Building Stronger Bridges: Strategies for Improving Communication and Collaboration Between Industry and Academia in Software Engineering

Rico, Sergio

2023

Document Version:
Publisher's PDF, also known as Version of record

Building Stronger Bridges: Strategies for Improving Communication and Collaboration Between Industry and Academia in Software Engineering

Sergio Rico

Doctoral Dissertation, 2023
Department of Computer Science
Lund University

This thesis is submitted to the Research Education Board of the Faculty of Engineering at Lund University, in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Engineering.

LU-CS-DISS: 2023-04
Doctoral Dissertation 73, 2023
ISBN 978-91-8039-661-5 (printed version)
ISBN 978-91-8039-662-2 (electronic version)
ISSN: **1404-1219**

Department of Computer Science
Lund University
Box 118
SE-221 00 Lund
Sweden

Email: sergio.rico@cs.lth.se
WWW: `http://cs.lth.se/`

Printed in Sweden by Tryckeriet i E-huset, Lund, 2023

ABSTRACT

Background: The software engineering community has expressed growing concern regarding the need for more connections between research and practice. Despite the large amount of knowledge researchers generate, its impact on real-world practice is uncertain. Meanwhile, practitioners in industry often struggle to access and utilize relevant research outcomes that could inform and enhance their work. Collaboration between industry and academia is seen as a potential solution to bridge this gap, ensuring that research remains relevant and applicable in real-world contexts.

Objective: This research aims to explore challenges in communication and collaboration between industry and to design, evaluate, and implement strategies that foster this collaboration.

Methodology: The design science paradigm inspires this research, as we aim to obtain knowledge about industry-academia communication and collaboration by studying challenges and solutions in context. The thesis includes case studies; some are exploratory, while others focus on evaluating specific strategies.

Results: In terms of problem understanding, we identified challenges that impact communication and collaboration, such as different expectations, perspectives, and ways of working. Furthermore, we pinpointed factors facilitating communication, including long-term projects, research relevance, and practitioners' involvement. Regarding how to improve communication and collaboration, we investigated two strategies. The first strategy involves using the SERP-taxonomy approach in a project on software vulnerability management in IoT systems. The second strategy involves the proposal of interactive rapid reviews, conducted in close collaboration with practitioners. We share the lessons from conducting two reviews (one in testing machine learning systems and the other in software component selection). The benefits of conducting interactive rapid reviews include mutual understanding, the development of networks, and increased motivation for further studies.

Conclusion: The thesis emphasizes the importance of industry-academia collaboration as a key aspect in closing gaps between research and practice. The strategies discussed provide tools to understand industry-academia partnerships better and support future collaborations.

POPULAR SUMMARY

Sergio Rico, Department of Computer Science, Lund University, Sweden

When we think about society's challenges, like the COVID-19 pandemic, we count on academia and industry to help us to face them. We rely on academic institutions to provide the scientific knowledge and expertise to understand the virus and develop vaccines and treatments. Similarly, we expect universities and other academic institutions to have trained medicine, public health, epidemiology, and healthcare professionals. Meanwhile, we expect industry to produce vaccines, treatments, and medical equipment and supplies. Essentially, we want scientific knowledge to be transformed into real-world solutions that can help us tackle the pandemic and require exchange between academia and industry.

Moving to the IT industry, the exchange between academia and companies is a key component of the innovation that has led to the development of disruptions like semiconductors, the internet, communication protocols, and, most recently, artificial intelligence (AI). In software engineering, the exchange between academia and industry is essential due to the importance of software in our daily lives, and because teams developing software are the best place to study and improve how software is developed.

One way to foster innovation and strengthen the industry-academia exchange is through collaboration, where researchers and practitioners work together to solve problems and create new knowledge. Although collaborative projects can vary in size, scope, level of collaboration, and duration, they have common characteristics, including the project stages, the roles of the participants, and the challenges that can arise during the project.

This thesis identifies challenges that make collaboration between industry and academia difficult in software engineering and proposes strategies to overcome them. Effective communication between researchers and practitioners is critical, as they are the main actors in these partnerships.

To conduct our research, we studied real projects where companies and academic research groups collaborated. We analyzed projects at different stages, from the project's inception to the final evaluation of results. Consequently, we proposed Interactive Rapid Reviews (IRRs) to increase interaction between researchers and

practitioners. IRRs involve answering questions from practitioners based on research results.

Our research has two main contributions. First, it offers a comprehensive view of challenges faced by researchers and practitioners grouped into three categories: communication distances, participants-related challenges, and research-related challenges. This comprehensive view aims to help researchers and practitioners collaborate more effectively.

Second, our thesis proposes two strategies to overcome some challenges. Develop and use common terminologies to describe and connect what researchers do and use Interactive Rapid Reviews (IRRs) to increase interaction between researchers and practitioners in the early stages of collaborative research projects. Inspired by rapid reviews in medicine, where research results inform medical treatments and practices, we adapted the rapid review methodology for software engineering. In two case studies, we found that IRRs were useful in developing a shared understanding of each other's needs and problems and piloting how to work together.

This research provides valuable insights into improving communication and collaboration between industry and academia in software engineering. Our ultimate goal is to create more synergies between these two sectors and ultimately impact the software products and services we use daily.

ACKNOWLEDGEMENTS

This research was funded by the Swedish government through the strategic research environment ELLIIT.

At the end of this stage in my career, it is an excellent opportunity to express my gratitude:

To society in general, for keep believing in science and research and to create the conditions to do it. It is thanks to taxpayers that Universities, research groups, and research funding exist. To the people and institutions of Sweden for enabling me to conduct my research in a free and open environment and for placing their trust in researchers. I believe our work pays back.

To my supervisors, Emelie Engström and Martin Höst, for being such masters and examples of researchers. I particularly value their understanding and integrity among their many qualities as individuals and researchers. I am grateful for the years spent together, during which I have learned from their example. I hope to give back to the community and society as they have done.

To my research group, the Software Engineering Research Group (SERG), I am grateful for providing a nurturing environment where the social aspect of software engineering is a top priority and where we deeply care about how research is conducted. SERG has been my academic home for the last five years, a place where we've conducted research and teaching. It has been a privilege not only to learn about conducting research but also about being a researcher. I want to extend my gratitude to Professors Björn Regnell, Martin Höst, and Per Runeson for their guidance and support. Teaching software engineering, requirements, and testing was a rewarding experience. I thank Markus, Alma, and Elizabeth for sharing teaching duties with me. I am also thankful to my colleagues in the group - Adha, Daniel, Masoumeh, and Song - for the enriching FIKA times and stimulating discussions.

To my co-authors for generously sharing their wisdom, feedback, and collaborative spirit. Thanks to Elizabeth Bjarnason, working and learning from your experience was fun. A special shout-out to Nauman bin Ali for his availability and invaluable insights.

A los Rico, Rangel, y Ordóñez por ser un ejemplo del poder de la educación para transformar vidas. A mis papas Moncho y Roslay, mis hermanos Jonnatan,

Carlos Mario y Laura. También a nuestros amigos que han hecho nuestra vida más divertida en Suecia y a todos quienes nos han visitado y desde Colombia nos siguen compartiendo su cariño. Sentimos su calidez tan necesaria en los días fríos.

To Juliana, Sergio Luis, and Hilda, for such a journey of unforgettable experiences. We have had both tough and good moments. Yet, we keep the good energy, vibes, love, and fun.

Tack! Gracias! Thanks!
Sergio Rico– Lund, 2023

LIST OF PUBLICATIONS

This thesis comprises an introduction and a compilation of six papers. The introduction provides an overview of the research topic and briefly describes the papers and their contributions. The papers are listed below:

Papers included in the thesis

I **A case study of industry-academia communication in a joint software engineering research project**
Sergio Rico, Elizabeth Bjarnason, Emelie Engström, Martin Höst and Per Runeson
Journal of software: Evolution and Process, 33(10), 2021.
DOI: 10.1002/smr.2372

II **Challenges and strategies in industry collaboration for Ph.D. students in software engineering**
Sergio Rico, Martin Höst, Emelie Engström and Nauman bin Ali
Under review in a conference, 2023.

III **A taxonomy for improving industry-academia communication in IoT vulnerability management**
Sergio Rico, Emelie Engström and Martin Höst
45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), 2019.
DOI: SEAA.2019.00014

IV **Guidelines for conducting interactive rapid reviews in software engineering–from a focus on technology transfer to knowledge exchange**
Sergio Rico, Nauman bin Ali, Emelie Engström and Martin Höst
Technical report, 2020.
DOI: 10.5281/zenodo.4327725

V **Exploring ML testing in practice – Lessons learned from an interactive rapid review with Axis Communications**

Qunying Song, Markus Borg, Emelie Engström, Håkan Ardö and Sergio Rico
1st International Conference on AI Engineering – Software Engineering for AI. 2022.
DOI: 10.1145/3522664.3528596

VI **Experiences from conducting interactive rapid reviews – two industrial cases.**
Sergio Rico, Nauman bin Ali, Emelie Engström and Martin Höst
Under review in a journal, 2023.
DOI: SSRN.4367321

CONTRIBUTION STATEMENT

The papers presented in this thesis are the result of collaborative efforts among researchers. All the co-authors have reviewed and approved the final versions of the papers. My individual contributions to each paper are detailed below:

Paper I: As the first author, I collaborated with other authors to design the study. I conducted the survey, and the second author prepared and lead the retrospective meeting. Along with the second author, I analyzed the data and was primarily responsible for writing the paper. Co-authors provided input and edits.

Paper II: As the first author, I designed the study and survey, conducted the data analysis, and wrote most of the paper. Co-authors provided feedback and validation for the research. I was primarily responsible for writing the paper, with input and edits from co-authors.

Paper III: As the first author, I collaborated with co-authors to design the study and co-led the development of the taxonomy with the second author. I was primarily responsible for writing the paper, with input and edits from co-authors.

Paper IV: As the first author, I proposed the initial idea for the study and conducted the literature review to design the interactive rapid reviews. I discussed the design with co-authors and was responsible for describing the method in the paper.

Paper V: As the fifth author, I contributed to the methodology for the study, based on the method proposed in Paper IV. I participated in discussions about the paper and contributed to writing the section about the method. Co-authors 1-3 were responsible for most of the writing, with input and editing from co-author four and myself.

Paper VI: As the first author, I proposed the idea for the study, designed and conducted the interviews, and was responsible for data analysis and writing most of the paper. Co-authors contributed by writing sections of the paper and participated in discussions about analysis and interpretation.

CONTENTS

INTRODUCTION

1 Introduction

Software has become a crucial element of modern society. Often described as ubiquitous [29], it permeates various aspects of our lives. The applications range from everyday devices and industrial systems to critical systems that ensure the operation of essential services. With the increasing importance of digitalization, automation, IoT, and the integration of artificial intelligence components, software relevance, and impact continue to expand. Ensuring that software is developed and maintained reliably, securely, efficiently, and in a manner that meets stakeholders' needs is a significant challenge. Software engineering aims to accomplish this by systematically applying scientific and engineering principles to software development, operation, and maintenance. Establishing a strong connection between academia and industry in software engineering is essential to ensure that academic research is well-informed by real-world challenges and that industry benefits from the latest advances in software engineering research [9].

The symbiotic relationship between academia and industry is crucial, as it enables the exchange of knowledge and expertise between the two sectors [3]. In this thesis, academia refers to higher education and research institutions where researchers advance the frontiers of knowledge and understanding through rigorous methods. Academic institutions are responsible for developing programs and curricula that prepare a critical mass of professionals to drive innovation and progress in various fields, including software engineering. Conversely, we refer to the industry as the domain of commerce, business, production, and public institutions working with software, where companies and organizations play a central role in creating jobs and providing essential goods and services. Numerous links and interactions exist between academia and industry, such as the transfer of technology and knowledge, recruitment, and the provision of training opportunities [37]

Industry-academia collaboration links the two sectors, enabling them to work together on research projects that combine academic research results, knowledge

exchange, and industrial needs. Although collaboration in the software engineering research community has gained attention [19,31,39], there is a need for clearer approaches, strategies, tools, and guidance to support industry-academia collaboration effectively. In response to this need, this thesis focuses on industry-academia communication and collaboration in software engineering. The research seeks to facilitate better integration of academic research and industry practice, ultimately benefitting both sectors. The thesis has two main goals, as follows:

RG1: To explore the challenges of industry-academia collaboration in software engineering and better understand the communication between researchers and practitioners.

- **RQ 1.1**: What are the key challenges researchers and practitioners face when collaborating on industry-academia projects in software engineering?
- **RQ 1.2**: What factors influence communication and collaboration between researchers and practitioners in industry-academia collaborations in software engineering?
- **RQ 1.3**: How do Ph.D. students in software engineering experience and overcome challenges in industry-academia collaborations?

RG2: To design and evaluate strategies to improve communication and collaboration between industry and academia in software engineering.

- **RQ 2.1**: How effective is the SERP-taxonomy approach in supporting communication about practical challenges and research results in software engineering?
- **RQ 2.2**: How can interactive rapid reviews support communication and collaboration between researchers and practitioners in industry-academia collaborations in software engineering?
- **RQ 2.3**: What are the benefits and limitations of using interactive rapid reviews in industry-academia collaborations in software engineering, and how can the approach be improved?

Industry-academia collaboration in software engineering has the potential to bring benefits to both academia and industry, including enhancing research relevance and impact, improving the quality of education, research utilization, and strengthening engagement between the two sectors [1, 10]. Despite this potential, effective collaboration between academia and industry remains challenging [19,31]. Therefore, this thesis's first goal is to better understand the factors that influence communication and collaboration in software engineering and how these challenges can be addressed. Although all the included papers in this thesis contribute to understanding the challenges, papers I and II primarily contribute by

characterizing researcher-practitioner communication and exploring how Ph.D. students experience and overcome communication and collaboration challenges when working with practitioners.

The second goal of this thesis is to design and evaluate strategies to improve communication and collaboration between academia and industry. The proposed strategies aim to promote knowledge exchange and collaboration between researchers and practitioners. Specifically, Paper III uses a framework to link research results with practitioners' challenges. Paper IV proposes interactive rapid reviews (IRR) as a way to improve communication and collaboration (Papers IV, V, VI). By evaluating and refining these proposed strategies, this thesis seeks to contribute to the ongoing efforts to improve industry-academia collaboration to the end of closing gaps between research and practice.

The findings of this thesis have practical implications for industry-academia collaborations, specifically in the planning and execution of research projects. For example, the identified success factors for communication and the strategies to overcome challenges can guide researchers and practitioners in establishing effective collaborations. The proposed strategies, such as interactive rapid reviews and SERP taxonomies, can also serve as tools for improving communication and collaboration between the two sectors. However, it is important to note that the effectiveness of these solutions may depend on specific collaboration settings, and further research is necessary to explore their generalizability. Overall, this thesis provides valuable insights into how industry and academia can collaborate more effectively to generate research outcomes that are more impactful and relevant to real-world problems.

This introduction (*kappa*) is structured as follows: Section 2 provides an overview of the research approach, explaining the methods and processes used throughout the thesis. In Section 3, the background and related work are presented, focusing on industry-academia collaboration and secondary studies in software engineering. The results of the included papers are summarized in Section 4, showcasing the main findings and their implications. Section 5 highlights the contributions made by this research. The threats to validity are discussed in Section 6. Potential future work is outlined in Section 7, offering and finally, Section 8, concludes the thesis.

1.1 List of Papers

This thesis is based on the following papers:

I. Paper I is a case study that explores communication and collaboration in a joint project. The study focuses on identifying the context where researchers and practitioners communicate and finding factors that influence communication and collaboration.

II. Paper II investigates the challenges faced by Ph.D. students when collaborating with practitioners in industry-academia collaborations in software engineering. The study identifies impactful challenges and strategies to overcome them based on a literature review, author suggestions, and a questionnaire completed by Ph.D. students.

III. Paper III focuses on developing a taxonomy, SERP-MENTION, to support communication and collaboration between industry and academia in IoT vulnerability management. The study employs the SERP architecture and multiple steps, including a review of existing taxonomies, interviews, and a workshop.

IV. Paper IV introduces a set of guidelines for conducting interactive rapid reviews in software engineering to facilitate knowledge exchange between researchers and practitioners. The method consists of five main steps: preparation, involvement, search, analysis, and dissemination. The process emphasizes close collaboration between researchers and practitioners.

V. Paper V focuses on establishing a common understanding of the machine learning testing domain by initiating collaboration between industry and academia. The study utilized an interactive rapid review (IRR) and the SERP taxonomy to support communication and collaboration.

VI. Paper VI presents two case studies that applied Interactive Rapid Reviews (IRRs) to promote knowledge exchange between researchers and practitioners in software engineering. The two case studies, Case-SoftSelection and Case-MLTest, explored software component selection and machine learning testing, respectively.

1.2 Communication and Collaboration

The terms communication and collaboration play a significant role in this thesis. However, given that there are no universal definitions for these concepts, we provide an overview of how they are employed in this work.

Communication refers to exchanging information, knowledge, ideas, and messages between individuals or groups. This exchange can occur through various means, such as face-to-face conversations, written messages, or other communication forms. The primary objective of communication is to create and foster a shared understanding among people. Examples of communication include a senior researcher and a manager defining a research project topic, a Ph.D. student collecting data through interviews, a researcher and a practitioner discussing study results, and a researcher presenting at a developer's conference. These examples illustrate that communication may happen within a collaborative project or independently.

Collaboration, on the other hand, refers to the process of working collectively with others to accomplish a common goal or objective. This process typically involves coordination and cooperation between individuals or groups and may entail sharing resources, expertise, or other assets to achieve a shared purpose. In this thesis, we concentrate on industry-academia collaborations in software engineering. The studies included in this thesis were conducted in various contexts, such as university-industry partnerships to develop new technologies in software testing, a collaboration between a research group with companies on tool development, industrial Ph.D. projects, and researchers working with practitioners on literature reviews on machine learning testing. These examples illustrate the diverse nature of collaboration in terms of formality, financing, duration, levels of commitment, and other aspects. Regardless of these variations, the underlying principle of industry-academia collaboration is the joint pursuit of a common goal by researchers and practitioners.

This thesis focuses on examining communication and collaboration between researchers and practitioners in software engineering, particularly within the context of industry-academia partnerships. Working together, researchers and practitioners with diverse backgrounds, experiences, and perspectives exchange knowledge and expertise, creating unique opportunities and challenges for communication and collaboration. By exploring the interactions between researchers and practitioners in these partnerships, we aim to gain a deeper understanding of the challenges and to describe strategies to overcome them.

2 Research Approach

The research approach of this thesis is inspired by the design science research paradigm [41]. Within this paradigm, the primary goal is to gain knowledge about how to solve similar problems by investigating solutions in context. The four main activities of this approach are problem conceptualization, solution design, solution implementation, and evaluation, with the research process being iterative.

In terms of problem conceptualization, knowledge is gained through the various studies conducted within this thesis, primarily in Papers I and II. These papers explore the challenges in industry-academia communication and collaboration in software engineering. One of the proposed solutions, presented in Paper IV, is interactive rapid reviews as a potential solution to address communication and collaboration challenges in industry-academia partnerships in software engineering. Although the solutions proposed are not context-specific, their implementation for evaluation purposes is tailored to each specific situation, aiming that the acquired knowledge can be generalized and applied to various contexts.

The thesis further evaluates SERP (in Paper III) taxonomies and IRRs as strategies (solutions) to improve communication and collaboration between researchers and practitioners in software engineering, as shown in Papers V and VI. By inves-

tigating these strategies in context, the research aims to provide insights into the challenges faced in industry-academia collaborations and offer practical strategies for overcoming these challenges, ultimately contributing to a better understanding of the problem and the development of effective solutions.

2.1 Thesis Overview

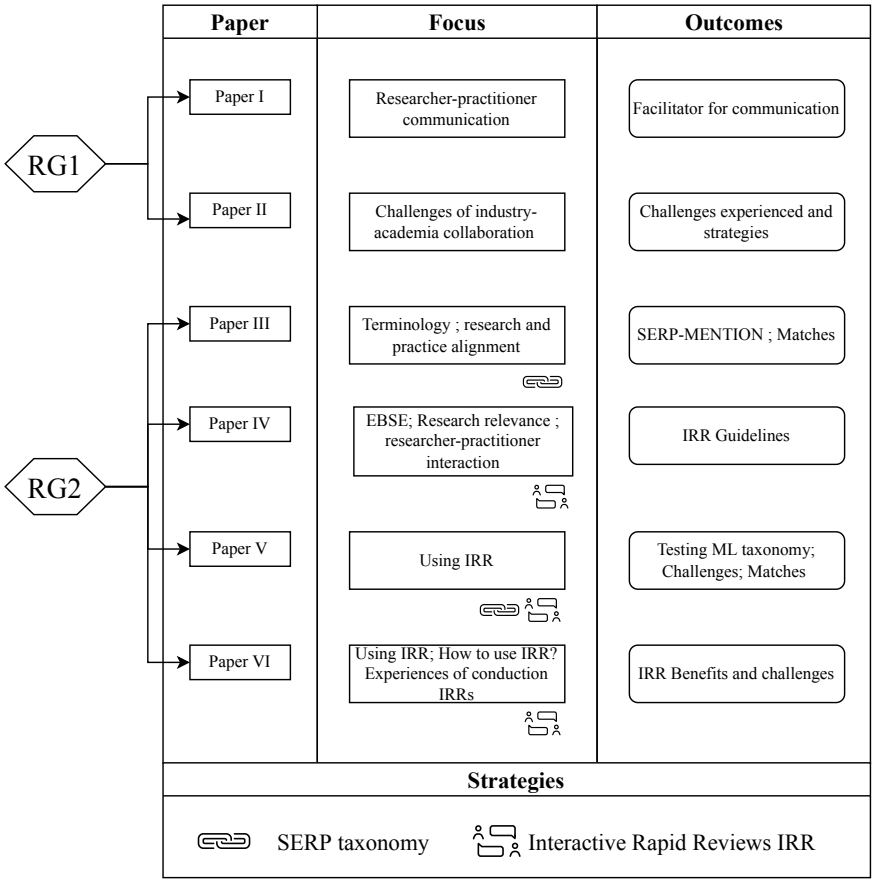


Figure 1: Research overview showing research goals, papers, focus, and main outcomes

Figure 1 provides an overview of the research, illustrating that Research Goal 1 (RG1) is primarily addressed in the first two papers, while Research Goal 2 (RG2) is mainly explored in the last four papers.

Table 1: Research design and data collection strategies for each study

Paper	Type of study	Data Collection Strategies
I	Case study	Timeline, Survey, Workshop
II	Survey study	Survey
III	Case study	Interviews, Workshop, Literature review
IV	Design proposal	Literature review
V	Case study	Literature review, Workshop
VI	Multi-case study	Interviews

The figure presents three vertical columns representing the papers, the focus of each study, and their outcomes. Paper refers to each of the six studies included in this thesis. Problem understanding denotes the key conceptual topics of interest in each study, and results represent the main outcomes of each paper. The two strategies (SERP taxonomy and IRR) are shown at the bottom and in each of the studies where they are used to address RG2.

The first goal of this thesis is to explore challenges that affect communication and collaboration in industry-academia collaborations. To achieve this goal, we conducted a case study investigating researcher-practitioner communication in a large industry-academia partnership (Paper I). Additionally, we surveyed Ph.D. students to identify their challenges while working with practitioners (Paper II). The outcomes of these studies are the facilitators for communication and the challenges that affect communication and collaboration in industry-academia collaborations.

The second goal, which is to design and evaluate solutions for improving communication and collaboration in industry-academia, led us to evaluate a previously proposed approach to improve communication called SERP-taxonomies [13] in the context of a software vulnerability management project (Paper III). We also proposed IRRs to foster knowledge exchange and collaboration between researchers and practitioners (Paper IV). The thesis details how an IRR was conducted in the context of a machine learning testing project (Paper V) where a SERP taxonomy was also used. Finally, we report the benefits and challenges of conducting IRRs by examining the case of Paper V and another case study on software component selection (Paper VI).

2.2 Types of Studies and Data Collection Techniques

To achieve the goals of this thesis, we used a mixed-methods research approach that combined qualitative and quantitative data collection techniques. Table 1 provides an overview of each paper's research design and data collection.

The research approach of this thesis is flexible and incorporates both qualitative and quantitative data collection techniques. Most of the studies in this thesis

are case studies. Case studies are an useful approach for exploring a phenomenon in a specific context and were used to evaluate the proposed strategies [42]. All the case studies were conducted in the context of industry-academia collaboration in software engineering, focusing on software testing, software vulnerability management, and software component selection topics.

We used various methods for data collection, including interviews, surveys, workshops, and literature reviews. In Paper I, we employed the evidence-based timeline method [6] to collect qualitative data, which involves creating timelines to evaluate projects retrospectively. This method allowed us to gather data about the participants, roles, interactions, project outcomes, and impacts. We surveyed to confirm our findings with a larger sample of participants. In Paper II, we surveyed to explore the challenges faced by Ph.D. students while working with practitioners. Paper III involved the development of a SERP taxonomy [13], which required us to review industry standards and scientific literature, and interview researchers. We also conducted a workshop with researchers and practitioners to describe research results and challenges using the SERP taxonomy in the project context.

In Paper IV, we proposed the design for IRRs and reviewed the literature on rapid reviews in medicine and software engineering to develop the proposal. Paper V included the development of another SERP taxonomy, which involved meetings with practitioners, literature reviews, and workshops with researchers and practitioners. Finally, for Paper VI, we conducted interviews with researchers and practitioners as the primary data collection method.

Overall, the use of a mixed-methods research approach enabled us to gain a comprehensive understanding of the challenges and possible solutions in industry-academia collaborations in software engineering. By combining both qualitative and quantitative data collection techniques, we were able to collect data from multiple sources and triangulate our findings.

3 Related Work

This section offers an overview of related work in two main areas. First, in Section 3.1, we examine the concept of industry-academia collaboration in software engineering. Second, since IRRs are a central topic in this thesis, we delve into the concept of Evidence-Based Software Engineering (EBSE) and secondary studies in Section 3.2.

3.1 Industry-Academia Collaboration in Software Engineering

Academia and industry have a long history of relationships and collaboration. Historically, universities were founded in places where there was a need for knowledge, and they have contributed to the development of society by providing knowl-

edge and education [14]. Today, well-developed economies and cultures rely on reputable institutions that can provide value in knowledge and education and act as a safeguard of society by providing a space for critical thinking and debate [45].

In recent years, increasing attention has been given to how universities and industry exchange knowledge and collaborate. Perkmann et al. [36] describe two different types of relationships between universities and industry: commercialization and academic engagement.

Commercialization refers to the efforts of academic institutions to commercialize or license their research results to industry [21]. This relationship includes patenting, academic entrepreneurship, research services, and intellectual property licensing. Many universities have created technology transfer, and patent offices, participated in science parks and incubators, and created spin-off companies to deal with these aspects [21, 30].

On the other hand, academic engagement refers to the knowledge-related interactions between academic and industrial organizations. This type of engagement can take many forms, both formal and informal. Examples include collaborative research, consulting, and informal activities like community participation and networking. This thesis emphasizes industry-academia collaboration, which can be seen as a form of academic engagement.

For researchers, engaging with industry has many benefits [35, 36]. It has been shown to enhance academic productivity. Researchers who collaborate with industry tend to publish as many or more papers as their peers who do not engage with the industry [5, 24]. Additionally, collaboration with industry often leads to new and innovative ideas, shapes the research agenda [17], and can also make educators more supportive of their students [7].

Industry-academia collaboration offers advantages to companies [1]. Collaborating with academic institutions can grant access to new knowledge, ideas, technologies, and talented individuals [9]. Additionally, these partnerships provide companies with opportunities to directly or indirectly influence the education of future employees [4]. By collaborating with universities, companies can better understand research advancements within their fields [28]. Finally, partnerships with reputable universities can enhance a company's image and reputation.

Moreover, industry-academia collaboration can contribute to innovation and economic growth, which is beneficial for society as a whole [45]. Universities play a crucial role in society by training new generations of professionals, conducting research, and developing new technologies [1]. Collaborating with academia can help companies stay at the forefront of technological advancements and improve their competitiveness [46]. In turn, this can lead to the creation of new jobs, products, and services, which can positively impact society. Finally, public-funded universities are expected to play a role in the enlightenment of society by promoting critical thinking and collective consciousness [11, 48]. In this regard, researchers and academic institutions may contribute better if they are aware of the recent developments, challenges, and opportunities in the industry.

The software engineering research community has been interested in industry-academia collaboration from the genesis of the discipline [22]. As a relatively young discipline, software engineering has been characterized by a strong focus on empirical research [15]. Therefore the need to collaborate with industry to validate and generalize the results of empirical studies is a key aspect of software engineering research.

Industry-academia collaboration has been a topic of interest in the software engineering community. Many models (see Table 2) have been developed to explain different aspects of collaboration and to provide advice to collaborate better and increase the knowledge transfer and exchange between academia and industry. One of the most cited models is the technology transfer model proposed by Gorshek et al. [23], which suggests a systematic approach to transferring technology from academic research to industry through a series of seven steps. This model emphasizes the importance of fostering close collaboration and cooperation between researchers and practitioners. It suggests that researchers observe the challenges faced by the industry and tailor their work accordingly, while practitioners help to adapt technology based on issues identified on-site.

Another model is the knowledge translation model proposed by Badampudi et al. [2]. This model suggests that research evidence needs to be translated into practitioners' context to be useful for practitioners. The authors propose a nine-step translation cycle that aims to comprehensively understand the organization's research findings, needs, and contextual factors. The model places importance on two core principles. The first is the need to take into account the specific context in which research will be applied. The second is the value of incorporating expert opinions. Additionally, the model stresses the need for an iterative and collaborative translation process customized to the unique context of the organization.

Industry-academia collaboration can have varying degrees of closeness and maturity levels. In a talk entitled "Software Engineering Research under the Lamppost" [49], Wohlin presented a maturity model with five levels of closeness between industry and academia. In level 5, the industry-academia collaboration is done in one team, where a specific industrial challenge is identified, draft solutions are evaluated and validated in iterations, and final solutions are usually implemented in practice. In level 4, the collaboration is offline and often remote. While a specific industrial problem is identified, the solution is done in academia, and a pre-packaged solution is offered that is challenging to adopt in the industry due to its generality. Levels 1-3 are characterized by non-existent or weak linkages between industry and academia and thus cannot be considered proper collaboration. Understanding the various levels of closeness in industry-academia collaboration can help researchers and practitioners identify the most effective strategies for collaboration and maximize the benefits for both industry and academia.

Other models have been proposed to structure and operationalize industry-academia collaboration. Mikkonen et al. [34] proposed the continuous and collaborative technology transfer, which suggests that the collaboration between academia

and industry should be ongoing, iterative, and mutually beneficial. The model advocates for a shift in the mindset from technology transfer to co-creation between the two parties and highlights the importance of continuous communication, knowledge exchange, and shared understanding. Additionally, Runeson and Minör [44] proposed the 4+1 View Model, which offers a comprehensive view of industry-academia collaboration by breaking it down into five distinct views, including stakeholder view, process view, data view, deployment view, and architectural view. Finally, the Certus Model proposed by Marijan and Gotlieb [31] provides a framework to structure and operationalize industry-academia research collaborations. The model suggests a structured process for initiating, implementing, and evaluating collaborations, focusing on goal definition, partner selection, collaboration design, and performance evaluation. The Certus Model aims to enable productive collaboration between industry and academia, generating high-quality research and fostering innovation.

Garousi et al. [20] presented a generic process view of industry-academia collaboration that includes four phases. The first phase, inception, focuses on team building and topic selection. The second phase, planning, involves defining the goal, scope, and expectations of the collaboration. The third phase, operational, comprises running, controlling, and monitoring the collaborative project. The final phase, transition, centers on technology and knowledge transfer, as well as the impact of the collaboration. It is important to note that this process view is not a model of collaboration per se but can be used to describe collaborative projects.

Table 2: Comparison of Industry-Academia Collaboration Models

Model & Focus	**Characteristics**	**Stages**	**Key Activities**
Technology transfer [23]	Transfer of research results to industry	3 stages: initiation, negotiation, and implementation	Identification of research results, engagement with industry, and transfer of results
Knowledge translation [2]	Facilitating knowledge exchange between academia and industry	4 stages: knowledge creation, dissemination, adoption, and implementation	Identification of knowledge gaps, knowledge transfer, adoption of knowledge, and integration of knowledge

Continued on next page

Table 2 – continued from previous page

Model & Focus	Characteristics	Stages	Key Activities
Continuous and collaborative technology transfer [34]	Continuous exchange of knowledge and collaboration	3 stages: alignment, collaboration, and evaluation	Co-creation of knowledge, continuous collaboration, and evaluation
4+1 View Model [44]	Viewpoints and perspectives to facilitate collaboration	5 viewpoints: process, data, logic, physical, and scenarios	Collaboration on process, data, logic, physical, and scenarios
CERTUS model [31]	Collaboration in software engineering research	3 stages: joint framing, joint research, and joint knowledge utilization	Mutual understanding, co-creation of research, and knowledge utilization
Generic process [20]	Describing collaborative projects	4 phases: inception, planning, operational, and transition	Team building, goal definition, running and monitoring, technology transfer and impact

3.2 EBSE and Secondary Studies

In response to the gap between research and practice in software engineering, evidence-based software engineering (EBSE) has been proposed as a systematic approach to bridge the divide. The purpose of EBSE is to inform decision-making in software engineering practice by incorporating the best available evidence, ensuring that solutions are grounded in rigorous research findings. EBSE involves five steps: (1) converting the need for information into an answerable question, (2) tracking down the best available evidence, (3) critically appraising the evidence for validity, impact, and applicability, (4) integrating the appraisal with software engineering expertise and stakeholder values, and (5) evaluating the decision-making process [12, 27]. This process emphasizes the importance of evidence-based decision-making in software engineering practice, allowing for the development of customized solutions that consider the unique context of software engineering practice.

EBSE uses secondary studies to synthesize evidence from primary studies and inform software engineering practice. However, accessing and synthesizing evidence from primary studies is time-consuming and labor-intensive [32]. Secondary studies are a valuable tool in EBSE, providing a systematic approach to synthesizing existing research and identifying gaps in current knowledge. Secondary studies enable informed decision-making and help bridge the gap between research and

practice by providing practitioners with a comprehensive and up-to-date understanding of a topic. These studies can make research results more accessible and understandable to practitioners, facilitating the integration of research evidence into software engineering practice.

Systematic literature reviews are one of the most common forms of secondary studies in software engineering research [26]. Systematic literature reviews involve a rigorous and systematic process of searching, selecting, and analyzing primary studies' evidence to answer a research question. This process consists in defining the research question, identifying relevant studies, selecting studies based on inclusion and exclusion criteria, assessing study quality, and synthesizing the evidence [25]. The goal of SLRs is to provide a comprehensive and unbiased summary of the evidence on a specific topic.

Systematic mapping studies are another form of secondary studies used in software engineering research [40]. Mapping studies aim to provide an overview of the research landscape and identify research gaps in a specific field. They involve a systematic process of identifying, selecting, and categorizing primary studies based on predefined inclusion and exclusion criteria [38]. The output of mapping studies is an overview of the research landscape, which can be used to identify areas of research saturation and research gaps.

Rapid reviews are a newer type of secondary study that aims to provide a timely and pragmatic synthesis of evidence [8]. Rapid reviews involve a streamlined process of identifying and summarizing the evidence on a specific topic. They may involve a reduced search scope, inclusion criteria, and appraisal process compared to systematic literature reviews and mapping studies. Rapid reviews may be useful when a timely synthesis of evidence is needed, and resources are limited.

Multivocal literature reviews are another form of secondary study that aims to include non-peer-reviewed sources of evidence, such as reports, conference proceedings, and technical papers [18]. These reviews involve a process of searching, selecting, and analyzing grey literature sources to answer a research question. Multivocal literature reviews are helpful when the relevant evidence is not published in peer-reviewed journals.

The choice of secondary study depends on the research question, the available resources, and the timeline for completion. Systematic literature reviews and mapping studies are suitable when a comprehensive and unbiased summary of the evidence is needed. At the same time, rapid reviews and multivocal literature reviews are helpful when time and resources are limited or when non-peer-reviewed sources of evidence are relevant.

Some researchers in software engineering have dedicated efforts to improve the quality and rigor of secondary studies. These efforts include developing guidelines on how to perform snowballing [50], how to perform quality assessment [47, 51], when to update secondary studies [33], and providing tools that make the search and selection of studies easier [16].

4 Summary of Results

This section summarizes the results of the papers included in this thesis.

4.1 Paper I

Title: A case study of industry–academia communication in a joint software engineering research project

This study aimed to identify the factors that facilitate communication between researchers and practitioners. We conducted this case study as part of the evaluation of the EASE research program[1]. We considered every time a researcher or practitioner communicated with the other party during the project as a communication instance. For each communication instance, we identified the communication parties, the content, and the context in which the communication occurred. As part of the analysis, we identified common characteristics of the communication context that promoted communication. Similarly, we identified project outcomes that were promoted by communication. The communication between researchers and practitioners occurred in three main contexts: the industry-academia environment, project-related meetings, and individual studies.

The study found that five main facilitators promoted communication: research relevance, practitioner's attitude towards research, active practitioner involvement, frequency of communication, and long-term collaboration. Long-term collaboration was identified as a significant facilitator of industry-academia communication in the industry-academia environment, leading to the expansion of social networks, new studies, and mutual learning between researchers and practitioners. In addition, communication in the social environment within the long-term research partnership stimulated knowledge exchange, promoting further and improved industry-academia collaboration, e.g. through new projects and joint supervision of MSc projects.

In the context of project-related meetings, the study found that the frequency and style of meetings and the active involvement of practitioners played a crucial role in promoting IA communication. Project meetings were a crucial context for IA communication, leading to good collaboration, jointly detailing and agreeing on the research direction, and initiating new research studies.

The third category of communication contexts that the study identified was individual studies. Two main types of individual studies were identified in the research project, which were industrial MSc projects and research studies. The industrial MSc projects were under the supervision of researchers and were related to the research project. The research studies, on the other hand, were conducted by researchers and were relevant to the research project. The communication that

[1]The Industrial Excellence Centre for Embedded Applications Software Engineering was active between 2008-2018

occurred in the individual studies was primarily related to the supervision of industrial MSc projects. The researchers and industrial supervisors involved in the MSc projects communicated about the research direction and the outcomes. This context was found to be mainly related to the supervision of industrial MSc projects, with the communication focusing on the research direction and outcomes.

Table 3 summarizes the facilitators and outcomes identified in the study.

Table 3: Facilitators and outcomes of industry-academia communication

Facilitator	**Description**	**Outcomes**
Research relevance	Ensuring research is relevant to industry needs	New knowledge, changes in practice, social networks
Practitioner attitude towards research	Positive attitude towards research among industry partners	Good IA collaboration, new scientific venues
Active practitioner involvement	Active involvement of industry partners in research project	New knowledge, changes in practice, social networks
Frequency of communication	Regular and frequent communication between researchers and practitioners	Good IA collaboration, new scientific venues
Long-term collaboration	Long-term research partnership between academia and industry	New knowledge, changes in practice, social networks, new scientific venues

Overall, the study provides insights into the importance of effective communication between researchers and practitioners and how it can be facilitated in the context of a software testing project. The findings can be valuable for future research and practice in promoting effective industry-academia collaboration.

4.2 Paper II

Title: Industry-Academia Collaborations in Software Engineering: Identifying Challenges and Strategies from Ph.D. Students' Experience

Paper II aimed to explore the challenges that Ph.D. students faced when collaborating with practitioners in industry-academia collaborations. The study presented a total of 58 challenges, which were categorized into 13 categories. These challenges were identified through a literature review and complemented with suggestions from the authors. To collect data, we used a questionnaire in which respondents were asked if they had experienced or observed each challenge and the level of impact it had on the collaboration. We received 12 responses, which helped us identify the most impactful challenges and those not experienced or observed

by Ph.D. students. Additionally, from the open questions, we identified strategies that could be used to overcome the difficulties.

The top five challenges with the highest impact were:

1. Researchers focus on long-term thinking, while practitioners prioritize short-term goals.
2. Researchers struggle to move from research prototypes to production-ready solutions.
3. Practitioners have little interest in participating in collaborative research projects
4. Researchers face problems contacting the right practitioners in the organization.
5. Practitioners take a long time to respond to researchers.

