Together We Are Stronger: Evidence-Based Reflections on Industry-Academia Collaboration in Software Testing

Published in:
Software Quality

Document Version:
Peer reviewed version

Queen's University Belfast - Research Portal:
Link to publication record in Queen's University Belfast Research Portal

Together We Are Stronger: Evidence-Based Reflections on Industry-Academia Collaboration in Software Testing

Michael Felderer[1] and Vahid Garousi[2]

[1] University of Innsbruck, Austria
michael.felderer@uibk.ac.at
[2] Queen's University Belfast, UK
v.garousi@qub.ac.uk

Abstract. For a highly relevant and applied research area like software testing industry-academia collaboration is of uttermost importance. In this paper we reflect on how industry-academia collaboration can be improved based on evidence from four empirical studies. We therefore first present four studies providing evidence on the (1) perceived level of challenges in testing activities, (2) focus areas in industrial and academic software testing conferences, (3) synergies between industrial and academic software testing conferences, as well as (4) the need for consideration of grey literature. Then, we reflect on issues, which we think can improve the link and synergies between industry and academia in software testing, i.e., research topics, guidelines and evidence, value and risk, context and scalability, action research and education as well as grey literature and open science.

Key words: software testing; software quality; industry academia collaboration

1 Introduction

In an applied research area like software engineering industrial impact and relevance are crucial [1]. This holds especially for software testing, which is an area of high scientific and practical importance as it comprises a critical set of activities to enable the development of high-quality software and systems [2].

This paper provides reflections on industry-academia collaboration in software testing, which aim to improve collaboration and as a follow-up impact of research in specific contexts and its transfer to practice. For that purpose, we first present studies providing evidence on industry-academia collaboration especially in software testing, but also its conferences and literature. We think that a holistic strategy to improving industry-academia collaboration requires to not only consider research topics, but also the research ecosystem, which includes conferences and literature, but also education.

This paper is structured as follows. Section 2 provides an overview of test activities. Section 3 provides available specific evidence on industry-academia collaboration in software testing as well as in its conferences and literature. Section 4 presents reflections on industry-academia collaboration based on the provided evidence. Finally, Section 5 concludes the paper.

2 Software Test Activities

According to the international testing standard ISO/IEC/IEEE 29119 [3], software testing comprises a set of activities conducted to facilitate discovery and/or evaluation of properties of one or more test items, i.e., a software system or parts of it. Following [4] we consider the nine test activities test-case design (criteria-based), test-case design (based on human expertise), test scripting, test execution, test evaluation, test-result reporting, test management, test automation, and other test activities. These test activities are defined in Table 1.

Activity Type	Description
Test-case design (criteria-based)	Designing test suites (set of test cases) or test requirements to satisfy coverage criteria, e.g., line coverage
Test-case design (based on human expertise)	Designing test suites (set of test cases) based on human expertise (e.g., exploratory testing) or other engineering goals
Test scripting	Documenting test cases in manual test scripts or automated test code
Test execution	Running test cases on the system under test (SUT) and recording the results
Test evaluation	Evaluating results of testing like assigning test verdicts
Test-result reporting	Reporting test verdicts and defects to developers, e.g., via defect (bug) tracking systems
Test management	Encompasses activities related to test management, e.g., planning, control, monitoring, etc.
Test automation	Automating any test activity
Other testing activities	Includes activities other than those discussed above, e.g., regression testing or test prioritization

Table 1. Definition of test activity types [4]

3 Evidence on Industry-Academia Collaboration in Software Testing as well as in its Conferences and Literature

Software testing is an important area when investigating industry-academia collaboration in software engineering as it is a common topic covered in papers

on industry-academia collaboration in software engineering in general as shown by a recent systematic literature review [5]. However, there are only a few studies focusing on industry-academia collaboration in software testing and providing evidence in that area. In this section we present four studies providing evidence on the perceived level of challenges in testing activities (Study 1), focus areas in industrial and academic software testing conferences (Study 2), synergies between industrial and academic software testing conferences (Study 3), as well as the need for consideration of grey literature (Study 4).

3.1 Study 1: Perceived Level of Challenges in Testing Activities

Garousi et al. [4] performed a survey among experienced practitioners to find out what industry wants from academia in software testing. For that purpose, practitioners were asked for the perceived level of challenges in testing activities as well as concrete research topics related to testing activities (see Table 1).

Figure 1 shows the the level of challenges in each of the nine testing activities as perceived by the 105 participants of the survey.

