Industry-Academia Research Collaborations During and After COVID-19

Dusica Marijan
Simula Research Laboratory
Norway
dusica@simula.no

Chetan Bansal
Microsoft Research
USA
chetanb@microsoft.com

Tamara Lopez
The Open University
UK
tamara.lopez@open.ac.uk

DOI: 10.1145/3485952.3485957
https://doi.org/10.1145/3485952.3485957

ABSTRACT

There exists a long-standing challenge of building successful research collaborations between industry and academia in software engineering. There are multiple reasons contributing to this issue, including different timelines, impact metrics, expectations, and perceptions of researchers and practitioners, altogether leading to the issue known as the *industry-academia collaboration gap*. After the onset of the COVID-19 pandemic, most researchers and practitioners were sent to work from home, relying on virtual collaboration with their peers, which might have given rise to an even wider industry-academia collaboration gap. At the 8th International Workshop on Software Engineering Research and Industrial Practice, held virtually at the International Conference on Software Engineering 2021, we investigate the impact that working from home has had on research collaborations between industry and academia. We also take a look at how remote work may change research collaborations in the future. In this report, we present takeaways from keynote talks on this matter, share insights from the panel of six experts on the topic of industry-academia collaborations, and finally summarize findings from related paper presentations.

Categories and Subject Descriptors

D.3.2 [**Software Engineering**]

General Terms

Management, Measurement, Economics, Experimentation, Human Factors, Legal Aspects.

Keywords

Technology transfer, innovation, industry-academia collaboration, research collaboration, software engineering, COVID-19, social distancing, work-from-home, remote work.

1. INTRODUCTION

Despite numerous efforts over the years to develop effective models for industry-academia collaborations [1][2][3], research partnerships between software engineering academia and software industry have remained challenging due to the number of reasons. For example, it is often difficult to make the output of academic research more appealing to practitioners [4], however, without accessing such research, practitioners may miss out on innovative ideas for improving their daily practice. Similarly, there is a disconnect between what problems researchers find interesting to focus on and what solutions practitioners find useful. Researchers often care about creating publishable and generalizable results, while practitioners very much care about the solution being fit for their industrial context, which implies narrowed generalizability. Furthermore, industry and academia usually operate on different time-scales [5], and it is often difficult to evaluate the success of software engineering research in practice [6], given that objective measures are hard to implement, while subjective measures may be prone to bias. These and similar issues have the potential to widen the industry-academia collaboration gap, especially during the times of remote work, which became the norm for many organizations during the COVID-19 pandemic.

Software Engineering Research and Industrial Practice (SER&IP) workshop series is a dedicated forum to discuss the challenges of industry-academia collaborations. Specifically, this year, the workshop focused on the ways of maintaining existing and building new research collaborations challenged by remote work brought upon by COVID-19.

The 8th SER&IP workshop was held at the virtual International Conference on Software Engineering (ICSE) 2021. The workshop featured two eminent keynote talks from industry and academia, 10 paper presentations, and the panel session. The panelists discussed strategies for *socially distant* university-academia collaborations. This report summarizes the findings and discussions from the workshop, with the goal to provide a better understanding of how to improve remote research collaborations for the coming hybrid way of work.

2. TAKEAWAYS FROM KEYNOTE TALKS

2.1 Emulating zebras and oxpeckers: developing a practitioner-academic mutualistic relationship

Prof Ita Richardson from Lero, the Irish Software Engineering Research Centre summarized best practices for developing a practitioner-academic mutualistic relationship, by making an analogy with zebras and oxpeckers [7]. Oxpeckers land on or zebras' back and get food by eating parasites that live on zebras' skin, while zebras get pest control. In her talk, Prof Richardson pointed out that industry and academia, much like zebras and oxpeckers, need to develop a relationship where they can make use of each other in a way that provides mutual benefits. She exemplified her experience with two types of research projects she participated in – exploratory research and action research. The exploratory research project, in their context, was organized around industry asking questions to be addressed in the collaboration, which ultimately lead to new action research projects.

In conclusion, the keynote reflected on various ways by which academics and practitioners can work together to provide mutual benefits. She discussed initiatives such as workshops, action research, case studies, commercialization, and industry fellowships, showing how these have been successful both for industry at a software engineering and commercial level, and for academics, through significant research outputs.