The challenges that were not experienced or observed as having a medium or high impact by any Ph.D. student were:

1. Practitioners do not value qualitative research.
2. Practitioners see academic research as a waste of time since it does not apply to business.
3. Research is not relevant to practitioners.
4. Researchers and practitioners do not speak the same language, e.g., Swedish and English.
5. Issues with digital tools, e.g., slow internet connection, unstable connection, lack of audio/video

For a complete list of challenges and the strategies to overcome them, please refer to Paper II. These findings highlight the diverse range of challenges faced by Ph.D. students in industry-academia collaborations, with some challenges having a high impact on the collaboration. In contrast, others were not observed or experienced by Ph.D. students. Both researchers and practitioners can use this information to anticipate potential issues and consider the strategies to address them effectively, ultimately enhancing the collaboration between academia and industry.

4.3 Paper III

Title: A taxonomy for improving industry-academia communication in IoT vulnerability management

In this study, our goal was to evaluate SERP-taxonomies as a strategy for supporting industry-academia communication and collaboration, specifically in the

context of IoT vulnerability management. This was important due to the emergent nature of IoT and the lack of standards, which can cause fragmentation in the field.

We built upon the SERP architecture [13], previously used to support communication in software testing, and applied it to the development of a taxonomy called SERP-MENTION. The development process involved multiple steps, including a review of existing taxonomies, interviews with industry and academia representatives, and a workshop to identify challenges and solutions related to IoT vulnerability management. This enabled identifying and refining the key categories and entities needed for the taxonomy.

The results of the study showed that SERP-MENTION was perceived as valuable by potential users for describing and communicating research outputs and practical challenges in software vulnerability management. The taxonomy was also found to be useful for describing challenges and solutions in IoT vulnerability management by the project participants. Additionally, the taxonomy showed the potential to impact collaboration by linking challenges from industry to solutions in academia and vice versa.

Overall, the study highlighted the importance of developing taxonomies to support communication and collaboration in emergent fields like IoT vulnerability management. By providing a shared technical language, taxonomies like SERP-MENTION can help to bridge the gap between industry and academia, enabling more precise and unified descriptions of practical challenges and research outcomes.

4.4 Paper IV

Title: Guidelines for conducting interactive rapid reviews in software engineering–from a focus on technology transfer to knowledge exchange

Paper IV presents a set of guidelines for conducting interactive rapid reviews in software engineering. The objective is to facilitate knowledge exchange between researchers and practitioners. The guidelines are based on a literature review of rapid reviews and stakeholder engagement in medicine, as well as the authors' experience using secondary studies in software engineering. The proposed method emphasizes close collaboration between researchers and practitioners, allowing for an agile literature review process.

The interactive rapid review method consists of five main steps: (1) preparation, (2) involvement, (3) search, (4) analysis, and (5) dissemination. In the preparation step, the research questions and criteria are defined, and the involvement of practitioners is negotiated. In the involvement step, the practitioners contribute their knowledge and expertise to refine the research questions and provide context. In the search step, shortcuts are applied to speed up the process, and strict exclusion criteria are set to limit the number of papers. In the analysis step, the data is extracted and synthesized using narrative synthesis. Finally, in the dissemination step, the results are presented in a practitioner-friendly format, agreed upon with

the practitioners, who may also present or co-present the results. The interactive rapid review method aims to facilitate knowledge exchange by involving practitioners in the review process and streamlining the review process.

4.5 Paper V

Title: Exploring ML testing in practice: lessons learned from an interactive rapid review with axis communications

Paper V aimed to establish a common view of the problem domain in machine learning testing by initiating collaboration between industry and academia. To achieve this, the authors applied an interactive rapid review of the state of the art, involving four researchers from Lund University and RISE Research Institutes and four practitioners from Axis Communications. In addition, the SERP taxonomy architecture was used to guide the alignment of terminology and interests within the review team, resulting in a helpful taxonomy for communication.

The study utilized the IRR approach to trigger and align communication between different stakeholders, identify and describe current challenges in the case context, and rank them by their perceived importance for the target organizational unit within the case company. An in-depth analysis of the 35 primary studies that matched the most important review question, "How to test the dataset?", was conducted. Nine technological rules on data testing, extracted from five papers, were identified and discussed.

Overall, the study demonstrated how IRR and SERP could support communication and collaboration between industry and academia in ML testing. The study allowed the stakeholders to establish a standard view of the problem domain, align their terminology, and identify and prioritize relevant research questions. These results could pave the way for future joint projects and collaborations.

4.6 Paper VI

Title: Experiences from conducting interactive rapid reviews – two industrial cases

The paper presents two case studies that applied the method of IRRs to promote knowledge exchange between researchers and practitioners in software engineering. Case Study 1, named Case-SoftSelection, focused on the selection of software components and involved one researcher and one practitioner. The search was performed using keywords in the search engine Scopus. The researchers found criteria for software selection from the literature and developed a preliminary model for software component selection by exchanging and discussing with the practitioner. Case Study 2, named Case-MLTest, involved four practitioners and four researchers and explored machine learning testing. The researchers used a set of systematic literature reviews as initial input and analyzed the studies included in the secondary studies. A SERP taxonomy was used in the analysis, and a set of matches between problems and solutions were identified.

The benefits of conducting IRRs were analyzed and classified into four categories: mutual learning, an overview of the field, usage of research, and future collaboration. The mutual understanding, the co-creation of knowledge, and researchers learning from an industrial context were some of the benefits found in the mutual learning category. The overview of the field was another benefit for the researchers to understand the industrial perspective of the problem they investigated. Usage of the results of the review was found to be positive in providing an understanding of the research state of the art, identifying research literature, and finding relevant solutions. Finally, IRRs helped develop networks for future collaborations, where researchers and practitioners formed new relationships and had ideas for future collaboration projects.

In Case-SoftSelection, the motivation was to explore ways to collaborate with academia and assess its feasibility. In contrast, in Case-MLTest, the motivation was to understand the industry practice and network with practitioners in the field. Both IRRs had a positive impact on the participants, including new networks, new knowledge, and new ideas for future collaboration projects. Overall, the IRR approach seemed to be a way for researchers and practitioners to collaborate and work on specific problems while looking at the horizon for future collaborative work.

5 Contributions

This section presents the contributions of this thesis organized according to the research questions. Figure 2 visually represents the contributions, with each research question corresponding to one distinct contribution. We have summarized a takeaway for each contribution corresponding to the main point or recommendations.

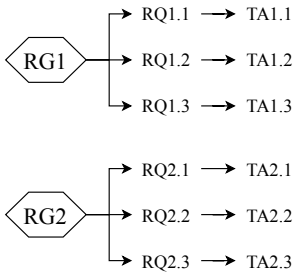


Figure 2: Overview of thesis contributions

5.1 Goal 1

The first goal is to delve into the challenges associated with industry-academia collaboration in software engineering, specifically focusing on understanding and improving communication between researchers and practitioners. Then, to effectively highlight the contributions of this research, we will present them in relation to the research questions that have guided this thesis.

RQ1.1 What are the key challenges researchers and practitioners face when collaborating on industry-academia projects in software engineering?

Paper II includes a selection of 13 challenges that were identified in the literature review and completed with the author's experience in industry-academia collaborations. The challenges can be classified into three categories (1) challenges related to communication distances between researchers and practitioners; (2) challenges related to people; and (3) challenges related to the research work.

- Communication distances: CH1 geographical distance, CH2 time available for communication, CH3 differences in values, social norms, and culture, CH4 organizational differences, CH5 different time horizons, CH6 communication tools
- People: CH7 availability and accessibility, CH8 motivation and willingness to exchange, CH9 differences in knowledge and skills, CH10 beliefs and expectations,
- Research work: CH11 terminology and language, CH12 research relevance, and CH13 maturity of research.

Table 4 shows in which papers there is also a reference to each challenge.

In Paper I, five factors were identified that facilitate communication between researchers and practitioners. We can map how these factors are related to the challenges in Table 4. Firstly, the relevance of the research is directly related to the challenge of research relevance (CH12). In that study, it was observed that the involvement of industry at the management level and in the research studies boosted the relevance of the research. Active involvement and a positive attitude of practitioners were also identified as positive factors. The direct application of some outcomes kept the motivation of the practitioners (CH8) as they saw the value of working with the researchers. Similarly, having the collaboration program as a framework made it easier to find the right people to communicate (CH7) with and start new projects, reducing the organizational friction that could emerge without a framework (CH4). The frequency and communication style during the meetings were also identified as factors that facilitate communication since people were available (CH2) and resources were allocated to the collaboration program.

Table 4: Challenges and their Presence in the Papers

Challenge	**PI**	**PII**	**PIII**	**PIV**	**PV**	**PVI**
CH1: Geographical distance		X				
CH2: Time available for communication	X	X				
CH3: Differences in values, social norms, and culture		X				X
CH4: Organizational differences	X	X				X
CH5: Different time horizons		X				
CH6: Communication tools		X				
CH7: Availability and accessibility	X	X				
CH8: Motivation and willingness to exchange	X	X				
CH9: Differences in knowledge and skills		X				
CH10: Beliefs and expectations		X				X
CH11: Terminology and language		X	X		X	
CH12: Research relevance		X	X	X		
CH13: Maturity of research		X	X		X	

Paper II surveyed participants on their experiences collaborating with industry and found that all challenges except CH6 were mentioned as having a low, medium, or high impact on collaboration. The survey results indicate that these challenges are potential problems that could arise in industry-academia collaborations.

Paper III mainly addresses the challenge of terminology (CH11) in emerging domains, where many approaches are being proposed and terminology is not well defined due to the lack of standards. The study aimed to address this challenge by describing challenges and research outcomes in a common language and linking them to bridge gaps and make research more relevant (CH12) and ensure that the outcomes solve the challenges (CH13).

In Paper IV, the proposed IRR aims to address the challenge of the research practice gap (CH12) by bringing together researchers and practitioners to discuss the research problem and its outcomes. While there are no further challenges explicitly addressed in this paper, Papers V and VI demonstrate that the IRR method can be used to address other challenges in industry-academia collaborations.

In paper V, a key challenge was the maturity of research (CH13). Since data testing for computer vision is a new field, the research is still in its early stages. The review team was aware of the terminology differences (CH11) and then it was not a challenge when conducting the review.

Paper VI identified challenges related to conducting IRRs, including roles and responsibilities (CH4, CH6), the lack of available research results (CH13), and time constraints (CH2).

The 13 category challenges examined in Papers II offer a wide perspective on various aspects that can influence industry-academia collaborations. As explored in other papers, these challenges can be addressed by implementing proposed solutions. Our primary contribution to research question RQ1.1 is the recognition of 13 challenges, which can be categorized into three distinct groups: communication

distances, people, and research work. By acknowledging these challenges, researchers and practitioners can better prepare to address potential issues, improving communication and collaboration within industry-academia projects.

TA1.1

To enhance communication and collaboration within industry-academia collaborations, researchers and practitioners must recognize the potential challenges that can emerge in such partnerships.

RQ1.2: What factors influence communication and collaboration between researchers and practitioners in industry-academia collaborations in software engineering?

Various factors facilitate collaboration between researchers and practitioners in industry-academia projects, as outlined in Table 3. These factors can lead to positive outcomes, such as new knowledge generation, changes in practice, expanding social networks, and the development of new scientific venues.

For successful collaboration, it is crucial to align research with industry needs (Research relevance). Fostering a positive attitude toward research among industry partners (Practitioner attitude towards research) is also vital for effective collaboration. Encouraging active involvement of industry partners in research projects (Active practitioner involvement) and maintaining regular, frequent communication between researchers and practitioners (Frequency of communication) are additional key facilitators. Long-term research partnerships between academia and industry (Long-term collaboration) can further contribute to successful outcomes.

Considering the factors identified in Table 3, researchers and practitioners should focus on establishing solid relationships, ensuring research relevance, actively involving industry partners, and promoting regular communication. Moreover, cultivating long-term partnerships is essential for achieving successful outcomes.

TA1.2

To ensure effective collaboration in industry-academia projects, prioritize research relevance, establish strong relationships, encourage the active involvement of industry partners, maintain regular communication, and cultivate long-term partnerships.

RQ1.3 How do Ph.D. students in software engineering experience and overcome challenges in industry-academia collaborations?

Ph.D. students and other junior researchers often face challenges collaborating with industry partners. Based on a survey of Ph.D. students, we have found various strategies to address these challenges, which can be found in Paper II. These strategies offer valuable insights and tips for junior researchers, Ph.D. supervisors and enhance collaboration with industry partners by adapting to each project's specific needs and requirements. In addition, the strategies identified in the study can guide Ph.D. students to improve their collaboration, including relationship building, effective communication of research goals and outcomes, and overcoming obstacles inherent in industry-academia collaborations. In conclusion, engaging with industry partners can present challenges but also serves as an invaluable learning experience for Ph.D. students in software engineering. We believe that by cultivating effective communication and collaboration skills, Ph.D. students can contribute to their research area, forge meaningful professional relationships, and acquire vital skills for their future careers.

TA1.3

To address communication and collaboration challenges with industry partners, adopt the strategies outlined in Paper II.

5.2 Goal 2

The second goal is to develop and assess strategies to enhance communication and collaboration in software engineering, particularly within the context of industry-academia partnerships. To effectively showcase the contributions of this goal, we will present them in connection with the research questions that have guided this thesis.

RQ2.1 How effective is the SERP-taxonomy approach in supporting communication about practical challenges and research results in software engineering?

In Paper III, we developed SERP-MENTION to support communication between industry and academia in IoT software vulnerability management. The development process included reviewing existing taxonomies, conducting interviews with representatives from both industry and academia and holding a workshop to identify challenges and solutions. The study revealed that potential users perceived the SERP-MENTION taxonomy as valuable and showed promise in connecting industry challenges with academic solutions.

The researchers in Paper V utilized the interactive rapid review approach and employed the SERP taxonomy architecture to establish a common perspective

among the researchers while investigating academic research on testing machine learning systems. This approach enabled researchers to harmonize their terminology, categorize research findings, and identify and prioritize pertinent research questions. The SERP framework was useful for navigating research results, familiarizing researchers with the data, and guiding terminology alignment within the team.

Though the primary focus of both studies was not the development of SERP taxonomies, they both demonstrated the potential of using SERP to support communication and collaboration between industry and academia in emerging fields such as IoT vulnerability management and machine learning testing. By offering a shared technical language, taxonomies like SERP-MENTION can help bridge the gap between industry and academia, enabling more accurate and cohesive descriptions of practical challenges and research outcomes.

TA2.1

To enhance mutual understanding and align research results with practitioners' challenges in industry-academia collaborations, researchers can develop taxonomies following the SERP architecture.

RQ2.2: How can interactive rapid reviews support communication and collaboration between researchers and practitioners in industry-academia collaborations in software engineering?

Based on Papers V and VI, IRRs might be a valuable approach to enhance communication and collaboration between researchers and practitioners in industry-academia collaborations by encouraging knowledge exchange, fostering a shared understanding of the problem domain, aligning terminology, identifying relevant research questions, and paving the way for future collaborations. IRRs involve iterative, interactive, and collaborative work between researchers and practitioners to formulate research questions, identify inclusion/exclusion criteria, perform keyword searches to identify relevant primary studies, analyze primary studies to develop preliminary models or problem-solution matches, and disseminate findings to practitioners. The potential of IRRs in promoting dialogue and collaboration was illustrated in the case studies, which led to new projects, master thesis proposals, and collaborations between researchers and practitioners. Some scenarios where IRRs might be beneficial include:

- Exploring emergent fields: IRRs could be used to establish a common understanding of an emergent field, where there may be limited existing research or fragmented information. For example, as shown in Paper V, IRRs can be used to explore the challenges and solutions in machine learning testing.

- Bridging the gap between industry and academia: IRRs could help bridge the gap between industry and academia by aligning terminology, promoting mutual learning, and providing an overview of the field from both perspectives. This can facilitate communication and collaboration between the two groups, as demonstrated in Paper VI.
- Developing preliminary models: IRRs could be used to develop preliminary models or problem-solution matches based on the analysis of primary studies. This can help both researchers and practitioners to understand the current state of the art and identify gaps that need to be addressed.
- Prioritizing research questions: IRRs could be used to prioritize relevant research questions by analyzing the primary studies identified through keyword searches. This can ensure that future research is focused on the most pressing issues and can benefit both researchers and practitioners.
- Evaluating the usage of research: IRRs could be used to evaluate the usage of research by identifying and disseminating relevant primary studies to practitioners. This can help to improve industry practices and can also benefit researchers by ensuring that their work is being applied in the real world.

As a practical approach, IRRs might be a useful method for establishing communication and collaboration between researchers and practitioners, especially when exploring emergent fields in industry-academia collaborations.

TA2.2

To potentially improve communication and collaboration when exploring an emergent field, consider conducting interactive rapid reviews.

RQ2.3: What are the benefits and challenges of using interactive rapid reviews (IRRs) in industry-academia collaborations in software engineering?

Benefits of using (IRRs) in industry-academia collaborations, as shown in Papers V and VI, include improved communication and collaboration between researchers and practitioners, promoting knowledge exchange, creating a shared understanding of the problem domain, aligning terminology, identifying relevant research questions, and paving the way for future collaborations. IRRs also offer a quick and efficient way to identify relevant primary studies and analyze them to develop preliminary models or problem-solution matches, providing both researchers and practitioners with a better understanding of the current state of the art and potential gaps that need to be addressed.

However, there are also challenges associated with using IRRs in industry-academia collaborations. These include challenges related to roles and responsibilities, such as researchers and practitioners needing to know their duties. Other challenges include the need for results matching the needs and expectations of the review team and issues related to the timeliness of the reviews.

From the two cases analyzed in paper VI, we identified recommendations to improve the effectiveness of IRRs in industry-academia collaborations (See Table 2 in Paper VI). These include clarifying roles and responsibilities at the outset of the collaboration, providing clear guidelines on the IRR process and expectations, and setting realistic timelines for the review. It is also important to ensure that the IRR team has the necessary skills and expertise to conduct the review effectively and that a clear communication plan is in place to facilitate dialogue between researchers and practitioners throughout the process.

TA2.3

To improve the effectiveness of interactive rapid reviews in industry-academia collaborations, clarify roles and responsibilities, provide clear guidelines and expectations, set realistic timelines, ensure the necessary skills and expertise, and establish a clear communication plan.

6 Threats to Validity

In this section, we discuss the threats to the validity of the thesis from a broader perspective, considering the four types of validity [43] construct, internal, external (generalizability), and reliability.

6.1 Construct Validity

Construct validity refers to the degree to which a research study operationalizes the concepts it intends to study.

The definition of communication and collaboration concepts may create threats since their use varies across disciplines and contexts. To minimize this threat, we shared the terminology we used with the participants to ensure a common understanding of the concepts and mitigate potential misunderstandings or misinterpretations.

Regarding the challenges investigated in this thesis, they may not capture all the nuances and complexities associated with communication and collaboration. To address this threat, we surveyed continuously the literature to identify challenges and contrasted them with the researcher's experience working in industry-academia collaborations. It is important to note that the studies were conducted in Sweden, which may limit the generalizability of the findings. We also ensured to

make complete descriptions to researchers and practitioners when we present the challenges and strategies in the data collection phase.

Another aspect of construct validity that we have considered is the measurement and results interpretation. The methods we used to evaluate the proposed strategies, like IRR and SERP taxonomy, may be subject to bias and error. To mitigate this threat, we used triangulation in several instances, such as in Paper I, where we surveyed a broader group of practitioners to confirm our findings in the case study. Furthermore, in Papers I, III and VI, we asked the participants to validate the results section in the manuscripts to ensure that the results were correctly interpreted. Finally, peer review of the analysis was a common practice in all the papers to maintain the quality and rigor of our research.

6.2 Internal Validity

Internal validity concerns the extent to which the results of a study can be attributed to the intervention or treatment rather than to other factors. To address selection bias, we conducted multiple case studies using a wide range of cases of industry-academia collaborations, such as software vulnerability management (Paper III), software component selection (Paper VI), and testing machine learning (Papers V,VI). We also considered various sizes of collaborations, from small groups of researchers to large collaborations, and studied collaborations at different phases, from inception to finalization.

To minimize the influence of researchers in the data collection, we tried to keep one author responsible for the meta-perspective and other authors more involved in applying the strategies like in Paper V. This approach allowed us to maintain a balance between the researchers' involvement and the accuracy of the data collected.

6.3 External Validity

External validity, also known as generalizability, refers to the extent to which the results of a study can be generalized to other contexts. In this thesis, we have considered external validity by conducting literature reviews to capture a broader perspective of the challenges and to further study the challenges identified in the case studies. This allowed us to maintain a comprehensive view that could be interesting for a larger audience and focus on the challenges that were more relevant to the case studies. For example, in Paper II, we investigated the challenges faced by Ph.D. students when collaborating with practitioners in industry-academia collaborations in software engineering, identifying impactful challenges and strategies to overcome them.

We acknowledge that cultural factors in the environment where the collaborations take place may have influenced the generalizability of the results. To mitigate this threat, we have included a discussion of the limitations of the case studies in

the papers. For example, in Paper IV, we introduce a set of guidelines for conducting interactive rapid reviews in software engineering to facilitate knowledge exchange between researchers and practitioners, emphasizing the importance of considering the specific context and limitations when applying the guidelines.

To improve the transferability of the results, we included detailed descriptions of the methods and how they were applied in the papers. In both SERP taxonomies and IRRs, we described the approaches as flexible and capable of being tailored to specific contexts. For instance, in Paper III, we describe the steps to build SERP-MENTION, in Paper IV, we describe the details of the IRR process, and in Paper V, we describe the details of conducting IRR. By providing a detailed description of the methodology, we aimed to make the taxonomy adaptable to other contexts.

6.4 Reliability

Reliability refers to the consistency and stability of the research findings when replicated under similar conditions. To address the threats to reliability, such as researcher bias and the influence of researchers in the data collection, we used peer review in critical steps of the research process. For example, the interviews in Paper VI were coded by one author and then reviewed by another author to ensure the accuracy of the results. Similarly, other critical steps in the analysis were reviewed by other researchers to ensure the quality of the results. Additionally, we used triangulation to validate the results, for instance, in Paper VI, where we presented two case studies that applied IRR, we also had access to the project documents and the results of the IRRs.

7 Future Work

While this thesis has provided insights into the challenges and benefits of industry-academia collaborations and proposed strategies to improve communication and collaboration, it opens up several avenues for future research. In this section, we outline potential directions for future research that could build upon the findings of this thesis and extend the knowledge and impact of industry-academia collaborations.

Organizational factors in industry-academia collaboration: Future research should investigate the influence of different types of organizations, such as start-ups, small and medium enterprises (SMEs), consulting firms, software companies, universities, and research institutes, on the success of industry-academia collaborations. By examining these factors, researchers can identify specific conditions and characteristics that promote or hinder collaboration. *FRQ1: What organizational factors influence industry-academia collaborations?*

Challenges in industry-academia collaboration across different regional contexts: This thesis has explored the challenges and benefits of industry-academia

collaboration based on existing literature and case studies in Sweden. Future research could investigate how these challenges are experienced in other regional and cultural contexts and if the strategies proposed in this thesis apply to other regional contexts where cultural factors may influence collaboration. *FRQ2: How are challenges in industry-academia collaborations experienced in other regional contexts?*

Synergies between industry and academia to influence teaching: One interesting area of research involves exploring the synergies between industry and academia in shaping teaching and curricula. By aligning curricula, courses, and materials with the needs of the industry, academic institutions can ensure that graduates possess the necessary skills and knowledge to succeed in their chosen fields, creating a win-win situation for both parties. *FRQ3: How can synergies be developed between industry and academia to influence teaching?*

Interactive Rapid Reviews (IRRs) to enhance industry-academia collaborations: The potential of IRRs to close gaps between industry and academia has been demonstrated in this thesis, but more IRRs should be conducted to understand their benefits better and suggest recommendations for improving their effectiveness in industry-academia collaborations. Some scenarios where IRRs can be utilized include 1) capturing practitioners' insights before conducting a systematic literature review, 2) incorporating IRRs into academic courses to familiarize future practitioners with academic research, and 3) using IRRs at the beginning of Ph.D. projects to become acquainted with state of the art. *FRQ4: What can we learn from conducting more IRRs in industry-academia collaborations?*

8 Conclusion

This research addressed the challenges and benefits of industry-academia collaborations and proposed strategies to improve communication and collaboration. The research identified a range of challenges that can affect the success of such collaborations, including time constraints, differences in knowledge and skills, and communication barriers related to values, social norms, and culture, among others. The thesis proposes strategies like developing SERP taxonomies and conducting interactive rapid reviews to build stronger bridges between industry and academia. These strategies foster mutual understanding, facilitate identifying relevant research questions, and promote knowledge exchange.

Part of the research outcomes includes factors that positively influence communication in industry-academia collaborations. These factors include establishing a good relationship between researchers and practitioners, prioritizing regular communication, ensuring the active involvement of industry partners, and establishing long-term partnerships. Additionally, the research has pinpointed success factors that can positively influence communication, such as being aware of the roles and responsibilities of each party and setting realistic timelines.

This thesis offers insights into industry-academia collaborations' challenges and benefits and proposes strategies to improve communication and collaboration. Moreover, the findings can guide researchers and practitioners in establishing effective partnerships and generating research outcomes that are impactful and relevant to software engineering practice.

References

[1] Samuel Ankrah and AL-Tabbaa Omar. Universities–industry collaboration: A systematic review. *Scandinavian Journal of Management*, 31(3):387–408, 2015.

[2] Deepika Badampudi, Claes Wohlin, and Tony Gorschek. Contextualizing research evidence through knowledge translation in software engineering. In *Proceedings of the Evaluation and Assessment on Software Engineering*, pages 306–311. 2019.

[3] Victor Basili, Lionel Briand, Domenico Bianculli, Shiva Nejati, Fabrizio Pastore, and Mehrdad Sabetzadeh. Software engineering research and industry: a symbiotic relationship to foster impact. *IEEE Software*, 35(5):44–49, 2018.

[4] Kathy Beckman, Neal Coulter, Soheil Khajenoori, and Nancy R Mead. Collaborations: closing the industry-academia gap. *IEEE software*, 14(6):49–57, 1997.

[5] Michaël Bikard, Keyvan Vakili, and Florenta Teodoridis. When collaboration bridges institutions: The impact of university–industry collaboration on academic productivity. *Organization Science*, 30(2):426–445, 2019.

[6] Elizabeth Bjarnason and Björn Regnell. Evidence-based timelines for agile project retrospectives–a method proposal. In *International Conference on Agile Software Development*, pages 177–184. Springer, 2012.

[7] Dhruba Borah, Khaleel Malik, and Silvia Massini. Are engineering graduates ready for R&D jobs in emerging countries? teaching-focused industry-academia collaboration strategies. *Research Policy*, 48(9):103837, 2019.

[8] Bruno Cartaxo, Gustavo Pinto, and Sergio Soares. *Rapid Reviews in Software Engineering*, pages 357–384. Springer International Publishing, 2020.

[9] Jeffrey C Carver and Rafael Prikladnicki. Industry–academia collaboration in software engineering. *IEEE Software*, 35(5):120–124, 2018.

[10] Esther de Wit-de Vries, Wilfred A Dolfsma, Henny J van der Windt, and MP Gerkema. Knowledge transfer in university–industry research partnerships: a review. *The Journal of Technology Transfer*, 44(4):1236–1255, 2019.

[11] Gerard Delanty. The university in the knowledge society. *Organization*, 8(2):149–153, 2001.

[12] Tore Dybå, Barbara A. Kitchenham, and Magne Jørgensen. Evidence-based software engineering for practitioners. *IEEE software*, 22(1):58–65, 2005.

[13] Emelie Engström, Kai Petersen, Nauman bin Ali, and Elizabeth Bjarnason. Serp-test: a taxonomy for supporting industry–academia communication. *Software Quality Journal*, 25(4):1269–1305, 2017.

[14] Henry Etzkowitz and Loet Leydesdorff. The triple helix–university-industry-government relations: A laboratory for knowledge based economic development. *EASST review*, 14(1):14–19, 1995.

[15] Michael Felderer and Guilherme Horta Travassos. The evolution of empirical methods in software engineering. In *Contemporary Empirical Methods in Software Engineering*, pages 1–24. Springer, 2020.

[16] Katia Romero Felizardo and Jeffrey C. Carver. Automating systematic literature review. In Michael Felderer and Guilherme Horta Travassos, editors, *Contemporary Empirical Methods in Software Engineering*, pages 327–355. Springer, 2020.

[17] Renato Garcia, V Araújo, S Mascarini, EG Santos, and AR Costa. How long-term university-industry collaboration shapes the academic productivity of research groups. *Innovation*, 22(1):56–70, 2020.

[18] Vahid Garousi, Michael Felderer, and Mika V Mäntylä. Guidelines for including grey literature and conducting multivocal literature reviews in software engineering. *Information and software technology*, 106:101–121, 2019.

[19] Vahid Garousi, Kai Petersen, and Baris Ozkan. Challenges and best practices in industry-academia collaborations in software engineering: A systematic literature review. *Information and Software Technology*, 79:106–127, 2016.

[20] Vahid Garousi, Dietmar Pfahl, João M Fernandes, Michael Felderer, Mika V Mäntylä, David Shepherd, Andrea Arcuri, Ahmet Coşkunçay, and Bedir Tekinerdogan. Characterizing industry-academia collaborations in software engineering: evidence from 101 projects. *Empirical Software Engineering*, 24(4):2540–2602, 2019.

[21] Eloïse Germain, Magnus Klofsten, Hans Löfsten, and Sarfraz Mian. Science parks as key players in entrepreneurial ecosystems. *R&D Management*, 2022.

[22] Robert L. Glass, Iris Vessey, and Venkataraman Ramesh. Research in software engineering: an analysis of the literature. *Information and Software technology*, 44(8):491–506, 2002.

[23] Tony Gorschek, Per Garre, Stig Larsson, and Claes Wohlin. A model for technology transfer in practice. *IEEE software*, 23(6):88–95, 2006.

[24] Magnus Gulbrandsen and Jens-Christian Smeby. Industry funding and university professors' research performance. *Research policy*, 34(6):932–950, 2005.

[25] Barbara Kitchenham and Stuart Charters. Guidelines for performing systematic literature reviews in software engineering. 2007.

[26] Barbara A. Kitchenham, Pearl Brereton, David Budgen, Mark Turner, John Bailey, and Stephen G. Linkman. Systematic literature reviews in software engineering–a systematic literature review. *Information and software technology*, 51(1):7–15, 2009.

[27] Barbara A Kitchenham, Tore Dybå, and Magne Jørgensen. Evidence-based software engineering. In *Proceedings. 26th International Conference on Software Engineering*, pages 273–281. IEEE, 2004.

[28] Keld Laursen and Ammon Salter. Searching high and low: what types of firms use universities as a source of innovation? *Research policy*, 33(8):1201–1215, 2004.

[29] Kalle Lyytinen and Youngjin Yoo. Ubiquitous computing. *Communications of the ACM*, 45(12):63–96, 2002.

[30] Inés Macho-Stadler, David Pérez-Castrillo, and Reinhilde Veugelers. Licensing of university inventions: The role of a technology transfer office. *International Journal of Industrial Organization*, 25(3):483–510, 2007.

[31] Dusica Marijan and Arnaud Gotlieb. Industry-academia research collaboration in software engineering: The certus model. *Information and software technology*, 132:106473, 2021.

[32] Christopher Marshall and Pearl Brereton. Tools to support systematic literature reviews in software engineering: A mapping study. In *ACM/IEEE international symposium on empirical software engineering and measurement*, pages 296–299. IEEE, 2013.

[33] Emilia Mendes, Claes Wohlin, Katia Felizardo, and Marcos Kalinowski. When to update systematic literature reviews in software engineering. *Journal of Systems and Software*, 167:110607, 2020.

[34] Tommi Mikkonen, Casper Lassenius, Tomi Männistö, Markku Oivo, and Janne Järvinen. Continuous and collaborative technology transfer: Software engineering research with real-time industry impact. *Information and Software Technology*, 95:34–45, 2018.

[35] Markus Perkmann, Rossella Salandra, Valentina Tartari, Maureen McKelvey, and Alan Hughes. Academic engagement: A review of the literature 2011-2019. *Research policy*, 50(1):104114, 2021.

[36] Markus Perkmann, Valentina Tartari, Maureen McKelvey, Erkko Autio, Anders Broström, Pablo D'este, Riccardo Fini, Aldo Geuna, Rosa Grimaldi,

Alan Hughes, et al. Academic engagement and commercialisation: A review of the literature on university–industry relations. *Research policy*, 42(2):423–442, 2013.

[37] Markus Perkmann and Kathryn Walsh. University–industry relationships and open innovation: Towards a research agenda. *International journal of management reviews*, 9(4):259–280, 2007.

[38] Kai Petersen, Robert Feldt, Shahid Mujtaba, and Michael Mattsson. Systematic mapping studies in software engineering. In *12th International Conference on Evaluation and Assessment in Software Engineering (EASE) 12*, pages 1–10, 2008.

[39] Kai Petersen, Cigdem Gencel, Negin Asghari, Dejan Baca, and Stefanie Betz. Action research as a model for industry-academia collaboration in the software engineering context. In *Proceedings of the 2014 international workshop on Long-term industrial collaboration on software engineering*, pages 55–62, 2014.

[40] Kai Petersen, Sairam Vakkalanka, and Ludwik Kuzniarz. Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and software technology*, 64:1–18, 2015.

[41] Per Runeson, Emelie Engström, and Margaret-Anne Storey. The design science paradigm as a frame for empirical software engineering. In *Contemporary empirical methods in software engineering*, pages 127–147. Springer, 2020.

[42] Per Runeson and Martin Höst. Guidelines for conducting and reporting case study research in software engineering. *Empirical software engineering*, 14(2):131, 2009.

[43] Per Runeson, Martin Höst, Austen Rainer, and Björn Regnell. *Case Study Research in Software Engineering: Guidelines and Examples*. Wiley Online Library, 2012.

[44] Per Runeson and Sten Minör. The 4+1 view model of industry–academia collaboration. In *Proceedings of the 2014 international workshop on Long-term industrial collaboration on software engineering*, pages 21–24, 2014.

[45] Ammon J Salter and Ben R Martin. The economic benefits of publicly funded basic research: a critical review. *Research policy*, 30(3):509–532, 2001.

[46] Anna Sandberg, Lars Pareto, and Thomas Arts. Agile collaborative research: Action principles for industry-academia collaboration. *IEEE software*, 28(4):74–83, 2011.

[47] Muhammad Usman, Nauman Bin Ali, and Claes Wohlin. A quality assessment instrument for systematic literature reviews in software engineering. *e-Informatica Software Engineering Journal*, 17(1):230105, 2023.

[48] Carol H Weiss. The many meanings of research utilization. *Public administration review*, 39(5):426–431, 1979.

[49] Claes Wohlin. Software engineering research under the lamppost. In José Cordeiro, David A. Marca, and Marten van Sinderen, editors, *ICSOFT 2013 - Proceedings of the 8th International Joint Conference on Software Technologies, Reykjavík, Iceland, 29-31 July, 2013*, pages IS–11. SciTePress, 2013.

[50] Claes Wohlin. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *Proceedings of the 18th international conference on evaluation and assessment in software engineering*, pages 1–10, 2014.

[51] Lanxin Yang, He Zhang, Haifeng Shen, Xin Huang, Xin Zhou, Guoping Rong, and Dong Shao. Quality assessment in systematic literature reviews: A software engineering perspective. *Information and Software Technology*, 130:106397, 2021.

INCLUDED PAPERS

PAPER I

A CASE STUDY OF INDUSTRY-ACADEMIA COMMUNICATION IN A JOINT SOFTWARE ENGINEERING RESEARCH PROJECT

Sergio Rico, Elizabeth Bjarnason, Emelie Engström, Martin Höst and Per Runeson. Journal of software: Evolution and Process, 33(10), 2021.

Abstract

Empirical software engineering research relies on good communication with industrial partners. Conducting joint research both requires and contributes to bridging the communication gap between industry and academia in software engineering. This study aims to explore communication between the two parties in such a setting. To better understand what facilitates good industry-academia (IA) communication and what project outcomes such communication promotes, we performed a case study, in the context of a long-term IA joint project, followed by a validating survey among practitioners and researchers with experience of working in similar settings. We identified five facilitators of IA communication and nine project outcomes related to this communication. The facilitators concern the relevance of the research, practitioners' attitude and involvement in research, frequency of communication and longevity of the collaboration. The project outcomes promoted by this communication include, for researchers, changes in teaching and new scientific venues, and for practitioners, increased awareness, changes to practice, and new tools and source code. Besides, both parties gain new knowledge and develop social-networks through IA communication. Our study presents empirically-based

insights that can provide advise on how to improve communication in IA research projects and thus the co-creation of software engineering knowledge that is anchored in both practice and research.

1 Introduction

Companies developing software, or software-intensive products and services, constantly strive to acquire software engineering competence to stay competitive. This involves getting access to people with relevant competence and developing the current knowledge within the company. Universities aim to be a source for both aspects of competence through graduating software engineers that can be employed in industry, and by conducting academic research that may add to the existing knowledge in the industry and contribute to improved industrial practices. Although the interplay between academic research and industry has been recognised as a way to exchange knowledge and innovate [2], little is known about how to manage mutual expectations and interaction [31]. Particularly in applied research disciplines like software engineering, the degree of interaction with industry is expected to be high as the research cannot be conducted in isolation in a university lab, but has to be – at least partially – conducted in real-world settings. Joint research projects, therefore, may provide mutual benefits for industry and academia. While industry gets access to competence, researchers gain insight into and access to real-world settings for their research [4].

Despite these potential mutual benefits, researchers have identified challenges in connecting research and practice [15]. The research topics and outcomes need to be relevant for industry [7, 14]. Research results should present practical advice to software engineering practitioners [25]. The time perspectives and incentives of industry and academia may be conflicting [34]. Industry and academia have to develop a symbiotic relationship to bridge the gap between the two parties [4].

Our research goal is to understand, within the context of a joint project between industry and academia (IA), 1) what factors can facilitate IA communication and 2) what outcomes that IA communication may contribute to. By *communication* we refer to the exchange of information between people, including verbal, written and visual information, and in what context this communication takes place, for example meetings, reports, and e-mail. Further, we acknowledge that information is different from knowledge, implying that communication is a means to promote outcomes of an IA research project, not a goal in itself as could be the case for exchanging knowledge. However, we hypothesise that communication is indeed an important factor for IA projects. Further, as engineering researchers, we focus on the organizational and practical aspects of communication in the context of an IA collaboration, rather than from a purely communication science perspective.

Researchers and practitioners communicate in different contexts and for different purposes throughout an IA research project [16]. Before officially start-

ing a project, the discussions are usually focused on selecting the research topic and building the team. Once the project starts, the participants jointly define the project plan which may be more or less flexibly defined. During the operation of the project, two types of communication take place, one is related to the research work where researchers, for example, collect empirical data, and practitioners get involved in the research process. Another type of communication concerns the management and reporting of the project. Finally, the knowledge is encapsulated in scientific publications and solutions that are disseminated among researchers and practitioners. By studying the communication between researchers and practitioners, we aim to gain knowledge on how to manage communication in future IA projects.

In this study, we investigate the following research questions:

RQ1
: Can we identify certain conditions, activities, relations, or practices that facilitate mutual communication between researchers and practitioners?

RQ2
: Given IA communication, as observed in RQ1, what outcomes for practitioners and researchers can be identified that are promoted by this IA communication?

The relation between communication and outcomes is complex, probably bi-directional, and includes many confounding factors embedded in the context. To enable us to study this complex phenomenon, we chose to conduct a case study of an IA research project to answer these questions. Case study methodology allowed us to perform "an empirical enquiry that draws on multiple sources of evidence to investigate one instance ... of a contemporary software engineering phenomenon [i.e. IA communication] within it's real-life context... when the boundary between phenomenon and context cannot be clearly specified" [33].