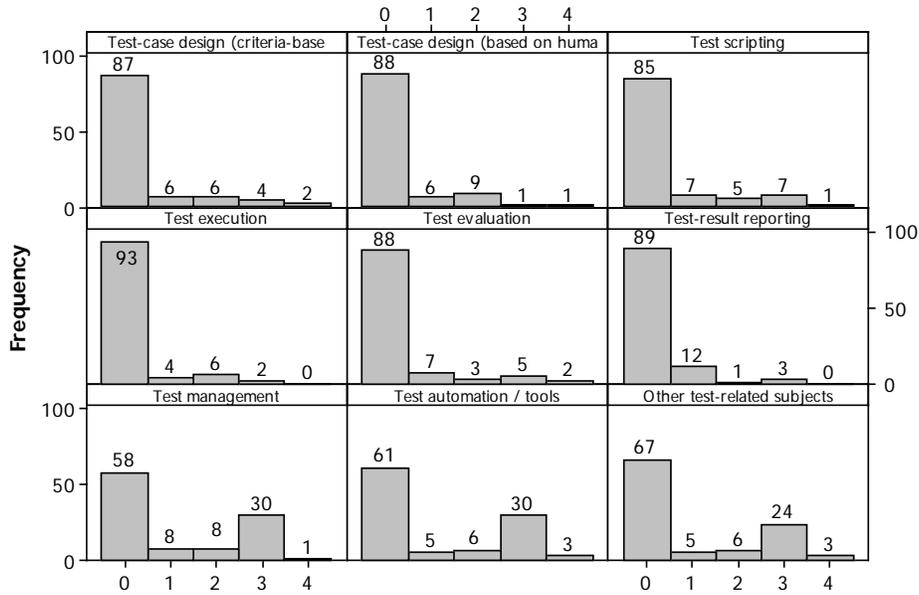**Fig. 1.** Level of challenges in each testing activity (0=no challenges at all, . . . , 4=lots of challenges) [4]

Only for the activities of test management, test automation and other testing activities a significant amount of challenges is perceived by the participants. For these three activities, for instance the following concrete practically relevant topics, where solutions from research are required, are mentioned:

– *Test management*: test ROI calculation; risk metrics; test size and effort esti-
  mation; balance between test efficiency and effectiveness
– *Test automation*: metrics for test automation; fully-automated test script gen-
  eration; usability of test automation; automated recommendation support for
  test execution
– *Other activities*: regression testing of complex legacy software; adoption of
  open source tools; test training and awareness

3.2 Study 2: Focus Areas in Industrial and Academic Software Testing Conferences

Garousi and Felderer [6] compared presentation titles from several leading indus-
trial and academic conferences. The top three terms were "automation", "mobile",
and "agile" in industrial presentation titles and "model", "combinatorial", and
"automated" in academic presentation titles, respectively. Both communities ap-
pear to focus on test automation. However, the presentation titles reveal that
when practitioners refer to test automation, they mostly mean automating test
execution. In contrast, academics focus mostly on automating other activities like
test-case design or evaluation. Further industrial focus areas are test management
and other activities like domain-specific aspects (e.g., mobile, robotics or cloud
testing) as well as non-functional aspects like performance testing. This finding
is inline with the industrial challenges raised in Fig. 1. Common presentation
topics at academic conferences are search-based test-case design, combinatorial
testing, mutation testing and model-based testing, i.e. topics at academic testing
conferences seem to be mainly related to automated test-case design.

3.3 Study 3: Synergies between Industrial and Academic Software Testing Conferences

Beszedes and Vidacs [7] compiled a comprehensive list of 63 academic and 38
industrial conferences that focus on testing aspects and analyzed their industry-
academia synergies. For that purpose, they compute an index which considers
chairs, presenters and keynotes from the other community (i.e., industry and
academia, respectively) to measure synergies.