2.2 Work-from-home during and after COVID 19

Dr Sonia Jaffee from Microsoft reflected on the experience of work-from-home during and after COVID 19 [8]. Dr. Jaffe insights are based on a large cross-company study investigating how remote work has affected individuals, teams, and companies. The study used a combination of interviews, surveys, telemetry data, and diary notes for data collection. The initiative consisted of over 50 research projects, conducted by teams that span a range of disciplines (including engineering, research, marketing, human resources, and facilities) and divisions (including Microsoft Research, Office, Windows, Azure, Xbox, GitHub, and LinkedIn). In terms of short term productivity measures, the reported benefits of working from home include focused time, less interruptions, and less time spent on commute, but on the flip side people are missing social interactions, have less motivation, and are experiencing poor office setup. Furthermore, as offices provide spatial boundaries and commutes provide temporal boundaries between life and work, there is a challenge to replace such spatial and temporal boundaries with technological boundaries.

The keynote concluded, that as for the broader implications of remote work, it can worsen existing inequalities, because offices are traditionally considered equalizers. Furthermore, females are found to be more likely to leave work to do child-care. For the work situation going forward, the study found that 2/3 of workers support hybrid work, with a mixture of office and home-work.

3. INSIGHTS FROM THE PANEL

The panel session titled *Strategies for "Socially Distant" University-Company Collaborations*, and moderated by Steven Fraser from Innoxec, discussed how COVID-19 inspired innovation ecosystem changed university-industry collaborations. The panelists included Sheri A. Brodeur, Director of Corporate Relations at MIT, Randy Katz, Vice-Chancellor for Research at UC Berkeley, Xue (Steve) Liu from McGill University, and Sheng-Ying (Aithne) Pao from NTHU Taiwan.

The panelists shared their perspectives on the number of topics, including how has the fidelity of relationships changed, have there been any negative consequences of remote work for industry-academia collaborations, funding support from the government, implications on (global) talent acquisition, the impact on online education, etc. The panelists concluded that industry-university ecosystem is very robust and despite the challenges brought by COVID-19, there remained quality interactions between the two communities, and there will remain to be hybrid means of interaction between the two as we go forward.

4. SUMMARIES OF PAPER PRESENTATIONS

4.1 What we can learn from how programmers debug their code

Though researchers have proposed numerous techniques to support debugging, Hirsch and Hofer [9] claim they are not frequently taken up in practice. To understand more about debugging in practice, the authors conducted an online survey with 102 developers recruited through LinkedIn and personal networks. Participants were asked to share information about a recent bug fixing experience, answering questions about the bug's impact and root cause, the changes that were made, the time spent and perceived difficulty. The authors found that roughly 36% of the reported bugs could be fixed with a single-line change, that semantic errors due to the inconsistency of requirements or a disconnect between intention and implementation were more plentiful but required less time to fix than memory or concurrency errors. In the main, participants indicated that finding faults took more time than fixing them.

One key takeaway from the study is the observations made by Hirsch and Hofer is that participants overwhelmingly employed a "natural reasoning process" of replication, observation and deduction when undertaking bug-fixing tasks. To promote uptake in practice, Hirsch and Hofer argued that tools should support this process, and offered a number of other suggestions to tool developers, recommending that tools provide focused support for fault localization, and to integrate bug-fixing support with IDEs. For researchers working in similar areas, the data for this study are available at: https://doi.org/10.5281/zenodo.4449045

4.2 Towards a Systematic Engineering of Industrial Domain-Specific Languages

Domain-specific languages (DSLs) enable practitioners to contribute to engineering solutions in a number of industrial contexts. In this paper Gupta et al. [10] argue that prior research addresses the reuse of languages or produces guidelines for modelling, but does not comprehensively address the needs of industry users. These users do not frequently perform modelling tasks, and therefore need usable, reusable components that provide support for common modelling tasks within their domains. To address these challenges, the authors presented an industry report about their effort to build a systematic approach to develop and customize graphical DSLs. Within the Building Block Approach, developers define constraints and methodical steps, and help users refine business requirements. Developers then create a DSL Building block definition and visual representation of elements that can be extended and adapted to meet different requirements. The approach was evaluated in a series of focus groups run with industrial partners and researchers. Focus group participants affirmed that the solution facilitates the re-use of DSL components and provides greater support to users that includes documentation and tailored representations and notations that support use. In future, the authors intend to identify ways to further improve the user experience (UX) for domain experts, to make the approach accessible to automation, and to facilitate cross-domain reuse.