We explored the characteristics and outcomes of the communication within our case project, and validated our findings through a survey. As our case, we studied a 3-year project within a 10-year research program including collaboration between two Swedish universities and local branches of three industrial corporations with international outreach. Our main data collection consisted of a project retrospective that was conducted using a time-line based method [5] at the closing stage of the research program. The retrospective was conducted as a focus group meeting using a time-line as a catalyst for the data collection. The time-line visualised key events within the project and was prepared before the meeting. The audio recording from the focus group meeting was transcribed, coded, and thematically analysed in line with our research questions. Later, the results from the analysis were validated through a survey with a broader population. The survey was based on the communication facilitators and related project outcomes identified in the case study.

The main contributions of this paper are twofold. Firstly, we explore the role of IA communication within a joint research project and what characteristics of the project that facilitated this communication. Secondly, we identify some outcomes of the IA research project that were promoted by the IA communication within the project.

We describe related work in Section 2 and our case study in Section 4, including the case and research method. Our results from the case study and the validation survey are presented in Section 5 and discussed in Section 7. Finally, Section 8 concludes the paper.

2 Background and Related Work

In this section, we present an overview of the research related to our study. Firstly, we describe some perspectives considered by researchers when analysing IA research. Mainly we are interested in how IA communication has been investigated. Secondly, we present research findings in software engineering related to IA collaboration and the role of communication. We observed that our view of IA communication might be broader than the one adopted by other researchers in the field. Consequently, we expect that this section contributes to present the IA communication perspective used in this paper.

Researchers across disciplines have investigated IA research from different perspectives. One example is a review published by Salter et al. [36], where the authors investigated the economic impact of public-funded research. The authors identified six types of contribution to economic growth related to the extension of useful knowledge, training of graduates, new scientific methods, networks and social interaction, increased scientific and technological problem solving, and new companies. Good et al. conducted a literature review from an organisational perspective of technology transfer ecosystems [19], i.e. university-affiliated organisations that are involved in technology transfer activities. Specifically, the authors analysed technology transfer offices, science parks, incubators, and university venue funds. The authors concluded that those structures have been studied in isolation and highlighted the need for a holistic approach. In another review, published by Perkmann and Walsh, the topic is the interaction channels between industry and academia and the contribution to open innovation [31]. In this review, IA research projects, like the one in our case study, are identified as one of the connections between industry and academia, namely research partnerships. Other identified connections are research services, commercialisation of intellectual property rights, and people exchange. Perkmann and Walsh also found that IA research has mainly been focused on effects and less on how IA research is performed, which is what our study investigates for the aspect of communication.

Among the research about how industry and academia work together, communication has been identified as an essential factor. Ankrah and Al-Tabbaa [2]

conducted a systematic review of 109 papers on industry-academia collaboration across different research disciplines. They present a model to represent IA projects that covers motivations to collaborate, how the collaboration is formed and operated, the factors that enhance and inhibit the joint work, and the outcomes. In their model, communication is mentioned twice. First, as an activity throughout the joint project. Participants communicate formally or informally by voice, email, video calls, etc, and by publishing written material such as reports, booklets, newsletters, bulletins, and research papers. Second, as a factor that facilitates or inhibits the organisational management and, therefore, the joint work. Both these aspects of communication are included in our study.

Similarly, from a source of 103 papers about industry-academia collaborations, Rybnicek and Königsgruber [35] recognise the importance of communication as a factor that influences the relationship. They started their analysis based on the facilitating factors for IA collaboration identified by Ankrah and Al-Tabbaa [2] and identified the following facilitating factors of communication. Frequent communication is essential for developing a shared understanding of cultural differences [21], backgrounds, and interests that may affect communication [26]. Personal contacts and relationships are vital for developing work networks. These contacts are important both on the management and operational level. Companies often select partners based on their expertise and the research reputation of the institution [11]. Adequate communication channels and regular face-to-face contact have a positive influence on the relationship [10]. The use of common languages and mutual understanding also affect IA communication since researchers and practitioners may use different terminologies [3, 17]. Each partner needs to be aware of the other partner's terminology. Cultural differences, e.g., the way of working, may hinder IA research [8]. Consequently, distances between the partners need to be identified and addressed early on in joint projects [8]. While some of the factors described above coincide with our findings, we identified some additional factors that may foster IA communication.

In software engineering, Garousi et al. [15] conducted a systematic literature review on IA collaborations, with a final set of 33 primary studies. The authors identified challenges and best practices in IA collaboration. They adopted the model proposed by Ankrah and Al-Tabbaa [2] to represent the formation and operation of IA projects. Challenges related to communication were identified in all project phases. Some of the challenges related to communication included gaps in time horizon, areas of interest, and responsibilities; difficulties at handling multiple collaborators; lack of standard terminologies; and low pre-existent networks before the projects. Following, we briefly mention some examples of best practices in line with the results of our study. Negotiate and elicit research topics with practitioners before conducting industrial experiments is an example of a best practice that may increase the trust between the participants and contribute to select industry-relevant problems [30]. Another example is to have a local champion, i.e., an engaged practitioner. This brings benefits to the joint project including

initiating studies faster, access to data, contact with business units and stakeholders [20]. To attract top management support is a best practice that balances the joint project's research objectives and brings value to each participant [24]. Another best practice is to conduct weekly meetings that may enable practitioners to more quickly test and give feedback on new ideas [13]. Finally, since innovation and impact on practice take time, establishing long-term relationships is a good practice. High-quality and relevant research results tend to be supported by long-term research, rather than being the result of a short-term research project [23]. An interesting observation from this study is that among the 33 papers included, 17 were by Scandinavian authors (14 by Swedish researchers). For software engineering, this factor may indicate Scandinavian countries' willingness to develop IA research and conduct research on this topic in order to further improve it.

In a follow-up study of IA projects, Garousi et al. [16] surveyed 64 respondents around the world and identified which of the challenges and patterns identified in their previous study that impacted the projects described by the respondents (101 projects). The authors found a high impact of challenges related to mismatches between industry and academia, human and organisational challenges and lack or drop of interest/commitment, and less impact of communication-related challenges. Notice that for communication challenges, the authors limited their inquiry to communication channels used during the execution of the project, e.g., problems with Skype or dealing with several partners. Our view of IA communication goes beyond communication channels and communication only during the execution of specific projects. We consider each researcher–practitioner interaction a communication instance regardless of whether it happens when defining a research topic, as part of a research study, or when diffusing research results.

Researchers have investigated and proposed several models related to IA research in software engineering. Sandberg et al. [37] presented a relational model that includes ten principles for managing IA collaborations. The model is based on research on collaborative practices [28]. Marijan and Gotlieb [27] presented the Certus model to reflect IA knowledge co-creation. The model relies on the idea that research needs to be performed jointly by researchers and practitioners, and that this requires continuous dialogue and alignment between the participants. Similarly, Mikkonen et al. [29] published a model describing continuous and collaborative technology development. Their model supports the idea that innovation is not developed in academia and transferred to industry. Instead, it is joint research between industry and academia that leads to innovation. The first two models were derived from research programs similar to the one in our case study, one in Sweden and one in Norway, and the third from a national research program in Finland. Although these models do not explicitly model IA communication, they model IA research, which we believe relies on and creates IA communication.

Wohlin et al. [38] surveyed industry and academia representatives in Sweden and Australia about success factors for IA collaboration in software engineering. Having support from top management and a champion (contact person) on the

industrial site were considered the top factors for success, both by industry and academia respondents. Communication factors were not ranked explicitly (except for "regular meetings"), but they are inherent in several of the involved factors.

In summary, previous research identifies communication as one important factor in IA projects. However, there are very few studies explicitly investigating the role of IA communication.

3 Research Method

The research was conducted in two main phases with a total of eight steps, as visualised in Figure 1. In the first phase, we performed a case study of an IA research project. In the second phase, we conducted a survey to validate the findings from this case study. The survey was conducted with a broader set of participants than those included in the case study.

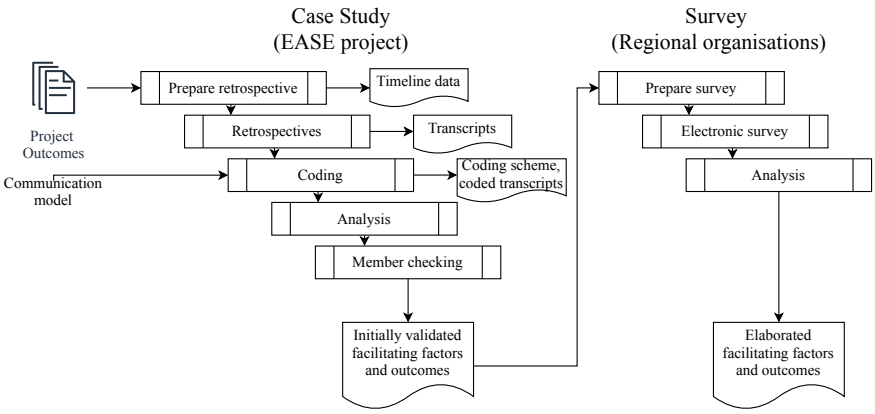


Figure 1: Overview of research method

3.1 Case study

The objective of the case study was to investigate our research questions, i.e. to identify factors that can facilitate IA communication and outcomes promoted by such IA communication in a joint project. As our research goal was to investigate this type of complex phenomenon, where the borderline between the phenomenon and the context is not clear, we chose to perform a case study since this which would allow us to study communication in its context [33]. Case study methodology is by definition based on studying one, or a small number of instances, where generalization can not be derived statistically but rather analytically by comparing

case characteristics and assessing the relevance of the findings for other contexts. Consequently, case study findings are not focused on quantitative outcomes, but on the qualitative understanding of a complex phenomenon in its context. Studying a specific case allows us to gain in-depth insight into IA communication. Our unit of analysis is a research project, as described in the next Section.

Case Description

The case study was conducted within The Industrial Excellence Centre for Embedded Applications Software Engineering, EASE – a 10-year research program performed 2008–2018 in close IA collaboration. The program involved two academic partners and three industrial partners. The partners are all active in Southern Sweden, within a 2-hour drive. The industrial partners all operate on an international market, and the two larger ones are either part of, or owned by, Japanese multinational corporations. The program budget comprised 10.5 MSEK ($\approx$ 1 M€ or 10 full-time equivalents) per year, and was jointly funded by industry (50%), academia (33%), and a national innovation agency (Vinnova) (17%). The overall goals of the program were threefold:

- availability of competent personnel,
- making results useful for industry, and
- research excellence.

While these goals may be considered contradictory, industrial and academia partners agreed on that they were fully compatible through the conduct of applied software engineering research, published in highly ranked publication outlets.

The research program included three to four projects in parallel, organised around different topics in software engineering. A board of directors, composed of representatives for the funding organisations, made the decisions on which themes to explore, and the budget for each project. The detailed scope and deliverables of the projects were defined in an agile manner, focusing on mutually agreed outcomes over comprehensive documentation. Thus, the project members jointly and continuously discussed and defined what and how research should be performed within the project. Within the program, PhD students, postdocs, and faculty were funded to a varying degree throughout the program. Master of Science (M.Sc.) student projects were also executed within the program, although financed by separate sources. Decisions about the acceptance of new PhD students into the program were taken at the program board level, while at the project level, specific research activities were decided. Parts of the contributions from industrial were in kind, with company employees working in, and interfacing with the research program.

The researchers involved in the program were active in the fields of software engineering, software technology, and computer engineering. The majority of the senior researchers were Swedish natives, as were about half of the PhD student

while others originated from other European and Asian countries. Industry participants were dominantly Swedish, although for two of the partner companies their corporate language was English, since these companies are international and employ international staff also in Sweden. Thus, the project internal language of the studied research program was English, while the communication and management culture was dominantly Scandinavian.

The collaboration practices during the formation phase of the program are previously published [32], while we herein focus on a specific project that operated during the third phase, comprising the last three years of the program. The joint projects executed during the last three years of the research program had the following themes[1]:

- A. Configuration and interaction in internet-of-things
- B. Parallel execution for embedded systems using machine learning
- E. Increased efficiency in software development through decision-support in the testing process

For each of the projects in the program, a reference group was set up with one or more representatives from each company involved in the specific project. The reference groups met regularly with the researchers within the project to share progress reports and discuss the next research steps. Once a year, a two-day conference was held off-site to report progress across the program and to discuss and plan the research in more depth. In addition to these management meetings, industry and academia representatives met, to work on developing research prototypes, for interviews and empirical observations, and for planning purposes. In total, 500 IA meetings were recorded during the 10-year duration of the program, eight PhD theses were examined, and more than 200 scientific papers were published.

In our case study, we investigate one of the projects that was active during the final three years of the research program (theme E above), in which the four last authors of this paper were part. This project focused on decision-support in the testing process. The project group consisted of 6–13 researchers and 3–5 practitioners, where most of the senior researchers had been involved in a previous project within the same research program. The high proportion of researchers is due to an increase in the number of PhD students. The wide range in number of researchers is due to that PhD students and faculty members funded by other projects, also participated in the activities of the case study project in order to benefit from the IA environment provided by the research program. Research activities in this project included literature studies and synthesis, problem conceptualisation through interviews and observations, development of solutions and evaluation of these in industrial contexts. Studies conducted by faculty, postdocs and M.Sc. students could be run over a couple of months, while PhD student projects had

[1]The enumeration scheme comes from projects C and D of phase 2 being merged into project E.

a longer time perspective in order to fit into the thesis work. However, specific studies within the frame of PhD student projects may have shorter timelines.

Research results from the case project include systematic literature reviews (one of which included a perspective that was particularly relevant to industry, namely industrially evaluated regression testing methods [1]), practical guidance to industry on specific software engineering methods, for example, test scoping [12], automated bug assignment [22], and exploratory testing [18], and theory to explain and improve communication within software engineering [6]. Some articles were published in practitioner-oriented magazines, while most papers were published in high-ranked journals and conferences. One of the sub-projects is presented by Carver and Prikladnicki [9] as an example of a successful IA collaboration in software engineering. Regularly, researchers were invited to companies to present their results, or practitioners were invited to the universities for seminars.

Preparing Data Collection through Retrospectives

The main data collection for this case study was conducted at a retrospective meeting based using evidence-based timelines that facilitated reflecting on how industry and academia had worked together within the research program. A retrospective method called EBTR (evidence-based timeline retrospective) [5] was used. This method enables designing a retrospective to focus on specific areas or topics through specifying goals that are then detailed into a) focus questions and b) aspects to visualise on timelines based on data, or evidence. Both the focus questions and the visualised aspects of project history are selected with the aim of triggering and supporting group reflections in-line with the goals defined for the retrospective. In this case, the retrospective's overall goal was defined as understanding the value of the IA partnership by exploring how joint work was performed within the research program and what benefits had been gained both short and long term. Since this included considering by whom and how the work had been performed in each project, the material allowed us to study the communication between industrial and academic partners in the context of a research project and connected to the outcomes and benefits of that project.

Based on the goal of the retrospective, four timelines were defined. Each timeline represented an aspect of the studied research project's history. These aspects captured

- people involved in the project,
- interaction events, e.g. project meetings and workshops,
- needs and activities, e.g. industrial needs and research activities, and
- outcomes such as industrial impact, research results etc.

Prior to the retrospective meeting the available project documentation was studied and evidence-based timelines were constructed from available project data

(evidence) thereby providing a visualisation of the project history. An extract from the timelines used in the retrospective is shown in Figure 2. For each project, one timeline per aspect was defined. The second author of this paper collected the data used to populate these timelines and produced the timelines based on a selection of this data. The data was collected from project reports, minutes of meetings, and publication lists. Prior to the retrospective meeting, the project participants of the studied case project were asked to complement this data by providing information not directly available in the documentation, e.g. about M.Sc. projects.

The evidence-based timelines were validated prior to the retrospective meeting by sending them out to the participants of the studied research project together with a list of key publications. The participants were asked to prepare for the retrospective meeting by skimming the material and reflecting on the main topics researched in the project, the industrial needs, gains and impacts. In addition to providing quality assurance of the timelines, involving the participants in the preparations motivated and prepared them for active participation in the retrospective event.

One moderator per project was recruited in this preparatory stage. The moderators all had previous experience of leading retrospectives and focus group sessions, but had not been actively involved in the research project that they were to moderate.

Data Collection through Retrospectives

Half-day retrospective meetings were held for each of the three projects active in the final phase of the research program. The retrospectives were held with project members attending a physical two-day event using a set of four timelines visualising project history. Prior to the meeting, three of the four timelines were populated based on evidence found in project documentation, namely people, interaction events and outcomes (see above). The fourth timeline, with needs and activities, was populated during the meeting as part of the retrospective discussions. In the retrospective meeting, the pre-prepared evidence-based timelines supported the participants in remembering past project events, and thus triggered and enabled a fact-based discussion guided by the pre-defined focus questions [5].

During the retrospective meeting, the participants were presented with the partly populated timelines printed on 2 x 1 m cloth, placed on a large table around which the participants gathered, see Figures 2 and 3. The retrospective participants worked for about three hours, analysing and discussing the research project based on the timelines. The moderator guided the retrospective meeting using the pre-prepared focus questions (see above). At the meeting, the participants alternated between individual reflection and group discussions. During the meeting, the participants populated the timelines with more details, and when necessary, corrected or adjusted pre-printed timeline data. All project members, past and present, were invited to the retrospective meeting, and for the project reported in this paper, there were eight participants from academia and three participants from industry. One

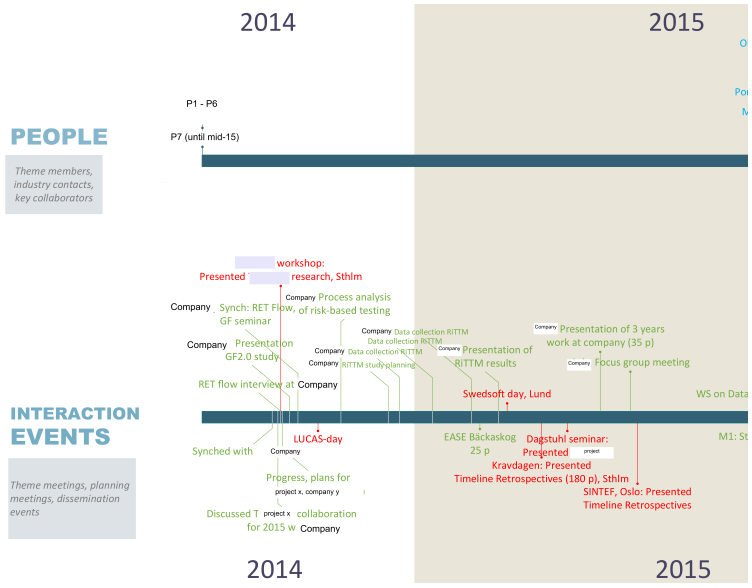


Figure 2: Example of a minor part of the timelines used at the retrospective meeting (anonymized for confidentiality).

of the academics acted as the moderator and led the retrospective meeting. The moderator also ensured that both the industrial and the academia perspective were equally voiced during the meeting, although they were imbalanced in numbers. The participants from industry had all been actively involved in the project under study, and all played an active role throughout the retrospective meeting. Among the academic participants, all had been involved in the project to varying degrees, and their active participation to the retrospective varied with the extent of their involvement in the project. Three to four of the academic participants were active in research studies, while the rest were involved as supervisors and in various managerial roles, thus boosting the number of academic participants.

The main outcome of this step consists of transcriptions of the retrospective meeting. The meeting was recorded using both video and audio, and the audio files were transcribed word-by-word by a professional transcriber. In addition, the participants made notes on the timelines of additional events, connections etc.

Coding

The transcripts were imported into QSR International's NVivo 12 qualitative data analysis software for coding and analysis. Coding was conducted in two steps. Initially, four researchers (authors 1–4) read through the material and independently

Figure 3: Discussions around the timelines (placed on the table) at the retrospective meeting with case project.

identified themes and proposed codes. The initial codes were proposed based on our pre-understanding of IA communication. A common coding scheme was then agreed on in a joint meeting. We formulated a communication model, Figure 4, based on the main categories from our code scheme consisting of communicating parties, communication context, content of the communication, and outcomes. The main categories included sub-categories and nodes. For example, the main category "Communication Party" included the sub-category "University" and the node "Researcher".

In the next step, two researchers (1st and 2nd author) coded the material according to this scheme identifying all the communication instances according to the model. By a communication instance, we refer to a segment of text in the transcription that explicitly mentions communication between two communicating parties. For each instance, we coded all the categories according to the model when possible. For example, the node "researcher" in the category "Communication Party" was used all times that in the transcription, one researcher communicated with someone else.

Figure 4: The communication model used as the basis for the coding of communication instances

Analysis

After coding, further analysis was conducted to identify patterns in the material based on the codes and combinations of these. For example, we analysed the frequencies of codes for occurrences of communication contexts including both industrial and academic communicating parties. We settled on describing the case project using the communication contexts and the *project outcomes* since these provided interesting insights. During the analysis, factors *facilitating* communication were identified by the researchers. These facilitators and outcomes were analysed and discussed further by the researchers and are described in Section 5.

Member checking

We conducted a first validation by sending out the results in the form of an early version of Section 5 to participants of the studied case project. We asked two representatives from industry and two senior researchers that had been active in the case project, to give feedback on the results. In particular, we asked them to comment on results that agreed with their experiences; what results that did not agree

with their experiences; and if possible, also what results that were new or surprising to them. The feedback was collected and used to validate the list of facilitating factors and outcomes.

3.2 Survey

We performed a survey to validate the results of our case study with a broader set of participants beyond that of the studied case project. For this reason, we constructed a survey and invited survey participants through a mailing list managed by our research group. The mailing list covers our broad IA network on software engineering and consists of practitioners from different industrial organisations in Sweden. Since this list also contains other mailing lists, it is difficult to state the exact number of participants that were invited to participate in the survey. We estimate that the mailing list reaches at least 500 email addresses and at least 60 companies. The sample was chosen because it was seen as a natural extension of the sample we worked within the case study.

The survey instrument consisted of questions based on the case study results. The participants were asked which of the identified facilitators and outcomes they have experienced themselves in previous IA collaborations, and to note additional ones. The main questions were:

- Characterisation questions, mainly in which sector they work, i.e. industry, public sector, research institute or academia
- What outcomes they have experienced from IA projects (selected from the identified outcomes). They were also able to add new outcomes that were not listed (in free text form)
- What outcomes the participants thought were important in IA communication (selected from the identified outcomes)
- What facilitators the participants believe are valid (selected from a list of the identified facilitators). They were also able to add new facilitators that were not listed (in free text form)

In total 50 respondents completed the survey. We grouped the respondents into two groups, one group consisting of 17 researchers (13 from academia and 4 from research institutes) and one group consisting of 33 practitioners (2 from public sector and 31 from industry). In the analysis, we explain the patterns found in their responses.

For all three survey questions (confirm the validity of facilitator, experience of outcomes, and importance of outcomes), the respondents could mark any number of alternatives, from 0 to 5 for the facilitators, and from 0 to 9 for the outcomes. Since we did not ask for invalid facilitators or unimportant outcomes, we believe that the respondents only marked the alternatives that were the most valid and/or important to them.

4 Results

The results of this case study consist of findings derived through analysis of the evidence-based timeline retrospectives and validation of these findings in a survey.

Table 1: Overview of identified facilitators of IA communication

Code	Name	Description
F1	Research Relevance	The degree to which the research represents real problems faced by companies
F2	Practitioner's Attitude towards Research	The predisposition of practitioners to participate in joint research
F3	Active Practitioner Involvement	The degree to which practitioners participate in joint research
F4	Frequency of Communication	Particular characteristics of meetings that define a way of conducting meetings
F5	Long Term Collaboration	A formal joint research project that takes places over a longer time period, e.g. beyond 2–3 years.

4.1 Case study findings

In the analysis, we identified factors that facilitate IA communication and project outcomes promoted by this communication, based on communication instances described in the retrospective meeting. When analysing the coded transcript of the retrospective meeting, we noticed similarities between the contexts in which the communication instances were described to have taken place. Therefore, we grouped the communication instances by their communication context. We identified three main contexts, namely the **IA environment as a whole**, **project-related meetings**, and individual **studies**. In the context of the IA environment, we observe IA communication beyond that of project-related meetings and individual studies. In the context of project-related meetings, project members communicate on project-related topics through meeting physically or online. Finally, in the context of studies, we observed communication in all phases of individual research studies including M.Sc. projects. We have identified characteristics of each context that facilitate communication between industry and academia, and that promote project outcomes. The facilitators are referred to as F1–F5, see Table 1, and the related project outcomes are referred to as O1–O9, see Table 2.

In the following subsections, the identified facilitators and outcomes are presented with respect to the communication contexts in which they have been observed. Curly brackets are used in the text to denote relations identified between factors facilitating IA communication, and subsequently promoted outcomes. For

Table 2: Overview of identified outcomes promoted by IA communication

Code	Name	Description
O1	New Knowledge	New knowledge that is produced in the joint project e.g papers, code, tools
O2	Awareness	A sense of general informed knowledge about ongoing research and results
O3	Changes in Practice	Changes that take place in companies motivated by research results
O4	Tools and Source Code	Source code or tools that are implemented in products or the value chain of companies based on research results
O5	Social Networks	Social and contact networks that arise and develop within IA projects and that remain beyond the time frame of a specific project
O6	New Studies	New research studies and M.Sc. projects that emerge from an IA project
O7	Good IA Collaboration	Improvements to the ecosystem to facilitate joint work
O8	Changes in Teaching	Changes in the content of courses at University level introduced by researchers involved in an IA project
O9	New Scientific Venues	New forums where researchers exchange with other researchers, sprung out of IA projects

example, "{F1 $\xrightarrow{C}$ O1}" denotes a relation between F1 and O1 in that factor F1 (Research Relevance) facilitates IA communication and thereby promotes the project outcome O1 (New Knowledge). The letter C over the arrow indicates that it is an indirect relation over communication in the IA project. In some cases, more than one factor in combination were identified, and in some cases, more than one outcome were identified, which is marked by listing a set of factors/outcomes in parentheses. A summary of the relations is shown in Figure 5.

IA Environment

The IA Environment refers to the whole research program as a context where communication occurred beyond the context of a specific project or study. Some of the identified outcomes are not directly related to specific events or meetings. Rather, the participants expressed that the program in itself acted as *"an engine for generating more and more collaboration on all different levels"*.

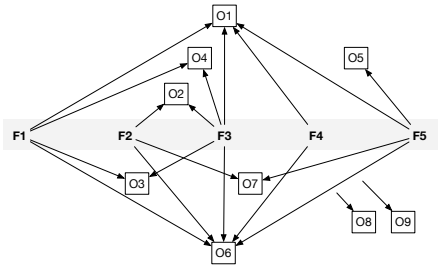


Figure 5: A summary of the indirect relations via communication ($\{\text{Fx} \xrightarrow{C} \text{Oy}\}$) between facilitators (Fx) and outcomes (Oy)

The **long-term collaboration (F5)** supported by the 10-years research program facilitated communication between researchers and practitioners. Within the long-term horizon of the program, the participants' **social networks (O5)** were expanded $\{\text{F5} \xrightarrow{C} \text{O5}\}$. The participants expressed that this long-term aspect of the context, in some cases longer than the research projects, was inductive to initiating **new studies (O6)**, including master thesis projects and research studies $\{\text{F5} \xrightarrow{C} \text{O6}\}$. Similarly, the context provided junior researchers with an environment through which they had access to and could work with industry. The participants expressed that the continuous way of working and delivering value to the industrial partners motivated them to participate and thus led to **improved IA collaboration (O7)** $\{\text{F5} \xrightarrow{C} \text{O7}\}$. The industrial partners expressed that **the long-term collaboration (F5)** facilitated communication with academia and yielded benefits in the form of **new knowledge (O1)** that was useful both in the short-term and the long-term perspective $\{\text{F5} \xrightarrow{C} \text{O1}\}$. This relation concerning the long-term aspect of the collaboration was expressed by one participant from industry : *"We could apply the results directly ... we got long term proof that enabled us to see that, yes, we are doing the right things"*. One participant also pointed out that the **long-term collaboration (F5)** facilitated staying focused on the agreed long-term plans without being affected by the company's operational priorities. Thus, the research project was shielded from short-term industrial perspectives. Overall, the long-term collaboration led to mutual learning about each other, whereby the IA communication was further facilitated.

The communication between industrial and academic partners in the IA environment led to developing a **social network (O5)** where personal contacts, even beyond organisational affiliations, were established and kept active. During the project, some of these industrial discussion partners became actively involved in the research as formal company contacts. Both researchers and practitioners expressed that the informal environment around the project was very positive and facilitated IA communication, which in turn generated additional IA research.

Even further, the participants described that through participating in the project they strengthened their ability to communicate with industry and academia, and that this, in turn, promoted the identification of novel ideas for further **new studies (O6)** and joint projects. In summary, communication in the social environment within the long-term research partnership stimulated knowledge exchange that promoted further and **improved IA collaboration (O7).**

Through our case study, we observed two outcomes on the academic side that were promoted through the communication with industry, one regarding teaching and one related to scientific forums. Several academic participants described that their involvement in the case project and communication with industrial partners led to **changes in teaching (O8)**, in particular within the courses for which they were responsible. The awareness of industrial needs and the new knowledge gained through IA communication in the project thus promoted improvements to university courses. Furthermore, the communication between industrial and academic partners around requirements and testing, created an awareness of the relevance and importance of this topic that stimulated the establishment of a **new scientific venue (O9)** in the shape of a scientific workshop series on this topic[2]. This new international forum provides researchers and practitioners with the opportunity to exchange knowledge and experience around one of the leading research topics of the case project.

Project-Related Meetings

Based on our empirical data, we have identified two main types of project-related meetings where industrial and academic partners communicated about research and industrial needs at a general level (as opposed to meetings related to specific research studies, see next section). These two types of meetings were either of a creative nature or related to the project organisation. The creative meetings observed in our material took place during the formation phase of the project, when senior researchers met with industrial contacts. The communication at these meetings promoted **good IA collaboration (O7)** in jointly defining the research direction for the project. Through brainstorming sessions involving project members from both industry and academia, the main areas of interest were identified. As stated by one of the senior researchers involved in the management of the project, these jointly agreed areas *"formed a frame for what was actually done"* . By **involving practitioners (F3)** also in this formation phase, and by basing the scope on industrial needs, thus ensuring **research relevance (F1)** further facilitated IA communication in the project. There were multiple meetings with various companies during the formation phase. For some of these meetings, the relevance of the research and the involvement of practitioners facilitated IA communication that led to initiating joint **M.Sc. projects (O6)** $\{(\text{F1, F3}) \xrightarrow{C} \text{O6}\}$, even for companies that did not become formal partners in the research project.

[2] `https://ret.cs.lth.se`

The most common type of project-related meetings were project meetings. For our case project, such meetings were held regularly every 6–8 weeks with all the involved researchers and the industrial contact persons. Most of the times participants were present in person at these meetings, with the exception of researchers from one of the university sites that occasionally attended via Skype. At these project meetings, status and intermediate research results were presented and discussed, and the industrial partners shared new or changed needs from their perspective. The communication at these meetings played an essential role in promoting **good IA collaboration (O7)** in jointly detailing and agreeing to the research direction, and in initiating **new research studies (O6)**. The **frequency and style (F4)** of these meetings and the **active involvement of practitioners (F3)** created a positive communication climate where ideas, needs and intermediate results were shared and discussed. For example, early on in the project, the industrial contacts expressed a preference for focusing on decision-making specifically for testing when *"the companies said, we want to look at testing"*. This was agreed as the direction in which the research then proceeded, thereby strengthening the **relevance of the research (F1)** for the industrial partners. This relevance was further stimulated when *"the specific [industrial] needs became studies"* and thus the IA communication led to jointly defining **new studies (O6)** $\{(\text{F1, F3, F4}) \xrightarrow{C} \text{O6}\}$. An example of this is a systematic literature study that was initiated when industrial partners expressed a need to understand the state of the art regarding test case selection and prioritisation [1]. Due to the industrial interest in this topic, one of the company contacts were actively involved in reviewing articles in this literature review and thereby acquired **new knowledge (O1)** $\{(\text{F1, F3, F4}) \xrightarrow{C} (\text{O1, O6})\}$ through participation in the research.

Studies

The research project included both research studies and industrial M.Sc. projects related to the topics covered by our case project. The research studies were initiated based on joint agreement at the project meetings (see above) and were relevant to the industrial partners, thus ensuring research relevance for these joint studies. Similarly, the M.Sc. projects were highly relevant to industry since companies directly initiated these projects, sometimes with a researcher within the project. These M.Sc. projects thus stimulated further IA communication in the shape of joint-supervision. These industrial M.Sc. projects applied scientific methods to design and validate solutions to industrially relevant problems for the companies.

Research Studies The research studies within our case project were performed with industrial partners through **active practitioner involvement (F3)** in all phases of the studies, including research design, data collection and analysis. This active involvement, in combination with the **style of meetings (F4) w.r.t. regularity** and

open discussions facilitated frequent and regular communication between the researchers and practitioners involved in each study. This factor related to the style of the meetings was also observed to facilitate communication at the project level meetings. Thus, the IA communication promoted that the company contacts gained **new knowledge (O1)** $\{(\text{F3, F4}) \xrightarrow{C} \text{O1}\}$ and deep insights into the research results through early access to results from the ongoing studies. This in turn enabled the practitioners to improve processes and tools within their companies. Thus, the IA communication in these meetings also promoted **changes in practice (O3)** $\{(\text{F3, F4}) \xrightarrow{C} (\text{O1, O3})\}$. For example, two of the participating companies implemented changes to their test strategies based on results obtained and communicated within the project. One company representative expressed that *"when I saw some benefits, I implemented that."* Thus, the fact that the research was **relevant (F1)** to the industrial partners facilitated the IA communication and led to **changes in practice (O3)** $\{\text{F1} \xrightarrow{C} \text{O3}\}$.

Most of the research studies within our case project were performed as case studies, and included activities at the companies such as *data collection* and *research seminars*. Some of the data collection methods that were used had the added benefit of disseminating **new knowledge (O1)** directly to the participating practitioners. In particular, this was the case for *focus groups* and *interactive posters* where the informants were presented with research ideas and topics, and asked to reflect and give their views on these either at a meeting or individually by marking their viewpoints on a publicly available poster. This approach created a win-win situation, where active **practitioner involvement (F3)** in the data collection facilitated IA communication which then led to the practitioners gaining insights in the shape of **new knowledge (O1)** $\{\text{F3} \xrightarrow{C} \text{O1}\}$. For example, a set of focus groups were held around the topic of exploratory testing where different templates for expressing exploratory test cases were presented to the participants who then got to try them out [18]. These focus groups, and the IA communication that took place there lead to **changes in practice (O3)** for the participating test team who *"modified [their test practices] and have seen the direct impact"*. This team also spread their new knowledge to *"related teams within neighbouring areas"* within the company. Similarly, within a case study of ten teams, the team members were asked to assess the ease of working with other teams through voting by noting their viewpoints on posters, so called interactive posters. This approach of active **practitioner involvement (F3)** in the data collection facilitated IA communication and promoted an increased **awareness (O2)** $\{\text{F3} \xrightarrow{C} \text{O2}\}$ of the research topic (in this example, communication gaps) and an interest in the ongoing research among company employees. This involvement also enabled the researchers to spread **new knowledge (O1)** $\{\text{F3} \xrightarrow{C} \text{O1}\}$ of the underlying theoretical model to the entire studied department consisting of around 200 people. In this case, the company contact described that the use of interactive posters had promoted a new **awareness (O2)** and insight within the organisation regarding potential causes of

communication gaps that helped people to be more tolerant of each other and being proactive in how they communicate with "difficult" teams.

In the case project, research results were disseminated and communicated to industry in several ways, including through *seminars* at the companies. The seminars led to the practitioners gaining **new knowledge (O1)** and increased **awareness (O2)** in general. As one researcher stated, *"some things are tacit, in the sense that you get more informed ... not necessarily a specific method, but you have awareness"*.

Industrial M.Sc. Projects The industrial M.Sc. projects provided a context where communication promoted establishing personal contacts and **social networks (O5)** between practitioners and researchers. For example, one of the case project's company representatives first became acquainted with one of the researchers when they co-supervised an M.Sc. project at the company, and this then led to participating in the case project. The practitioner's previous experience of working with the researcher positively influenced the **practitioner's attitude (F2)**, which further facilitated the practitioner's communication with researchers and **improved the IA collaboration (O7)** $\{\text{F2} \xrightarrow{C} \text{O7}\}$. Therefore, the practitioner was more **aware (O2)** $\{\text{F2} \xrightarrow{C} (\text{O2, O7})\}$ of ongoing research and available to participate in **new studies (O6)** $\{\text{F2} \xrightarrow{C} (\text{O2, O6, O7})\}$. The **research relevance (F1)** and the **practitioner involvement (F3)** in the project played an important role for the scope and impact of the M.Sc. projects. Given that topics of the M.Sc. projects were of interest to the researchers who actively participated in the project, researchers and practitioners could define the scope of these M.Sc. projects jointly in order to become more relevant and useful to the companies and to the researchers. Furthermore, through communication of M.Sc. projects within the IA project, similar and overlapping interests were identified in other areas of the company, which led to broadening the outreach of the results from the M.Sc. projects.

Continuous communication between researchers and practitioners involved in industrially relevant research, provided a direct impact on practice within the participating companies. Industrial M.Sc. projects often provided direct value in the shape of **tools and source code (O4)**, and this relevance facilitated the adoption of these results within the companies. For example, one M.Sc. project resulted in a tool for automatically prioritising issues in the company's issue management system. This tool was used as is in the company's software development organisation and thereby saved time and effort in issue prioritisation. Another example is an M.Sc. project that implemented an automatic checker for architectural rules that removes the need for manual reviews and thereby contributes to increasing the quality of the code. This tool was integrated into the company's development toolchain and, thus enabled a **change in practice (O3)**. We see in our case study that the **research relevance (F1)** and high **practitioner involvement (F3)** developed a favourable environment that stimulated communication and contributed to con-

crete gains and values including industrially-relevant new **tools and source code (O4)** and **changes in practice (O3)** $\{(\text{F1, F3}) \xrightarrow{C} (\text{O3, O4})\}$.

4.2 Results from the survey

To validate the results from the case study, we conducted a survey with within our collaboration network, and thus with a broader sample of participants than for the case study. Note that due to the limited survey format, we could only validate the facilitators and outcomes, not the complete relational graph emerging from the rich qualitative data collected in the retrospective meeting.

The results of the survey with respect to facilitators are shown in Figure 6. The figure indicates how many researchers and practitioners agreed with the marked item as a facilitator in the IA communication for each facilitator. In the survey, all the identified facilitators were confirmed by at least half of the respondents. On average, the researcher respondents marked 3.35 and the practitioners 3.14 facilitators. As discussed in Section 3.2, we do not interpret un-marked facilitators as generally invalid, but rather as being less valid to the respondents. On the other hand, the fact that many of the respondents confirm a certain facilitator is interpreted as an indicator that this factor is a valid facilitator also for a broader sample of IA project beyond the studied case.

The participants in the survey mentioned some additional facilitators. Researchers mentioned frequency of meetings, experience of the "other side", personal connections, and the attitude of the researcher (should be to transfer research, not collect empirical data). Sharing information with more frequency, and researcher's attitude were also mentioned by practitioners. They also mentioned the importance of an understanding of the basic and relevant needs of both sides.

The results of the survey concerning outcomes are shown in Figure 7. The bars marked 'Experience' show how many of the respondents, researchers and practitioners, recognise the marked item as an outcome of IA projects. The bars marked 'Importance' show how many of the respondents, researchers and practitioners, view the outcome as important to them when working in IA projects.

According to our survey participants, both researchers and practitioners, the most prevalent outcome of IA research is new knowledge (O1), as is shown by the responses both based on experience and with respect to the importance of the outcome. This aligns well with the case study findings, where four out of five facilitators promote new knowledge $\{\text{F1,F3–F5} \xrightarrow{C} \text{O1}\}$. Both groups had experienced awareness (O2) and good collaboration (O7) as outcomes promoted by IA communication. However, the practitioners found awareness (O2) be more important than the researchers did. In contrast, respondents of both groups responded that IA collaboration (O7) is less important.

Changes in teaching (O8) and new scientific venues (O9) are more of a concern for researchers, but interestingly enough only considered important by 10–15% of researchers. Likely, this outcome is not considered to be among the most important

outcomes, which does not mean that it is unimportant, as discussed in Section 3.2. The fact that researchers have experienced changes in practice (O3) as an outcome promoted by communication is to be expected, as the surveyed researchers are involved in IA research projects. However, the change in practice is considered important to a lower degree, and only one in four practitioners consider this to be an important outcome.