  Figure 2 shows the presence of industrial chairs, PC members and keynote
speakers in academic conferences and vice versa. Notable findings from this
comparison are as follows:

– Academic conferences more often invite industrial members as program chairs
  than industrial conferences invite academic ones
– Synergies regarding the membership of program committees are similar
– There are more keynotes with industrial background on academic conferences
  than academic keynotes on industrial ones
– Considering other factors as well, most notably the existence of a synergistic
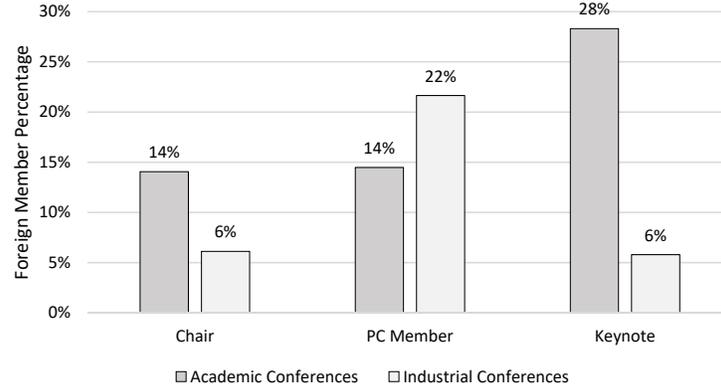 track, academic conferences show more ambitions to synergies

Fig. 2. Presence ratio of industrial members in academic conferences and vice versa) [7]

– In case of academic conferences, it seems that younger conferences are more industry oriented than the more mature ones

Testing: Academia-Industry Collaboration, Practice and Research Techniques (TAIC PART) and Software Quality Days (SWQD) are rated and highlighted as distinguished venues to foster industry-academia collaboration in software testing and examples of mutual recognition between the two communities.

3.4 Study 4: Need for Consideration of Grey Literature

Garousi et al. [8] investigated the need for multivocal literature reviews in software engineering. Multivocal literature reviews [9] take grey literature into account in addition to formally published academic literature. Grey literature is typically neither formally peer-reviewed nor formally published, which implies uncertainty of the status of the covered information. Grey literature sources comprise for instance blog-like documents, videos, and white papers and are often written by pracitioners to share their knowledge and opinions. The authors investigate, mainly based on examples from the domain of software testing, what types of knowledge are missed and what the software engineering community can gain when explicitly considering grey literature in literature studies.

For instance, Fig. 3, which is covered by the study, shows based on a multivocal literature review on test process assessment and improvement [10], what information from practice would have been missed if we were to exclude grey literature sources. Overall, 57 different test process assessment and improvement models were identified in the formal and grey literature sources of the performed multivocal literature review. From these sources 14 were grey literature reporting test maturity models such as TMap, Agile TMM or Test Maturity Index which would have been lost in a regular systematic literature review (by not including the grey literature). Furthermore, Fig. 3 shows the number of papers per model using or extending a source model. Without grey literature, the usage of TMap and some other models would not have been considered.
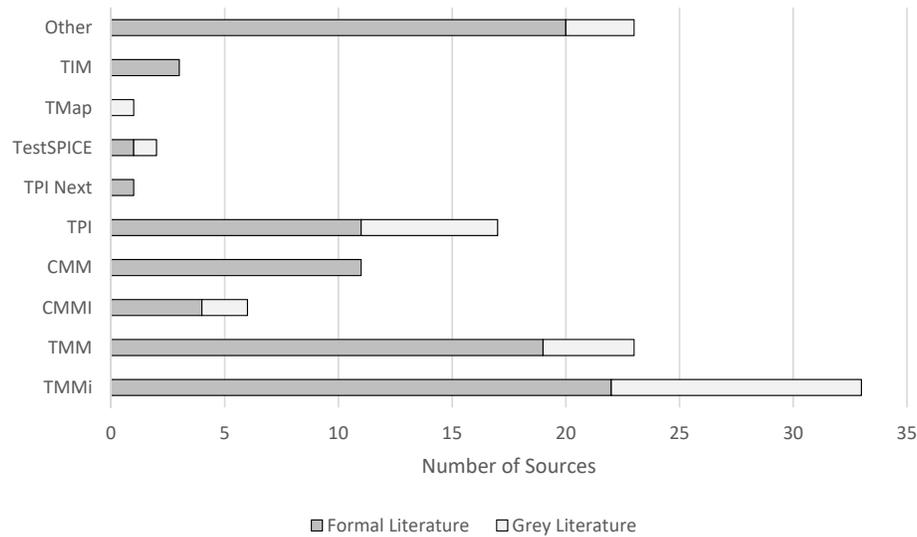
Fig. 3. Formal and grey literature per test process improvment model [10]

Based on a second multivocal literature review on test automation [11], also covered in the study by Garousi et al. [8], it is once more shown that a significant body of experience and knowledge from practicing test engineers on the topic of test automation would have been missed. To put this in quantitative terms, in the multivocal literature review factors to be considered for deciding when and what to automate in testing were analyzed by the type of source where they were mentioned. In total, 15 factor categories were identified, grey literature sources contributed a total of 219 occurrences of factors, while academic sources discussed only 67 occurrences. Furthermore, two factor categories (test oracle and development process) would not have been identified in the study if grey literature would not have been considered.