4.3 Globally Distributed Development during COVID-19

In this paper, NicCanna et al. [11] describe how Ocucu Ltd, a medium-sized organization with headquarters in Ireland, managed software development teams distributed throughout Ireland, Europe, Asia and America during the COVID-19 pandemic in 2020. After conducting a series of interviews and administering a survey to employees in March, 2020 the authors designed a series of interventions to address and promote wellness within the workforce, as they worked from home during the pandemic. The analysis of the data identified concerns among employees in three areas. Personal concerns included the need to manage home and work life and the requirement to support social interactions for newly hired and newly remote workers; technology concerns were raised about the infrastructure in the workplace and in homes, and; concerns were identified about the difficulty in managing the social aspects of the software development process at a distance. The authors subsequently made changes in the company to address the personal concerns, finding that technology and process changes were minimal, an advantage of the company's prior experience in managing globally distributed teams.

An evaluation of the changes made in 2020 suggests that productivity increased, new customers were gained, and employees reported feeling well supported by management. The authors provide a series of recommendations to small, medium and large organizations wishing to mitigate the negative effects of remote working. These include being flexible, providing proactive, supportive leadership, offering financial support to workers and finding ways for people to stay connected.

4.4 Use and Perceptions of Multi-Monitor Workstations: A Natural Experiment

A working environment that includes multiple monitors is commonly considered to improve productivity among software developers. To investigate this received wisdom, Amir et al. [12] conducted a survey with 101 professional and student developers working at home during the COVID-19 pandemic. The survey asked users to comment on their use and perceptions of multi-monitor workstations. Additional questions were asked to allow the authors to compare experiences in different situations, asking about the number of computer monitors and physical setup at home, the environment participants had in the workplace, and perceptions of how productivity was impacted by the changes in working environment brought by the COVID-19 pandemic. Analysis showed that developers perceive that the use of multiple monitors benefits their productivity and identified that having two monitors in the workplace is common. Participants also indicated that one monitor is used more than the other, and that they typically assign specific tasks to different monitors.

The study also provided a look at working environments in the home during the pandemic. In the home environment, fewer participants reported having access to two monitors, and desk sizes were smaller, due in part to limitations on space and cost. The authors found that while working at home resulted in an increase in distractions that were not work related, participants also reported that it was the lack of interaction with co-workers that had the most profound effect on productivity. Interactions with co-workers, while sometimes a source of distraction, were perceived by participants to be useful and helpful, providing compelling insight into the factors that contribute to a truly optimal work environment for developers.

4.5 All Researchers Should Become Entrepreneurs

In this paper, Cabot et al. [13] challenged researchers to take an entrepreneurial approach to tool development as a way to overcome long standing difficulties in forming and maintaining strong research-industry collaborations. Drawing upon. their experiences in industrializing Xatkit, a model-based approach to chatbot development, the authors propose that researchers adopt a commercial open-source business model. In typical collaborations, researchers are not directly connected to users and depend upon the committed engagement of a small or medium enterprise (SME). In contrast, by following a commercial open-source model, prototypes are released on an open-source platform, improvements are made to make the software usable, a community of use can be grown, and a commercial extension can be spun off for commercial users. In this way, the authors claim, the research community stands to benefit. A commercial product requires researchers to make robust technological choices that meet the needs of actual industrial clients, and gives them access to immediate, direct feedback on the utility and potential impact of their work.

4.6 Exploring the Dimensions of University - Company Collaborations: Research, Talent, and beyond in a Chaotic COVID-19 World

University-company partnerships are complex and fragile. In this paper, Fraser et al. [14] explored the benefits of university-company collaborations beyond research and talent – primarily from a US and Canadian perspective. To build effective and enduring partnerships, authors discuss various scenarios: incubating collaborations, connecting experts, assessing and communicating collaborative value, and growing the relationship. A key takeaway from the paper is that university-company partnership benefits have many "dimensions" beyond just research innovation and talent acquisition. A company that is able to leverage other dimensions (marketing collateral, sales, policy influence, government incentives) will build a more profitable collaboration and will produce more business value for the partners over time. If a partnership is unbalanced, with little or no rewards or incentives, participants will eventually disengage. Through this paper, the authors stimulated discussion and incubated new opportunities for valued university-company research collaborations.