For Tools and source code (O4), social networks (O5), and new studies (O6), our respondents have experienced these as outcomes of IA communication to a higher degree than they consider them to be important outcomes. This applies to both researchers and practitioners. This is particularly worth noticing regarding new studies (O6), as the cases study findings indicate that all facilitators promote this outcome $\{$(F1–F5) $\xrightarrow{C}$ O6$\}$. Still, we interpret the responses to the question of importance as a ranking rather than an absolute assessment. Thus, new studies (O6) may be important, but, for example, our respondents view new knowledge (O1) as more important.

Some additional outcomes were mentioned by the participants in the survey. The participants in the survey mentioned some other outcomes. For participants from academia, the additional outcomes may be seen as related to awareness (O2), e.g., industrial trends, real-world problems, industrial challenges, vocabulary, and terms used in industry. Participants from industry, mentioned additional outcomes such as access to international experience, improved company-to-company cooperation through research projects, and recruitment, e.g. through contacts with students.

5 Discussion

We now discuss the results regarding the facilitators (RQ1) and outcomes (RQ2) of IA communication. For an overview of our results, see Table 1 and Table 2 and the observed indirect relations via communication in Figure 5.

5.1 Facilitators of IA Communication (RQ1)

Our study identifies five facilitators (F1–F5) that contributed to productive IA communication in the case project. These facilitators can be viewed as characteristics of the context where the communication occurs that contribute positively to the outcome of the project.

The relevance of the topics under study (F1) and the long-term horizon of the program (F5) facilitated IA communication within the project. The involvement of industry at the management level and in the research boosted the relevance of the research. From our perspective, the project benefited from previous joint work, due to that the people involved had already established good practices for IA communication within the long-term program before initiating the studied project. This

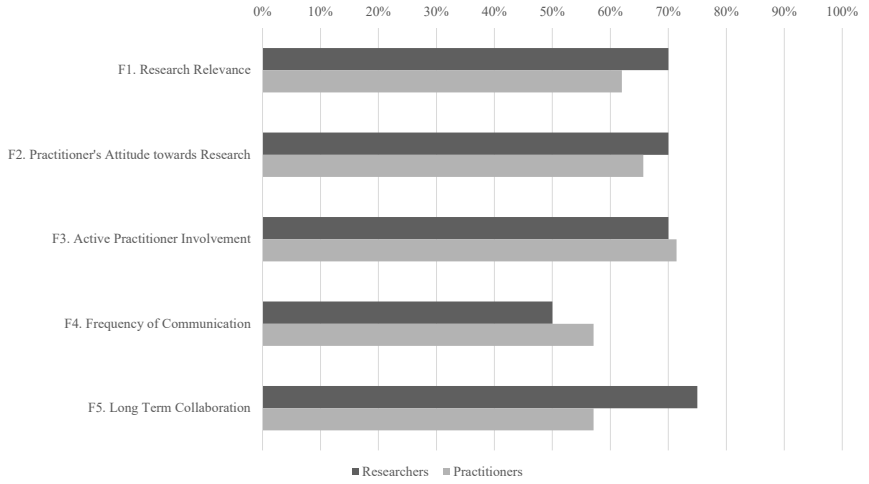


Figure 6: Survey responses to facilitators of IA communication. On average, researchers marked 3.35 facilitators and practitioners 3.14

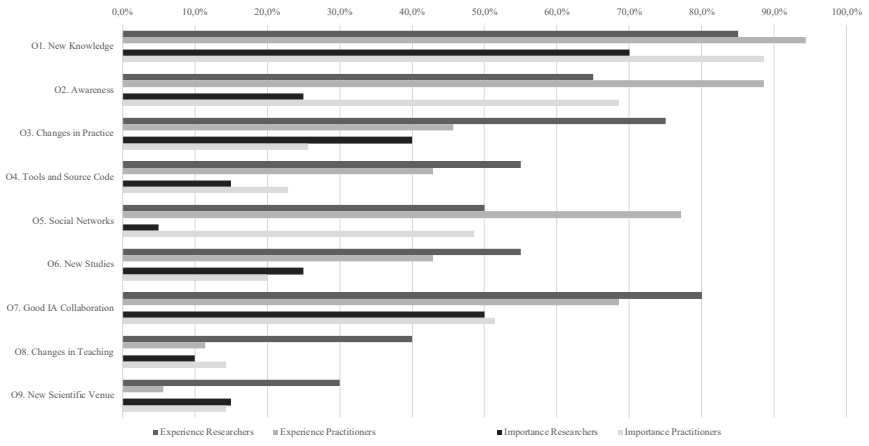


Figure 7: Survey responses to project outcomes promoted by IA communication. On average, researchers marked 5.35 outcomes as experienced and 4.77 outcomes as important. On average, practitioners marked 2.55 outcomes as experienced and 3.54 outcomes as important.

included the style and regularity of the meetings, the practice of ensuring research relevance of studies, and active practitioner involvement throughout each study. In the literature, the long term perspective in IA collaborations is connected to a stronger level of commitment [15]. In our case, the long-term nature of the IA research project provided the participants with the freedom to collaborate over a longer time period consisting of several years. Within the long-term agreements, the participants had the flexibility to define studies without any additional formalities.

One major challenge when working with practitioners is the "lack or drop of interest/commitment" [15]. We identified active involvement (F3) and the attitude on the practitioner side (F2) as a key facilitators of IA communication. We hypothesise that these two facilitators are due to two factors. Firstly, the relevance of the research performed motivates and stimulates practitioner involvement. Examples of this from our case project, are the impacts on practice observed in relation to the adoption of output from research and from M.Sc. projects. Secondly, the trust, respect, and mutual understanding of an existing project network facilitate communication between parties. In our case project, communication between industry and academic partners flowed naturally, and people knew whom to contact and how to work with their counterparts.

In our study, the frequency of communication (F4) was identified as one facilitating factor for IA communication. The frequency of communication was also identified as a facilitating factor for collaboration, for example by Rybnicek and Königsgruber [35]. Similarly, we found that active involvement by practitioners and practitioners' attitude towards research are critical to ensure the relevance of research results. These results are in line with the models proposed for joint research in software engineering [27, 29] that require a high degree of involvement from practitioners.

Finally, we identified the style of meetings as a facilitator of IA communication, and associate this to the long-term nature of the project (F5). Even if previous systematic literature reviews have not specifically identified this factor, Ankrah and Al-Tabbaa [2] conclude that "meetings and networking" are important for collaboration.

In the final project phase of our case project, the participants were familiar with each other and had an established way of working together. Each research study within the project shaped its own patterns and forms of communication; however, as new people joined the project and new studies were initiated the established ways of communicating were passed on, or inherited. We observed a well-divided hierarchy of meetings, and the group involved in each study had internal and informal discussions. In each meeting, it was clear what type of concerns were addressed, e.g. on the topic, on the study, or on the whole project. This allowed for focused discussions of each concern at the relevant level. To some extent, the facilitators that we have identified for IA communication correspond well to the facilitators identified in previous studies for collaboration in general.

5.2 Project Outcomes (RQ2)

We identify the project outcomes promoted by IA communication for academia and industry respectively. For the academics, working with industry can impact teaching (O8) and research (O6), and for practitioners, the impact can be seen in changes to practice (O3). For both parties, the communication promotes increased knowledge (O1). Given that researchers are often teaching university courses, the knowledge exchange with industry has an indirect effect on the students and, therefore, on future software engineers. Suppose the education of future practitioners receives the input from research conducted with the input from the industry. In that case, this enriches a critical mass of (new) professionals and entrepreneurs who could then quickly become involved in the industry or develop new business ideas.

An important benefit for researchers of IA projects is access (O5) to and insights into industry, which enables researchers to collect empirical data and validate research findings. Furthermore, the case project facilitated exchanges with researchers in general, both those directly involved in the project and others through personal contacts. These exchanges are valuable since they enable validating results and considering other viewpoints. Researchers and practitioners all benefited from these exchanges.

For practitioners, the outcomes of working with academia are both direct and indirect. Direct outcomes include changes in practice motivated by research findings (O3), and tools and source code originating from the research (O4) that can be used at the companies. Industry often view these contributions as the main gain and outcome of an IA project. These two outcomes were also in line with the overall goals of the EASE program, in particular the goal of results useful for industry, see Section 3.1. However, the survey results indicate that these outcomes are less valued than new knowledge in general.

We have identified an additional indirect outcome of the IA communication in the shape of increased awareness of research among practitioners (O2). Our analysis indicates that this awareness, in contrast to knowledge that has a direct industrial application, may impact practitioners in several ways e.g. inspiration for new products, bench-marking with other practices, and increased confidence gained from selecting practices based on research findings. Overall, both types of benefits need to be considered when evaluating the benefits of IA research projects, since the potential gains influence industrial partners' willingness to commit and actively participate in IA research projects. The possibility to reach these objectives is an important factor in facilitating industry participation.

We identify knowledge exchange between industry and academy (O1 and O2) as an outcome of the communication that occurs within individual studies and throughout the entire project. As is expected, new knowledge is built-in research studies, and communication contributes to achieving the goals of the studies. In addition, IA research projects can contribute to a positive cycle that leads to further

studies and mutual learning. Professional relationships are cultivated through IA project activities and exchanges during meetings. Many of these relationships go beyond the project lifetime and may lead to additional future IA interactions. In general, IA communication fosters more collaboration.

In the survey, many of the outcomes received high scores, both regarding the degree to which they have been experienced, and to what extent participants think they are important. However, there are some outcomes that did not receive high scores with respect to both aspects. Outcomes related to industrial practice (O3 and O4) were considered less important than outcomes related to knowledge and awareness (O1, O2). This difference in ranking indicates that, in general, there is more interest in outcomes related to knowledge than an immediate practical impact. Another possible view is that research results rarely are directly applicable to a specific industry setting but needs to be generally understood first and then adapted to the specific setting. Outcomes related to impact on research and teaching (O6, O8, O9) were, by a majority of the respondents, not marked as important, either by researchers or practitioners. As we see, outcomes that impact research and teaching may not be perceived by the participants as a priority or experienced to the same degree as other direct outcomes. However, these outcomes are indirect and visible in the long-term.

As described in Section 4.2, the survey participants mentioned some additional outcomes. However, many of these correspond to the need for knowledge and awareness (O1, O2). Participants mentioned, for example, the need for knowledge about industrial trends and real-world problems. In the same way, participants mentioned, for example, having access to international experience and recruitment of personnel as outcomes.

5.3 Validity of Contribution

Our main contribution is the identification of facilitators of IA communication in the context of an IA research project and outcomes promoted by such communication. We assess this contribution by discussing threats to validity and steps taken to mitigate these.

Construct Validity is about the concepts of the study, particularly IA communication and context. Our empirical data was collected from a retrospective meeting that had the goal of reflecting on the IA research project based on a timeline visualising projects events and outcomes. The objective of the retrospective was to investigate how industry and academia had worked together within the research program, not specifically focusing on communication in isolation. There is a risk that the retrospective did not focus enough on communication for this study. However, a large share of the timeline data was focused on communication, which is one reason why we selected to study IA communication for this case project and we found that the data was useful for studying communication due to the variety of communication instances found in the material. Furthermore, the survey helped us

to mitigate this risk by confirming the results with project participants and survey respondents.

Internal Validity relates to the suggested relationships between data entities, in this case, facilitators and outcomes. Propositions of these relationships are based on an aggregation of assumed connections between the entities in our coding scheme. These connections were identified in the data, and need to be further tested. There is a risk of researcher bias in the analysis that may affect the reliability of our results. We partly mitigated this risk by working in pairs and by systematically applying thematic coding. Our familiarity with the project is both a risk and a strength. The risk is that of confirming our prior beliefs without considering the data. This risk is partly mitigated by using a bottom-up approach in the coding (i.e. the facilitators were derived after the coding), and partly by asking other project members to read and comment on the results. This validation was performed by sending the manuscript to three practitioners and two senior researchers involved in the case project, four of which responded. Furthermore, the risk of misinterpretations is partly mitigated by the researchers being familiar with the case project, and knowing the people involved.

External Validity describes the generality of our results. We formulate our contribution to be applicable in any IA research project, and our findings can, thus, be tested also in other contexts. Our results are derived from observations in a single case study. The survey was an initial step towards external validity where additional people from industry and academia confirmed the identified facilitators and outcomes. Survey participants mentioned additional factors, e.g. mutual trust and understanding, style of communication, researchers attitude, and recruitment of graduating students. However, in general, the results of the survey, as described in Section 4.2, strengthen the generality of our findings. Future research may investigate these factors and further strengthen the generality of our results.

6 Conclusions

Communication plays a crucial role in any collaboration, so also in industry–academia (IA) research projects, both in facilitating the project as such and in creating a shared understanding of the goals and the outcomes of the project. In this study, we have analysed the communication within a 3-year IA research project, which in turn was part of a 10-year research program. The overall goals of the program were, from the industrial side, to increase the competence of personnel, and, from the research side, to perform research of high scientific quality that is relevant and useful to industry. Thus, knowledge sharing and knowledge co-creation were expected outcomes of bringing the researchers and practitioners together in various projects, both of which rely on communication between the parties.

We collected empirical data that was analysed according to a simplified model of communication (Figure 4) describing instances of communication where each

instance represents IA communication between two parties, within a context, and having explicit communication outcomes. Through analysis of such communication instances, we identified elements that facilitate communication between industrial and academic partners (RQ1) and examples of project outcomes that were promoted by IA communication (RQ2). These facilitators and outcomes, as reported in Section 5, provide empirically-based insights that may be used to guide the setup of similar joint projects and thereby improve IA communication. Furthermore, the extended of IA communication, including the observed contexts of communication, facilitators and project outcomes, may inspire future research on the characteristics and relationships between these proposed constructs of IA communication, which we find much needed.

In summary, the following recommendations may facilitate IA communication in joint research projects and subsequently stimulate the project outcomes identified as being promoted by IA communication:

Ensure that research is relevant to all participants by discussing and jointly agreeing to the scope of IA research programs, projects and studies. Practitioners will be more willing to engage in research activities, if the research topics and results are relevant and applicable to their work challenges. We noticed how addressing problems experienced by practitioners contributed to developing a favourable IA collaboration climate supported by communication that stimulated and led to changes in practice and new knowledge.

Foster a positive attitude towards research by listening to the needs and interests of industry, and aiming to provide value to practitioners through research. The view and attitude of practitioners towards research, researchers and research results influence their involvement in, and commitment to, research activities. We noticed that practitioners with trust in and previous positive experiences of collaborating with researchers had a positive attitude towards further such collaboration, which facilitated the communication with researchers.

Promote active practitioner involvement by openly discussing plans and emerging research results, and by inviting practitioners to take an active role, e.g. in reviewing papers and writing articles. An active engagement of practitioners in research projects contributes to identifying and addressing industrially-relevant problems in research studies. Furthermore, these engaged practitioners are critical in leading and promoting changes in practice based on research results. We noticed that the active involvement of practitioners was a critical factor that led to having discussions around industrially relevant topics with researchers. From these dialogues, new studies emerged around industrial challenges, and practitioners were made aware of research in the field.

Regularly hold both formal and informal meetings with a clear focus and adapted to the specific needs, e.g. of overall project synchronisation versus work meeting. IA communication and goal achievement are stimulated by a combination of formal meetings for project management, and open and informal meetings where creativity flourishes.

Establish a long-term collaboration between industry and academia through joint projects and networking events. A long-term collaboration contributes to creating social networks, identifying more research studies, and the possibility to apply results in the academic and industrial contexts. In addition, the long-term aspect of a collaboration allows researchers and practitioners to gain insight into each other's spheres and to develop good practices and ways of working together.

References

[1] Nauman Bin Ali, Emelie Engström, Masoumeh Taromirad, Mohammad Reza Mousavi, Nasir Mehmood Minhas, Daniel Helgesson, Sebastian Kunze, and Mahsa Varshosaz. On the search for industry-relevant regression testing research. *Empirical Software Engineering*, 24(4):2020–2055, 2019.

[2] Samuel Ankrah and Al-Tabbaa Omar. Universities–industry collaboration: A systematic review. *Scandinavian Journal of Management*, 31(3):387–408, 2015.

[3] Yasunori Baba, Masaru Yarime, and Naohiro Shichijo. Sources of success in advanced materials innovation: the role of" core researchers" in university–industry collaboration in japan. *International Journal of Innovation Management*, 14(02):201–219, 2010.

[4] Victor Basili, Lionel Briand, Domenico Bianculli, Shiva Nejati, Fabrizio Pastore, and Mehrdad Sabetzadeh. Software engineering research and industry: a symbiotic relationship to foster impact. *IEEE Software*, 35(5):44–49, 2018.

[5] Elizabeth Bjarnason, Anne Hess, Richard Berntsson Svensson, Björn Regnell, and Joerg Doerr. Reflecting on evidence-based timelines. *IEEE Software*, 31(4):37–43, 2014.

[6] Elizabeth Bjarnason, Kari Smolander, Emelie Engström, and Per Runeson. A theory of distances in software development. *Information and Software Technology*, 70:204–219, 2016.

[7] Lionel Briand, Domenico Bianculli, Shiva Nejati, Fabrizio Pastore, and Mehrdad Sabetzadeh. The case for context-driven software engineering research: Generalizability is overrated. *IEEE Software*, 34(5):72–75, 2017.

[8] Ana Isabel Canhoto, Sarah Quinton, Paul Jackson, and Sally Dibb. The co-production of value in digital, university–industry r&d collaborative projects. *Industrial Marketing Management*, 56:86–96, 2016.

[9] Jeffrey C Carver and Rafael Prikladnicki. Industry–academia collaboration in software engineering. *IEEE Software*, 35(5):120–124, 2018.

[10] Thomas Clauss and Tobias Kesting. How businesses should govern knowledge-intensive collaborations with universities: An empirical investigation of university professors. *Industrial Marketing Management*, 62:185–198, 2017.

[11] Alan Collier, Brendan J Gray, and Mark J Ahn. Enablers and barriers to university and high technology sme partnerships. *Small Enterprise Research*, 18(1):2–18, 2011.

[12] Emelie Engström, Mika Mäntylä, Per Runeson, and Markus Borg. Supporting regression test scoping with visual analytics. In Laurie Williams and Claes Wohlin, editors, *Seventh International Conference on Software Testing, Verification and Validation*, pages 283–292. IEEE Computer Society, 2014.

[13] Eduard Paul Enoiu and Adnan Causevic. Enablers and impediments for collaborative research in software testing: an empirical exploration. In Radu Dobrin, Peter Wallin, Ana C. R. Paiva, and Myra B. Cohen, editors, *WISE'14, Proceedings of the 2014 ACM International Workshop on Long-term Industrial Collaboration on Software Engineering, Vasteras, Sweden, September 16, 2014*, pages 49–54. ACM, ACM, 2014.

[14] Vahid Garousi, Markus Borg, and Markku Oivo. Practical relevance of software engineering research: synthesizing the community's voice. *Empirical Software Engineering*, 25:1687–1754, 2020.

[15] Vahid Garousi, Kai Petersen, and Baris Ozkan. Challenges and best practices in industry-academia collaborations in software engineering: A systematic literature review. *Information and Software Technology*, 79:106–127, 2016.

[16] Vahid Garousi, Dietmar Pfahl, João M Fernandes, Michael Felderer, Mika V Mäntylä, David Shepherd, Andrea Arcuri, Ahmet Coşkunçay, and Bedir Tekinerdogan. Characterizing industry-academia collaborations in software engineering: evidence from 101 projects. *Empirical Software Engineering*, 24:2540–2602, 2019.

[17] Aleksandra Gawel. Business collaboration with universities as an example of corporate social responsibility-a review of case study collaboration methods. *The Poznan University of Economics Review*, 14(1):20, 2014.

[18] Ahmad Nauman Ghazi, Kai Petersen, Elizabeth Bjarnason, and Per Runeson. Levels of exploration in exploratory testing: From freestyle to fully scripted. *IEEE Access*, 6:26416–26423, 2018.

[19] Matthew Good, Mirjam Knockaert, Birthe Soppe, and Mike Wright. The technology transfer ecosystem in academia. an organizational design perspective. *Technovation*, 82:35–50, 2019.

[20] Paul Grünbacher and Rick Rabiser. Success factors for empirical studies in industry-academia collaboration: a reflection. In Margaret M. Burnett, Holger Giese, Tien Nguyen, and Yuriy Brun, editors, *Proceedings of the 1st International Workshop on Conducting Empirical Studies in Industry, CESI 2013, San Francisco, California, USA, May 20, 2013*, pages 27–32. IEEE, IEEE Computer Society, 2013.

[21] Jianzhong Hong, Johanna Heikkinen, and Kirsimarja Blomqvist. Culture and knowledge co-creation in r&d collaboration between mncs and chinese universities. *Knowledge and Process Management*, 17(2):62–73, 2010.

[22] Leif Jonsson, Markus Borg, David Broman, Kristian Sandahl, Sigrid Eldh, and Per Runeson. Automated bug assignment: Ensemble-based machine learning in large scale industrial contexts. *Empirical Software Engineering*, 21(4):1579–1585, 2016.

[23] Hermann Kaindl, Sjaak Brinkkemper, Janis A. Bubenko Jr., Barbara Farbey, Sol J. Greenspan, Constance L. Heitmeyer, Julio Cesar Sampaio do Prado Leite, Nancy R. Mead, John Mylopoulos, and Jawed I. A. Siddiqi. Requirements engineering and technology transfer: Obstacles, incentives and improvement agenda. *Requirements Engingeering*, 7(3):113–123, 2002.

[24] Ali Kanso and Denis Monette. Foundations for long-term collaborative research. In Radu Dobrin, Peter Wallin, Ana C. R. Paiva, and Myra B. Cohen, editors, *WISE'14, Proceedings of the 2014 ACM International Workshop on Long-term Industrial Collaboration on Software Engineering, Vasteras, Sweden, September 16, 2014*, pages 43–48. ACM, ACM, 2014.

[25] Claire Le Goues, Ciera Jaspan, Ipek Ozkaya, Mary Shaw, and Kathryn T Stolee. Bridging the gap: From research to practical advice. *IEEE Software*, 35(5):50–57, 2018.

[26] Kyoung-Joo Lee. From interpersonal networks to inter-organizational alliances for university–industry collaborations in japan: the case of the tokyo institute of technology. *R&D Management*, 41(2):190–201, 2011.

[27] Dusica Marijan and Arnaud Gotlieb. Industry-academia research collaboration in software engineering: The certus model. *Information and Software Technology*, page 106473, 2021.

[28] Lars Mathiassen. Collaborative practice research. *Information Technology & People*, 15(4):321–345, 2002.

[29] Tommi Mikkonen, Casper Lassenius, Tomi Männistö, Markku Oivo, and Janne Järvinen. Continuous and collaborative technology transfer: Software engineering research with real-time industry impact. *Information and Software Technology*, 95:34–45, 2018.

[30] Ayse Tosun Misirli, Hakan Erdogmus, Natalia Juristo Juzgado, and Oscar Dieste. Topic selection in industry experiments. In Xavier Franch, Nazim H. Madhavji, P. C. Anitha, and Andriy V. Miranskyy, editors, *Proceedings of the 2nd International Workshop on Conducting Empirical Studies in Industry, CESI 2014, Hyderabad, India, June 2, 2014*, pages 25–30. ACM, ACM, 2014.

[31] Markus Perkmann and Kathryn Walsh. University–industry relationships and open innovation: Towards a research agenda. *International journal of management reviews*, 9(4):259–280, 2007.

[32] Per Runeson. It takes two to tango – an experience report on industry–academia collaboration. In Giuliano Antoniol, Antonia Bertolino, and Yvan Labiche, editors, *Fifth IEEE International Conference on Software Testing, Verification and Validation, ICST*, pages 872–877. IEEE, 2012.

[33] Per Runeson, Martin Höst, Austen Rainer, and Björn Regnell. *Case Study Research in Software Engineering – Guidelines and Examples*. Wiley, 2012.

[34] Per Runeson, Sten Minör, and Johan Svenér. Get the cogs in synch – time horizon aspects of industry–academia collaboration. In Radu Dobrin, Peter Wallin, Ana Cristina Ramada Paiva, and Myra B Cohen, editors, *International Workshop on Long-term Industrial Collaboration on Software Engineering (WISE)*. ACM, 2014.

[35] Robert Rybnicek and Roland Königsgruber. What makes industry–university collaboration succeed? a systematic review of the literature. *Journal of Business Econonomic*, 89(2):221–250, 2019.

[36] Ammon J Salter and Ben R Martin. The economic benefits of publicly funded basic research: a critical review. *Research policy*, 30(3):509–532, 2001.

[37] Anna Sandberg, Lars Pareto, and Thomas Arts. Agile collaborative research: Action principles for industry-academia collaboration. *Software, IEEE*, 28(4):74–83, july-aug. 2011.

[38] Claes Wohlin, Aybüke Aurum, Lefteris Angelis, Laura Phillips, Yvonne Dittrich, Tony Gorschek, Håkan Grahn, Kennet Henningsson, Simon Kågström, Graham Low, Per Rovegard, Piotr Tomaszewski, Christine Van Toorn, and Jeff Winter. The success factors powering industry-academia collaboration. *IEEE Software*, 29(2):67–73, 2012.

PAPER II

CHALLENGES AND STRATEGIES IN INDUSTRY COLLABORATION FOR PH.D. STUDENTS IN SOFTWARE ENGINEERING

Sergio Rico, Martin Höst, Emelie Engström and Nauman bin Ali. Under review in a conference, 2023.

Abstract

Background and Context: Ph.D. projects in software engineering in Sweden often involve collaboration with industry partners. Harmonious collaboration is critical to the success of these projects. While there is existing literature about challenges in industry-academia collaboration, there is limited information about the specific challenges faced by Ph.D. students and their strategies to address them.

Objective: We aim to explore Ph.D. students' challenges when collaborating with practitioners in research projects and identify the strategies they use to address these challenges.

Methodology: This study focuses on Ph.D. projects in software engineering from two Swedish universities. Data was collected by surveying the universities' current and recently graduated Ph.D. students.

Results: Findings reveal that researchers face various challenges related to communication distances, people, and research work. We identified strategies to address these challenges, such as synchronization sessions, establishing a common ground, and providing short-term benefits to industry partners while working towards long-term goals.

Conclusion: The insights gathered in this research can help improve collaboration in software engineering Ph.D. projects. By understanding and addressing these challenges using the suggested strategies, both parties can enhance collaboration, leading to more successful outcomes and stronger partnerships.

1 Introduction

Software engineering knowledge is context-bound, meaning context plays a crucial role in shaping the research investigations and the knowledge created in the process [3, 4, 9, 13]. Thus, the collaboration between industry and academia is essential for software engineering research and practice.

Working together makes researchers and practitioners aware of each other's field, which is mutually beneficial. It allows researchers to gain access to real-world problems and data [1] while providing practitioners with access to the latest research and tools.

The exchange of knowledge and expertise between researchers and practitioners leads to the development of new tools, processes, and methodologies that can potentially improve practice [6]. Moreover, industry collaboration provides the natural settings necessary for validating proposed solutions.

Collaboration with industry partners is also essential for software engineering Ph.D. students' training and research. Successful collaboration between industry and academia in the context of Ph.D. students' work will ensure the training of the next generation of researchers who are well-versed in collaborating with industry and developing solutions that positively impact software engineering practice.

However, to the best of our knowledge, the challenges faced by Ph.D. students when collaborating with industry and strategies to overcome these challenges have not yet been thoroughly investigated. Nonetheless, there are some studies in this area. Wohlin and Regnell [14] highlighted the importance of industry-academia collaboration in Ph.D. students' education and identified several strategies for managing industrial relevance and maintaining close contact with industry partners. Recently, Song and Runeson [12] shared their experiences as a Ph.D. student and supervisor working with industry, analyzing different collaboration scenarios and providing insights and recommendations to facilitate future collaborations. In this study, we provide a more comprehensive investigation of challenges and mitigation strategies for successful collaboration between Ph.D. students and industry partners.

Our study focuses on Ph.D. student projects in Sweden. By better understanding the challenges and strategies to overcome them, we aim to help Ph.D. students, supervisors, and practitioners improve collaboration in these projects. The research questions are:

RQ1: What challenges do Ph.D. students face when collaborating on software engineering research projects?

RQ2: What strategies do Ph.D. students use to overcome collaboration challenges?

Our study's contributions include the following: (1) a list of challenges, (2) an analysis of the most impactful challenges, and (3) suggestions for strategies and practical recommendations to overcome these challenges.

The remainder of the paper is structured as follows. Section 2 describes the methodology used in this study, while Section 3 presents the study's findings. Section 4, we discuss the results and offer recommendations for Ph.D. students, researchers, and practitioners. Finally, Section 5 concludes the paper and suggests avenues for future research.

2 Methodology

We carried out this research in two steps. First, we compiled a list of challenges based on existing models and known challenges in industry-academia collaboration. Second, we surveyed a sample of Ph.D. projects to quantify the prevalence and impact of these challenges.

2.1 Identification of challenges

To compile the list of challenges, we initially reviewed four papers [5, 6, 8, 10] on industry-academia collaboration in software engineering research. We primarily relied on the systematic literature review by Garousi et al. [5], as it comprehensively reviews challenges and best practices in industry-academia collaboration in software engineering research. The remaining papers served as supplementary sources to identify additional challenges not covered in the literature review.

2.2 Survey of challenges

Population

We focused on Ph.D. student projects in software engineering in Sweden. These projects are typically fully funded, and students are employed either by the university (academic Ph.D. student) or companies (industrial Ph.D. student). Even in industrial Ph.D. student projects, at least one supervisor is formally affiliated with a university. During their Ph.D. studies, most Ph.D. students work with one or more companies. For industrial Ph.D. students, their employing company influences their research topic. Along with the Ph.D. studies, where the students are expected to conduct and publish research on a specific topic, the students also learn how to collaborate with industry partners. Thus we consider the Ph.D. student projects a good population to study the challenges of industry-academia collaboration in software engineering research.

Specifically, we selected the Ph.D. projects from the software engineering research groups at Lund University (LU) and Blekinge Institute of Technology

(BTH), two Swedish universities. These two groups are appropriate for this study because of the number of people engaging in software engineering research, research output and recognition, diversity of research topics, and industrial engagement. Our choice can be considered convenience sampling, as the authors of this paper are affiliated with these universities, which facilitated access to the research groups and participants.

Data Collection

At the time of the survey, there were 5 Ph.D. students at LU (including the first author of this paper) and 13 at BTH in the respective software engineering groups. To complement the sample, we additionally invited more experienced recent Ph.D. holders. In total, we invited 19 individuals to participate in the study, of which 12 accepted and completed the survey.

We collected data through an online[1] questionnaire, based on the list of compiled challenges. The survey was open for nine weeks, during which we sent reminders to participants to encourage them to complete the survey.

The survey consisted of two parts: the first part gathered general information about the participants, and the second part collected data on the challenges they faced and the strategies they used to address them.

We presented the 58 identified challenges, organized into 13 categories. For each challenge, participants were asked to indicate: (1) if they had experienced it, (2) if they had observed the situation we described as a challenge, but they did not consider it a challenge, and (3) for the challenge where a respondent agrees that it is indeed a challenge, we ask them to rate the impact of the challenge on collaboration on a three-point scale (low, medium, or high impact). Participants were also asked to share open-ended, free-text responses describing their strategies for addressing the challenges within each category.

Data analysis and ranking

We ranked the challenges based on their prevalence and impact, i.e., the number of participants who consider a given challenge to have either a medium or high impact on their collaborations. The rationale behind this approach was to identify challenges that significantly affect the respondents' experiences and distinguish them from those that respondents did not consider challenging.

To analyze open-ended questions, we used qualitative content analysis [11]. We coded, and grouped the strategies suggested by respondents to address the challenges. We began by reading through the responses to gain a comprehensive understanding of the data. Then, we formulated a strategy code (ST#) for each response describing the strategy used to address the challenge and mapped the

[1]We distributed the survey online using an online survey tool, Sunet Survey, available for researchers at Lund University.

strategies with the related challenge. Even though the questions were organized by challenge, this step was necessary to consolidate similar strategies and clarify any ambiguous strategies. The first author completed the coding and analysis, which the third author then reviewed.

2.3 Threats to validity

External validity is about the ability to generalize the findings of the study. As we aimed to identify challenges faced by Ph.D. students, the study's target population and sample surveyed comprised Ph.D. students. Even though the students get support from their supervisors and other colleagues, they still answer as rather junior researchers, which should be considered when interpreting the results.

Our primary focus on two research groups in software engineering may also lead to a sampling bias, as the participants from these groups could experience a subset of challenges compared to those from other institutions.

The small sample size and response rate might affect the generalizability of the results and introduce potential biases. Future research should address these limitations by expanding the sample to include diverse research groups and institutions, refining the list of challenges iteratively, and employing alternative data collection methods to increase the representativeness.

Concerning the criterion validity (e.g., [7]), the completeness of the list of challenges may be limited, as some challenges could have been missed or underrepresented. However, we based the list on the available literature, including a comprehensive systematic literature review [5] (for details see Section 2.1).

3 Results

In this section, we present the results organized into four subsections. Section 5.3describes the identified challenges.Section 3.2 provides an overview of the participants and their survey responses. Section 3.3, presents the most impactful challenges experienced by the participants. Finally, Section 3.4 presents the mitigation strategies suggested by the respondents.

3.1 Identified challenges

Describing the challenges began with mapping them to the model of communication distances between researchers and practitioners by Bjarnason et al. [2]. However, we soon realized that since we were also interested in more general aspects of collaboration, as explored by, for example, Garousi et al. [5] and Marijan and Sen [8], some of the challenges could not be attributed to communication distances. Instead, they were related to the people involved or the nature of the research performed. Thus we created three overarching groups of challenges:G1

– Communication distances, G2 – People involved in collaboration, and G3 – Research work. Within these three groups (G1 – G3), we defined 13 categories CH1 – CH13 (see Table 1) to group similar challenges.

After establishing the groups and defining the initial categories, we categorized challenges from the literature. We first categorized 48 challenges from Garousi et al. [5] and complemented them with additional challenges from Marijan and Sen [8] focusing on technology readiness (e.g., Ch 13.4 and Ch 13.5) and release cycles (e.g., Ch 2.3).

It is important to note that there is no one-to-one mapping between the challenges from the initial papers [5, 8] and the ones in this study, as some aspects are more detailed (e.g., motivation) in this study since we are focusing on Ph.D. projects. Similarly, some challenges were not included in the survey as they did not apply to Ph.D. projects, such as financial investment in academia or competition between internal and external researchers.

After creating an initial questionnaire with 53 challenges, we reviewed the list. We added five more challenges that considered relevant aspects, including language differences (Ch 11.2), ways of doing work (Ch 9.4), having managers on board (Ch 4.7), and time distance (Ch 2.1, 2.2, and 2.3). Detailed descriptions of the groups and categories are presented below.

G1: Categories related to communication distances

This group of categories focuses on challenges related to communication distances. This encompasses various factors that affect communication, such as geographical distance, time constraints, cultural values, and organizational factors [2].

Geographical distance CH1 can hinder face-to-face communication and relationship-building. The physical separation between researchers and practitioners and the perceived difficulties of moving between locations can create barriers affecting the project. Temporal distance CH2, involves differences in time available for communication due to varying time zones, working hours, and times of the year when participants are available. These factors can make synchronizing communication more difficult.

Socio-cultural distance CH3 refers to differences between researchers and practitioners in values, social norms, and culture. These disparities can make it challenging to establish effective communication and understanding between the parties.

Organizational distance CH4 arises from different structures and rules for researchers and practitioners. Challenges like divergent interests and goals, personnel changes or reassignments, and lack of resources allocated for research may create obstacles in establishing successful collaboration. Discrepancies in time horizons CH5 further complicate collaborative work. Researchers often focus on long-term objectives, while practitioners may prioritize short-term goals. Aligning

Code	Category	Description
G1: Communication Distances		
CH1	Geographical Distance	Geographical distance involves physical separation and the perceived challenges of traveling between locations.
CH2	Time Constraints	Differences in the time available for communication, including different time zones and working hours.
CH3	Values and Culture	Distances in values, social norms, and culture due to their different backgrounds, roles, and working environments.
CH4	Organizational Factors	Related to the organizations having different interests, goals, and policies affecting collaboration.
CH5	Different Time Horizons	Differences in time horizons of researchers and practitioners.
CH6	Communication Tools	Communication tools to support communication including access, availability, quality, and willingness to use the tools.
G2: People involved in the collaboration		
CH7	Availability and Accessibility	Availability and accessibility include identifying and contacting the right person in the organization.
CH8	Motivation and Willingness	Motivation or willingness to collaborate
CH9	Knowledge and Skills	Differences in knowledge and skills affecting communication and collaboration.
CH10	Beliefs and Expectations	Related to the pre-existing beliefs and expectations on how to work together, affecting communication.
G3: Research Work		
CH11	Terminology and Language	Differences in the terminology and language causing potential misunderstandings.
CH12	Research Relevance	Relevance of the research to the practitioners' needs and interests.
CH13	Maturity of Research	Maturity of research outcomes and practitioners' needs and expectations, including difficulties implementing and scaling outcomes

Table 1: Overview of challenge categories in industry-academia collaboration

the organizations, goals, and objectives may be challenging due to these differing perspectives.

Lastly, differences in communication tools CH6 can also be a challenge, as researchers and practitioners may use different tools or have other preferences for these types of tools making it harder to communicate and collaborate.

G2: Categories related to people involved in collaboration

The second group of categories addresses challenges related to the people involved in the collaboration.

Accessibility and availability CH7 refer to difficulties in coordinating schedules and finding mutually convenient communication times. Overcoming these barriers is essential for fostering effective communication and relationship-building between researchers and practitioners.

Motivation CH8, plays a crucial role in aligning the goals and interests of researchers and practitioners. Misaligned objectives can hinder the achievement of desired outcomes, making it hard to realize the benefits of collaboration.

Ensuring that researchers and practitioners possess the necessary knowledge and skills CH9 is another critical aspect of collaboration. Gaps in knowledge or skills can impede progress, potentially leading to a loss of motivation, interest, and lower quality outcomes. Lastly, managing beliefs and expectations CH10 is vital when working together to avoid misunderstandings and misalignment of objectives.

G3: Categories related to research work

The third group of categories addresses challenges related to the research work, encompassing terminology, language, research relevance, and technology readiness.

Terminology and language CH11 can pose challenges in two ways. Firstly, researchers and practitioners may develop distinct concepts and terminology to describe their domains, making it difficult to understand each other. Secondly, finding a common language (e.g., Swedish or English) or the level of fluency in a language can present barriers to communication.

Ensuring research relevance CH12 is crucial for a collaborative project. Aligning research goals and objectives with the company's needs can be challenging, as each party may have different perspectives on what is relevant to the other.

Technology readiness CH13 refers to ensuring that the technology used in the collaboration is appropriate and capable of meeting the research needs. Furthermore, the maturity of the technology plays a role in determining its suitability for real-world applications. For example, research prototypes or lab-based experiments may not be mature enough for production use or may require significant adaption to be integrated into practical applications.

3.2 Survey participants and data overview

We received 12 responses from Ph.D. students and recent Ph.D. holders. Six respondents belonged to Blekinge Institute of Technology (BTH) and six to Lund University (LU), indicating a balanced representation. Among the respondents, three were industrial Ph.D. students, and nine were academic Ph.D. students. All the respondents had previous industrial experience (at least 1 year) before starting their Ph.D. studies. The respondents have been working on their Ph.D. projects for 1 to 5 years, with an average of about 2.5 years.

Table 2 presents an overview of the responses for each challenge with eight columns: CH (column 1) for the challenge category CH1–CH13, Ch.Sub (column 2), as the identifier for each unique challenge, Challenge description (column 3), and the number of respondents for different levels of a challenge's impact are in columns 4–8. Here NE (column 4) indicates if the respondents have not experienced the stated challenge in practice. Similarly, SE (column 5) is a situation the respondent has observed but does not consider a challenge. While LI (column 6), MI (column 7), and HI (column 8) represent low, medium, or high impact challenges, respectively.

