computer Programming.*
- *Journal of Computer Science and Technology.*
- *Communications of the ACM*

Additional reviewer of International Conferences:

- 31st IEEE/ACM International Conference on Automated Software Engineering (ASE 2016), Singapore, Singapore.
- 30th IEEE/ACM International Conference on Automated Software Engineering (ASE 2015), Lincoln, Nebraska, USA.
- 22nd IEEE International Conference on Software Analysis, Evolution, and Reengineering (SANER 2015), Montreal, Canada.

Internships

- *From 27 May 2013 to 27 July 2013 he has been a visiting researcher at the Ecole Polytechnique de Montréal, Canada. Supervisor: Prof. Giuliano Antoniol*

External Reviewer of Grant Applications

- *External Reviewer of projects submitted in the Quebec-Flanders bilateral research cooperation program*

Research Meetings

- *Sebastiano Panichella was invited by the National Institute of Informatics (NII), Japan, to participate in NII Shonan Meeting entitled "Mobile App Store Analytics" (Japan)*
- *Sebastiano Panichella was invited by the Adesso company, Switzerland, to participate in "Adesso Quartalsmeeting" 24th feb 2016 (Zurich).*

GRANTS AND EU PROJECTS **EU projects**

- Sebastiano Panichella wrote an H2020 proposal (as technical coordinator) for the EU H2020-ICT-2018-20 call, entitled COSMOS, contract no. 957254. COSMOS was recently selected for funding by the H2020. Much of the increasing complexity of ICT systems is being driven by the more distributed and heterogeneous nature of these systems, with Cyber-Physical Systems accounting for an increasing portion of Software Ecosystems. This basic premise underpins the COSMOS proposal which focuses on blending best practices DevOps solutions with the development processes used in the CPS context: this will enable the CPS world to deliver software more rapidly and result in more secure and trustworthy systems. COSMOS brings together a balanced consortium of big industry, SMEs and academics which will develop enhanced DevOps pipelines which target development of CPS software. The COSMOS CPS pipelines will be validated against 5 use cases provided by industrial partners representing healthcare, avionics, automotive, utility and railway sectors. These will act as reference use cases when promoting the technology amongst Open Source and standardization communities. For the former a specific community building activity will be performed to stimulate engagement with Open Source; for the latter, the standards experience of the coordinator and partners will be employed to promote COSMOS technologies within heavily regulated sectors where there is an increasing need for well-defined software V&V solutions.

Total H2020 project 5MIL EUR, Sebastiano Panichella got direct funding for 720,000 EUR

Ack: We personal thank Dr. Sean Murphy and Marc Rennhard for the important personal and professional support provided, critical to make the original COSMOS project proposal more convincing.

- Sebastiano Panichella was partially funded with Gabriele Bavota, Gerardo Canfora, Massimiliano Di Penta, in the EU FP7-ICT-2011-8 project Markos, contract no. 317743. Specifically, the MARKOS project aimed to realize the prototype of a service and an interactive application providing an integrated view on the Open Source projects available the on web, focusing on functional, structural and licenses aspects of software code. My effort is focused on implementing relevant aspects of the Software System realized by Markos and and a generate new research results in the field of Software Engineering. Particular effort is spent on analysis of source code to study the evolution of software project to automatically extract reusable components from source code. From the other things I also extract licensing statements from the source code to monitor their evolution and avoid that changes in source code also generate the break of licenses.

SNF projects

- Sebastiano Panichella obtained funding for the SURF-MobileAppsData SNF (No. 200021–166275) project. The goal of the SURF-MobileAppsData project is mining mobile apps data available in app stores to support software engineers in better supporting maintenance and evolution activities for these apps (**Total SNSF (CHF) 349,926**).

Innosuisse participation

- Sebastiano Panichella was Team member of the *MOSAIC Innosuisse project* - Monitored Platform for Container Based Applications / Team member / 01.03.2017.

TALKS GIVEN

International Summer School on Software Engineering 2011

How identify Mentors in software projects? Discussion and perspectives *July 2011*.

FSE 2012

Who is going to Mentor Newcomers in Open Source Projects?, *November 2012*.

ICPC 2012

Mining source code descriptions from developer communications, *June 2012*.

ICSE 2013

YODA: Young and newcOmer Developer Assistant, *May 2013*.