4.7 Issue Auto-Assignment in Software Projects with Machine Learning Techniques

Issue assignment is a laborious and error-prone task which is mostly done manually today. In this paper, Oliveira et al. [15] presented an industrial case study on using machine learning to automate the issue assignment process in a large electronic company. The authors proposed intelligent models that could correctly assign new issues according to the teams' responsibility by leveraging historical data. To make it successful, the authors showed it is important to have a well-defined project, iterative processes, and an open communication channel between researchers and practitioners. Further, it is also crucial to develop additional features to support service maintenance and investigate the models' deterioration to define the suitable retraining frequency. The authors also demonstrated that even when accuracy is not so high, automated issue assignment can result in significant saving of time and effort for the teams. This case study provides several invaluable insights which will be useful for anyone developing machine learning models for automating software engineering tasks in an industrial setting.

4.8 Can GraphQL Replace REST? A Study of their Efficiency and Viability

In this paper, Vadlamani et al. [16] explored whether GraphQL is a viable alternative API architecture over Representational State Transfer (REST). The authors measured and compared the time performance efficiency of the two API architectures in terms of request response; and, by assessing their benefits and weaknesses, and gathering developer perspectives. REST has traditionally been the standard web service architectural style for API creation. However, its popularity has been challenged with the introduction of GraphQL, an open source query language for APIs introduced by Facebook. GraphQL has been quickly adopted by GitHub, Shopify, Airbnb, Twitter and more products are joining the list. While GraphQL promises a considerable improvement over REST, the authors showed that much remains unexplored with respect to its efficiency and feasibility in its application. A custom API client on GitHub was developed by the authors to check on the response times and the corresponding magnitude of difference between REST and GraphQL. Thereafter, they surveyed employees of GitHub to understand software developers' educated opinion and perceptions about REST and GraphQL based on their practical experience with APIs. The authors concluded that both API paradigms have their benefits and weaknesses, and one cannot replace the other, at least in the near future.

4.9 Leveraging Data Scientists and Business Expectations during the COVID-19 Pandemic

In this paper, Monteiro et al. [17] shared insights and experiences of a relatively new data science team during the COVID-19 pandemic. At the time the pandemic started, the team was positioning itself as a Center of Excellence in Advanced Analytics. With the pandemic, it had to keep up with the expectations from the stakeholders; manage current and upcoming data science projects within the methodology practiced in IT; and maintain a high level in the quality of service delivered. This paper discussed how the COVID-19 pandemic affected the team productivity and its practices as well as the lessons learned with it. Shortly before the pandemic, the team was starting to have an increased exposure within the organization and implementing new data science projects while it attempted to manage the expectations from the stakeholders as well as pressure coming both internally and externally. Due to its work culture, it embraced the adoption of face-to-face communication instead of an extensive documentation repository. With the pandemic and by forcibly working from home, this model proved to be a challenge. The data science team relied on the adoption of formalized processes and the heavy usage of collaboration tools to keep up with the demand. These tools included messaging and voice applications and digital Kanban boards. The insights shared by the authors will be useful even beyond the pandemic for remote and geographically distributed teams.

4.10 On the Gap Between Software Maintenance Theory and Practitioners' Approaches

In this paper, Ferreira et al. [18] investigated the gap between software maintenance techniques proposed by the research community and the actual practices. The authors carried out a survey with 112 practitioners from 92 companies and 12 countries. They concentrated on analyzing if and how practitioners understood and applied the following subjects: bad smells, refactoring, software metrics, and change impact analysis. The study showed that there is a large gap between research approaches and industry practice in those subjects, especially in change impact analysis and software metrics. There is a lack of proper tool support to perform change impact analysis. Moreover, bad smells and software metrics are the less known and applied concepts. The authors' study also revealed that participants considered the lack of system documentation, lack of development patterns, and legacy software as the leading software maintenance challenges. The results indicated that software maintenance demands even more community effort to develop and provide proper tools and methods for software maintenance, especially in change impact analysis and software measurement.

5. CONCLUSION

SER&IP 2021 brought together software engineering researchers and industry practitioners to discuss the impact of COVID-19 on industry-academia research collaborations. In particular, the workshop participants discussed what benefits and drawbacks working from home brought relative to the productivity and collaboration success, and what have we learned during the times of remote work that can be useful going forward and building new collaborations. The overall message is that while remote work brought some unique challenges not present in the traditional workplace, it also brought opportunities. While human touch has been lost, the barriers of time and distance have been reduced.

Overall, different perspectives on the impact of remote work on industry-academia research collaborations shared by the keynotes, panelists, and paper presenters were engaging and interesting for the participants from both industry and academia. We hope that the workshop attendees gained useful learning from the workshop discussions, that they can take forward in organizing their industry-academia collaborations in the future.