The cells for columns 4–8 have a background color that indicates the relative number of responses for each level of impact. A darker color indicates more responses, while a lighter color indicates fewer responses. For instance, Ch 10.4 and Ch 10.5 have a dark background color for the NE column, indicating that most respondents have not experienced these challenges (10 and 11 respondents, respectively). In the challenge description, some challenges have been marked with arrows (up or down) based on the ranking approach described in Section 3.3.

3.3 Challenges ranking and impact

In Table 2, we marked the challenges that have more than four respondents experiencing them as High Impact (HI) or Medium Impact (MI) with an upward arrow. Similarly, challenges with no respondents experiencing them as high or medium impact were marked with a downward arrow before the text of the challenge.

Challenges (Ch5.1, Ch13.4, Ch8.3, Ch8.2, Ch4.2) are the top 5 challenges, which had 6 or 5 respondents experiencing them as medium or high impact, as ranked according to the approach outlined in Section 2.2.

3.4 Strategies used to address the challenges

Strategies for challenges related to communication distances

Researchers and practitioners can conduct online meetings and use company VPN to remotely access resources (ST1) to bridge the geographical distance CH1. This is even possible for activities like data collection with interviews. As physical

meetings are important, organizing workshops to connect researchers and practitioners after transitioning to remote collaboration (ST2) can be beneficial. Attending conferences with companies of interest present (ST4) can also foster connections between researchers and practitioners.

Addressing time constraints and scheduling CH2 involves researchers prioritizing proper research over speedy results, taking the necessary time to prepare (ST8), remaining adaptive and flexible (ST9), and inquiring about working schedules in advance to create an appropriate research plan (ST10).

To overcome challenges related to sociocultural distance CH3, researchers can present their work as a perspective to consider rather than an absolute truth (ST11). They can also meet up at conferences, give presentations, engage in hallway chat, and get to know the company representatives (ST12). Reaching out to practitioners, being persistent, using personal contacts, and personalizing communication (ST13) can also help. Furthermore, respondents suggested encouraging practitioners to reach out to academia for collaboration (ST14), and consider working with an alternative company if the initial one is not responding (ST15).

Regarding organizational distances CH4, promoting research and articulating collaboration benefits (ST16) can strengthen connections. Trust can be built by signing NDAs to maintain the confidentiality of data, tools, and results. Involving one potential champion at every step of the research project (ST18) and working together to build a long-term relationship with practitioners (ST19) can further cement trust. Connecting with individuals not companies, and limiting manager involvement (ST20) can foster more genuine connections.

Managing different time horizons CH5 requires researchers to quantify the potential benefit of a long-term project to make it tangible and attractive (ST21). Open communication and clarity on progress and expectations (ST23) can help align time horizons. Both researchers and practitioners face time constraints, so planning articles' publication schedules is essential (ST24). Providing an early report with initial analysis to the case company as feedback (ST25) and framing the question so that it has value "now" (ST26) can help bridge the gap between different time horizons.

For communication tools CH6, video conferencing tools like Zoom (ST27) can facilitate communication and collaboration between researchers and practitioners.

Strategies for challenges related to the people

Collaborating with industry partners requires effective communication and a certain level of motivation and willingness to exchange on both sides. To address accessibility and availability challenges CH7, researchers can seek help from practitioners (ST28) and identify a company champion to help drive the collaboration from their side (ST29). Regular follow-ups to build relationships (ST30) and using personal contacts while being assertive when necessary to reach the right people (ST31) can help to maintain commitment from both parties.

To address challenges related to motivation and willingness to exchange CH8, researchers can be proactive, well-prepared for meetings, and minimize the workload for the other party(ST38). Facilitating collaboration for practitioners and requesting meetings to develop commitment were also identified as effective strategies (ST39, ST40).

Addressing differences in knowledge and skills CH9 may require strategic approaches, like an "onboarding" phase with a new collaborator, "shadowing" stakeholders to learn about the practitioners' context (ST36), adapting reporting styles based on the audience, and asking practitioners to teach and explain relevant knowledge (ST37, ST38).

Challenges related to beliefs and expectations challenges CH10 can be addressed by clearly defining the expectations, goals, and, timeline-plans for the research, and communicating them with industry partners (ST39). Using technical terminology, providing concise reports to the industry, explaining project timelines, and adjusting them to meet industry partners' deadlines were also identified as effective strategies (ST40-42).

Strategies for challenges related to the research work

To overcome terminology and language barriers CH11, holding synchronization sessions for researchers and practitioners to discuss and harmonize concepts is beneficial (ST44). Preparing terms and concepts and presenting them in early meetings helps establish a common ground for collaboration (ST43).

Collaborating with industry partners can also present challenges related to the research work. Ensuring research relevance to practitioners CH12 can be addressed by proposing highly relevant topics, identifying common interests at the beginning of the collaboration (ST45), and framing research outcomes in the language of the industry by quantifying them in terms of market share and revenue (ST46).

Finally, the challenge of research maturity CH13 arises when scaling research results to production, especially with cutting-edge technologies. Addressing this challenge involves employing software engineers from the company to develop tools for scaling research results to production (ST47). Another strategy is to adapt to the challenge and give short-term benefits to the industry while collecting data and making inferences for the long term (ST48).

4 Discussion

4.1 Strategies used to address the challenges

In this subsection, we discuss the suggested strategies by the Ph.D. students organized according to the phases of collaboration outlined by Garousi et al. [6]. These phases include inception, planning, operation, and transition or diffusion of results. Besides the phases, we also include communication and relationship-building.

The **inception phase** is crucial for establishing trust and initiating partnerships. Strategies promoting early engagement, such as workshops (ST2), conferences (ST4), research promotion(ST16), and benefit articulation, can help lay the foundation for successful collaborations. Adopting a flexible approach when presenting research and addressing immediate value questions can further engage practitioners.

During the **planning phase**, clearly defining expectations (ST23), goals, and timelines (ST39) can help align the interests of both parties, setting the stage for a fruitful partnership. Researchers can ensure more efficient and effective planning by being aware of costs, schedules, and the practicalities of collaboration (ST10).

As the collaboration progresses into the **operation phase**, managing the research process becomes increasingly important. Key strategies include conducting digital data collection (ST3), champion involvement(ST18), and being assertive. Additionally, considering practitioner workloads (ST10) and actively seeking their expertise (ST38) can help bridge knowledge gaps and create a more inclusive research environment.

Effective communication of research outcomes is vital in the transition or diffusion of results phase. Providing timely feedback (ST25), tailoring reporting methods to suit the audience (ST37), and presenting results in a user-friendly format (ST40) can enhance the value and impact of the research. In addition, by framing research outcomes in technical language (ST40) and quantifying them in concrete metrics(ST46), researchers can make their findings more appealing and relatable to practitioners.

Communication and relationship building are essential components of the collaboration process. Implementing strategies such as online meetings, remote access, video conferencing tools (ST3), and flexible scheduling (ST9) can be beneficial to cultivate a more collaborative environment. Furthermore, by tailoring communication styles to fit practitioner preferences (ST37) and maintaining open communication channels (ST23), the overall collaboration experience can be improved.

4.2 Recommendations for Ph.D. students, supervisors, and practitioners

For **Ph.D. students**, effective collaboration with practitioners begins with polishing communication and networking skills. This includes attending conferences, workshops, and industry events to create connections and foster relationships. Understanding technical language and practitioners' expectations is crucial to better-communicating research outcomes in a manner that resonates with practitioners. Being proactive and well-prepared can make collaborations more fruitful. Students should regularly follow up with practitioners and facilitate their involvement by minimizing the work required on the industry partners' end. Lastly, students

should strive to understand the industry context, learn about the sector where their research is being applied, and identify potential areas of mutual interest.

Supervisors support collaboration by facilitating networking opportunities, guiding communication, and navigating the differences between academic and industrial contexts. They can also create a supportive environment by fostering open dialogue between students and practitioners, helping to establish clear expectations and plans for collaborative research. Furthermore, supervisors can mentor students on relationship building, emphasizing the importance of trust and strong connections in collaborative research. Ultimately, this will contribute to forming a generation of well-trained researchers to collaborate with practitioners and drive impactful research that benefits both academia and industry.

Practitioners can actively support collaborative research by being open to collaboration, engaging with researchers, attending academic conferences, and reaching out to academia for potential partnerships. Their expertise and industry-specific knowledge can be invaluable to researchers, and their willingness to participate in discussions can bridge knowledge gaps. Practitioners should also communicate their expectations clearly, providing feedback on research goals, timelines, and deliverables to help researchers align their work with industry needs. Having a practitioner interested in the project, i.e., a "champion" to drive the research project, facilitate communication, and coordinate with academic partners, can significantly enhance the likelihood of success of the collaboration. Lastly, practitioners can support long-term relationships by fostering an environment that encourages ongoing collaboration and recognizes the value of both short-term and long-term research outcomes.

4.3 About the most and least experienced challenges

Most respondents faced issues related to differences in timeframes (CH5) and expectations between researchers and practitioners (CH10). Specifically, the challenge of researchers and practitioners having different timeframes, where practitioners expect the results quickly (Ch5.1), was highly impactful. This difference in timeframes and expectations can lead to misalignments in research objectives and hinder effective collaboration. Additionally, the challenge of researchers struggling to move from research prototypes to production-ready solutions (Ch13.4) was identified as another significant issue, indicating the difficulty in translating academic research into industry practices and tangible outcomes.

These challenges emphasize the importance of fostering a common understanding between researchers and practitioners to bridge the gap between their goals, expectations, and working methods. While research outcomes are often expected to be general, abstract, and theoretical, practitioners seek actionable results immediately applicable to their business context.

By aligning research objectives with industry needs and focusing on producing actionable results, the collaboration between academia and industry can be

strengthened, leading to more fruitful partnerships. Strategies such as establishing synchronization sessions, providing short-term benefits to industry partners, and actively engaging in communication to address differences in motivation, knowledge, and expectations should be considered.

In addition to these strategies, building mutual understanding and appreciation for each party's value to the projects is crucial. This involves researchers showing how theoretical contributions can be translated into practical applications and practitioners highlighting research benefits for their organizations. Ensuring that both parties commit to the collaboration is vital to overcome these barriers and achieving successful outcomes in the Ph.D. projects.

While our study identified particular challenges as less experienced by the respondents, they may still be significant in other contexts. For instance, challenges such as practitioners not valuing qualitative research (Ch10.5) and academic research being perceived as a waste of time since it does not apply to business (Ch10.4) were ranked lower in the list. These challenges may be context-dependent and could arise when there is a lack of understanding or appreciation for the value of academic research within the industry. These challenges probably received low ranks as many Ph.D. projects are developed in settings where the industry is involved. The impact of these challenges may vary depending on the specific collaboration, the industry sector, and the organization's culture.

It is important to note that some challenges, although less experienced overall, were marked by one respondent as having medium or high impact. This suggests that the severity of these challenges might be underestimated based on the survey results alone. In such cases, it is difficult to draw definitive conclusions about the extent of these challenges without further investigation. In addition, it is essential to consider the possibility that these challenges have a higher impact in different collaboration contexts or when working with diverse industry partners. Further research is necessary to explore these challenges in different settings and the strategies that can be employed to address them effectively.

4.4 Impact of new challenges

The survey conducted in this study aimed to assess the experiences of Ph.D. students concerning challenges previously identified in the literature [5], as well as some new challenges not covered in Garousi et al.'s study. Among these newly identified challenges, several were perceived to have a medium to high impact on collaborations.

For instance, one respondent considered the geographical distance between researchers and practitioners (Ch1.1) highly impactful. Time availability-related challenges, such as different time zones (Ch2.1) and varying release cycles (Ch2.3), were reported to have a medium impact by one and three respondents, respectively.

Another new challenge, getting managers on board (Ch4.7), was reported to have a medium impact by two respondents. This finding underscores the need for

researchers to demonstrate the value and relevance of their research to industry partners and secure managerial support for successful collaborations. However, some new challenges did not receive votes for medium or high impact, such as researchers not having a positive attitude toward industry partners (Ch 3.4) or language barriers between researchers and practitioners(Ch 11.2). Interestingly, additional challenges not included in the initial list by Garousi et al. [5], about technology readiness and the maturity of research results for industry use (Ch 13.4, 13.5) [8], were marked as highly impactful.

5 Conclusion

In this study, we have explored what challenges are perceived by Ph.D. students when collaborating with industry partners. We have also identified strategies that can be used to address these challenges. We identified various challenges related to communication distances, people, and research work. By systematically categorizing these challenges and analyzing the strategies suggested by the respondents, we provide a comprehensive framework for understanding and addressing the obstacles researchers and practitioners face when collaborating.

Our findings reveal that effective collaboration in software engineering Ph.D. projects requires overcoming numerous barriers, including geographical, sociocultural, organizational, and time constraints, and addressing differences in motivation, knowledge, and expectations among collaborators. By implementing the strategies identified in this study, researchers and practitioners can enhance collaboration, leading to more successful outcomes and stronger industry-academia partnerships.

Future research can investigate the effectiveness of the identified strategies in different contexts and explore additional factors influencing industry-academia collaboration in software engineering Ph.D. projects. Additionally, developing guidelines and best practices based on this study can support researchers and practitioners in navigating and overcoming collaboration challenges.

In conclusion, this study contributes to understanding industry-academia collaboration in software engineering Ph.D. projects and provides a valuable resource for researchers and industry practitioners. Furthermore, stakeholders can enhance collaborative efforts and foster stronger, more fruitful partnerships in software engineering by addressing the challenges and employing the suggested strategies.

Table 2: Number of responses for each challenge. NE Not Experienced; SE Experienced but not a challenge; LI Challenge with low impact; MI Challenge with medium impact; HI Challenge with high impact.

CH	Ch.Sub	CHALLENGES	NE	SE	LI	MI	HI
CH1	Ch1.1	Geographical distance between researchers and practitioners.	4	5	2	0	1
CH2	Ch 2.1	Researchers and practitioners are located in different time zones	7	4	0	1	0
	Ch 2.2	Practitioners are not available at the same time as researchers need them e.g. evenings and weekends	5	4	2	1	0
	Ch 2.3	Researchers and practitioners having different release cycles	2	4	3	3	0
CH3	Ch 3.1	Lack of prior relationships between researchers and practitioners makes communication difficult	3	4	2	2	1
	Ch 3.2	Practitioners are not open to discussing strengths and weaknesses of their practices	5	2	3	1	1
	Ch 3.3	Practitioners not having a positive attitude towards research/academia	5	2	2	2	1
	Ch 3.4	↓Researchers not having a positive attitude towards industry/companies	8	2	2	0	0
CH4	Ch 4.1	↑Different interests and goals among the organizations	5	2	1	3	1
	Ch 4.2	↑Change or turn over of practitioners and contact persons in the organizations	4	2	1	3	2
	Ch 4.3	↑Organizations not having resources to invest in research	6	1	1	3	1
	Ch 4.4	Organizations do not want to discuss what they consider to work well	8	2	0	1	1
	Ch 4.5	Organizations do not want to discuss what they consider to be their competitive advantage	9	2	0	0	1
	Ch 4.6	↓Difficult to handle communication with practitioners from multiple organizations	8	2	2	0	0
	Ch 4.7	Hard to get managers on board to support research projects	6	1	3	2	0
	Ch 4.8	Researchers do not have access to data due to intellectual property restrictions	7	1	1	2	1
	Ch 4.9	↑Researchers do not have access to companies' infrastructure due to technical restrictions	5	3	0	2	2
CH5	Ch 5.1	↑Researchers think and work in long-term and practitioners in short-term	3	2	1	4	2
	Ch 5.2	↑Hard to get managers to invest in long-term research projects	6	0	2	3	1
	Ch 5.3	↑Researchers need to work quickly to meet the needs of practitioners	4	2	2	3	1
CH6	Ch 6.1	↓Researchers and practitioners use different tools, e.g., slack vs. email	3	8	1	0	0
	Ch 6.2	↓Researchers do not have access to practitioners' tools, e.g., ms teams, slack, internal wiki	7	4	1	0	0
	Ch 6.3	↓Issues with digital tools, e.g., slow internet connection, unstable connection, lack of audio/video,	8	3	1	0	0
CH7	Ch 7.1	Researchers face problems contacting the right practitioners in the organization	2	3	4	1	2
	Ch 7.2	Practitioners are not available to talk to researchers	4	4	1	1	2
	Ch 7.3	↑Practitioners take a long time to respond to researchers	3	2	3	2	2
	Ch 7.4	↑Communication is not frequent enough	4	4	0	4	0
CH8	Ch 8.1	Researchers have little interest in participating in collaborative research projects	9	1	0	2	0
	Ch 8.2	↑Practitioners have little interest in participating in collaborative research projects	4	1	2	3	2
	Ch 8.3	↑Problems to keep the motivation of practitioners in the research project	4	0	3	3	2
CH9	Ch 9.1	↑Researchers need time to learn the business context and tools used by practitioners	2	5	1	3	1
	Ch 9.2	Researchers do not know the company processes and development cycle	2	7	1	1	1
	Ch 9.3	Researchers or practitioners are not updated on the latest technology innovations	5	4	2	1	0
	Ch 9.4	↓Researchers and practitioners have different ways of doing tasks, e.g., modeling	5	3	3	0	0
	Ch 9.5	Practitioners do not have time to teach researchers about their practices	5	5	1	1	0
	Ch 9.6	Practitioners lack training in software engineering theory	5	5	0	1	1
	Ch 9.7	Practitioners lack skills to work with research results	3	5	1	1	2
CH10	Ch 10.1	Researchers overestimate practitioners' research participation.	4	4	3	0	1
	Ch 10.2	Practitioners expect researchers to know about everything and be able to solve all problems	8	1	1	2	0
	Ch 10.3	↑Practitioners expect research to be conducted in the same way as companies work e.g agile	6	1	1	4	0
	Ch 10.4	↓Practitioners see academic research as a waste of time since it does not apply to business	10	2	0	0	0
	Ch 10.5	↓Practitioners do not value qualitative research	11	1	0	0	0
	Ch 10.6	↓Practitioners prefer white papers and blogs over research papers	7	4	1	0	0
	Ch 10.7	Researchers and practitioners having different views on research evidence	7	2	2	1	0
CH11	Ch 11.1	Researchers and practitioners have different concepts and terminologies	4	5	1	1	1
	Ch 11.2	↓Researchers and practitioners do not speak the same language e.g English vs Swedish	8	3	1	0	0
	Ch 11.3	↓Researchers and practitioners do not speak the same dominant language e.g English vs Swedish	5	6	1	0	0
CH12	Ch 12.1	↓Research is not relevant to practitioners	8	3	1	0	0
	Ch 12.2	Researchers struggle to understand the relevant problems for practitioners from a business perspective	8	1	1	1	1
	Ch 12.3	Selected topics for research are not relevant to practitioners	7	2	1	1	1
	Ch 12.4	Practitioners want solutions that could be easily adapted to their context	4	2	3	3	0
	Ch 12.5	Researchers do not consider actual needs of industry practice	8	2	0	2	0
	Ch 12.6	Tension between research quality and practical applicability	7	0	2	2	1
CH13	Ch 13.1	↑Research outcomes are hard to implement or scale in the organizations	3	3	2	1	3
	Ch 13.2	Research results are not exploitable by practitioners	5	1	3	2	1
	Ch 13.3	Research results are not well described, or documented making it hard to implement in a new context	8	2	1	0	1
	Ch 13.4	↑Researchers struggle to move from research prototypes to production-ready solutions	4	1	1	3	3
	Ch 13.5	Differences in developing and deploying software in academia and industry	5	2	2	3	0

References

[1] Victor Basili, Lionel Briand, Domenico Bianculli, Shiva Nejati, Fabrizio Pastore, and Mehrdad Sabetzadeh. Software engineering research and industry: a symbiotic relationship to foster impact. *IEEE Software*, 35(5), 2018.

[2] Elizabeth Bjarnason. Distances between requirements engineering and later software development activities: a systematic map. In *International Working Conference on Requirements Engineering: Foundation for Software Quality*, pages 292–307. Springer, 2013.

[3] Lionel Briand, Domenico Bianculli, Shiva Nejati, Fabrizio Pastore, and Mehrdad Sabetzadeh. The case for context-driven software engineering research: generalizability is overrated. *IEEE Software*, 34(5):72–75, 2017.

[4] Tore Dybå. Contextualizing empirical evidence. *IEEE Software*, 30(1):81–83, 2013.

[5] Vahid Garousi, Kai Petersen, and Baris Ozkan. Challenges and best practices in industry-academia collaborations in software engineering: A systematic literature review. *Information and Software Technology*, 79:106–127, 2016.

[6] Vahid Garousi, Dietmar Pfahl, João M Fernandes, Michael Felderer, Mika V Mäntylä, David Shepherd, Andrea Arcuri, Ahmet Coşkunçay, and Bedir Tekinerdogan. Characterizing industry-academia collaborations in software engineering: evidence from 101 projects. *Empirical Software Engineering*, 24(4):2540–2602, 2019.

[7] Barbara A. Kitchenham and Shari L. Pfleeger. *Personal Opinion Surveys*, pages 63–92. Springer, 2008.

[8] Dusica Marijan and Sagar Sen. Industry–academia research collaboration and knowledge co-creation: Patterns and anti-patterns. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 31(3):1–52, 2022.

[9] Kai Petersen and Claes Wohlin. Context in industrial software engineering research. In *Proceedings of the Third International Symposium on Empirical Software Engineering and Measurement, ESEM 2009, USA*, pages 401–404. IEEE Computer Society, 2009.

[10] Sergio Rico, Elizabeth Bjarnason, Emelie Engström, Martin Höst, and Per Runeson. A case study of industry–academia communication in a joint software engineering research project. *Journal of software: Evolution and Process*, 33(10):e2372, 2021.

[11] Colin Robson and Kieran McCartan. *Real World Research*. John Wiley & Sons, Nashville, TN, 4 edition, December 2015.

[12] Qunying Song and Per Runeson. Industry-academia collaboration for realism in software engineering research: Insights and recommendations. *Information and Software Technology*, 156:107135, 2023.

[13] Claes Wohlin. Software engineering research under the lamppost. In *ICSOFT 2013 - Proceedings of the 8th International Joint Conference on Software Technologies, Iceland,*, pages IS–11. SciTePress, 2013.

[14] Claes Wohlin and Björn Regnell. Strategies for industrial relevance in software engineering education. *Journal of Systems and Software*, 49(2-3):125–134, 1999.

PAPER III

A TAXONOMY FOR IMPROVING INDUSTRY-ACADEMIA COMMUNICATION IN IOT VULNERABILITY MANAGEMENT

Sergio Rico, Emelie Engström and Martin Höst, In proceedings of 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), 2019.

Abstract

Background: In software engineering, industry-academia is a symbiotic relationship. Researchers need to be aware of the industry to produce relevant research, while practitioners are educated in academia and could take advantage of empirical research. The SERP taxonomy architecture is designed to support communication between practitioners and researchers in software engineering. **Objective:** The purpose of this study is to analyze to what extent the SERP taxonomy architecture is useful for improving communication between researchers and practitioners in IoT vulnerability management. **Method:** We developed a SERP taxonomy for IoT vulnerability management, SERP-MENTION, in an incremental way. Along the development, we evaluated the developed taxonomy in a project of industry academia collaboration. **Results:** In addition to the taxonomy itself we elaborate on the taxonomy development process and the potential of SERP-MENTION to support communication between researchers and practitioners within the area. **Conclusions:** The SERP architecture can be used in a new field, it is perceived as

useful by potential users to better describe and communicate research outputs and practical challenges in software vulnerability management.

1 Introduction

Empirical software engineering is an applied research area. Funding agencies and industry expect that research in the area of software engineering should affect and improve practices in industry. This is for example manifested in the "Triple Helix" model (e.g. [10]), emphasizing the interplay between university, industry, and government with funding agencies in innovation systems. Software engineering research can result in innovations in terms of new products, as well as in improved processes and tools that can be used in practice. Thus a shared understanding about practical challenges and proposed solutions between industry and academia is expected.

Shared understanding requires good communication in both directions of research results and perceived needs. If academia fails to understand what the actual problems in industry are, there is a risk of conducting irrelevant research in isolation. If communication fails in the other direction and industry is unaware of research in academia, there is a risk that improvement opportunities are missed. Poor communication between industry and academia may also lead to that improvement proposals are not sufficiently evaluated, e.g., through evaluations of research findings in form of tools and processes, as it requires collaboration. A basic assumption of this paper is that the communication between industry and academia can be improved, and that researchers and industrial organizations can benefit from the improvement.

Communication can take different forms. It can be direct communication, e.g. through meetings in joint research or discussions at conferences. It can also be indirect communication e.g. through published academic papers and technical reports. Regardless how communication is carried out, it is a problem if the communicating parts do not view the problem from the same abstraction level, not use the same terminology, or even understand each others' terminology. The construction and usage of a taxonomy can improve the communication by providing a common terminology and understanding of the domain and by catalyzing preciseness and completeness of problem descriptions. It can also support software process improvement when it comes to identifying relevant research results. Especially, in a SERP taxonomy [19] the scope of the classified research results are described in terms of which parts of the process they cover. In this paper we investigate if it is possible and meaningful to use a similar taxonomy approach to structure the area of security vulnerability management. We developed and evaluated a SERP taxonomy, SERP-MENTION (Software engineering research and practice in the management of vulnerabilities in the internet of things), in a joint research project between industry and academia. By developing the taxonomy we aimed to

study how researchers and practitioners perceived the use of this type of taxonomy to support the industry-academia communication, how SERP-MENTION can be used to describe challenges in the industry and solutions in academia, and finally the potential of the developed taxonomy to link the solutions and challenges. We report our experiences from applying the SERP approach as well as the resulting taxonomy.

The outline of this paper is as follows. In Section 2 background and related research is presented, and in Section 3 the research methodology is presented. The results from the execution of the research are presented in Section 5 and then analyzed and discussed in Section 7, before the main conclusions are summarized in Section 8.

2 Background and related research

2.1 The SERP approach

Many taxonomies have been developed to structure and understand the area of software engineering. Usman et al. [26] conducted a mapping study on taxonomy development in software engineering based on 270 primary studies. They conclude that there is a strong interest in creating software engineering taxonomies but few are extended and maintained. Bayona-Oré et al. reviewed literature on methods and guidelines for taxonomy development and propose a generic method for taxonomy development within software engineering [6]. Petersen and Engström proposed the SERP taxonomy architecture [19] for taxonomies, aiming at supporting the matching of software engineering challenges and solutions in context. Engström et al. further developed and validated a taxonomy, SERP-test, based on the SERP taxonomy architechture, using the guidelines proposed by Bayona-Oré et al. [6]. SERP-test has then been extended with details specific for regression testing to support the search for industry relevant regression testing evidence [2]. In this paper the *SERP approach* refers to both the taxonomy structure as proposed by Petersen and Engström [19] and the process of taxonomy development and evolution as proposed by Bayona-Oré et al. [6].

A SERP taxonomy covers three facets for describing practical challenges: 1) desired effect, 2) context factors, and 3) scope of change. Research solutions are described by these three facets and one additional facet, 4) intervention. For each facet a taxonomy of entities are built bottom-up within a community of practitioners and researchers having interest in the topic. Important steps in the taxonomy development are the definition of the scope and purpose, identification of important terms, increments of validations and updates against its purpose, and deployment in the community of users.

2.2 Managing vulnerabilities in IoT

Here, we use the SERP taxonomy structure to develop a taxonomy in the area of vulnerability management in Internet of Things (IoT). A vulnerability is an externally reported problem in software that should be considered to be removed, otherwise the security of the software can be decreased [18]. The NVD CVE (National Vulnerability Database, Common Vulnerabilities and Exposures) database has an increasing number of vulnerabilities listed [18]. In 2017 only there were more than 14,000 new vulnerabilities reported. A large share of the vulnerabilities that are reported in the CVE database describe vulnerabilities in Open Source Software (OSS) components. Since IoT products often are based on OSS, vulnerability management is important in IoT system development and management [17]. Management of vulnerabilities denotes the actions taken to identify vulnerabilities in code, evaluating their criticality, making changes, and deploying new versions in operational code [12]. Since it is often costly to deploy changes in operational products in IoT the ability to identify and analyse vulnerabilities in a reliable way is crucial, not the least because of the large number of published vulnerabilities.

2.3 Taxonomies in IoT security

SERP-MENTION was developed to support the communication between industry and research, the purpose differs from other taxonomies developed in the area of security for IOT, a field where vulnerability management is included. Dosemain et al. [8] proposed a taxonomy to define the connected objects to IoT, identifying energy,communication, functional attributes, local user interface and hardware, and software resources. The possible threats and attacks for IoT have also been addressed by researchers through taxonomies, Babar et al. [5] classified the possible threats by the use of IoT, while Nawir et al. [16] identified the network security attacks. Finally, Adat et al. [1] identified security challenges and provided a taxonomy of defense mechanism in IoT.

3 Research methodology

The methodology used in this study share similarities with action research in that we designed a solution to a problem in one specific case. However, the solution, SERP-MENTION, was developed and evaluated off-line, in parallel with the project under study, which was an ongoing industry-academia research collaboration project, and unlike action research we did not change the studied case context based on the findings.

3.1 Research questions

The overall goal of the study is to investigate the application of the SERP approach in a new area, i.e., IoT vulnerability management. Thus the contribution is twofold: 1) the resulting taxonomy (SERP-MENTION) as such, developed to support communication between researchers and practitioners in IoT vulnerability management and 2) a validation of the SERP-approach. The research questions are as follows:

RQ1 To what extent can the SERP-taxonomy architecture be reused to develop a taxonomy in the area of IoT vulnerability management?

RQ2 To what extent is SERP-MENTION useful for improving the communication about vulnerability management between researchers and practitioners?

RQ3 To what extent is SERP-MENTION useful for linking research outputs and practical challenges?

To answer the first question we apply the SERP-approach to develop SERP- MENTION and reflect on the procedure. The second question is answered by conducting interviews and a workshop, and by applying the taxonomy to a sample set of research papers. The third question is answered by mapping practical challenges identified in the workshop and literature with research results identified in the literature.

3.2 Project under study

Since we are investigating industry-academia communication, our case under study is a research project involving both practitioners and researchers. The goal of the studied research project was to develop support for working with vulnerabilities in industrial IoT software development and maintenance. The project was executed in a time period of about 3 years and consisted of partners both from the university, industrial organizations working with software development, and an institute taking a role resembling that of universities. In total two university research groups, one research institute, and six industrial organizations were involved. The project was funded by a national funding agency and the industrial organizations participated with in-kind funding. In the project, support was developed both in the form of software tools, and in the form of a process improvement model for working with questions related to vulnerability management.

3.3 Research process

We followed the steps in Figure 1. The first version of the taxonomy was developed starting with the SERP-taxonomy architecture [19]. We reviewed a set of

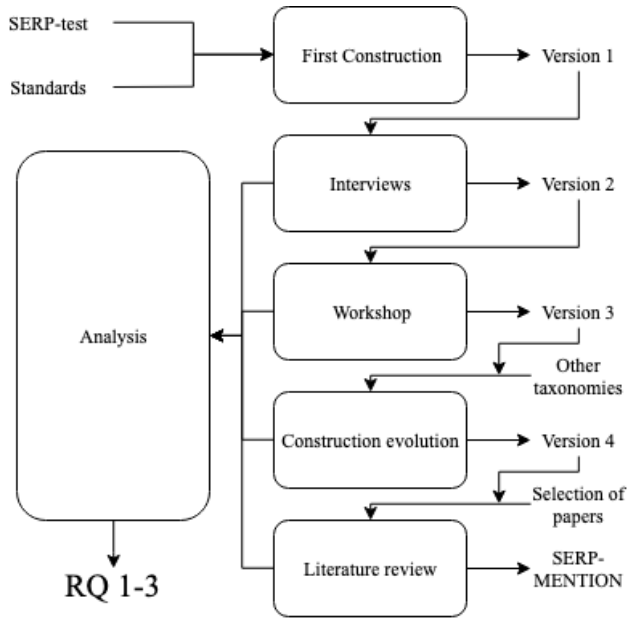


Figure 1: Research steps

standards, identified in a first literature search, and mapped the extracted requirements to the SERP-taxonomy architecture[1].

Two interviews were conducted with senior researchers who participated in the research project. The first interviewee is also the third author of this paper. The second senior researcher has more than 10 years of experience in cryptography and software security. The purpose of the interviews was to identify if the researchers recognized a communication gap in the project, to evaluate if the SERP approach seemed to be a way to bridge the gap and to evaluate the proposed taxonomy. The interviews were semi-structured with a set of questions formulated before the interviews. Questions covered the interviewees role and experience, foreseen challenges, comments about the current version of the taxonomy, expectations on the project, design decision during the project, and thoughts about possibilities and challenges when using the results of the project.

As the next step, a workshop was carried out with participants involved in the case research project. In total 9 people were involved, 4 persons from industry, 3 from academia, and 2 from research institutes. The workshop had two objectives, to analyze the usefulness of the taxonomy for describing challenges, and to identify entities that were lacking in the taxonomy. At the workshop we there-

[1]The interview and workshop protocol are available at `http://doi.org/10.5281/zenodo.3234676`

fore instructed the participants to carry out three activities. First they formulated improvement goals on the form "To achieve *effect* for *context* in *scope*" without using the taxonomy. Then they tried to rewrite them using entities provided in the taxonomy. This allowed us to compare the results. Finally, they were given the possibility to propose entities to the taxonomy that would have helped in specifying the challenges further. The feedback from the workshop was the main input for the third version of SERP-MENTION. After the workshop we incorporated entities from taxonomies identified in related work, see Section 2.3.

Finally, in the last phase the generality of the taxonomy was evaluated by using it to describe challenges and solutions derived from a sample set of academic papers. The papers were retrieved doing a one-level snowballing, taking as seed three prior papers produced in the research project (papers [7, 12, 17]). This resulted in the final version of the taxonomy. The set of challenges and solution derived from literature was then mapped to the challenges derived from the workshop based on the final version of SERP-MENTION to evaluate the ability of the taxonomy to link research outputs with real challenges, which is one of the purposes of a SERP taxonomy.

3.4 Limitations

In addition to constructing a SERP taxonomy for vulnerability management we have collected and analyzed data, from one industry-academia collaboration, about the approach as such, through interviews, a workshop and a literature review. Trustworthiness of this type of qualitative research can be assessed not only in terms of validity, but also in terms of generalizability and reliability [21].

Generalizability As this is a single study, we cannot draw any general conclusions about the SERP approach from this study alone. However, as a complement to previous and future studies on the SERP approach, it can provide support for its value. We support theoretical generalization by providing relevant details about the context and nature of project under study. The generality of the taxonomy as such, i.e., SERP-MENTION, was evaluated by applying it to a sample set of papers. Although this evaluation confirms that the taxonomy applies also to challenges and solution extracted in other contexts than our studied project, it does not confirm general completeness. This means that the structure of the classified entities may be reused as is, and in addition new entities may be added to the taxonomy as the identification of practical challenges and relevant research solutions emerges. To get a complete overview over the research on vulnerability management, a full systematic literature review is needed.

Validity To strengthen the validity of our conclusions we have applied a systematic procedure for collecting and analyzing the data, and we have been careful not to overgeneralize our findings. One threat to the validity is researcher bias, since

the second author of this paper had developed similar taxonomies before and the third author was involved in the studied industry-academia collaboration project, and was also one of the interviewees. This threat was mitigated by letting the first author lead the taxonomy construction process as well as the design of the interviews and the workshop and analysis of the data. All three authors were involved in validating the outcome of each step.

The conclusions drawn about the usefulness of the taxonomy is based on participants perceptions and indirect evidence regarding aspects of using the taxonomy that could be tested off-line, e.g. improved preciseness of challenge descriptions.

Reliability The reliability of the results refers to the consistency of interpretations of terms and concepts. This is strengthened by the fact that researchers were familiar with the tool (SERP) as well as the project under study. However, none of the researchers were experts on vulnerability management, which may have negative impact on the validity of the taxonomy. This is mitigated in accordance with the taxonomy development process [6] by involving domain experts in the development and evaluation of the taxonomy.

4 Results

4.1 Interview results

In the first interview, the researcher described the evolution in the project, starting from a potential need identified by researchers to the implementation of a tool that was used by industrial companies. The tool identifies and evaluates vulnerabilities in OSS components used in IoT systems. For the tool development, collaboration between researchers and practitioners was required. That is one reason why a common understanding about the objectives and the way of working of the tool was needed. Related to the communication gap, the researcher pointed out how the awareness, concerns, and challenges about security were different for each company, according to their size, culture, maturity, and type of product or service offered. However, the need to handle vulnerabilities was relevant for all the companies, which meant that communication was essential to understand needs and context in the project in order to develop a useful tool. The interviewee was asked to describe challenges related to IoT security, with and without the first version of SERP-MENTION. The preliminary result after the exercise was that using SERP-MENTION can improve the precision and clarity of the challenge descriptions.

The second interview followed the same questions and there were no disagreements, but some additional aspects were discussed. Given that the second interviewee is an expert in information security one purpose was to evaluate the scope and categories of the taxonomy. The main scope of IoT vulnerability management

was confirmed by the researcher as an interesting topic in academia, and also as relevant to companies according to the interviewee's previous experiences with industry.

The facets were analyzed and some changes were made. The effect facet was refined, deleting entities that were out of scope, too general or actually described activities instead of desired effects. The scope was limited to include only activities related to vulnerability management instead of the whole IoT product cycle.

4.2 Workshop results

Workshop participants described practical challenges related to IoT vulnerability management. In the analysis, challenges were classified into three groups A, B, and C, according to how well they followed the taxonomy after the second task, i.e., when they were asked to rewrite the challenges based on the taxonomy; A for correctly following the taxonomy, B for partial adherence, and C for those who did not follow it at all.

After the second task, it was clear that challenges in groups A and B were better than those in group C. A better description means that the desired effect, context and scope were more specific with less internal terminology. It was clear that the challenges described in group C were still too general. The challenges in group C also used words related to specific companies, which makes them more difficult for others to understand. Table 1 shows the A and B challenges described by the workshop participants.

Concerning the terminology used to describe the challenges, the terms from the effect facet were utilized, new terms suggested by the participants were added to the taxonomy in relation to efficiency and trust. From the context facet, just a few terms were used, while around half of the scope facet were used. Challenges from groups B and C mixed terms from the scope and the context, some participants described "the company" or "our project" to describe the scope. These inputs were taken into account for the new version of the taxonomy. For example, the entities of the scope facet were reduced to only cover the vulnerability management process and new entities were added to the other two facets.

For example, one challenge was formulated as "to achieve faster CVE evaluation during the software development for the project" before using the taxonomy, and as "to achieve quicker and more accurate vulnerabilities management during the product design for the organization" after using the taxonomy.

4.3 Literature review

To evaluate the generality of the taxonomy we applied it to a sample of relevant papers. As described in Section 3.3, the sample papers was derived as the relevant references of the papers produced in the case project. Examples of non-relevant

Table 1: Challenges described by the Workshop participants

Entry	Challenge Description
Ch1	To improve the communication with clients and partners.
Ch2	To improve the speed of decisions about vulnerabilities.
Ch3	To diagnose the importance of identified vulnerabilities.
Ch4	To automatically identify vulnerabilities.
Ch5	To improve the time to patch vulnerabilities.
Ch6	To improve the time to respond to vulnerabilities.
Ch7	To improve the efficiency in evaluating vulnerabilities .
Ch8	To improve the accuracy of the vulnerabilities evaluated.
Ch9	To define a process to handle vulnerabilities.
Ch10	To determine the need for urgency of the response.
Ch11	To more promptly address identified vulnerabilities.
Ch12	To achieve lower cost for identified vulnerabilities in the product life cycle.
Ch13	To achieve higher credibility for the company brand in the user community.

paper are research methodology papers and papers about cyber-security in general, e.g,. discussing specific vulnerabilities that have been found in products.

From the literature we derived an additional set of challenge-descriptions as well as a set of solution proposals. These are listed in Table 2. Table 3 shows the final taxonomy, grey marked, and the mapping of challenges and solutions derived from the workshop and the literature.

The purpose of this evaluation was to ensure that all entities and categories of the taxonomy had counterparts in real research outputs or challenge descriptions related to the project under study. For a taxonomy to be used it should be aligned with the terminology used by the intended users, in our case researchers and practitioners in IoT vulnerability management.