ICSM 2013

Empirical Investigation on Documentation Usage Patterns in Maintenance Tasks, *September*.

CSER 2013 - Concordia University downtown Montral (<http://concordia.ca>)

Supporting Developers, Mining of Software Repositories, *June*.

ICPC 2014

How the Evolution of Emerging Collaborations Relates to Code Changes: an Empirical Study, *June*.

ICPC 2014

CODES: mining souRce cOde Descriptions from developERs diScussions, *June*.

ICMSE 2014

How Developers' Collaborations Identified from Different Sources Tell us About Code Changes, *September*.

ASE 2014

Recommending Refactorings based on Team Co-Maintenance Patterns, *September*.

SANER 2015

Would Static Analysis Tools Help Developers with Code Reviews? *March*.

ICSME 2015

How Can I Improve My App? Classifying User Reviews for Software Maintenance and Evolution, *October*.

ICSME 2015

Supporting Newcomers in Software Development Projects, *October*.

ASE 2015

Development Emails Content Analyzer: Intention Mining in Developer Discussions, *November*.

EOSESE 2015

Textual Analysis or Natural Language Parsing? A Software Engineering Perspective, *December*.

“Adesso Quartalsmeeting” - 2016

Summarization Techniques for Code, Changes, and Testing, *February*.

Invited by Gran Sasso Science Institute, Center of Advanced Studies - 2016

Systematic Mining of Software Repositories, *July*.

ICSE 2016

The Impact of Test Case Summaries on Bug Fixing Performance: An Empirical Investigation, *May*.

FSE 2016

ARdoc: App Reviews Development Oriented Classifier, *November*.

FSE 2016

What Would Users Change in My App? Summarizing App Reviews for Recommending Software Changes, *November*.

ICSE 2017

SURF: Summarizer of User Reviews Feedback., *May*.

ICSE 2017

Analyzing APIs Documentation and Code to Detect Directive Defects, *May*.

VSS 2017

Summarization Techniques for Code, Change, Testing and User Feedback *December*.

VST (collocated with SANER 2018)

Summarization Techniques for Code, Change, Testing and User Feedback. *March*.

SBST 2019 (collocated with ICSE 2019)

DRONE: A Tool to Detect and Repair Directive Defects in Java APIs Documentation. *May*.

ICSE 2019

The Cloudification Perspectives of Search-based Software Testing *May*.

IC2E 2019

Quality and Feedback Techniques in Kubernetes Application Engineering *June*.

Did a talk at Cisco Systems GmbH 2019 - <https://www.meetup.com/it-IT/Microservices-Zurich/events/262000623/>

Cloud-based testing. *July*.

PHD STUDENTS
SUPERVISED

Muhammad Ilyas Azeem, PhD student at Laboratory for Internet Software Technologies, Institute of Software Chinese Academy of Sciences, Beijing 100190, China. .

- *Action-based Recommendation in Pull-request Development* (ICSSP 2020).

Carol V. Alexandru, PhD student at University of Zurich, Switzerland, 2016.

- *A Search-based Training Algorithm for Cost-aware Defect Prediction* (GECCO 2016).

- *What Would Users Change in My App? Summarizing App Reviews for Recommending Software Changes* (FSE 2016).

- *ARdoc: App Reviews Development Oriented Classifier* (FSE 2016).

- *Exploring Deep Learning Techniques for Supporting the Mining of information in Structured and Unstructured Data.*

- *Reducing Redundancies in Multi-Revision Code Analysis* (SANER 2017).

- *Replicating Parser Behavior using Neural Machine Translation* (ICPC 2017).

- *Redundancy-free Analysis of Multi-revision Software Artifacts.* EMSE 2018

- *On the Usage of Pythonic Idioms. Artifacts.* ONWARD 2018

Giovanni Grano, PhD student at University of Zurich, Switzerland, 2017.

- *Testing with Fewer Resources: An Adaptive Approach to Performance-Aware Test Case Generation.* *Transactions on Software Engineering (JSEP)*

- *Branch Coverage Prediction in Automated Testing.* *Journal of Software: Evolution and Process (JSEP).*

- *Exploring the Integration of User Feedback in Automated Testing of Android Applications* (SANER 2018).