6. ACKNOWLEDGMENTS

We thank to all the participants of the workshop, the keynote speakers, the authors and paper presenters for their contribution to the workshop. We thank to the members of the SER&IP 2021 program committee for helping prepare a great program, as well as the organizers of the ICSE 2021 conference, and the ICSE 2021 Workshop Chairs.

7. REFERENCES

[1] T. Gorschek, P. Garre, S. Larsson, C. Wohlin, A model for technology transfer in practice, IEEE software, 88-95, 2006.

[2] D. Marijan, Arnaud Gotlieb, Industry-Academia research collaboration in software engineering: The Certus model, Information and Software Technology, 132: 106473 (2021).

[3] V. Garousi, D. Pfahl, J. Fernandes, M. Felderer, M. Mantyla, D. Shepherd, Characterizing industry-academia collaborations in software engineering: evidence from 101 projects, Empirical Software Engineering, 24 (2019), pp. 2540-2602.

[4] D. Marijan, S. Sen, Good Practices in Aligning Software Engineering Research and Industry Practice, *ACM SIGSOFT Software Engineering Notes*, 44(3): 65-67 (2019).

[5] D. Marijan, T. Zimmermann, M. Ham, Bridging Software Engineering Research and Industrial Practice, *ACM SIGSOFT Software Engineering Notes*, 46(1): 30-32 (2021).

[6] D. Marijan, W. Shang, R. Shukla, Implications of Resurgence in Artificial Intelligence for Research Collaborations in Software Engineering, *ACM SIGSOFT Software Engineering Notes*, 44(3): 68-70 (2019).

[7] I. Richardson, Emulating zebras and oxpeckers: developing a practitioner-academic mutualistic relationship, Proceedings of the 8th Int. Workshop on Software Engineering Research and Industrial Practice, 2021.

[8] S. Jaffee, Work-from-home during and after COVID 19, Proceedings of the 8th Int. Workshop on Software Engineering Research and Industrial Practice, 2021.

[9] T. Hirsch, B. Hofer, What we can learn from how programmers debug their code, Proceedings of the 8th Int. Workshop on Software Engineering Research and Industrial Practice, 2021.

[10] R. Gupta, S. Kranz, N. Regnat, B. Rumpe, A. Wortmann, Towards a Systematic Engineering of Industrial Domain-Specific Languages, Proceedings of the 8th Int. Workshop on Software Engineering Research and Industrial Practice, 2021.

[11] C. NicCanna, M.A.Razzak, J. Noll, S. Beecham, Globally Distributed Development during COVID-19, Proceedings of the 8th Int. Workshop on Software Engineering Research and Industrial Practice, 2021.

[12] G. Amir, A. Prusak, T. Reiss, N. Zabari, D.G. Feitelson, Use and Perceptions of Multi-Monitor Workstations: A Natural Experiment, Proceedings of the 8th Int. Workshop on Software Engineering Research and Industrial Practice, 2021.

[13] J. Cabot, H. Bruneliere, G. Daniel, A. Gómez, All Researchers Should Become Entrepreneurs, Proceedings of the 8th Int. Workshop on Software Engineering Research and Industrial Practice, 2021.

[14] S. Fraser, D. Mancl, Exploring the Dimensions of University-Company Collaborations: Research, Talent, and Beyond in a Chaotic COVID-19 World, Proceedings of the 8th Int. Workshop on Software Engineering Research and Industrial Practice, 2021.

[15] P. Oliveira, R. Andrade, T. Nogueira, I. Barreto, L. Bueno, Issue Auto-Assignment in Software Projects with Machine Learning Techniques, Proceedings of the 8th Int. Workshop on Software Engineering Research and Industrial Practice, 2021.

[16] S. Vadlamani, B. Emdon, J. Arts, O. Baysal, Can GraphQL Replace REST? A Study of Their Efficiency and Viability, Proceedings of the 8th Int. Workshop on Software Engineering Research and Industrial Practice, 2021.

[17] W. Monteiro, M. Prado, G. Reynoso-Meza, Leveraging Data Scientists and Business Expectations during the COVID-19 Pandemic, Proceedings of the 8th Int. Workshop on Software Engineering Research and Industrial Practice, 2021.

[18] M. Ferreira, M. Bigonha, K. Ferreira, On The Gap Between Software Maintenance Theory and Practitioners' Approaches, Proceedings of the 8th Int. Workshop on Software Engineering Research and Industrial Practice, 2021.