Based on these study, we can conclude that grey literature covers a large amount of knowledge, opinions and experience from practitioners and it can contribute substantially to the body of knowledge and open challenges in certain areas of software testing.

4 Reflections on Industry-Academia Collaboration in Software Testing

Based on the results of the studies and our own experiences we reflect on some issues that we consider relevant to benefit from industry-academia collaboration, especially in software testing, and to further improve the link and synergies between industry and academia.

Research Topics Researchers should be aware of their industrial partners' challenges and take into account research topics that are novel, feasible, industrially relevant, and potentially impactful [12]. As highlighted in [4], problems, where practitioners require support from research, are especially related to test management and test automation. Garousi and Herkiloğlu [13] even propose a process to select suitable topics for industry-academia collaborations in software testing.

Guidelines and Evidence Researchers can benefit a lot from available guidelines (e.g., technology transfer models [14] or patterns and anti-patterns [5]) and evidence on industry-academia collaboration in software engineering [15]. For instance, the first author of this paper was inspired by an available technology transfer model [14] when planning and performing research on testing with defect taxonomies in close collaboration with industry [16].

Value and Risk Companies are under continuous pressure to make profit. Hence, researchers should take the business value into account and consider ideas from value-based software engineering [17]. Related to the concept of value is the concept of risk [18], which can be considered in all test activities [19]. For instance, risk (and therefore also value aspects) can be considered in test generation and selection approaches, which allows a much better business framing even of sophisticated test-generation approaches.

Context and Scalability Software testing research has to become more context-driven by focusing on problems driven by concrete needs in specific domains and development projects. The applicability and scalability, which are essential for industrial accpetance and adoption of research results, depend largely on contextual factors, whether human (such as engineers' background), organizational (such as cost and time constraints), or domain-related (such as the level of criticality and compliance with standards) [20]. This also requires that top-journals and conferences acknowledge that context-driven research is needed, valuable, and challenging. Such research focuses especially on scalability in realistic conditions, and therefore brings essential contributions for an applied research field like software testing.

Action Research and Education Action research [21], which seeks transformative change through the simultaneous process of taking action and doing research, is especially useful when collaborating with industry. It supports that the research problems are based on real industry needs and that research results are actually adopted in testing practice. Action research is often linked to education of practitioners. Especially software testing is considered as highly important, but with a considerable knowledge gap existing [22].

Grey Literature and Open Science The dissemination of results, i.e., papers, artifacts and data, plays an important role in industry-academia collaboration. In addition, to research papers, researchers could try describing their research in laymen's terms, making it available in more forms than conference articles and journal papers. Knowledge should be made available through various forms of

grey literature, e.g., by publishing slides, white papers, recorded videos, and blogs. On the other, hand researchers should also consider grey literature sources from practitioners as valuable sources to identify challenges, cases or even evidence. Researchers should also apply principles of open science [23], whose aim is to render all artifacts borne out of scientific research activities accessible, without any barriers, to any individual on Earth.

However, one should for sure also take into account that it is not achievable and even not desirable that industry and academia always collaborate very tightly. Wohlin [24] addresses this issues by defining five levels of of closeness between industry and academia, i.e., not in touch, hearsay, sales pitch, offline, and one team. One should be aware that industry has for obvious reasons shorter cycle times, which implies moving targets and may pose impediments for long-term investigations and innovations researchers are aiming for. However, in an applied research area such as software testing, there should always be a link between industry and academia, which may vary in in its intesnity. We hope that the raised issues support to keep continuously keep this link.

5 Conclusion

In this paper we first presented four studies providing evidence on the perceived level of challenges in testing activities (Study 1), focus areas in industrial and academic software testing conferences (Study 2), synergies between industrial and academic software testing conferences (Study 3), as well as the need for consideration of grey literature (Study 4). We then reflect on the following issues, which we think can improve the link and synergies between industry and academia in software testing: research topics, guidelines and evidence, value and risk, context and scalability, action research and education as well as grey literature and open science. Through papers such as this one, we're continuing our effort to bring practitioners and researchers in software testing closer to each other so that they can benefit each other much more than they do today.