- *BECLoMA: Augmenting Stack Traces with User Review Information* (SANER 2018).
- *How High Will It Be? Using Machine Learning Models to Predict Branch Coverage in Automated Testing*. MaLTesQuE (collocated with SANER 2018).
- *Android Apps and User Feedback: a Dataset for Software Evolution and Quality Improvement* (WAMA 2017).

Adelina Ciurumelea, PhD student at University of Zurich, Switzerland, 2016.

- *Exploring the Integration of User Feedback in Automated Testing of Android Applications* (SANER 2018).
- *BECLoMA: Augmenting Stack Traces with User Review Information* (SANER 2018).
- *Recommending and Localizing Code Changes for Mobile Apps based on User Reviews* (ICSE 2017).
- *Analyzing Reviews and Code of Mobile Apps for better Release Planning* (SANER 2017).

Carmine Vassallo, PhD student at University of Zurich, Switzerland, 2016.

- *How Developers Engage with Static Analysis Tools in Different Contexts . Empirical Software Engineering Journal*.
- *A Tale of CI Build Failures: an Open Source and a Financial Organization Perspective* (ICSME 2017).
- *Context is King: The Developer Perspective on the Usage of Static Analysis Tools* (SANER 2018).
- *How Developers Engage with Static Analysis Tools in Different Contexts. Empirical Software Engineering (EMSE)* 2019

Gerald Schermann, PhD student at University of Zurich, Switzerland, 2015.

- *Discovering Loners and Phantoms in Commit and Issue Data* (ICPC 2015).

Andrea Di Sorbo, PhD student at University of Sannio, Italy, 2016.

- *How Can I Improve My App? Classifying User Reviews for Software Maintenance and Evolution* (ICSME 2015).
- *Development Emails Content Analyzer: Intention Mining in Developer Discussions* (ASE 2015).
- *DECA: Development Emails Content Analyzer* (ICSE 2016).
- *What Would Users Change in My App? Summarizing App Reviews for Recommending Software Changes* (FSE 2016).
- *ARdoc: App Reviews Development Oriented Classifier* (FSE 2016).
- *SURF: Summarizer of User Reviews Feedback* (ICSE 2017). - *Android Apps and User Feedback: a Dataset for Software Evolution and Quality Improvement* (WAMA 2017).

MASTER
STUDENTS
SUPERVISED

Bill Bosshard, Master student at University of Zurich, Switzerland, 2019.

Atif Ghulam, Master student at University of Zurich, Switzerland, 2019.

Rafael Kallis, Master student at University of Zurich, Switzerland, 2019.

- *Ticket Tagger: Machine Learning Driven Issue Classification* (ICSME 2019).

Timofey Titov, Master student at University of Zurich, Switzerland, 2017.

- *Branch Coverage Prediction in Automated Testing* (JSEP 2018).

- *BECLoMA: Augmenting Stack Traces with User Review Information* (SANER 2018).

Alessandro Rigamonti, Master student at University of Zurich, Switzerland, 2017.

Develop search-based approaches to better predict change and defect prone classes. Zurich, Switzerland. 2016.

Carmine Vassallo, Master student at University of Sannio, Italy.

CODES: mining source code descriptions from developers discussions. (ICPC 2014)

Te Tan, master student at University of Zurich, Switzerland, 2017.

Advised on a Work on App Store Mining..

Simon Taennler, master student at University of Zurich, Switzerland, 2017.

Advised on a Work on App Store Mining..

BACHELOR
STUDENTS
SUPERVISED

Farul Acibal, bachelor student at University of Zurich, 2018.

Nik Zaugg, bachelor student at University of Zurich, Switzerland, 2018.

Gulshan Kundra, master student at LUT, Finland, 2018.

Ivan Taraca, bachelor student at University of Zurich, Switzerland, 2017.

Tool-support for Test Cases Summaries generator and Enhancements.

Alexander Hofmann, bachelor student at University of Zurich, Switzerland, 2017.

ChangeAdvisor - A tool for Recommending and Localizing Change Requests for Mobile Apps based on User Reviews..

Antonio Galluccio, Bachelor student at University of Zurich, Switzerland, 2017.

Toward Generating Test Case Summaries..

Lucas Pelloni, Bachelor student at University of Zurich, Switzerland, 2017.

- *BECLoMA: Augmenting Stack Traces with User Review Information* (SANER 2018).