In this exercise we could see that all categories below level 3 for all facets but 'intervention' was useful and sufficient, as all categories were mapped to at least one of the challenges or solutions, and that no additional categories were needed at that level to classify the entries. However, the taxonomy at that stage also included entities at higher levels of detail that were not fully covered. Categories or entities that were not covered by any challenges or solutions extracted from literature or the workshop were removed from the taxonomy.

In summary, our literature review and mapping confirmed stability of parts of the taxonomy as shown in Table 3 and indicated a mismatch between the literature, standards and industrial needs regarding the details. This mismatch would require an extensive systematic literature review to be proven and understood, which is out of the scope for our study. The taxonomy proposed here may however guide such review.

Table 2: Practical challenges and research solutions derived from the selected papers in the literature review

Src	Entry	Entry Description
[13]	Ch14	To prevent the intruder's access to the objects that may cause physical damage or change their operation.
	Ch15	To assure security measures for the transmitted data from devices and prevent it from external interference or monitoring
	Ch16	To guarantee the data integrity at the information processing unit
[22]	Ch17	To attestate efficiently in a large dynamic and heterogeneous network.
[23]	Ch18	To evaluate identified vulnerabilities to identify relevance and impact.
[12]	Ch19	To identify relevant vulnerabilities among the huge amount of information about vulnerabilities.
	Ch20	To evaluate identified vulnerabilities to identify relevance and impact.
[4]	Ch21	Developers perceive system availability more important than confidentiality.
[24]	Sol1	To improve the Instruction Detection Systems, SecAMI calculates a relationship between attack spreads, detection, and consequences on the availability.
[13]	Sol2	To be able to identify potential vulnerabilities, in any company developing IoT systems with OSS, track versions of used OSS or COTS versions in the products.
	Sol3	To facilitate correctness in the evaluation of vulnerabilities in in any company developing IoT systems with OSS, track possible threats in software products.
	Sol4	To achieve faster and more robust management of vulnerabilities in any company developing IoT systems with OSS, have a well defined process for identifying and monitoring sources of vulnerabilities.
	Sol5	To achieve a more cost efficient remediation of vulnerabilities in any company developing IoT systems with OSS, evaluate severity and relevance of vulnerabilities and make decisions for handling and reacting to identified vulnerabilities.
	Sol6	To allow a more robust and transparent vulnerability process in any company developing IoT systems with OSS, communicate vulnerability and security information, internally and externally in a structured way.
	Sol7	To improve transparency, effectiveness and awareness of the vulnerability management process in any company developing IoT systems with OSS, use HAVOSS.
[3]	Sol8	To increase the vendor's patch release speed, disclose vulnerability information.
[14]	Sol9	To diagnose the importance of vulnerabilities, evaluate with respect to the CVSS score.
[11]	Sol10	To identify vulnerabilities automatically, apply fuzzing and penetration testing.
	Sol11	To detect overflows, follow a combination of automatic approaches.
	Sol12	To improve effectiveness of vulnerability, use code review.
	Sol13	To respond quickly, vulnerabilities should be reported to companies.
	Sol14	To improve reputation, companies should respond more quickly to reported vulnerabilities.
[7]	Sol15	To improve effectiveness and efficiency of the of vulnerability identification and assessment in any company developing IoT systems with OSS, use the tool for mapping vulnerabilities to code.

4.4 SERP-MENTION

In this subsection, we present SERP-MENTION. As described in Section 3, the taxonomy was developed incrementally. Here the fifth and latest version is presented. SERP-MENTION enables classification of research results and practical

Table 3: Mapping of challenges and solutions to SERP-MENTION

	Effect															Scope						Context										Intervention	SERP -MENTION
	Improve											Solve		Diagnose								People	IoT system						Business					
	Access control	Secure data communication	Resilience to attacks	Availability	Process transparency	Process effectiveness	Process efficiency	Vendor's patch release speed	Awareness of security	Trust	Define vulnerability process	Automatic identification	Importance of vulnerabilities	Identification	Communication	Patch management	Assessment	Configuration management	Design-Development	Development culture	Objects spread geographically	Large number of objects	Lightweight encryption	Heterogeneus networks	Solaris	Open soce software	Lack of resources	Open business environment	Vendors		
Sol1			✓											✓							✓									Use the risk assessment tool SecAMI	Mapping
Sol2											✓			✓												✓				Track versions of used OSS or COTS	
Sol3						✓											✓									✓				Track possible threats in software products	
Sol4						✓	✓							✓	✓		✓	✓								✓				Have process for identifying and monitoring sources of vulnerabilities	
Sol5							✓										✓									✓				Evaluate severity and relevance of vulnerabilities and make decisions for handling and reacting	
Sol6					✓	✓								✓		✓	✓									✓				Communicate vulnerability and security information, internally and externally	
Sol15					✓	✓			✓					✓	✓	✓	✓	✓	✓							✓				Use HAVOSS	
Sol8								✓								✓													✓	Disclose vulnerability information	
Sol9													✓				✓													Evauate with respect to the CVSS score	
Sol10												✓		✓																Apply fuzzing and penetration testing	
Sol11						✓								✓																Folllow a combination of automatic approaches.	
Sol12						✓								✓																Use code review.	
Sol13					✓										✓												✓			Vulnerabilities should be reported to companies.	
Sol14										✓						✓											✓			Respond more quickly to reported vulnerabilities	
Sol15						✓	✓							✓			✓									✓				Use the vulnerability tool by Cableigh et al. for mapping vulnerabilities to code.	
Ch1										✓						✓										✓						
Ch2							✓							✓	✓	✓	✓	✓	✓							✓						
Ch3													✓				✓									✓						
Ch4												✓		✓												✓						
Ch5							✓									✓										✓						
Ch6							✓							✓	✓	✓	✓	✓	✓							✓						
Ch7						✓											✓									✓						
Ch8						✓											✓									✓						
Ch9											✓															✓						
Ch10											✓						✓									✓						
Ch11							✓							✓	✓	✓	✓	✓	✓							✓						
Ch12							✓							✓												✓						
Ch13										✓																✓						
Ch14	✓															✓					✓											
Ch15		✓	✓													✓						✓										
Ch16	✓															✓							✓									
Ch17	✓																		✓					✓								
Ch18				✓															✓						✓							
Ch19				✓										✓													✓					
Ch20				✓													✓										✓					
Ch21									✓										✓	✓												

challenges in IoT vulnerability management. Each entry can be described and classified using the facet-based SERP architecture [19].

The main facets of the taxonomy are *intervention*, *effect*, *scope* and *context*. Each SERP facet is the root of a taxonomy of entities grouped in categories (or nodes). The first and second level of each such category are visible in Table 3. SERP *entries* refer to descriptions of practical challenges or research outputs on a format including *entities* from the SERP facets. Research results can for example be expressed like:

> To achieve *effect* during *scope* in *context* do *intervention*.

Challenges are expressed in a similar way but do not include an entity from the intervention facet. An example of a practical challenge is:

> It is a challenge to *improve the efficiency* of the *vulnerability evaluation* when *OSS is used in the IoT system*.

An example of a research result [25] is:

> To *improve the access control* during *the patch management*, when *having a large number of objects*, do *implement a security manager on top of the centralized IoT hub*.

Intervention

An intervention is an act performed, to diagnose, solve a problem or improve vulnerability management. The interventions listed in Table 3 were extracted from the research proposals in our literature review. We did not find it meaningful to categorize this list further. For SERP-MENTION version 1, we added categories based on requirements derived from the IoT security standards such as: "provide automated support" and "secure design". However, during the literature review and mapping, we found that this classification was not useful as it did not match the extracted interventions nor was it orthogonal.

Effect

An effect is a target, i.e. what is to be achieved by an intervention. Inspired by SERP-test [9], we identified three relevant types of effects: *improve, solve,* and *diagnose*, where improve refers to measurable improvements of the current state.Solve refers on the other hand to a request for solutions to unsolved problem, e.g. no current solution exist to compare with. Finally, diagnose refers to requests for support in assessing the current situation We identified 10 improvement goals, 2 unsolved problems and 1 diagnose target, listed in Table 3.

Scope

The scope entities in SERP-MENTION are activities of the vulnerability management process. For a solution, it refers to the activity where the intervention is applied, while for a challenge it refers to the activity for which the effect is desired. We identified 6 such activities, listed in Table 3. *Design and development* are activities carried out before the IoT system is deployed; *vulnerability identification*, *assessment* and *patch management* after the IoT system is deployed; and *communication* with customers, partners, etc. along the whole process.

Context

The context entities are factors that either motivate the need for an intervention or affect the applicability and effect an intervention, e.g, the use of open source code when IoT products are developed. The context factors extracted in this study are categorized to be either *people*-related, *business*-related, or *system*-related. People factors are related to humans like the culture in the company. Business factors are constraints given by the business environment or business decisions. System factors are related to the nature of the IoT systems. We identified one people-related factor, five system-related factors and four business-related factors, listed in Table 3.

5 Discussion

IoT is an emergent topic both in industry and academia. An indication of this is the IoT ecosystem fragmentation and a lack of standards [15]. When a new terminology is starting to be established, taxonomies are useful. They allow to reason about classes of problems instead of specific instances. They can also support communication by providing concepts and a technical language [20]. In this research this was seen, e.g., when participants listed challenges. Even though they had experience and knowledge about security they lacked a common terminology.

To cover the needed terminology, SERP-MENTION was developed with focus on vulnerability management. We considered both the technical, methodological and organizational dimensions of IoT vulnerability management. When reviewing existing taxonomies for IoT security [1, 5, 8, 16] (Section 2.2), we found that they were partly useful also for our purposes. and thus we decided use some of their categories to structure the facets in the SERP-MENTION. SERP-MENTION can be reused and adapted, adding more categories and entities to the facets, also parts of other taxonomies can be included. A key difference between SERP-MENTION and previous taxonomies is that SERP-MENTION is designed to support communication between researches and practitioners by providing a way to link challenges from industry to solutions in academia while the other taxonomies were focused in describing or gain understanding about specific IoT security topics.

SERP-MENTION was developed in the context of an industry-academia collaboration project. The need for this type of taxonomy was identified, at least by the researchers, in the project and it was developed in parallel with the project. The final version of the taxonomy as presented in this paper was completed in the end of the collaboration project and thus not explicitly used in the project. However, in retrospect the usage of this type of taxonomy in the project had it already been developed could probably have helped especially researchers to get a more complete understanding of important research questions. Furthermore, in presenting results, a taxonomy like this could probably be useful as a guide, not the least in communication in academic articles. It would probably also have given a richer and more consistent terminology in communication within the project.

SERP-test is another SERP taxonomy, also aimed to support the communication between researchers and practitioners but in the area of software testing [9]. Both similarities and differences are observed. In both cases the scope seems to be the facet with the highest agreement between how industrial need and research results are communicated. Similarly, in both cases the intervention facet remains unrefined as it is hard to find a general and orthogonal classification of interventions. This facet is not needed for matching purposes, but could be useful for comparing several solutions to the same challenges. An example of how this is done for a special case of testing, regression testing, is provided in a systematic literature by bin Ali et al. [2]. The first two levels of the 'effect target' is identical with the first two levels of SERP-test, but there is more variation in the details. The case is similar when it comes to context factors.

To develop taxonomies in software engineering Bayona-Oré et al. [6] have proposed a method and Usman et al. [26] reviewed that method suggesting some improvements. The method considers the phases of planning, identification, and extraction, design and construction, validation, and deployment. In the development of SERP-MENTION we followed the phases of the method: Planning is part of the research steps, the design and construction approach was incremental, where for each increment activities of identification and extraction were developed. The validation of the taxonomy was carried out in the literature review and the mapping of the entries to be classified.

A taxonomy can be developed top-down or bottom-up. While developing SERP-MENTION we combined the two approaches. The top-down approach was followed when we started from SERP architecture, reviewing standards, and reusing taxonomies. The bottom-up approach while adding entities that were actually used when describing challenges and solutions in the workshop and reviewed literature.

6 Conclusions

A contribution of this study is SERP-MENTION, see Sec 4.4, a taxonomy developed to support communication between industry and academia in IoT vulnera-

bility management by enabling holistic, precise and unified descriptions of practical challenges and research outputs. By developing SERP-MENTION we can reflect on the usefulness of the SERP architecture for this purpose (RQ1). SERP-MENTION shares the four main facets with the SERP architecture (intervention, effect, scope, and context). SERP architecture also allows integrating other taxonomies partially or completely to describe a specific facet.

A mapping between research and practice is useful in several phases of a research project: Initially, in a collaborative research project, SERP-MENTION can be used to support expressing the challenges (or research questions) in a precise and holistic way and to ensure that everyone involved have a shared understanding of the problem to solve. Further it may guide a search for relevant literature and when reporting results it ensures that this is done consistently with other practitioners and researchers in the community.

The participants in the project, both from industry and academia, were during the workshop able to describe challenges using the taxonomy, in a more precise way than without. This is a first indication of the usefulness of the taxonomy to improve the communication in the project (RQ2), although further research is needed.

During the literature review we mapped, using SERP-MENTION, the research results identified in the literature and the industrial challenges derived from the workshop. The mapping helped to validate the developed taxonomy. Furthermore, it shows the potential of SERP-MENTION to link research and practice or, in case such links are missing, to visualize a gap between research and practice (RQ3).

Finally, we share some reflections about the development method. An incremental method helped us to quickly incorporate feedback from the previous steps. A combination of approaches, top-down and bottom-up was useful to map and validate the taxonomy. Involving practitioners in the development process contributed to giving the taxonomy practical relevance.

SERP-MENTION is not complete but mirrors the main aspect of the research project under study and its related research literature. It may be used as is or extended in other projects with similar scope.

References

[1] Vipindev Adat and BB Gupta. Security in Internet of Things: issues, challenges, taxonomy, and architecture. *Telecommunication Systems*, 67(3):423–441, 2018.

[2] Nauman bin Ali, Emelie Engström, Masoumeh Taromirad, Mohammad Reza Mousavi, Nasir Mehmood Minhas, Daniel Helgesson, Sebastian Kunze, and Mahsa Varshosaz. On the search for industry-relevant regression testing research. *Empirical Software Engineering*, 2019.

[3] Ashish Arora, Ramayya Krishnan, Rahul Telang, and Yubao Yang. An empirical analysis of software vendors' patch release behavior: impact of vulnerability disclosure. *Information Systems Research*, 21(1):115–132, 2010.

[4] Mikael Asplund and Simin Nadjm-Tehrani. Attitudes and perceptions of iot security in critical societal services. *IEEE Access*, 4:2130–2138, 2016.

[5] Sachin Babar, Parikshit Mahalle, Antonietta Stango, Neeli Prasad, and Ramjee Prasad. Proposed security model and threat taxonomy for the internet of things (IoT). In *International Conference on Network Security and Applications*, pages 420–429, 2010.

[6] Sussy Bayona-Oré, Jose A Calvo-Manzano, Gonzalo Cuevas, and Tomas San-Feliu. Critical success factors taxonomy for software process deployment. *Software Quality Journal*, 22(1):21–48, 2014.

[7] Alexander Cobleigh, Martin Hell, Linus Karlsson, Oscar Reimer, Jonathan Sönnerup, and Daniel Wisenhoff. Identifying, prioritizing and evaluating vulnerabilities in third party code. In *Proceedings International Enterprise Distributed Object Computing Workshop (EDOCW)*, pages 208–211, 2018.

[8] Bruno Dorsemaine, Jean-Philippe Gaulier, Jean-Philippe Wary, Nizar Kheir, and Pascal Urien. Internet of things: a definition & taxonomy. In *Proceedings Next Generation Mobile Applications, Services and Technologies*, pages 72–77, 2015.

[9] Emelie Engström, Kai Petersen, Nauman bin Ali, and Elizabeth Bjarnason. Serp-test: a taxonomy for supporting industry–academia communication. *Software Quality Journal*, 25(4):1269–1305, 2017.

[10] Henry Etzkowitz and Loet Leydesdorff. The dynamics of innovation: from national systems and Mode 2 to a Triple Helix of university-industry-government relations. *Research Policy*, 29(2):109 – 123, 2000.

[11] Munawar Hafiz and Ming Fang. Game of detections: how are security vulnerabilities discovered in the wild? *Empirical Software Engineering*, 21(5):1920–1959, 2016.

[12] Martin Höst, Jonathan Sönnerup, Martin Hell, and Thomas Olsson. Industrial practices in security vulnerability management for iot systems–an interview study. In *Proceedings of Software Engineering Research and Practice (SERP)*, pages 61–67, 2018.

[13] Rafiullah Khan, Sarmad Ullah Khan, Rifaqat Zaheer, and Shahid Khan. Future internet: the Internet of Things architecture, possible applications and key challenges. In *Proceedings Frontiers of Information Technology (FIT)*, pages 257–260, 2012.

[14] Peter Mell, Karen Scarfone, and Sasha Romanosky. A complete guide to the common vulnerability scoring system version 2.0. In *FIRST-Forum of Incident Response and Security Teams*, volume 1, page 23, 2007.

[15] Shahid Mumtaz, Ahmed Alsohaily, Zhibo Pang, Ammar Rayes, Kim Fung Tsang, and Jonathan Rodriguez. Massive Internet of Things for industrial applications: Addressing wireless IIoT connectivity challenges and ecosystem fragmentation. *IEEE Industrial Electronics Magazine*, 11(1):28–33, 2017.

[16] Mukrimah Nawir, Amiza Amir, Naimah Yaakob, and Ong Bi Lynn. Internet of things (iot): Taxonomy of security attacks. In *Proceedings Electronic Design (ICED)*, pages 321–326, 2016.

[17] Pegah Nikbakht Bideh, Martin Höst, and Martin Hell. HAVOSS: A maturity model for handling vulnerabilities in third party oss components. In *Proceedings International Conference on on Product-Focused Software Process Improvement (PROFES)*, 2018.

[18] NIST. National vulnerability database. https://nvd.nist.gov/. (visited on: 2018-05-15).

[19] Kai Petersen and Emelie Engström. Finding relevant research solutions for practical problems: The SERP taxonomy architecture. In *Proceedings of International Workshop on Long-term Industrial Collaboration on Software Engineering*, WISE '14, pages 13–20, 2014.

[20] Paul Ralph. Toward methodological guidelines for process theories and taxonomies in software engineering. *IEEE Transactions on Software Engineering*, 2018.

[21] Colin Robson. *Real World Research, 2:nd ed.* Blackwell, 2002.

[22] Ahmad-Reza Sadeghi, Christian Wachsmann, and Michael Waidner. Security and privacy challenges in industrial internet of things. In *Proceedings Design Automation Conference (DAC)*, pages 1–6, 2015.

[23] Muhammad Shahzad, Muhammad Zubair Shafiq, and Alex X Liu. A large scale exploratory analysis of software vulnerability life cycles. In *Proceedings International Conference on Software Engineering (ICSE)*, pages 771–781, 2012.

[24] Tawfeeq Shawly, Jun Liu, Nathan Burow, Saurabh Bagchi, Robin Berthier, and Rakesh B Bobba. A risk assessment tool for advanced metering infrastructures. In *Proceedings of International Conference on Smart Grid Communications*, pages 989–994, 2014.

[25] Anna Kornfeld Simpson, Franziska Roesner, and Tadayoshi Kohno. Securing vulnerable home IoT devices with an in-hub security manager. In *Proceedings International Conference on Pervasive Computing and Communications Workshops*, pages 551–556, 2017.

[26] Muhammad Usman, Ricardo Britto, Jürgen Börstler, and Emilia Mendes. Taxonomies in software engineering: A systematic mapping study and a revised taxonomy development method. *Information and Software Technology*, 85:43–59, 2017.

PAPER IV

GUIDELINES FOR CONDUCTING INTERACTIVE RAPID REVIEWS IN SOFTWARE ENGINEERING – FROM A FOCUS ON TECHNOLOGY TRANSFER TO KNOWLEDGE EXCHANGE

Sergio Rico, Nauman Bin Ali, Emelie Engström and Martin Höst, Technical Report, 2020.

Abstract

Evidence-based software engineering (EBSE) aims to improve research utilization in practice. It relies on systematic methods (like systematic literature reviews, systematic mapping studies, and rapid reviews) to identify, appraise, and synthesize existing research findings to answer questions of interest. However, the lack of practitioners' involvement in the design, execution, and reporting of these methods indicates a lack of appreciation for knowledge exchange between researchers and practitioners. Within EBSE, the main reason for conducting these systematic studies is to answer the practitioner's questions and impact practice. However, in many cases, academics have undertaken these studies without any direct involvement of practitioners. This report focuses on the rapid review guidelines and presents practical advice on conducting these with practitioner involvement to facilitate knowl-

edge co-creation. Based on a literature review of rapid reviews and stakeholders engagement in medicine and our experience of using secondary studies in software engineering, we propose extensions to an existing proposal for rapid reviews in software engineering to increase researchers-practitioners knowledge exchange. We refer to the extended method as an interactive rapid review. An interactive rapid review is a streamlined approach to conduct agile literature reviews in close collaboration between researchers and practitioners in software engineering. This report describes the process and discusses possible usage scenarios and some reflections from the proposal's ongoing evaluation. The proposed guidelines will potentially boost knowledge co-creation through active researcher-practitioner interaction by streamlining practitioners' involvement and recognizing the need for an agile process.

1 Introduction

Software engineering research aims to establish software development practice on scientific foundations. This ambition requires that research is relevant and accessible for practice. Evidence-based software engineering (EBSE) is one such initiative to provide the best available evidence to support software development and maintenance. Often, a single empirical study provides insufficient confidence in the strength of evidence. There is a need to synthesize available research (where individual studies often have contradictory results) on a topic of interest. The EBSE) [31] approach has the following five steps: (1) convert a practical information need to an answerable question, (2) identify available evidence to help answer the question, (3) critically appraise the evidence, (4) make evidence-informed decisions, and (5) evaluate the effectiveness and efficiency of steps 1-4.

The EBSE community has developed several systematic secondary study methods for steps 2-3, including systematic literature reviews (SLRs) [32], systematic mapping studies (SMS) [43], and rapid reviews (RRs) [11]. Similarly, several authors have proposed solutions to facilitate step 4 in the EBSE process by introducing knowledge translation [7] or the technology transfer models [38].

Among the secondary study methods, mainly RR and SLRs are intended to support changes in practice. The SMSs only develop an overview of existing research on a topic. They are not intended to provide actionable insights for practice. SLRs risk being less attractive for practitioners because of the time frame needed to complete them. The time limitation of SLRs is overcome with the use of RRs. RRs are a variant of SLRs that simplify several steps of SLRs to provide information under time restrictions.

However, secondary studies are often conducted without any participation of practitioners. This lack of involvement can be partly explained by the implied objectivist view of knowledge [26] in the five-step EBSE process. In steps 2-3, knowledge is treated as objective, disembodied from the context, and codified,

which in step 4 is transferred or communicated to practice. We overcome this limitation by extending the guidelines for RRs guided by the following principles: 1) *Prioritize exchange* between researchers and practitioners. 2) The review is conducted to be *relevant for practitioners* according to their context. 3) A *close collaboration* is expected while doing the review.

This report presents an extension to the existing guidelines for designing and conducting RRs in SE [11]. It includes an emphasis on iterative and flexible design and ways to increase practitioner involvement in RRs, we refer to this extended version as interactive rapid review (IRR).

Like agile software development, IRR aims to bring the stakeholders (practitioners and researchers) of the product (in this case, literature syntheses) closer together with shorter lead times, increased communication, and flexibility in the process. The iterative and flexible design recognizes that the information need will be refined and may change during an IRR. Similarly, the interaction is critical to developing a deeper understanding of the context where practical information need is situated and to improve the relevance of the results.

The extension is based on a literature review from evidence-based medicine (EBM) where rapid reviews are extensively used [28,30,39,50]. We further supplement these with our own experience of having conducted several SLRs targeting industrial needs (e.g., [1, 3, 15, 16]) and several industry-academia collaboration projects.

We envision that conducting an IRR based on the proposed guidelines may foster knowledge co-creation, bringing several benefits. The IRR results tailored for the practitioners' needs, improve research utilization in practice. Besides, conducting the IRR favors mutual understanding between practitioners and academics that paves the way for further collaboration.

The remainder of the report is structured as follows: we describe the related work and our approach for developing the IRR guidelines in Section 2. In Section 3, we describe the complete proposed guidelines for interactive rapid reviews. We further discuss the use and implication of IRRs in Section 4 and conclude the report in Section 5.

2 Background

2.1 Secondary studies in software engineering

Researchers in software engineering have widely adopted the use of secondary studies as a means to synthesize software engineering knowledge [5]. A large number of SLRs and SMS have been published in software engineering. Also the process itself, to conduct these secondary studies, has been a research topic, and some researchers have proposed improvements to the methods and new strategies. Some examples are snowballing as a search strategy [57], reporting guidelines for

search [4], study selection procedures [2,42], use of machine learning for automation of search and selection [46], and studies about when to update SLRs [37].

Recently, Felizardo et al. [19] published a systematic mapping study and a survey on the value of using secondary studies in software engineering. They observed that secondary studies mainly have been used in academic environments, for teaching purposes and to identify gaps in research. The value of conducting the studies is described in terms of ability to develop research skills in students and junior researchers and to provide insights to plan future research. Little is mentioned about the interaction with practitioners while conducting the studies or about the impact of secondary studies in industry.

Some voices in the software engineering research community have claimed that secondary studies need to connect more with practice. Budgen et al. [6] suggested aspects to improve when reporting systematic reviews to make the results more meaningful for teachers and practitioners. Le Goues et al. [34] reflected on the advantage to connect research evidence with recommendations for practitioners.

2.2 Rapid reviews in software engineering

Rapid reviews were introduced in software engineering by Cartaxo et al. with the primary goal to transfer knowledge from academia to industry [8–10]. Like previously introduced EBSE methods the rapid review term originates from evidence-based medicine. Cartaxo et al. [11] describe rapid reviews as secondary studies that aim to "provide evidence to support decision-making towards the solution, or at least attenuation, of issues practitioners face in practice". The reviews may be seen as a variation of systematic literature reviews where some steps are omitted or simplified to reduce completion time. In medicine, there are variations of the method to conduct a rapid review, however, the approaches share the following common aspects:

- The review is conducted in collaboration with practitioners and refers to practical problems in their context.
- The review is conducted in a short time and at a low cost.
- The review's results are "reported through mediums appealing to practitioners."

Rapid reviews should not be misunderstood as ad-hoc literature reviews or lax reviews. Instead, rapid reviews are a systematic approach with a transparently documented process. Cartaxo et al. propose rapid reviews in software engineering to be lightweight secondary studies to deliver evidence to practitioners in a short time to support decision making [11].

Rapid Reviews have two characteristics that make them a good candidate for connecting research and practice. First, they are conducted in a short period of

time, which is probably appreciated by practitioners. Second, the studies are framed in the context of practitioners making the results relevant for them. This report elaborates on the researcher-practitioner interaction in such studies and describes the procedure for conducting interactive rapid reviews (IRRs).

2.3 Stakeholder engagement in secondary studies

In EBM, rapid reviews are used to support policy decision [30,40,41,53], support decision-making under tight schedule restrictions [25,44,49,52,53] and to a lesser extent to identify areas for further research [39]. Deverka et al. [14] investigated the engagement of stakeholders in secondary studies, and concluded that stakeholder engagement contributes to developing a shared understanding of the knowledge and increasing the outcomes' relevance. In their study stakeholder refers to any person or organization with a direct interest in the secondary studies' process or outcomes and stakeholder engagement as "an iterative process of actively soliciting the knowledge, experience, judgment, and values of individuals selected to represent a broad range of direct interests in a particular issue". In 2017, the world health organization (WHO) published a guide about rapid reviews to strengthen health policy [51]. The guide was compiled by researchers and provide practical advice regarding various aspects of rapid reviews. Among other things, the guide addresses how to engage policymakers and health system managers in conducting rapid reviews.

3 Interactive Rapid Reviews

In this section we describe the preliminary steps for conducting an IRR and propose ways for researchers and practitioners to interact throughout the process. We base the proposal on a literature review of the use of rapid reviews in EBM, including 48 meta-studies and reflections on the method. The presented procedure is aligned with the one proposed by Cartaxo et al. [11] and reflects our own experiences of conducting interactive literature reviews [1]. Fig. 1 shows the activity flow to conduct the review.

Our proposal for IRR consists of five steps that are described in more detail later in this Section. The first step is to prepare the IRR and identify information needs based on a practical problem. In the second step, the research questions are identified, and an initial version of the IRR protocol is developed. The protocol keeps track of decisions and activities throughout the IRR. The third step consists of searching and selecting papers to find a limited set of papers to answer the research questions. Decisions about terminology and relevance are validated with practitioners. Based on the selected set of papers, the IRR report and dissemination documents are co-designed and developed during the fourth step. Finally, in the fifth step, the results are disseminated among the practitioners. Notice in Fig. 1.

that the steps are conducted interactively with practitioners and that the general flow is iterative, where according to the feedback, the step outcomes are refined.

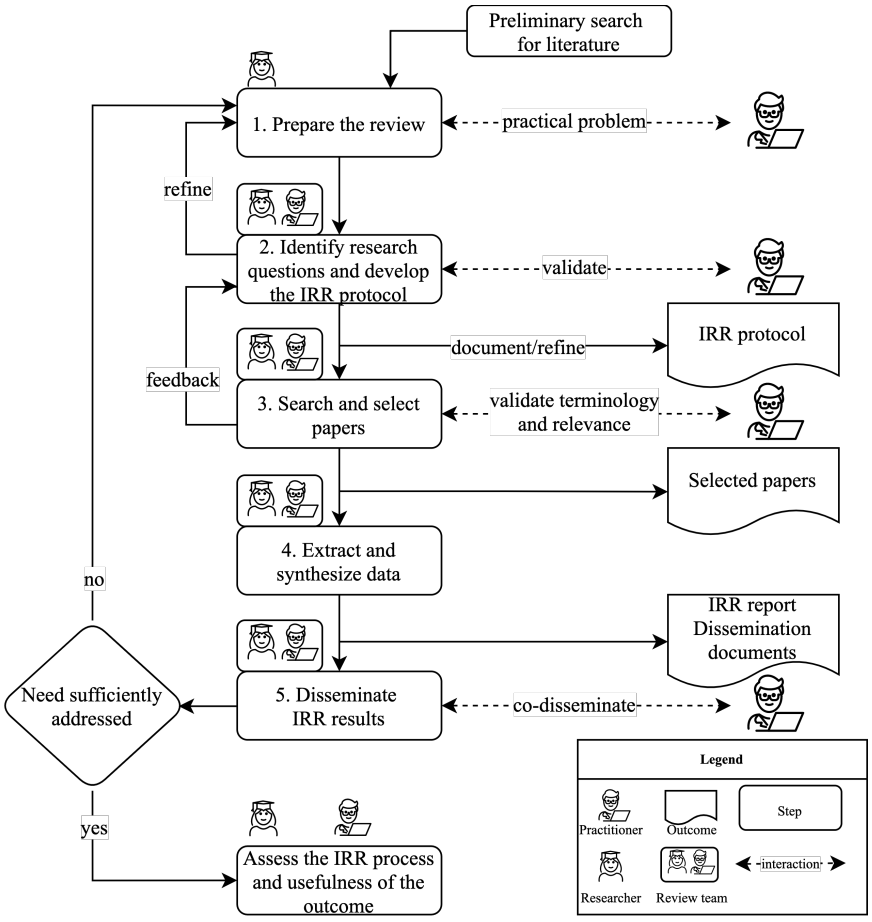


Figure 1: Workflow for performing an IRR

Table. 2. shows the central steps of an IRR in the first column (these have been adapted from Cartaxo et al. [11]). The second column highlights the contribution of our proposal with activities suggested to promote interaction with practitioners, and the third column lists the outcomes for each step. In the rest of this section we discuss each of these steps and possible interaction in more detail. Note that, when conducting an IRR the following general aspects should be considered:

- An IRR can be conducted in many scenarios throughout the researcher-

practitioner relationship. The main goal of this type of review is not to publish a research paper, but to align communication between stakeholders and gain relevant knowledge to solve a practical problem.

- An IRR is preferably lead by researchers as they have more experience dealing with the scientific literature. Practitioners provide insights to keep the IRR relevant for practice with a consideration of their context.
- Conducting an IRR is an agile process. Similar to agile software development, our proposal for IRR embraces the following principles: smooth communication between researchers and practitioners; meaningful results in context; joint work with practitioners; and response to change and flexibility.

3.1 Prepare the review

Fig. 2. shows the activities to prepare the review. In this step, the review team is formed, and information need is identified and described in context. The interaction between researchers and practitioners aims to get a commitment to performing the IRR and identifying a context-relevant problem for the IRR.

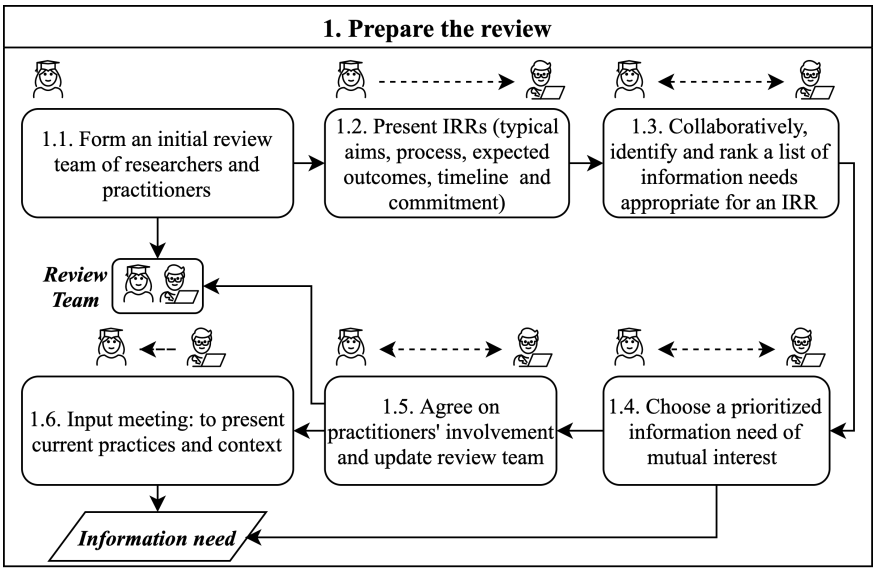


Figure 2: Prepare the review aims at get a shared understanding of what is an IRR, the expected outcomes, and to plan the work ahead

Step	Activity to promote interaction	Outcomes
1. Prepare the review	1.1. Form an initial review team of researchers and practitioners. 1.2. Present IRRs (typical aims, process, expected outcomes, timeline and commitment). 1.3. Collaboratively, identify and rank a list of information needs appropriate for an IRR. 1.4. Choose a prioritized information need of mutual interest. 1.5. Agree on practitioners' involvement and update the review team. 1.6. Input meeting: to present current practices and context.	Review team Description of information need Review topic
2. Identify research questions and develop the IRR protocol	2.1. Jointly, define the research questions. 2.2. Prepare and validate with practitioners the search strategy and inclusion/exclusion criteria.	IRR Protocol
3. Search and select papers	3.1. Perform the search. Present and validate the search results. 3.2. Apply inclusion/exclusion criteria 3.3. Update / extend the search	Papers to analyze
4. Extract and synthesize data	4.1. Co-design IRR reports and dissemination documents 4.2. Extract information and elaborate reports 4.3 Reaction meeting: present the initial results to the practitioners involved	Reports and dissemination documents
5. Disseminate IRR results	5.1. Identify the audience and medium of communication 5.2. Disseminate results to practitioners 5.3. Practitioners disseminate to other practitioners 5.4. Disseminate results to academic audiences	Reports and dissemination documents

Table 2: A list of activities proposed to increase the involvement of practitioners in rapid reviews (the steps in the first column are adapted from Cartaxo et al. [11]

Researchers lead the process to conduct the IRR. First, they form an initial review team based on the broad SE knowledge area (like software testing or re-

quirements engineering) and the practitioners' interests. Ideally, the review team should comprise at least two researchers, but it may be formed only by one researcher. Having at least two researches enriches the discussion and helps to improve the reliability of the study. It is even better if one of the researchers has experience conducting a systematic secondary study like SLR, SMS, or RR. During the review, the review team performs the search, selects papers, extracts, and synthesizes knowledge. Practitioners may or may not directly participate in these tasks depending on their degree of involvement. However, throughout the IRR, they are expected to, at the very least, have communication channels open with the review team to answer questions and provide feedback related to the relevance and context. Before starting with the review, researchers and practitioners need to clarify mutual expectations, agree non-disclosure agreements if applicable, and define roles and responsibilities [29].

In an initial presentation meeting, researchers introduce an overview of the IRR method, outcomes, roles, and responsibilities. This presentation helps to develop a shared understanding of expected outcomes and commitment. Before, the meeting, researchers do a preliminary search to get a sense of the literature in the field and support the dialogue with practitioners. Secondary studies are especially useful for this purpose [30, 36].

When practitioners have proposed the IRR topic concerning a practical problem, researchers and practitioners continue to identify context elements and research questions. Although, they have identified a practical problem they may need to specify the IRR scope further. To narrow the review topic, researchers may propose a shortlist of topics to the practitioners based on the results of the preliminary search and the practical problem [14]. With the list of topics, the practitioners rank the suggested topics according to their problem in context or suggest other directions. This exchange helps to agree on the IRR topic and contributes to making it interesting for both researchers and practitioners.

After the meeting, the review team may be updated with practitioners or new researchers. According to the practitioners' interest and familiarity with scientific literature, their participation may vary from being part of the review team to only provide feedback at specific points, e.g., clarifying terminology or the relevance of specific studies. The review team defines practical aspects like communication channels, file sharing, meetings calendar, and estimate the practitioners' time required to conduct the review, including both meetings and time required to answer questions.

Researchers need to get a good understanding of the practical problem and context variables. Researchers and practitioners may have an input meeting. During the input meeting, practitioners present the current practices in their context [14]. This meeting allows the review team to get a first approach to the research questions and keywords when preparing search queries.

At the end of this step, a team for the review has been formed. The team has an initial view of the problem in the context of practitioners. The review team has

a preliminary sense of research in the field and defined some practicalities like communication channels, meetings calendar, and follow-up meetings.

3.2 Develop the IRR protocol

For this step, we suggest two activities (see Fig. 2) related to define research questions with practitioners and prepare and validate the search strategy and inclusion/exclusion criteria.

The IRR protocol keeps track of the decisions and steps to conduct the review [21, 23]. During this step, the review team develops the protocol. However, this step may be revisited and the protocol updated in several iterations as new insights about both the context and the literature are gained [24, 35]. This favors the rigor of the study and the trust in the results. The protocol should contain at least [24]:

- Problem definition
- Research questions
- Search strategy
- Exclusion criteria
- Synthesis methods
- Initial proposals on how to disseminate the results

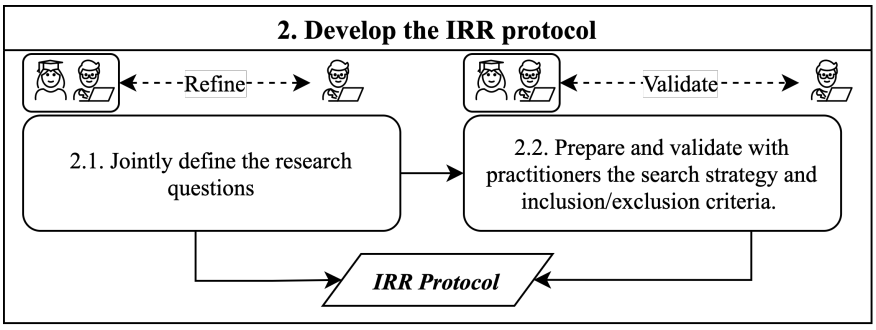


Figure 3: IRR follows a protocol that keeps track of decisions during the method

Research questions are crucial in the review because the search and knowledge synthesis is based on them. Practical questions are more suitable for this type of review, instead of general and broad questions [20]. Compared with SLRs, the

research question's scope is narrower as the questions in IRR address practical questions in a specific industry context [17, 53].

Researchers are used to working with research questions; thus, they may guide the formulation. They frame preliminary questions based on available literature and the practical problem. When defining research questions for IRR, it is essential to ensure alignment with practitioners' terminology. Questions are refined based on the exchange between the review team and practitioners to ensure that the final questions are relevant and include the particular practitioners' context [24, 36]. After a preliminary search, the review teams should evaluate if the research questions are suitable for an IRR according to the existent primary studies. If a preliminary search does not find related studies, it is probably unsuitable to continue with this approach.