6 Acknowledgments

References

1. Lo, D., Nagappan, N., Zimmermann, T.: How practitioners perceive the relevance of software engineering research. In: Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering, ACM (2015) 415–425
2. Harrold, M.J.: Testing: a roadmap. In: Proceedings of the Conference on the Future of Software Engineering, Citeseer (2000) 61–72

3. ISO: 29119-1-2013 - ISO/IEC/IEEE international standard - software and systems engineering –software testing –part 1:concepts and definitions. (2013)

4. Garousi, V., Felderer, M., Kuhrmann, M., Herkiloğlu, K.: What industry wants from academia in software testing?: Hearing practitioners' opinions. In: Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering, ACM (2017) 65–69

5. Garousi, V., Petersen, K., Ozkan, B.: Challenges and best practices in industry-academia collaborations in software engineering: A systematic literature review. Information and Software Technology **79** (2016) 106–127

6. Garousi, V., Felderer, M.: Worlds apart: industrial and academic focus areas in software testing. IEEE Software **34**(5) (2017) 38–45

7. Beszédes, Á., Vidács, L.: Academic and industrial software testing conferences: Survey and synergies. In: 2016 IEEE Ninth International Conference on Software Testing, Verification and Validation Workshops (ICSTW), IEEE (2016) 240–249

8. Garousi, V., Felderer, M., Mäntylä, M.V.: The need for multivocal literature reviews in software engineering: complementing systematic literature reviews with grey literature. In: Proceedings of the 20th international conference on evaluation and assessment in software engineering, ACM (2016) 26

9. Garousi, V., Felderer, M., Mäntylä, M.V.: Guidelines for including grey literature and conducting multivocal literature reviews in software engineering. Information and Software Technology **106** (2019) 101–121

10. Garousi, V., Felderer, M., Hacaloğlu, T.: Software test maturity assessment and test process improvement: A multivocal literature review. Information and Software Technology **85** (2017) 16–42

11. Garousi, V., Mäntylä, M.V.: When and what to automate in software testing? a multi-vocal literature review. Information and Software Technology **76** (2016) 92–117

12. Begel, A., Zimmermann, T.: Analyze this! 145 questions for data scientists in software engineering. In: Proceedings of the 36th International Conference on Software Engineering, ACM (2014) 12–23

13. Garousi, V., Herkiloglu, K.: Selecting the right topics for industry-academia collaborations in software testing: an experience report. In: 2016 IEEE International Conference on Software Testing, Verification and Validation (ICST), IEEE (2016) 213–222

14. Gorschek, T., Garre, P., Larsson, S., Wohlin, C.: A model for technology transfer in practice. IEEE software **23**(6) (2006) 88–95

15. Garousi, V., Pfahl, D., Fernandes, J.M., Felderer, M., Mäntylä, M.V., Shepherd, D., Arcuri, A., Coşkunçay, A., Tekinerdogan, B.: Characterizing industry-academia collaborations in software engineering: evidence from 101 projects. Empirical Software Engineering (2019) 1–63

16. Felderer, M., Beer, A.: Mutual knowledge transfer between industry and academia to improve testing withdefect taxonomies. Software-engineering and management 2015 (2015)

17. Ramler, R., Biffl, S., Grünbacher, P.: Value-based management of software testing. In: Value-based software engineering. Springer (2006) 225–244

18. Felderer, M., Schieferdecker, I.: A taxonomy of risk-based testing. International Journal on Software Tools for Technology Transfer **16**(5) (2014) 559–568

19. Felderer, M., Ramler, R.: Integrating risk-based testing in industrial test processes. Software Quality Journal **22**(3) (2014) 543–575

20. Basili, V., Briand, L., Bianculli, D., Nejati, S., Pastore, F., Sabetzadeh, M.: Software engineering research and industry: a symbiotic relationship to foster impact. IEEE Software **35**(5) (2018) 44–49
21. Santos, P.S.M.d., Travassos, G.H.: Action research use in software engineering: An initial survey. In: Proceedings of the 2009 3rd International Symposium on Empirical Software Engineering and Measurement, IEEE Computer Society (2009) 414–417
22. Garousi, V., Giray, G., Tüzün, E., Catal, C., Felderer, M.: Aligning software engineering education with industrial needs: a meta-analysis. Journal of Systems and Software (2019)
23. Fernández, D.M., Graziotin, D., Wagner, S., Seibold, H.: Open science in software engineering. arXiv preprint arXiv:1904.06499 (2019)
24. Wohlin, C.: Software engineering research under the lamppost. In: ICSOFT. (2013) IS–11