Andreas Schaufelbuhl, Bachelor student at University of Zurich, Switzerland, 2016.

Analyzing Reviews and Code of Mobile Apps for better Release Planning. (SANER 2017).

Stefano Giannantonio, Bachelor student at University of Molise, Italy.

YODA: Young and newComer Developer Assistant. (ICSE 2013)

SKILLS,
COMPETENCIES
GAINED DURING
THE PHD

Statistics:

During the PhD experience, because of his work in “Empirical software engineering”, he gained good experience in Statistics (the R environment was the main tool used for such purposes). He widely used several statistical tests (parametric and non) for formulating hypothesis and demonstrating the statistical significance (or superiority) of the proposed techniques.

Main Programming Languages:

He currently uses for his work programming languages like Java (high level), Perl (base level). He is very skilled in scripting languages like R (high level), Matlab (medium

level), Weka, RWeka.

Main Competencies Gained:

1) Machine Learning, Text Analysis and Natural Language Processing

He is an expert in Mining of Software repositories and successfully adopted/conceived tools based on Machine Learning (ADTree, Logistic Regression etc.) methods, Natural Language Processing (Stanford NLP parser, Stanford NLP POS Tagger etc.) techniques and Text Analysis (e.g. Vector Space Model, Latent Dirichlet Allocation, Latent Semantic Indexing Jensen and Shannon Model etc.) techniques. For example, a specific example of application of such competencies is represented by the implementation of the tool **ARdoc** (App Reviews Development Oriented Classifier) which is a Java tool that automatically recognizes natural language fragments in user reviews that are relevant for developers to evolve their applications. Specifically, natural language fragments are extracted according to a taxonomy of app reviews categories that are relevant to software maintenance and evolution. The categories were defined in our previous paper entitled *“How Can I Improve My App? Classifying User Reviews for Software Maintenance and Evolution”* and are: (i) Information Giving, (ii) Information Seeking, (iii) Feature Request and (iv) Problem Discovery. ARdoc implements an approach that merges three techniques: (1) Natural Language Processing, (2) Text Analysis and (3) Sentiment Analysis to automatically classify app reviews into the proposed categories. The purpose of ARdoc is to capture informative user reviews (requesting a new feature, description of a problem, or proposing a solution) and consequently to allow developers to better manage the information contained in user reviews.

2) Genetic Algorithms in SE

His research has yielded approaches to predict future defects in software artifacts based on historical information, thus assisting companies in effectively allocating limited development resources and developers in reviewing each others code changes. Developers are unlikely to devote the same effort to inspect each software artifact predicted to contain defects, since the effort varies with the artifacts size (cost) and the number of defects it exhibits (effectiveness). He adopted Genetic Algorithms (GAs) for training prediction models to maximize their cost-effectiveness. The evaluation of the approach was performed on on two well-known models, Regression Tree and Generalized Linear Model, and predict defects between multiple releases of six open source projects. The achieved results show that regression models trained by GAs significantly outperform their traditional counterparts, improving the cost-effectiveness by up to 240%. Often the top 10% of predicted lines of code contain up to twice as many defects.

3) Social Network Analysis

He is also an expert in Social Network Analysis (SNA) and has successfully used such information for profiling developers/expert in developers' SNA. See for example the papers *How the Evolution of Emerging Collaborations Relates to Code Changes: an Empirical Study* and *Who is going to Mentor Newcomers in Open Source Projects?* and download the related tool **Yoda** (Young and newcOmer Developer Assistant) which is an Eclipse plugin (available in <https://spanichella.github.io/tools.html>) able to profile expert in developers' SNA.

4) Other technologies

Other languages that he used during his academic experience are C, C++, Perl, Scilab, Pascal, Visual basic, Prolog, Lisp, PHP, JSP and Servlet. I also have strong experience with scientific software and tools, such as Matlab, R, Weka, that are widely used to build mathematical models through machine learning techniques (including defect prediction models). Other technologies and tools that he used during the academic years include SVN/GIT and DBMS, PostgreSQL, Gerrit code review Tool.

He works currently without problem with different Operating Systems, like Windows, Mac OS, and Linux (I know very well the Ubuntu distribution).

He is also very familiar with SQL (He currently use for his research work PostgreSQL). He proficiently use GIT/SVN as versioning systems. He also wrote a series of research paper using Latex tool as main reference.