The IRR protocol includes the search strategy and the inclusion/exclusion criteria. To define the search strategy, the review teams may consider insights from the preliminary search, the terminology extracted from the interaction with practitioners, and the identified context elements.

In an IRR, the review team uses shortcuts to reduce the number of sources to analyze and find more specific papers. Some of the shortcuts include [17, 20, 27, 33, 36, 50]:

- Base the review only in secondary studies
- Use only one search engine e.g., Scopus, Google Scholar
- Limit to only studies published in English
- Limit to specific journals and conferences
- Limit from some specific date range
- Limit according to the methodology of the study e.g., case studies.

If the review team may consult researchers with experience in the IRR topic, they can conduct peer review on the search queries to verify that all related terms are included [36, 47]. Some other search strategies like snowballing [20] or including grey literature may be considered if the review team has experience with these techniques. Regarding the inclusion/exclusion criteria, fixing strict exclusion criteria reduces the number of papers and thus favors rapidness [50].

This step should result in a preliminary version of the IRR protocol containing research questions, and a preliminary version of the search strategy, and inclusion/exclusion criteria. In addition, the review team may have initial ideas about how results will be communicated and the type of reports and documents to develop.

3.3 Search and select papers

Through the activities in this step (see Fig. 4), the review team performs the search and selection of papers. These activities require high interaction with practitioners to validate specific aspects such as terminology, the relevance of specific studies, and context elements. The review team may decide to update or extend the search of sources by conducting snowballing or manual search [20]. These decisions need to be updated in the IRR protocol.

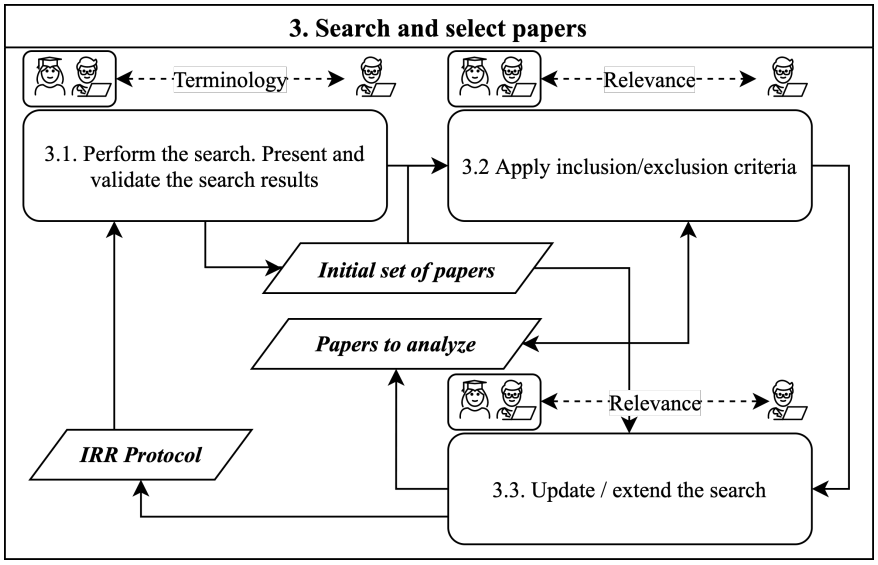


Figure 4: The search and selection of papers is a critical step to ensure the rapidness and relevance of the IRR

With the search results, the review team applies the exclusion criteria to select the set of papers included in the review. As in SLRs, the papers' selection may be divided into the following activities: Review the titles, read the title and full abstract, and read the full paper. A common practice in medicine is that only one team member make decisions about inclusion/exclusion of studies. Leaving the responsibility to only one reviewer reduces the time and avoids solving discrepancies about including/excluding specific studies [17, 23, 30, 50].

During this step, the review team may use tools to support the selection of papers. Felizardo and Carver [18] conducted a systematic search for approaches and tools to automate the SLR process. They found that selection of studies is the activity with most tool support. In their study, the authors analyze the different approaches and provide references to tools. At this point, the review team has a set of papers to analyze to answer the research questions.

3.4 Extract and synthesize data

The activities in this step, see Fig.5, aim to prepare and develop the material that will be used to disseminate the IRR results.

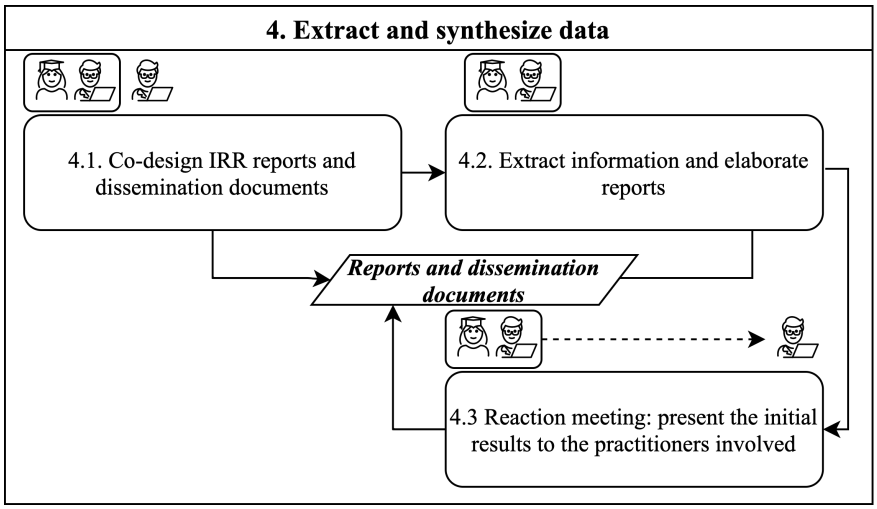


Figure 5: During this task, the review team extracts and synthesizes information from the selected paper to answer the research questions.

Before extracting information from research papers, the review team designs initial reports that will be shared with practitioners. This allows the reviewers to focus on what to search for in the papers. We suggest presenting the result as narrative summaries. A narrative summary is a text that summarizes the findings of the synthesis. More advanced methods like thematic analysis [13] may be used only when having a large number of primary studies, and the process will not impact the time to completion. The synthesis is mainly oriented to describe research results through a narrative summary [22, 45].

In a reaction meeting [14], the review team presents the IRR results to the initial group of practitioners. The practitioners provide feedback and suggestions on how to communicate them to a larger audience. Keep in mind that software engineers, with few exceptions, do not read scientific papers. Thus, the reports need to be designed in a practitioner friendly manner [30]. Some alternatives are visual abstracts [48], evidence briefings [12], presentations, seminars, and posters.

3.5 Disseminate IRR results

Fig. 6. shows the suggested activities in this step to disseminate the IRR results.

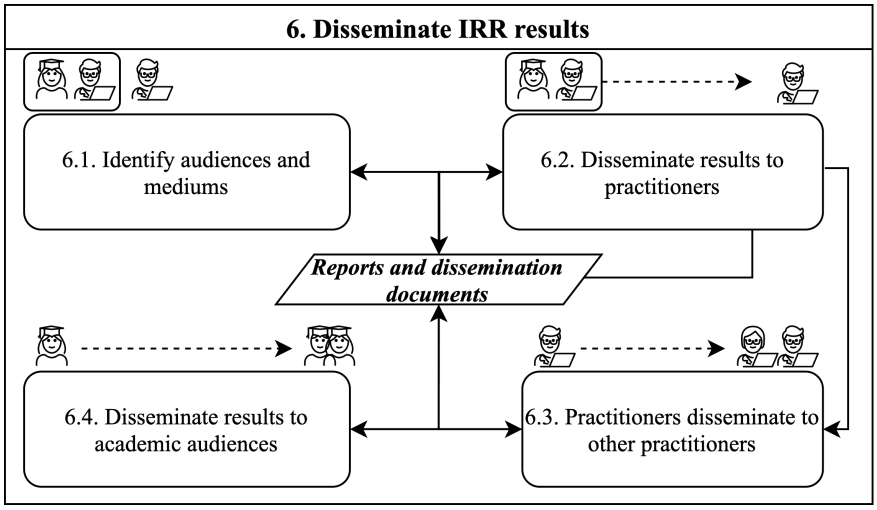


Figure 6: The last step in conducting the review is to disseminate the results.

Initially, the results are communicated to the practitioners involved in the review. Later, the results may be shared with other practitioners in the same organization. For some groups, the diffusion may require to adapt or create new ways to share the results. For example, one group may need less scientific details, while others may require only to present tools or source code. These strategies and diffusion actions need to be coordinated with practitioners who know their context and colleagues better.

Although an IRR's main goal is not to produce a scientific publication, some results may be relevant for academic audiences [33, 39]. If it is the case, the researcher may find the appropriate medium and publish the results. Otherwise, and following non-disclosure agreements, the results may be shared via social networks or in other academic spaces such as workshops, university courses, and online discussion.

3.6 IRR evaluation

Once the IRR results have been disseminated, the review team and the practitioners evaluate if the IRR results support the initial information needs. A possible result is that researchers and practitioners want to explore further a specific topic or take another perspective. Thus, they identify new research questions and apply the steps again. Another possible result may be the identification of a gap in research. If it is the case, the results are a starting point to design and support new research.

In our view, conducting an IRR is an opportunity for mutual understanding between research and practice. When evaluating the IRR, consider besides the outcomes the learnings by participating in the review. By getting involved in the IRR, practitioners develop an awareness of research results and their application in practice while researchers better understand industry challenges and their context.

4 Discussion

RR emerged in medicine as a faster approach than systematic reviews to synthesize knowledge from primary studies. While systematic reviews are well-defined, rapid reviews is an umbrella term that includes a spectrum of related methods. An important aspect of the approach presented in this work is the knowledge exchange between researchers and practitioners. In medicine, there are review groups that work on synthesizing knowledge for decision-making by following standardized protocols accepted by the community. In software engineering, knowledge synthesis is done by the knowledge-users themselves, either researchers or engineers, with different approaches and varying degrees of rigor.

In medicine, practitioners rely on and expect input from academia, while in software engineering, new ideas may be more important than evidence for practitioners approaching academia [55]. Proposed interventions need to be adapted to and re-evaluated in the new context [54]. This can be seen as an argument for allowing synthesizing knowledge in an earlier stage. However, to enable the validity assessment of the conclusions drawn, transparency and context-dependency is key.

RR lack a unique method, but there are some similarities to traditional systematic reviews. Even if the RR approaches are expeditious, they follow a structural set of steps where the research questions are defined at the beginning of the review, making it possible to track the review process and, if necessary, repeat it. Transparency is important since the processes and decision making are faster than in systematic reviews. For these reasons, all the decisions are documented and reported.

Interactive Rapid reviews are conducted in less time than systematic reviews since there is a requirement to have shorter feedback cycles when working with practitioners who want to receive knowledge to affect their products, processes, etc. One way of shortening the time in an IRR is to keep a narrow scope. Here a balance must be decided between answering all relevant questions for a subject and answering only the questions of interest in the collaboration between the practitioners and the researcher. Compared to a traditional review, the selection of subject scope is probably more dependent on practitioners' interests. To what extent this means that relevant and important areas in the literature is not prioritized can be a question for further research.

Another way to decrease the time of IRR is to use shortcuts to expedite the process. To satisfy the time restrictions, rapid reviews skip steps carried out in tra-

ditional systematic reviews or limit some steps. Some examples are: avoiding analysis of inter reviewer agreement, not conducting quantitative analysis, and limiting the search, e.g. by language, time, or the number of databases. Here, a balance must be decided between traditional rigor and obtaining information in a timely way.

Rapid reviews have the potential to bring researchers closer to practitioners and improve communication between them. IRRs aim to maintain professionals' interest and commitment during the review and provide them with useful results. For researchers, we see in IRR an opportunity to get closer to the industry, gather data and information, which we believe is essential in software engineering research.

We consider, like Wohlin [56], that working with industry is more about knowledge exchange than about knowledge transfer. Consequently, our proposal for IRRs is base on the idea that conducting a rapid review with practitioners is an opportunity to establish a bidirectional dialogue where researchers and practitioners get the chance to learn from each other. This interaction facilitates mutual understanding, favors research relevance, and paves the way for future collaborations.

5 Conclusion and Future Work

Our proposal for IRR reinforces the interaction between researchers and practitioners while performing the review. We believe such researcher-led, interactive reviews may improve the knowledge exchange between researchers and software engineering professionals. An IRR starts from a specific knowledge need from practitioners, which implies that the topic is relevant for practitioners from the beginning. During the review, practitioners are highly involved in refining the research questions and defining the protocol, which increases the researchers' understanding of the specific context. Throughout the selection of studies and information extraction, researchers and practitioners keep communicating, contributing to learning from each other. IIR results are disseminated in a practitioner friendly way, making them easier to use.

According to the points mentioned above, we included in our proposal opportunities to focus on the researcher-practitioner exchange during the review. Overall, we recognize in conducting rapid reviews an opportunity to establish a bi-directional exchange between researchers and practitioners that enables future joint work.

Finally, we identified the some potential benefits and challenges of conducting rapid reviews in software engineering. We envision that conducting rapid reviews in collaboration with practitioners may: 1) incentivize a dialogue between researchers and practitioners, 2) provide research results to the industry that are relevant for their context, 3) provide researchers opportunities to learn about the practitioner's problems and their context, and 4) develop networks that could be the base for new collaborative projects.

Similarly, we find the following points as challenging while conducting a rapid review. 1) Time constraints can influence the quality of the review. 2) There is a lack of clear guidance on how to perform rapid reviews and tools to verify the review's quality. 3) There could be misunderstandings about the depth and breadth of a rapid review. 4) There may be a lack of research results on the topic selected. 5) Practitioners' involvement may lead to bias due to practitioners' oriented results.

To address these challenges, we suggest to: 1) keep a protocol that contain all the decisions made in the review to evaluate the strength of conclusions, 2) follow the guidelines proposed in this paper, 3) reinforce transparency as an essential practice when working with industry, and 4) conduct a preliminary search and refine the research questions to identify when there is no available literature in the area, and 5) declare expectations from the beginning about the goals and role of researchers.

As future work, we plan to validate this proposal empirically by studying actual cases of rapid reviews with the industry and evaluate how rapid reviews impact researcher-practitioner communication within and beyond a research collaboration.

* Emojis representing researchers and practitioners designed by OpenMoji – the open-source emoji and icon project. License: CC BY-SA 4.0

References

[1] Nauman Bin Ali, Emelie Engström, Masoumeh Taromirad, Mohammad Reza Mousavi, Nasir Mehmood Minhas, Daniel Helgesson, and Sebastian Kunze. On the search for industry-relevant regression testing research. *Empirical Software Engineering*, 24(4):2020–2055, February 2019.

[2] Nauman Bin Ali and Kai Petersen. Evaluating strategies for study selection in systematic literature studies. In Maurizio Morisio, Tore Dybå, and Marco Torchiano, editors, *Proceedings of the International Symposium on Empirical Software Engineering and Measurement, ESEM '14*, pages 45:1–45:4. ACM, 2014.

[3] Nauman Bin Ali, Kai Petersen, and Claes Wohlin. A systematic literature review on the industrial use of software process simulation. *Journal of Systems and Software*, 97:65–85, 2014.

[4] Nauman Bin Ali and Muhammad Usman. Reliability of search in systematic reviews: Towards a quality assessment framework for the automated-search strategy. *Information & Software Technology*, 99:133–147, 2018.

[5] Nauman Bin Ali and Muhammad Usman. A critical appraisal tool for systematic literature reviews in software engineering. *Information & Software Technology*, 112:48–50, 2019.

[6] David Budgen, Pearl Brereton, Sarah Drummond, and Nikki Williams. Reporting systematic reviews: Some lessons from a tertiary study. *Information and Software Technology*, 95:62–74, 2018.

[7] David Budgen, Barbara A. Kitchenham, and Pearl Brereton. The case for knowledge translation. In *Proceedings of the International Symposium on Empirical Software Engineering and Measurement, ESEM*, pages 263–266, Baltimore, Maryland, USA, 2013. IEEE Computer Society.

[8] Bruno Cartaxo, Gustavo Pinto, Baldoino Fonseca, Márcio Ribeiro, Pedro Pinheiro, Maria Teresa Baldassarre, and Sérgio Soares. Software engineering research community viewpoints on rapid reviews. In *Proceedings of the International Symposium on Empirical Software Engineering and Measurement, ESEM*, pages 1–12, Porto de Galinhas, Recife, Brazil, 2019. IEEE.

[9] Bruno Cartaxo, Gustavo Pinto, and Sergio Soares. The role of rapid reviews in supporting decision-making in software engineering practice. In *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering EASE*, pages 24–34, Christchurch, New Zealand, 2018. ACM.

[10] Bruno Cartaxo, Gustavo Pinto, and Sergio Soares. Towards a model to transfer knowledge from software engineering research to practice. *Information and Software Technology*, 97:80–82, 2018.

[11] Bruno Cartaxo, Gustavo Pinto, and Sergio Soares. Rapid Reviews in Software Engineering. In Michael Felderer and Guilherme Horta Travassos, editors, *Contemporary Empirical Methods in Software Engineering*, pages 357–384. Springer International Publishing, Cham, 2020.

[12] Bruno Cartaxo, Gustavo Pinto, Elton Vieira, and Sérgio Soares. Evidence briefings: Towards a medium to transfer knowledge from systematic reviews to practitioners. In

Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, page 57. ACM, 2016.

[13] Daniela S Cruzes and Tore Dyba. Recommended steps for thematic synthesis in software engineering. In *Proceedings of the 2011 International symposium on empirical software engineering and measurement*, pages 275–284. IEEE, 2011.

[14] Patricia A Deverka, Danielle C Lavallee, Priyanka J Desai, Laura C Esmail, Scott D Ramsey, David L Veenstra, and Sean R Tunis. Stakeholder participation in comparative effectiveness research: defining a framework for effective engagement. *Journal of comparative effectiveness research*, 1(2):181–194, 2012.

[15] Henry Edison, Nauman Bin Ali, and Richard Torkar. Towards innovation measurement in the software industry. *Journal of Systems and Software*, 86(5):1390–1407, 2013.

[16] Emelie Engström, Per Runeson, and Mats Skoglund. A systematic review on regression test selection techniques. *Information & Software Technology*, 52(1):14–30, 2010.

[17] Robin M Featherstone, Donna M Dryden, Michelle Foisy, Jeanne-Marie Guise, Matthew D Mitchell, Robin A Paynter, Karen A Robinson, Craig A Umscheid, and Lisa Hartling. Advancing knowledge of rapid reviews: An analysis of results, conclusions and recommendations from published review articles examining rapid reviews. *Systematic Reviews*, 4(1), 2015.

[18] Katia R. Felizardo and Jeffrey C. Carver. *Automating Systematic Literature Review*, pages 327–355. Springer International Publishing, Cham, 2020.

[19] Katia Romero Felizardo, Érica Ferreira de Souza, Bianca Minetto Napoleão, Nandamudi Lankalapalli Vijaykumar, and Maria Teresa Baldassarre. Secondary studies in the academic context: A systematic mapping and survey. *Journal of Systems and Software*, 170:110734, 2020.

[20] Rebecca Ganann, Donna Ciliska, and Helen Thomas. Expediting systematic reviews: Methods and implications of rapid reviews. *Implementation Science*, 5(1), 2010.

[21] Chantelle Garritty, Adrienne Stevens, Gerald Gartlehner, Valerie King, and Chris Kamel. Cochrane rapid reviews methods group to play a leading role in guiding the production of informed high-quality, timely research evidence syntheses. *Systematic Reviews*, 5(1), 2016.

[22] Liliana Guzmán, Constanza Lampasona, Carolyn Seaman, and Dieter Rombach. Survey on research synthesis in software engineering. In *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering*, pages 1–10, 2014.

[23] Julie Harker and Jos Kleijnen. What is a rapid review? a methodological exploration of rapid reviews in health technology assessments. *International Journal of Evidence-Based Healthcare*, 10(4):397–410, 2012.

[24] Lisa Hartling, Jeanne-Marie Guise, Susanne Hempel, Robin Featherstone, Matthew D Mitchell, Makalapua L Motu'apuaka, Karen A Robinson, Karen Schoelles, Annette Totten, Evelyn Whitlock, et al. Fit for purpose: Perspectives on rapid reviews from end-user interviews. *Systematic Reviews*, 6(1), 2017.

[25] Lisa Hartling, Jeanne-Marie Guise, Elisabeth Kato, Johanna Anderson, Suzanne Belinson, Elise Berliner, Donna M Dryden, Robin Featherstone, Matthew D Mitchell, Makalapua Motu'Apuaka, et al. A taxonomy of rapid reviews links report types and methods to specific decision-making contexts. *Journal of Clinical Epidemiology*, 68(12):1451–1462.e3, 2015.

[26] Donald Hislop, Rachelle Bosua, and Remko Helms. *Knowledge management in organizations: A critical introduction*. Oxford University Press, 2018.

[27] Eva Kaltenthaler, Katy Cooper, Abdullah Pandor, Marrissa Martyn-St James, Robin Chatters, and Ruth Wong. The use of rapid review methods in health technology assessments: 3 case studies. *BMC Medical Research Methodology*, 16(1), 2016.

[28] Shannon E Kelly, David Moher, and Tammy J Clifford. Defining rapid reviews: a modified delphi consensus approach. *International Journal of Technology Assessment in Health Care*, 32(4):265–275, 2016.

[29] S. Khangura, J. Polisena, T.J. Clifford, K. Farrah, and C. Kamel. Rapid review: An emerging approach to evidence synthesis in health technology assessment. *International Journal of Technology Assessment in Health Care*, 30(1):20–27, 2014.

[30] Sara Khangura, Kristin Konnyu, Rob Cushman, Jeremy Grimshaw, and David Moher. Evidence summaries: The evolution of a rapid review approach. *Systematic Reviews*, 1(1), 2012.

[31] Barbara A. Kitchenham, Tore Dybå, and Magne Jørgensen. Evidence-based software engineering. In *Proceedings of the 26th International Conference on Software Engineering (ICSE)*, pages 273–281, 2004.

[32] Barbara Ann Kitchenham, David Budgen, and Pearl Brereton. *Evidence-based software engineering and systematic reviews*, volume 4. CRC press, 2015.

[33] Robyn Lambert, Thomas D Vreugdenburg, Nicholas Marlow, N Ann Scott, Lynda McGahan, and David Tivey. Practical applications of rapid review methods in the development of australian health policy. *Australian Health Review*, 41(4):463–468, 2017.

[34] Claire Le Goues, Ciera Jaspan, Ipek Ozkaya, Mary Shaw, and Kathryn T Stolee. Bridging the gap: From research to practical advice. *IEEE Software*, 35(5):50–57, 2018.

[35] Jessica Tajana Mattivi and Barbara Buchberger. Using the amstar checklist for rapid reviews: Is it feasible? *International Journal of Technology Assessment in Health Care*, 32(4):276–283, 2016.

[36] Heather M McIntosh, Julie Calvert, Karen J Macpherson, and Lorna Thompson. The healthcare improvement scotland evidence note rapid review process: Providing timely, reliable evidence to inform imperative decisions on healthcare. *International Journal of Evidence-Based Healthcare*, 14(2):95–101, 2016.

[37] Emilia Mendes, Claes Wohlin, Katia Felizardo, and Marcos Kalinowski. When to update systematic literature reviews in software engineering. *Journal of Systems and Software*, page 110607, 2020.

[38] Tommi Mikkonen, Casper Lassenius, Tomi Männistö, Markku Oivo, and Janne Järvinen. Continuous and collaborative technology transfer: Software engineering research

with real-time industry impact. *Information and Software Technology*, 95:34–45, 2018.

[39] Gabriel Moore, Sally Redman, Sian Rudge, and Abby Haynes. Do policy-makers find commissioned rapid reviews useful? *Health Research Policy and Systems*, 16(1), 2018.

[40] Denise F O'Leary, Mary Casey, Laserina O'Connor, Diarmuid Stokes, Gerard M Fealy, Denise O'Brien, Rita Smith, Martin S McNamara, and Claire Egan. Using rapid reviews: an example from a study conducted to inform policy-making. *Journal of Advanced Nursing*, 73(3):742–752, 2017.

[41] Carrie D Patnode, Michelle L Eder, Emily S Walsh, Meera Viswanathan, and Jennifer S Lin. The use of rapid review methods for the u.s. preventive services task force. *American Journal of Preventive Medicine*, 54(1):S19–S25, 2018.

[42] Kai Petersen and Nauman Bin Ali. Identifying strategies for study selection in systematic reviews and maps. In *Proceedings of the 5th International Symposium on Empirical Software Engineering and Measurement*, pages 351–354. IEEE Computer Society, 2011.

[43] Kai Petersen, Sairam Vakkalanka, and Ludwik Kuzniarz. Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology*, 64:1–18, 2015.

[44] Julie Polisena, Chantelle Garritty, Chris Kamel, Adrienne Stevens, and Ahmed M Abou-Setta. Rapid review programs to support health care and policy decision making: A descriptive analysis of processes and methods. *Systematic Reviews*, 4(1), 2015.

[45] Catherine Pope, Nicholas Mays, and Jennie Popay. *Synthesising qualitative and quantitative health evidence: A guide to methods: A guide to methods*. McGraw-Hill Education (UK), 2007.

[46] Rasmus Ros, Elizabeth Bjarnason, and Per Runeson. A machine learning approach for semi-automated search and selection in literature studies. In Emilia Mendes, Steve Counsell, and Kai Petersen, editors, *Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering*, pages 118–127. ACM, 2017.

[47] Carolyn Spry and Monika Mierzwinski-Urban. The impact of the peer review of literature search strategies in support of rapid review reports. *Research Synthesis Methods*, 9(4):521–526, 2018.

[48] Margaret-Anne Storey, Emelie Engström, Martin Höst, Per Runeson, and Elizabeth Bjarnason. Using a visual abstract as a lens for communicating and promoting design science research in software engineering. In *Proceedings of the 11th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, ESEM '17, page 181–186. IEEE Press, 2017.

[49] Sian Taylor-Phillips, Julia Geppert, Chris Stinton, Karoline Freeman, Samantha Johnson, Hannah Fraser, Paul Sutcliffe, and Aileen Clarke. Comparison of a full systematic review versus rapid review approaches to assess a newborn screening test for tyrosinemia type 1. *Research Synthesis Methods*, 8(4):475–484, 2017.

[50] Andrea C Tricco, Jesmin Antony, Wasifa Zarin, Lisa Strifler, Marco Ghassemi, John Ivory, Laure Perrier, Brian Hutton, David Moher, and Sharon E Straus. A scoping review of rapid review methods. *BMC Medicine*, 13(1), 2015.

[51] Andrea C Tricco, Etienne Langlois, Sharon E Straus, World Health Organization, et al. *Rapid reviews to strengthen health policy and systems: a practical guide*. World Health Organization, 2017.

[52] Andrea C Tricco, Wasifa Zarin, Jesmin Antony, Brian Hutton, David Moher, Diana Sherifali, and Sharon E Straus. An international survey and modified delphi approach revealed numerous rapid review methods. *Journal of Clinical Epidemiology*, 70:61–67, 2016.

[53] Amber Watt, Alun Cameron, Lana Sturm, Timothy Lathlean, Wendy Babidge, Stephen Blamey, Karen Facey, David Hailey, Inger Norderhaug, and Guy Maddern. Rapid reviews versus full systematic reviews: An inventory of current methods and practice in health technology assessment. *International Journal of Technology Assessment in Health Care*, 24(2):133–139, 2008.

[54] Roel J Wieringa. *Design science methodology for information systems and software engineering*. Springer, 2014.

[55] Ashley Williams. Do software engineering practitioners cite research on software testing in their online articles? a preliminary survey. In *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018*, pages 151–156, 2018.

[56] Claes Wohlin. Empirical software engineering research with industry: Top 10 challenges. In *2013 1st International Workshop on Conducting Empirical Studies in Industry (CESI)*, pages 43–46. IEEE, 2013.

[57] Claes Wohlin. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *Proceedings of the 18th international conference on evaluation and assessment in software engineering*, pages 1–10, 2014.

PAPER V

EXPLORING ML TESTING IN PRACTICE – LESSONS LEARNED FROM AN INTERACTIVE RAPID REVIEW WITH AXIS COMMUNICATIONS

Qunying Song, Markus Borg, Emelie Engström, Håkan Ardö and Sergio Rico. 1st International Conference on AI Engineering – Software Engineering for AI. 2022.

Abstract

There is a growing interest in industry and academia in machine learning (ML) testing. We believe that industry and academia need to learn together to produce rigorous and relevant knowledge. In this study, we initiate a collaboration between stakeholders from one case company, one research institute, and one university. To establish a common view of the problem domain, we applied an interactive rapid review of the state of the art. Four researchers from Lund University and RISE Research Institutes and four practitioners from Axis Communications reviewed a set of 180 primary studies on ML testing. We developed a taxonomy for the communication around ML testing challenges and results and identified a list of 12 review questions relevant for Axis Communications. The three most important questions (data testing, metrics for assessment, and test generation) were mapped to the literature, and an in-depth analysis of the 35 primary studies matching the most important question (data testing) was made. A final set of the five best matches were

analysed and we reflect on the criteria for applicability and relevance for the industry. The taxonomies are helpful for communication but not final. Furthermore, there was no perfect match to the case company's investigated review question (data testing). However, we extracted relevant approaches from the five studies on a conceptual level to support later context-specific improvements. We found the interactive rapid review approach useful for triggering and aligning communication between the different stakeholders.

1 Introduction

Artificial intelligence (AI) applications have grown in popularity and pervasiveness. Among the AI applications currently in use, machine learning (ML) is the dominant technique with active communities in academia and industry [27]. Enterprises across diverse industry domains want to harness the new possibilities promoted by ML. However, due to their impact and increasing use in safety-critical domains, we need to develop ways to build trust in these applications. Bosch *et al.* calls for increased research on *AI engineering* [10], i.e., an evolution of software engineering practices and processes to meet the needs of systems development that incorporate trained ML models. These systems, in contrast to most traditional systems, have a probabilistic behavior [29]. Therefore, we need new approaches and solutions or adapt the existing solutions to new challenges [2, 8].

In this paper, we focus on ML testing, emphasizing applications of ML-based computer vision. ML testing has been a popular research topic in the last few years. Secondary studies show a rapidly increasing publication trend [33,37,46] and dedicated academic workshops and conferences have been established. As novel ML testing results are constantly published, both researchers and practitioners need ways to organize the information and sift through the massive academic output. Furthermore, there is a need for effective ways to match research proposals with application-specific industry needs [13].

An approach helpful in the inception of a collaborative project is interactive rapid reviews (IRRs) [34]. An IRR is a collaborative effort between researchers and practitioners that aims to identify and synthesize relevant research outcomes for the practitioners in their context. Apparently, an IRR could be a beneficial tool for new collaborative projects to explore interests, facilitate the exchange of ideas, and promote mutual understanding.

We conducted an IRR on ML testing with Axis Communications (hereafter Axis). The long-term goal of the IRR was to initiate a collaboration on ML testing between researchers from Lund University, RISE Research Institutes of Sweden, and practitioners at Axis. As a means to that end, and a short-term goal, the IRR should identify the solution proposals from the academic community that are the most likely to provide value for Axis.

The contributions of this paper are the following. First, we developed a taxonomy about practical challenges and available research results on ML testing that helped us learn about the domain and navigate the research results (Sec. 4.1). Second, we compiled a list of twelve practical challenges, identified during the IRR at Axis, related to ML testing (Sec. 4.2). Third, we proposed a preliminary mapping, i.e, a potential connection between research results and practical challenges for three prioritized challenges at Axis (Sec4.3). Finally, we conducted an in-depth review of the 35 primary studies mapped to the highest priority topic, i.e., "How to test the dataset?" We extracted nine technological rules and identified context factors impacting the application of ML testing solutions found in the academic sources (Sec. 4.5).

2 Background and related work

This section presents our position on AI quality and its connection to ML testing. Moreover, we introduce IRRs and the industrial case context.

2.1 AI quality and ML Testing

Quality is a multi-faceted concept that is notoriously difficult to nail down. Adding AI on top of this further exacerbates the challenge. Still, we posit that AI quality is going to be an increasingly important concept to ensure the trustworthiness that future AI systems must provide. AIQ is a regional effort to gather interested parties on AI quality, with a particular focus on the subset of AI that realizes functionality through supervised or unsupervised machine learning, i.e., MLware.

We adhere to the definition of AI quality as "the capability of MLware to satisfy stated and implied needs under specified conditions while the underlying data satisfy the requirements specific to the application and its context" [5]. The definition stresses that MLware combines data and conventional source code; thus, its quality is defined as an amalgamation of corresponding quality definitions from the IEC/ISO 25000 series [21, 22]. The definition is in line with discussions by Felderer *et al.* in the context of testing data-intensive systems [18]. Moreover, the emphasis on data quality assurance is central in this paper.

Inspired by Bjarnason *et al.*'s work on requirements engineering (RE) and software testing [3], our position is that AI quality assurance must be tackled from two directions. RE and testing must support MLware development projects as two bookends. As MLware is sensitive to changes, as Sculley *et al.* put it "changing anything changes everything" [36], aligning RE and testing is perhaps even more important than for conventional software engineering. Within AIQ, we have addressed RE for ML [7,44], ML testing [9,16,30], and MLOps from the perspective of alignment [6]. In this paper, we again focus on ML testing.

ML testing is a rapidly growing research area that evolves software testing to meet the novel characteristics of ML-based systems. We used three secondary

studies of ML testing [33, 37, 46] as the basis for the current work. The three secondary studies were considered the latest in the field when we initiated the current study. Given that the area ML testing is fairly new, we did not expect very old publications. However, we did not explicitly exclude them either. The endpoint in the range was when we started the work, and we included all secondary studies we were aware of. Among the secondary studies we selected, two were published in 2020 [33, 46] and the other one in 2019 [37]. They used a systematic approach for searching, extracting, and synthesizing relevant studies on the topic of ML testing. It is also worth noting that Zhang *et al.* [46] and Ricco *et al.* [33] have also included arXiv pre-prints to be more extensive, and have identified 138 and 70 primary studies, respectively. In contrast, Sherin *et al.* restricted the literature search to peer-reviewed publications only and have identified 37 papers in their study [37]. In total, we have collected 180 unique primary studies based on the three secondary studies.

The three secondary studies report an increasing number of ML testing papers in recent years. The trend is the increased use of ML in various application domains and the importance of techniques for testing such applications. This is particularly evident as ML-based applications are deployed in both safety-critical and mission-critical contexts. Specifically, the majority of the studies that have been surveyed in Zhang *et al.* [46] are focusing on testing the correctness and robustness of supervised machine learning systems, while other type of learning such as unsupervised learning and reinforcement learning, and testing perspectives such as interpretability, efficiency, or privacy are much less studied. The analysis is consistent with the observations from Sherin *et al.* [37] that further attention is required to test the non-functional perspectives and different types of learning for ML systems. Sherin *et al.* [37] also highlighted that there is no adequate empirical evidence to evaluate the effectiveness of the available testing techniques, even though the area of ML testing keeps growing rapidly. In contrast, Ricco *et al.* [33] concluded that the most active research in ML testing has been dedicated on solving automatic test input generation and test oracle creation. Further studies are required to address numerous open challenges such as inventing proper testing metrics as well as benchmarks for ML systems.

2.2 Interactive rapid reviews

In this study, we use the guidelines for IRRs in software engineering proposed by Rico *et al.* [34]. Rapid reviews are a form of knowledge synthesis widely used in medicine to provide information quickly for decision-making. As an example, during the COVID-19 pandemic, a considerable amount of rapid reviews were conducted to support decision-making in many areas of medicine [41]. A group of researchers in software engineering proposed the use of rapid reviews to support decision-making in software projects [14]. A difference with the guidelines adopted in this study is the focus on the interaction between researchers and prac-

titioners to make the reviews more relevant in the practitioners' context and foster the knowledge exchange.

The guidelines are presented as a series of steps. The first step is to *prepare the review*. In this step, an initial area and topic are identified, and the team is set. The second step is to *identify review questions and prepare the IRR protocol*, where based on the exchange, the IRR team formulate review questions that represent the common interest and plan the steps to conduct the IRR i.e, IRR protocol. Then, the third step is to *search and select papers*. In this step, shortcuts are used to reduce the space of search. For that reason, it is suggested that the review questions are narrowed to specific questions. Then, during the fourth step, the IRR team *extracts and synthesize data* from the research literature and prepare the actions to *disseminate IRR results* in the fifth step. It is important to clarify that the guidelines are flexible and may require adaption to the specific needs of the case.

The software engineering research community is familiar with systematic literature reviews (SLRs) as a form of knowledge synthesis. Although IRRs and SLRs are similar in many aspects, like methodology and the need for a systematic approach, it is important to clarify that IRRs do not pretend to be an alternative to SLRs when synthesising research literature. IRRs do not aim to be extensive, but provide rapid and valid input for the practitioners. IRRs address more narrow questions than SLRs. When conducting IRRs the review team applies shortcuts to narrow the search space and then save time in selecting and extracting relevant data. These shortcuts may result in missing relevant sources for the IRR. There are two main reasons to select an IRR for this study. Compared with SRLs, IRRs require less resources and can be completed in shorter time frames. Second, IRRs, as presented here, aim to promote exchange between researchers and practitioners, which is desirable at this phase of Axis.

2.3 Case context

Axis was the first industrial collaboration partner in AIQ. Within Axis, we identified a development team that develop solutions based on advanced ML-based computer vision. The team, develops people counting applications for dynamic environments such as shopping malls and public squares.

People counting is considered a "statistical application" that should be accurate on average. Corner cases are largely ignored, i.e., if a person wearing a "funny hat" is missed or double-counted is ok – as long as the counter is not incremented by an amount large enough to noticeably affect the hourly/daily statistics. This is in contrast to security surveillance applications for which corner cases are critical, e.g., possible intruders crawling under the camera. Still, accuracy over time is important to people counting. False positives (counting ghosts) and false negatives (missing people) are considered equally bad. Thus the F1-score (balanced harmonic mean of precision and recall) is the primary evaluation metric.

A set of test datasets representing scenarios in various operational environments is used for regression testing. As there are significant differences between operational environments, referred to as scenes, F1-scores are measured for individual scenes rather than for a single diverse test set. To provide reliable quality assurance, ensuring a high coverage of scenarios in the test dataset is essential. Differently sized regression test suites are running 1) in a continuous integration context, 2) on nightly builds, and 3) weekly. Two different test setups are used in the regression testing. One testing the algorithms involved only and one testing the actual hardware used.

3 Method

To initiate a collaboration on ML testing between researchers and practitioners at Lund University, RISE Research Institutes of Sweden, and Axis, we conducted an IRR [34] following the five steps in Table 1. The expected outcome of the review was threefold: 1) to establish a common view of the general problem domain – ML testing, 2) to gain a quick overview of how current research matches with the specific needs at Axis, and 3) to propose a study aiming at filling one of the identified gaps. The researchers' activities were carried out by the first three authors of this paper, while the fourth author and his colleagues represent the practitioner's side. Finally, the fifth author guided and monitored the research procedure.

The steps of an IRR are similar to the steps of other types of systematic literature reviews but adapted to meet the specific needs of an industrial stakeholder. In our case the stakeholder was Axis.

3.1 Preparing the review

The goal of the preparation step was to form a review team of both researchers and practitioners and to identify mutual interests and information needs with respect to the general research topic, ML testing. In this step, the interaction between industry and academia took place in an input meeting. To prepare for the input meeting, the second author put together an overview presentation of the state-of-the-art of ML testing, and the fifth author put together an overview presentation of the IRR approach. Four researchers and four practitioners took part in the meeting. The four practitioners had different roles (expert engineer, software test engineer, senior software engineer, and technical leader) at the company and thus different perspectives on the topic. At the meeting, after the presentations, the practitioners shared aspects of their practices and challenges of testing their ML applications.