RESEARCH TOOLS IMPLEMENTED **A complete list of implemented Tools and Dataset available at:**
<https://spanichella.github.io/tools.html>.

LANGUAGES Sebastiano Panichella currently speak three languages: Italian (mather tongue), English (*B2*) and German (*A1.2*). He is still studying German for achieving the level B2.

TEACHING EXPERIENCE **TEACHING (UZH and ZHAW) activities & Achievements:**
University of Zurich: - Lecturer and co-lecturer for the Software Maintenance and Evolution course in 2014, 2015, 2016, 2017, 2018, 2019, 2020
Learning Goals: During the course Sebastiano teach to the students the foundations of software evolution and maintenance, by integrating recent research in both cloud computing and software engineering fields, thus transferring to students also this recent research outputs (in form of papers, datasets, tools and prototypes). This includes successful aged (i.e. legacy software) or cloud-based software systems, object-oriented reengineering, refactoring, change patterns, empirical analysis of software, classification/prediction models, software quality analysis.
Zurich University of Applied Science:
 - Cloud Computing course - CCP2 2020
 - INF-Prog1 2020
 - Co-lecturer for the CAS Information Engineering in 2018, 2019, 2020
Learning Goals: The main features of the Python program language.
 - Lab Instructor for the Programming course in Java in 2018, 2019, 2020
Learning Goals: The main features of the Java program language.

University of Sannio:
 - *Lab Instructor* (December 2013) for the Programming Techniques course of Professor Gerardo Canfora
Learning Goals: The Languages and Grammars, JavaCC parser.
 - *Teaching Assistant* for the Software Engineering course of Prof. Massimiliano Di Penta:
Learning Goals: Recovering Traceability Links via Information Retrieval Methods

Teaching Achievements. Dr. Panichella received positive lectures grades, improving them over the time, as commented by some students following his course at the UZH:

- "Dear Sebastiano, Thank you for the mail [...] I really appreciate that you were so supportive during my project ... I am super happy with the result, as it is actually a working system that I can use even outside of the mostly virtual space of a typical UZH project...". - "Sebastiano promote during the course critical thinking on the teached research topics".

PROFESSIONAL MEMBERSHIPS

- IEEE Membership (2011–present)
- ACM Membership (2019–present)

REFERENCES

Harald C. Gall, Ph.D (e-mail: gall@ifi.uzh.ch)

- Full Professor, [University of Zurich](#)
- ◊ Binzmuhlestrasse 14, CH-8050 Zurich (Switzerland).
- ★ *Dr. Gall, co-author of some publications at the University of Zurich.*

Gerardo Canfora, Ph.D (e-mail: canfora@unisannio.it)

- Full Professor, [University of Sannio](#)
- ◊ Palazzo ex Poste, Via Traiano, I-82100 Benevento (Italy).
- ★ *Dr. Canfora*

Andrian Marcus, Ph.D (e-mail: amarcus@utdallas.edu; phone: (972-883-4246))

- Associate Professor, [University of Texas at Dallas](#)
- ◊ 800 W. Campbell Road; MS EC31 Richardson, TX 75080 U.S.A.
- ★ *Dr. Marcus, member of dissertation committee and co-author of some publications.*

Giuliano Antoniol, Ph.D (e-mail: antonio1@ieee.org)

- Professor, [Ecole Polytechnique de Montreal](#)
- ★ *Dr. Antoniol, co-author of some publications.*

Andy Zaidman, Ph.D (e-mail: A.E.Zaidman@tudelft.nl)

- Professor, [University of Delft](#)
- ★ *Dr. Zaidman, co-author of some publications.*

(Advisor During the PhD) Massimiliano Di Penta, Ph.D (e-mail: dipenta@unisannio.it)

- Associate Professor, [University of Sannio](#)
- ◊ Palazzo ex Poste, Via Traiano, I-82100 Benevento (Italy).
- ★ *Dr. Di Penta, PhD adviser.*

Andrea De Lucia, Ph.D (e-mail: adelucia@unisa.it; phone: +39 089963376)

- Professor, [University of Salerno](#)
- ◊ Via Giovanni Paolo II, 132 - 84084 Fisciano (SA), Italy.
- ★ *Prof. De Lucia, co-author of some publications.*