3.2 Identifying review questions and developing the review protocol

The goal of the second step was to agree on a list of prioritized review questions and an initial review protocol. Here interactions took place in a workshop and a follow-up ranking exercise. Before the workshop, the first three authors developed a preliminary SERP-taxonomy [31] based on the state-of-the-art of ML testing and previous SERP-taxonomies on software testing [1, 17]. A SERP-taxonomy includes four facets (i.e., scope, context, effect, and intervention) to align descriptions of research solutions and industry challenges. In that way, such a taxonomy may be used to facilitate communication between practitioners and support the mapping of challenges from the industry to available research [17].

During the workshop, including the full review team, we walked through all facets and entities of the taxonomy to trigger discussions about ML testing challenges and potential solutions from various perspectives. Based on the outcome of the workshop, the researchers updated the taxonomy and proposed a list of 12 potentially relevant review questions to Axis. This list was then sent out to all participants (researchers and practitioners) with a request to rank them in order of interest using an ordinal scale from 1 to 5. After summarizing the results of this exercise, we agreed to search for research relevant to the three highest ranked questions. Furthermore, we agreed to include research based on relevance and applicability for Axis, but did not specify this further at this point.

3.3 Searching and selecting primary studies

During this step, we successively refined the review protocol while searching for relevant studies and delimiting the scope (i.e., defined exclusion criteria). As part of this activity, the researchers conducted the search and selection while the practitioners gave feedback on relevance and applicability of a small sample of papers sent to them by email.

We limited the search to the research covered by three recent secondary studies on ML testing [33, 37, 46]. The researchers revisited the complete set of primary studies (180 papers) to map them to the three review questions. This screening was based on full-text scanning as it was impossible to do it based neither on the original classification (in the secondary studies) nor on a sole title and abstract screening. At this stage, we had an inclusive approach meaning that if any of the three researchers marked a paper as relevant for a review question, it was coded as such.

Due to the large number of papers coded as potentially relevant to at least one of the three review questions, we decided to descope further and focus the in-depth analysis only on one of the review questions. Thus, the continued selection focused only on the highest prioritized review question, i.e., “How to test the dataset?” 35 of the 180 studies were marked as potentially relevant for this question. After a

thorough review, only five of them remained. At this stage, the remaining candidates were tightly connected to the Axis' context, i.e., testing data for ML-based computer vision.

3.4 Data extraction and synthesis

From the selected papers, we extracted technological rules following the design science lens described by Storey *et al.* [39]. A technological rule is a structured way of describing research contributions with respect to their effect, context, and intervention. Technological rules can be extracted and presented at different levels of abstraction, and are used for communicating the research output in a simple and condensed way [39]. Our goal was to compare technological rules, identify research gaps and the specific needs at Axis. To allow for generalization, we extracted technological rules of different abstraction levels from the primary studies. Furthermore, we extracted the maturity of the rules in terms of empirical observations and analytical reasoning that supported the propositions. The technological rules were then presented to the review team in a short reaction meeting, where the industrial team provided their reflections about relevance and applicability at Axis.

3.5 Disseminating the review results

As the main goal of the review was to initiate a new collaboration, we did not have a plan for disseminating the results within Axis. Instead the most relevant technological rules provide input for new MSc thesis proposals. However, unplanned dissemination took place as new knowledge travel between teams. The technological rules from one of the included papers were evaluated for another purpose by another team at Axis. On the academic side, results were summarized and presented as visual abstracts, one for each technological rule, at a seminar and in this report.

3.6 Validity of contributions

The main goal of conducting an IRR is not to build general theory or publish rigorous research results, but to extract sufficient knowledge to act in a specific situation. In our case, our primary goal was to align terminology, match interests, and initiate industry-academia collaboration on ML testing with one case company. Still, we believe the reported contributions could be helpful for other researchers with similar goals but they must be adapted to their contexts.

The taxonomy builds on previous work on taxonomy development [31], general testing terminology [17] and recent ML testing syntheses [33, 37, 46]. Hence, they are well founded in the research literature. However, the industrial validation

is made from a narrow perspective in a single case context. Thus, when elaborating the taxonomy, the details mirror the review team's interests and experiences, including four practitioners and four researchers.

Similarly, the list of challenges represents a single case, and the ranking of importance represents the interests of the review team. At a high abstraction level, these challenges and interests match the interests of the research community, represented by the 180 primary studies.

A review may also be validated based on its coverage, i.e., is any important work missing? In our case, we did not conduct an extensive search on our own but relied on the rigorous searches made in three recent secondary studies on the same topic. Although their searches were extensive, the time delay caused by the publication process leads to the omission of the most recent publications, i.e., from 2020 and onward. Thus, more recent research may provide a better match to our review questions. This should be considered in future work. Nevertheless, conclusions regarding relevance and applicability of available research are still valid and may guide future reviews.

4 Results

In this section we present the outcome of each step of the IRR. Subsection 4.1 describes the entities of the taxonomy after validation, subsection 4.2 lists the open questions derived from the workshop, subsection 4.3 describes our mapping of primary studies to the review questions, subsection 4.4 presents the final exclusion criteria resulting from the iterative review of papers mapped to review question 1, subsection 4.5 presents the nine technological rules extracted from the five most relevant primary studies, and subsection 4.6 finally elaborates on what we did not find in the research literature, i.e., the gap between research and practice in our case.

4.1 ML testing taxonomy

As a result of the initial interaction between the case company and the researchers, we agreed on creating a taxonomy of ML testing to guide the collaboration further. We developed a general taxonomy for three out of four facets in the SERP-taxonomy architecture [31] (i.e., context, scope, and effect). Since we enter this review from the challenge perspective, we found that detailing these three facets were sufficient for the communication within the review team. The fourth SERP-facet, intervention, may be used to elaborate technical aspects that support abstraction and comparison of classes of solutions.

Our resulting taxonomy, presented in Figure 1, aims to guide the formulation of practical ML testing challenges at an appropriate abstraction level to support identification and design of relevant research.

Scope of ML testing interventions

The green sector in Figure 1 shows the details of the scope facet. Scope here refers to the testing activity, or part of the testing process, on which a potential intervention may focus. We identified four important aspects where two are derived from generic testing literature, i.e., testing process and testing levels, and the other two are specific to the ML context, i.e., parts of the ML-system to be tested and the mode of operations (online or offline testing, cf. Figure 1).

Effect of ML testing interventions

The effect of an intervention is described in terms of its observed or desired impact on the testing. It could for example be the reported result of an empirical evaluation or an identified practical need in an exploratory study or case description. The blue sector in Figure 1 shows the effect facet. The main types of desired effects are derived from the generic SERP-taxonomy architecture [31], i.e., solving an unsolved problem (solve), improving the current solution (improve), and assessing the current situation (diagnose).

Context of ML testing interventions

The context facet aims to capture factors in the context that impact the effect or applicability of an intervention. Several such factors were identified, reflecting the multidimensional variation of both ML systems and testing approaches. Eight aspects of the context were included as shown in the yellow sector of Figure 1: 1) programming languages used for the implementation of the system, 2) degree of access to the system components, 3) framework, 4) machine learning type, 5) testing setup, 6) application, 7) type and 8) domain of the system.

At the initial stages of our project, we used the taxonomy for structuring the information in the secondary studies and for triggering discussions about the topic within the review team. We believe the resulting taxonomy may be used in similar ways by others to view ML testing challenges and solutions from various perspectives and thus support communication between researchers and practitioners who approach the topic – in different ways and in many cases at different abstraction levels. While conducting this review, we experienced that considering all the facets and digging in to details of all the facets of the taxonomy helped the communication and, by extension, our common understanding.

Table 1: Description of the five steps and corresponding research activities for this IRR study

Step	**Activities for Researchers**	**Activities for Practitioners***
1. Prepare the review	Describe research area and preliminary research goals	Meeting to identify mutual information needs and agree on involvement.
2. Identify review questions and develop the IRR protocol	Propose SERP taxonomy, elaborate review questions and scope of search and selection.	Meeting to validate and refine taxonomy and elaborate on questions and scope. List and prioritize review questions.
3. Search and select papers	Map primary studies to review questions, iterate samples of selected studies with practitioners, update inclusion/exclusion criteria based on feedback.	Give feedback on relevance and applicability of selected papers.
4. Extract and synthesize data	Identify and assess maturity of technological rules.	Meeting to discuss results (esp. relevance, applicability) of technological rules. Discuss how and to whom results should be summarized and communicated.
5. Disseminate IRR results.	Design and present visual abstracts for the identified TRs. Propose new research studies.	Meeting to give feedback on results. Present results within company.

4.2 Prioritized list of open questions

Guided by the taxonomy and the feedback from practitioners in the initial meetings, we formulated 12 review questions potentially relevant to our case. In the following list, organized using the scope facet of the taxonomy in Figure 1, boxes represent the review team's highest priority questions. Verbs from the effect facet appear in italics.

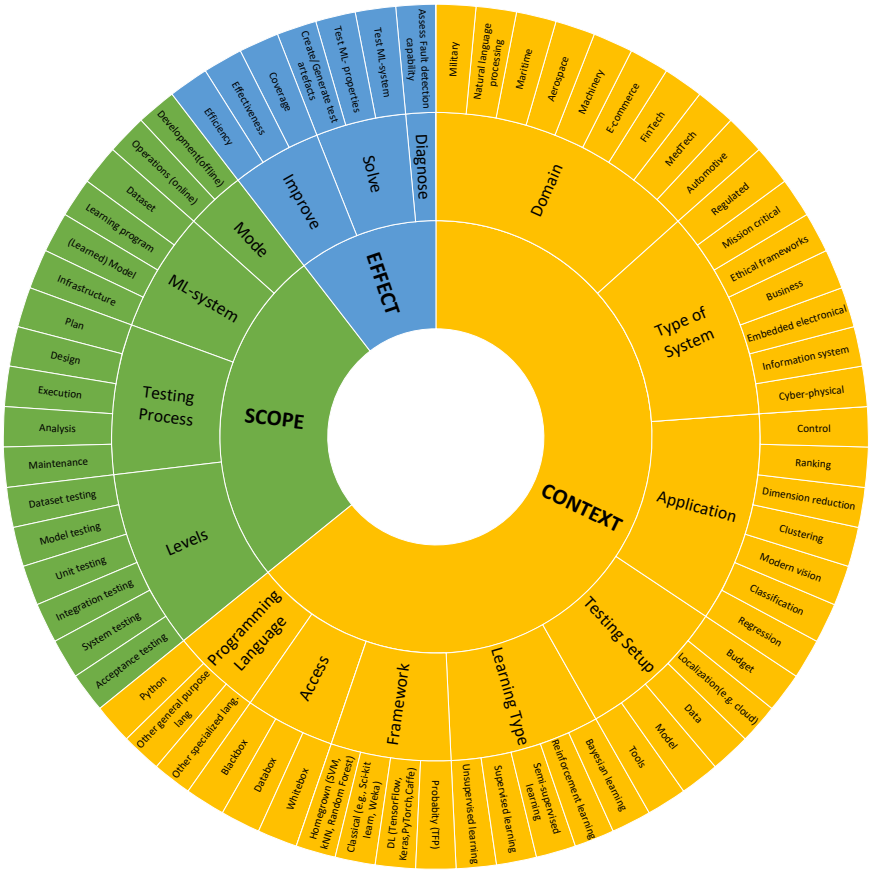


Figure 1: An ML testing SERP taxonomy with the facets context, scope, and effect.

- **ML system – Dataset**

(#1) How to test the dataset? The dataset is evolving, which motivates the following sub-questions:
  (a) How to identify mislabeled data in the dataset?
  (b) Adequacy testing, i.e., how to assess and improve data (scenario) coverage of the training and test datasets in terms of diversity?
  (c) How to assess potential bias after the training/test split?
  (d) How valid is the data and its use? Is the data used for testing within the operational design domain? Or did some of the data originate from another source?

- **ML System – Learning program**

(#2) How to *diagnose* (assess the fault detection capability) and *improve* (unit) testing (design and analysis) of the learning program?

(#3) How to *improve* testing of the learning program to detect more faults? (e.g., using unit testing)

- **ML system – Learned model**

(#4) Are there complementary metrics to assess model correctness (accuracy)? (e.g., edge case measures, uncertainty scores, aggregated metrics across scenes)

(#5) How to interpret and analyze the testing result of the ML model? (e.g., increased automation or visual analytics)

(#6) How to *diagnose* whether the current set of scenarios in the test dataset are appropriate for detecting faults on the model level?

(#7) How to *improve* coverage testing with respect to scenario diversity?

(#8) How to *improve* the test dataset to increase fault detection ability?

(#9) How to *improve* test prioritization to increase regression test efficiency?

(#10) How to generate new test cases for testing the model? (e.g, synthetic data, data augmentation, guided search)

- **Levels – System testing** (see Section 2.3)

(#11) How to *improve* (acceptance) testing of the ML-based system?

(#12) How to *diagnose* whether the current set of scenarios in the test dataset are appropriate for detecting faults on the system level?

Even though the above questions represent the interests of the review team, we argue that they constitute real challenges that generally deserve attention. We believe they may support other researchers trying to identify research gaps in ML testing for computer vision applications.

4.3 Mapping primary studies to the three most important open questions

The review team (i.e., four practitioners and three researchers) ranked the questions based on importance. Then the 180 primary studies covered in this review were mapped to the three review questions that were considered the most important (i.e., number #1, #4, and #10 (in the list of questions above). 35 primary studies were marked as potentially relevant for review question 1, 25 primary studies for review question 4, and 68 for review question 10. The details of this mapping can be found on Zenodo [38].

This mapping may help navigating the research literature and could be used as a starting point by researchers and practitioners facing similar challenges.

4.4 Selection of studies related to question #1

This section describes our analysis of relevance and applicability for Axis in relation to the review question #1. The final list of exclusion criteria is:

- *Purpose*. The proposed mechanism does not evaluate some properties of the data. In MLware, the training dataset is part of the system [5]. The purpose of the proposed mechanism shall be to test these data, not to generate synthetic data – unless the synthetic data are specifically used to validate the training data through comparison. Examples of excluded papers relate to:
  - Test data generation for ML systems such as Zhang *et al.* [47] and Tian *et al.* [40] that generate artificial driving scenes for testing ML-based autonomous driving functions.
  - Testing the model fitness like Zhang *et al.* [45] which validates the model relevance, and ML model underfitting as well as overfitting using a perturbed model validation technique.
  - Online monitoring of data prior to making predictions, e.g., Henriksson *et al.* [19], since it targets testing aspects of the operational environment and detects inputs that are outside the training dataset.
  - The proposed mechanism is intended to protect the neural network from antagonistic attacks. Antagonistic attacks are closer to cybersecurity research than data validation. Furthermore, adversarial attacks would typically target the system in operation and not the training/validation/test dataset. An example of such studies is Uesato *et al.* [42]

which evaluates learning systems in safety-critical domains by identifying the adversarial situations.

- *Applicability*. The proposed mechanism is not applicable in the Axis' context. Reasons for exclusion include:
  - *Not NN learning*. The mechanism is not applicable to supervised learning with neural networks. Axis uses neural networks for supervised learning and the proposed mechanism must be applicable. Thus, papers that explicitly address other learning mechanisms are considered out of scope, such as Krishnan *et al.* [26] on support vector machines and Uesato *et al.* [42] on reinforcement learning.
  - *Not images*. There is no explicit mention of how the proposed mechanism could work for images. Axis trains models for video sequences and the proposed mechanism must be applicable. Thus, interventions that validate only non-image data are excluded, e.g., spell or format checking, and named entity recognition on tabular data [11, 20, 35].

4.5 Best matches

After applying the criteria described above, five primary studies remained, partly answering the general review question (#1 in Section 4.2) "How to test the dataset?" Although none of the proposed solutions were directly applicable in the case context, Axis confirmed related problem formulations and shared potentially valuable ideas. In line with the design science lens [39], we extracted technological rules for each paper. A technological rule captures the mapping between a problem and a solution. We describe them in terms of: ***To achieve <effect> IN <context> DO <intervention>***. We describe the technological rules at different abstraction levels, i.e, some are more concrete and others are more general.

Paper I

Ma *et al.* [28] propose a mutation testing framework for deep learning systems to assess the quality of the test data. Specifically, a set of source-level mutation operators are defined to introduce faults to the training data and the training programs. In addition, a set of model-level mutation operators are defined to create mutants for the deep learning models without a training process. Eventually, the effectiveness of the test data can be evaluated from the analysis based on to what extent the injected faults could be detected. The authors have used the framework on two publicly available datasets (i.e., MNIST and CIFAR-10) with three popular deep learning models, and demonstrated the effectiveness of the framework for designing and constructing high-quality test datasets for deep learning MLware.

Two technological rules were extracted from this paper [28], i.e., a concrete one:

Technological rule 1

To measure the quality of test data for deep learning systems, use an adapted form of mutation testing.

and a general one:

Technological rule 2

To improve the generality and robustness of deep learning models, test the test dataset.

Both the source-level and model-level mutant operators are general and can be reused. Furthermore, the concept of the proposed framework, i.e., to use mutation testing, is not constrained by any specific type of deep learning application or data used. Thus, we consider this paper potentially relevant, and the synthesized findings can be transformed for testing the quality of test data for Axis as well. The response from Axis was positive, they found the approach interesting but questioned the scalability. Axis works with orders of magnitude larger datasets (about 10^5 to 10^7 images, which is 10 to 1,000 times larger than the MNIST and CIFAR-10 datasets) than the ones used for evaluating the approach in the paper. In addition, the industrial datasets are rather Full HD resolution than the 32×32 pixels targeted in many research papers. Another question is which mutant operators would work for the complexity of the industrial case, as MNIST and CIFAR-10 are trivial datasets in comparison. A pre-study in the Axis context will be initiated to investigate this further.

Paper II

Kim *et al.* [24] propose a concept of surprise adequacy as the test adequacy criterion for deep learning systems. Based on the trace of the neuron activation when executing the deep learning model on both the training data and the testing data, surprise adequacy can be calculated using either a likelihood-based approach or a distance-based approach. The resulting surprise adequacy indicates how surprising the test data is compared to the training data. The concept assumes that a good test input should be sufficiently, but not excessively, surprising to the training data. The authors also evaluate the effectiveness of using the surprise adequacy metric for sampling test input and improving the model accuracy via retraining, based on publicly available datasets such as MNIST and CIFAR-10, and deep learning systems for autonomous vehicles like Dave-2 and Chauffeur.

Two technological rules were extracted from this paper [24], the concrete one is:

Technological rule 3

To improve the classification accuracy of deep learning systems, retrain the model, systematically sampling inputs based on the surprise adequacy criterion.

while the general one is described as:

Technological rule 4

To test the correctness and robustness of deep learning systems, test the systems' behaviour with respect to their training data.

The proposed criterion – surprise adequacy – and the corresponding ways of computing such a criterion are transferable to different deep learning MLware as the training data, testing data, and neuron activations are inherent parts of such systems. The expected outcome is an indication of how good or how different the test is compared to the training data.

While the surprise adequacy metric was not considered relevant for the practitioners in the review team, it was explored by another team at Axis. Here it was explored for a slightly different purpose, i.e., to guide complementary data collection rather than for testing data. Collecting additional data that strives to maximize the surprise adequacy has also been proposed by the original inventors [25], as an approach to increase the diversity of both training and test datasets. However, Axis compared surprise adequacy to a set of other metrics and in the end they selected another option (recent work proposed by Pleiss et al. [32], not covered by the secondary studies used for our study selection). Part of the reason was that the proposed surprise adequacy calculations did not scale to size of the data set as described earlier (see subsection 4.5.1). On the other hand, Axis encourages additional research into surprise adequacy calculation for representative subsets of the data, i.e., aggregating measures for families of neuron activation traces.

Paper III

Byun *et al.* [12] introduce three different metrics for test input prioritization for deep neural networks, i.e., (1) confidence, measured by using the softmax function, (2) uncertainty, measured using Bayesian Networks, and (3) surprise as described in the previous paper by Kim *et al.* [24]. The authors apply these metrics to prioritize test inputs for two different systems (i.e., a digital classification system trained on the MNIST dataset, and TaxiNet) for image classification. They show the effectiveness of the metrics in indicating fault-revealing inputs and, by extension, for selecting test input and improving the model via retraining.

Two technological rules were extracted from this paper [12], including a concrete one:

Technological rule 5

To prioritize test input for deep neural networks, apply metrics of confidence, uncertainty, and surprise.

and a general one:

Technological rule 6

To increase test effectiveness when testing deep neural networks in safety-critical systems, prioritize test input.

Similar to paper II, this paper proposes metrics to measure the sentiment of the deep neural networks. Then, the test inputs can be validated, prioritized, and effectively selected for testing and retraining the model. The findings are considered generic and potentially relevant for Axis' needs. While "Surprise Adequacy" was investigated by another team at the company (see subsection 4.5) the other two metrics "confidence" and "uncertainty" have not yet been considered.

Paper IV

Cheng *et al.* [15] study a set of metrics to measure the dependability attributes of neural networks. The metrics include robustness, interpretability, completeness, and correctness. In our review, the paper was initially excluded due to its purpose (i.e., measuring the dependability of neural network models) and the application on autonomous driving. However, after a second review, the paper was included since the part related to the completeness of the training data seems relevant. The paper uses neuron k-activation and neuron activation patterns as a measure of scenario coverage and completeness of training data for NN-based autonomous driving systems. The assessment of completeness of the training data, which is used for testing the coverage of the training data but is general for testing of data regardless of its use, could support both quality assurance of training or test datasets.

One general technological rule was extracted from this paper [15] as only the part about the training data completeness is relevant. The technological rule is described as:

Technological rule 7

To measure dependability of neural networks, evaluate the completeness of the training data.

The part that involves measuring completeness of training data is relevant for data testing in general. While the paper sets the general focus in autonomous driving applications, further investigation on how it could be applied into Axis' context should be performed.

Paper V

Bolte *et al.* [4] construct a system framework for corner case detection in training datasets for autonomous driving systems. The framework consists of three parts: (1) a semantic segmentation model to partition and classify the image into different semantic parts; (2) an image prediction model that predicts the next image based on the previous set of images and counting the errors based on the real image; and (3) a detection system that detects the corner cases if an object is unpredictable given the error score counted in the previous model. The proposed framework can be used in both online and offline modes. The difference is that the offline mode takes a collected database of image data for training, and the online model uses live video frames collected by the camera installed on vehicles. The authors have trained and evaluated the framework on the Cityscapes dataset and achieved prominent results for detecting unusual situations for autonomous driving. Two technological rules were extracted from this paper [4], the concrete one is:

Technological rule 8

To detect corner cases for ML-based autonomous driving systems, use a system framework based on image segmentation and prediction.

while the general one is:

Technological rule 9

To improve the robustness of machine learning systems, identify critical situations.

The concept of the study is relevant for data testing for ML systems in general, although the proposed system framework works with image data. It predicts corner cases for autonomous driving, which can be used to test and retrain the ML model. Still, the actual applicability of using this framework in Axis needs to be further investigated.

While identifying corner cases is not so important for the application of people counting, which is the focus of the practitioners in the review team, it could be vital for the companies' other products in the security business segment. For example, a corner case of someone moving strangely to avoid detection would be critically important for a security application to detect. For security applications, detecting such anomalies are among the most important use cases. On the other hand, for people counting applications, it could be interesting to apply a broader definition of corner cases, i.e., not only very rare cases but rather underrepresented scenarios. Any mechanism that could support identification of such scenarios would be helpful for ML testing.

4.6 Identified gap

The five papers that we have presented in the previous subsection mainly include frameworks, metrics, constructed tools, and mechanisms for measuring the quality of the data and consistency of the test data with respect to the training data for ML systems. Based on the general focus and sub-questions of the first review question (i.e., how to test the dataset), we observed that the extracted technological rules can be used to address some, but not all, related perspectives.

In particular, there are no studies that provide relevant techniques for a) identifying mislabeled data in the dataset. Thus, this sub-question is still an open challenge and needs to be studied further. However, paper IV provides mechanisms for measuring the completeness of training data, which gives some insights and potential solutions for b) assessing and improving the training and testing data coverage – also from the perspective of scenario coverage. Note that the proposed mechanism focuses on the autonomous driving domain. In addition, no studies we identified target c) assessing biases between the training and testing data. However, papers II and III could be potentially relevant since they support measuring and filtering test data not represented in the training data using different metrics (i.e., confidence, uncertainty, and surprise adequacy). The same observation also applies for sub-question d) how valid the data used for testing is with respect to the training data, where the metrics proposed in papers II and III can be used for such purposes. As a result, we believe the findings we synthesized can ease some parts of the research question we focus on, whereas further investigation is needed to address the remaining gaps.

5 Discussion

AI engineering is an emerging field that is vital for AI quality. As argued in Section 2, ML testing is going to play a critical role in ensuring that future AI solutions are trustworthy. However, there is no established go-to model describing ML testing. Several previous studies propose dimensions to bring structure to the research area. This paper synthesizes a novel taxonomy based on three secondary studies. The taxonomy shall be considered work-in-progress, but it already has provided value for us in an emerging industry-academia collaboration.

The three secondary studies used three different classification strategies for their respective goals. Both Riccio *et al.* and Sherin *et al.* refer to their works as systematic mapping studies and focus on trends and gaps. The scope of Riccio *et al.* is functional testing of ML systems and they structure the 70 primary studies based on 1) system context, 2) testing approach, and 3) empirical evaluation [33]. Sherin *et al.*'s mapping, including also non-functional testing, provides less synthesis and rather extracts fine-granular information from the 37 primary studies. In the secondary study by Zhang *et al.*, referred to only as a survey, the authors explicitly specify their ambition to provide a comprehensive overview of ML test-

ing. The 138 included papers are organized into 1) testing properties, 2) testing components, 3) testing workflow, and 4) application scenarios. We believe that our proposed taxonomy combines the complementary perspectives provided in previous work.

Any taxonomy or model is developed for a specific communication purpose, targeting a defined group of people. In our case, the goal was to align terminology and identify shared interests within the group of researchers and practitioners. Thus, we found the SERP approach [31] applicable and useful. Furthermore, by building on SERP-test [17], comprising common testing terminology, and adding the ML perspective from the secondary studies [33,37,46] as well as from the case company, we got a solid basis for our communication on the topic – ML testing.

At a general level, we found a good match between the prioritized challenges of our case company and the research focus within the community. As a result, 92 out of 180 primary studies were classified as potentially relevant for at least one of the top three review questions, i.e., #1 How to test the dataset (35 primary studies), #2 How to assess model accuracy (metrics) (25 primary studies), and #3 How to generate test cases for testing the model (68 primary studies). However, as shown in the in-depth analysis of the first set of papers, no perfect matches exist.

Of the 35 primary studies initially considered relevant for data testing, we finally selected and reviewed five that best matched the specific context and needs at Axis. After analyzing and synthesizing the findings from the papers, we extracted nine technological rules. The practitioners were positive to the presented techniques (see subsection 4.5) and thought most of them could be relevant. Particularly, they have used the surprise adequacy for complementing data collection in the company as described in subsection 4.5.2. However, we found no perfect match directly transferable to the applications and data testing issues at Axis. As underlined by the definition of AI quality [5], finding feasible ways to perform data quality assurance is at the heart of the problem. Therefore, we are convinced that data testing will play an important role in the future of AI engineering. Also, it is significant in future research to explore further how to instantiate and evaluate the techniques in this industrial setting.

The concepts (e.g., metrics and criteria) and interventions (e.g., approaches and frameworks) presented in the five papers are quite generic for data testing in the ML field as to the extent of our interpretation. Hence, we believe those concepts and interventions can be reused and adapted for solving potential issues for different ML application domains. In the same way, the technological rules can be used to map solutions to challenges at different abstract levels, and support the communication and knowledge exchange between academic researchers and industrial practitioners in further studies.

6 Conclusions

We report outcomes and lessons learned from applying an interactive rapid review on machine learning testing. The review team consisted of four researchers from Lund University and RISE Research Institutes of Sweden and four practitioners from Axis. The primary goal of the study was to initiate collaboration and align terminology and interests between the partners.

Three secondary studies, covering 180 primary studies on machine learning testing, functioned as a starting point for the review. The classifications of research in the secondary studies were mapped to the SERP taxonomy architecture [31] to guide the alignment of terminology and interests within the review team. The resulting SERP taxonomy were further extended by general taxonomies on software testing [17] built on the same taxonomy architecture. Finally, we validated and updated the outcome based on discussions and reflections in the review team. While we plan to evolve the outcome, *this paper presents the latest version of the taxonomy.*

The new SERP taxonomy was used to identify and describe current challenges in the case context. *The complete list of challenges are presented in this report*. Furthermore, the review team ranked the challenges by their perceived importance for the target organizational unit within the case company.

The primary studies were mapped to the three most important questions. Moreover, we conducted an in-depth analysis of the 35 papers for the highest ranked question, i.e., "How to test the dataset?" *We present and discuss 9 technological rules on data testing*, extracted from 5 of the papers. *Finally, we report and discuss the relevance and applicability criteria used to filter out those 5 papers.*

As AI quality combines source code and data quality [5], we believe that data testing will be increasingly important within the field of AI engineering. Our findings call for more research on the topic, not the least for image data, targeting business-critical computer vision systems. Furthermore, convincing data testing for computer vision applications can potentially constitute a cornerstone in the safety argumentation in future assurance cases, e.g., for critical ML-based perception applications in automotive [8], avionics [43], and healthcare [23].

The motivation for this interactive rapid review was to identify research or research gaps of relevance for the case company. Thus, all steps in the process have been guided by Axis' specific needs. As our next step, we plan to design a joint solution-oriented study on the topic of data testing as well as a set of MSc thesis project proposals. Based on the discussions of the selected studied in Section 4.5, there are several promising directions for future collaborations. Our case (of industry-academia collaboration) is a single case and as such a proof-of-concept that may be extended with additional cases.

Acknowledgments

This initiative received financial support through the AIQ Meta-Testbed project funded by Kompetensfonden at Campus Helsingborg, Lund University, Sweden. In addition, this work was supported in part by the Wallenberg AI, Autonomous Systems and Software Program (WASP).

References

[1] Nauman Bin Ali, Emelie Engström, Masoumeh Taromirad, Mohammad Mousavi, Nasir Mehmood Minhas, Daniel Helgesson, Sebastian Kunze, and Mahsa Varshoaz. On the search for industry-relevant regression testing research. *Empirical Software Engineering*, 24(4):2020–2055, 2019.

[2] Saleema Amershi, Andrew Begel, Christian Bird, Robert DeLine, Harald Gall, Ece Kamar, Nachiappan Nagappan, Besmira Nushi, and Thomas Zimmermann. Software engineering for machine learning: A case study. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, pages 291–300, 2019.

[3] Elizabeth Bjarnason, Per Runeson, Markus Borg, Michael Unterkalmsteiner, Emelie Engström, Björn Regnell, Giedre Sabaliauskaite, Annabella Loconsole, Tony Gorschek, and Robert Feldt. Challenges and practices in aligning requirements with verification and validation: a case study of six companies. *Empirical software engineering*, 19(6):1809–1855, 2014.

[4] Jan-Aike Bolte, Andreas Bar, Daniel Lipinski, and Tim Fingscheidt. Towards corner case detection for autonomous driving. In *2019 IEEE Intelligent vehicles symposium (IV)*, pages 438–445. IEEE, 2019.

[5] Markus Borg. The aiq meta-testbed: Pragmatically bridging academic ai testing and industrial q needs. In Dietmar Winkler, Stefan Biffl, Daniel Mendez, Manuel Wimmer, and Johannes Bergsmann, editors, *Software Quality: Future Perspectives on Software Engineering Quality*, pages 66–77, Cham, 2021. Springer International Publishing.

[6] Markus Borg. Agility in software 2.0 – notebook interfaces and mlops with buttresses and rebars. In *Proc. of the International Conference on Lean and Agile Software Development*, pages 3–16. Springer, 2022.

[7] Markus Borg, Joshua Bronson, Linus Christensson, Fredrik Olsson, Olof Lennartsson, Elias Sonnsjö, Hamid Ebadi, and Martin Karsberg. Exploring the assessment list for trustworthy ai in the context of advanced driver-assistance systems. In *2021 IEEE/ACM 2nd International Workshop on Ethics in Software Engineering Research and Practice (SEthics)*, pages 5–12. IEEE, 2021.

[8] Markus Borg, Cristofer Englund, Krzysztof Wnuk, Boris Duran, Christoffer Lewandowski, Shenjian Gao, Yanwen Tan, Henrik Kaijser, Henrik Lönn, and Jonnas Törnqvist. Safely entering the deep: A review of verification and validation for machine learning and a challenge elicitation in the automotive industry. *Journal of Automotive Software Engineering*, 1(1):1–19, 2019.

[9] Markus Borg, Ronald Jabangwe, Simon Åberg, Arvid Ekblom, Ludwig Hedlund, and August Lidfeldt. Test automation with grad-cam heatmaps-a future pipe segment in mlops for vision ai? In *2021 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, pages 175–181. IEEE, 2021.

[10] Jan Bosch, Helena Holmström Olsson, and Ivica Crnkovic. Engineering ai systems: A research agenda. In *Artificial Intelligence Paradigms for Smart Cyber-Physical Systems*, pages 1–19. IGI Global, 2021.

[11] Eric Breck, Neoklis Polyzotis, Sudip Roy, Steven Whang, and Martin Zinkevich. Data validation for machine learning. In *MLSys*, 2019.

[12] Taejoon Byun, Vaibhav Sharma, Abhishek Vijayakumar, Sanjai Rayadurgam, and Darren Cofer. Input prioritization for testing neural networks. In *2019 IEEE International Conference On Artificial Intelligence Testing (AITest)*, pages 63–70. IEEE, 2019.

[13] Anita D Carleton, Erin Harper, Michael R Lyu, Sigrid Eldh, Tao Xie, and Tim Menzies. Expert perspectives on ai. *IEEE Software*, 37(4):87–94, 2020.

[14] Bruno Cartaxo, Gustavo Pinto, and Sergio Soares. *Rapid Reviews in Software Engineering*, pages 357–384. Springer International Publishing, Cham, 2020.

[15] Chih-Hong Cheng, Chung-Hao Huang, Harald Ruess, Hirotoshi Yasuoka, et al. Towards dependability metrics for neural networks. In *2018 16th ACM/IEEE International Conference on Formal Methods and Models for System Design (MEMOCODE)*, pages 1–4. IEEE, 2018.

[16] Hamid Ebadi, Mahshid Helali Moghadam, Markus Borg, Gregory Gay, Afonso Fontes, and Kasper Socha. Efficient and effective generation of test cases for pedestrian detection-search-based software testing of baidu apollo in svl. In *2021 IEEE International Conference on Artificial Intelligence Testing (AITest)*, pages 103–110. IEEE, 2021.

[17] Emelie Engström, Kai Petersen, Nauman Bin Ali, and Elizabeth Bjarnason. Serp-test: A taxonomy for supporting industry—academia communication. *Software Quality Journal*, 25(4):1269–1305, dec 2017.

[18] Michael Felderer, Barbara Russo, and Florian Auer. On testing data-intensive software systems. In *Security and Quality in Cyber-Physical Systems Engineering*, pages 129–148. Springer, 2019.

[19] Jens Henriksson, Christian Berger, Markus Borg, Lars Tornberg, Cristofer Englund, Sankar Raman Sathyamoorthy, and Stig Ursing. Towards structured evaluation of deep neural network supervisors. In *2019 IEEE International*

Conference On Artificial Intelligence Testing (AITest), pages 27–34. IEEE, 2019.

[20] Nick Hynes, D Sculley, and Michael Terry. The data linter: Lightweight, automated sanity checking for ml data sets. In *NIPS MLSys Workshop*, 2017.

[21] ISO/IEC. Iso 25012 systems and software engineering – systems and software quality requirements and evaluation (square) - data quality model, 2008.

[22] ISO/IEC. Iso 25010 systems and software engineering – systems and software quality requirements and evaluation (square) - system and software quality models, 2011.

[23] Fei Jiang, Yong Jiang, Hui Zhi, Yi Dong, Hao Li, Sufeng Ma, Yilong Wang, Qiang Dong, Haipeng Shen, and Yongjun Wang. Artificial intelligence in healthcare: past, present and future. *Stroke and vascular neurology*, 2(4), 2017.

[24] Jinhan Kim, Robert Feldt, and Shin Yoo. Guiding deep learning system testing using surprise adequacy. In *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*, pages 1039–1049. IEEE, 2019.

[25] Jinhan Kim, Jeongil Ju, Robert Feldt, and Shin Yoo. Reducing dnn labelling cost using surprise adequacy: An industrial case study for autonomous driving. In *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pages 1466–1476, 2020.

[26] Sanjay Krishnan, Jiannan Wang, Eugene Wu, Michael J. Franklin, and Ken Goldberg. Activeclean: Interactive data cleaning for statistical modeling. *Proc. VLDB Endow.*, 9(12):948–959, aug 2016.

[27] Lucy Ellen Lwakatare, Aiswarya Raj, Ivica Crnkovic, Jan Bosch, and Helena Holmström Olsson. Large-scale machine learning systems in real-world industrial settings: A review of challenges and solutions. *Information and Software Technology*, 127:106368, 2020.

[28] Lei Ma, Fuyuan Zhang, Jiyuan Sun, Minhui Xue, Bo Li, Felix Juefei-Xu, Chao Xie, Li Li, Yang Liu, Jianjun Zhao, et al. Deepmutation: Mutation testing of deep learning systems. In *2018 IEEE 29th International Symposium on Software Reliability Engineering (ISSRE)*, pages 100–111. IEEE, 2018.

[29] Dusica Marijan, Arnaud Gotlieb, and Mohit Kumar Ahuja. Challenges of testing machine learning based systems. In *2019 IEEE International Conference On Artificial Intelligence Testing (AITest)*, pages 101–102, 2019.

[30] Mahshid Helali Moghadam, Markus Borg, and Seyed Jalaleddin Mousavirad. Deeper at the sbst 2021 tool competition: Adas testing using multi-objective search. In *2021 IEEE/ACM 14th International Workshop on Search-Based Software Testing (SBST)*, pages 40–41. IEEE, 2021.

[31] Kai Petersen and Emelie Engström. Finding relevant research solutions for practical problems: The serp taxonomy architecture. In *Proceedings of the 2014 International Workshop on Long-Term Industrial Collaboration on Software Engineering*, WISE '14, page 13–20, New York, NY, USA, 2014. Association for Computing Machinery.

[32] Geoff Pleiss, Tianyi Zhang, Ethan Elenberg, and Kilian Q Weinberger. Identifying mislabeled data using the area under the margin ranking. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 17044–17056, 2020.

[33] Vincenzo Riccio, Gunel Jahangirova, Andrea Stocco, Nargiz Humbatova, Michael Weiss, and Paolo Tonella. Testing machine learning based systems: a systematic mapping. *Empirical Software Engineering*, 25(6):5193–5254, 2020.

[34] Sergio Rico, N. Ali, Emelie Engström, and Martin Höst. Guidelines for conducting interactive rapid reviews in software engineering – from a focus on technology transfer to knowledge exchange. 2020.

[35] Sebastian Schelter, Dustin Lange, Philipp Schmidt, Meltem Celikel, Felix Biessmann, and Andreas Grafberger. Automating large-scale data quality verification. *Proceedings of the VLDB Endowment*, 11(12):1781–1794, 2018.

[36] D. Sculley et al. Hidden Technical Debt in Machine Learning Systems. In *Proc. of the 28th Int'l Conf. on Neural Information Proc. Systems*, pages 2503–2511, 2015.

[37] Salman Sherin, Muhammad Uzair khan, and Muhammad Zohaib Iqbal. A systematic mapping study on testing of machine learning programs, 2019.

[38] Qunying Song, Markus Borg, Emelie Engström, Håkan Ardö, and Sergio Rico. Primary studies, January 2022.

[39] Margaret-Anne Storey, Emelie Engstrom, Martin Höst, Per Runeson, and Elizabeth Bjarnason. Using a visual abstract as a lens for communicating and promoting design science research in software engineering. In *2017 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, pages 181–186, 2017.

[40] Yuchi Tian, Kexin Pei, Suman Jana, and Baishakhi Ray. Deeptest: Automated testing of deep-neural-network-driven autonomous cars. In *Proceedings of the 40th international conference on software engineering*, pages 303–314, 2018.

[41] Andrea C Tricco, Chantelle M Garritty, Leah Boulos, Craig Lockwood, Michael Wilson, Jessie McGowan, Michael McCaul, Brian Hutton, Fiona Clement, Nicole Mittmann, et al. Rapid review methods more challenging during covid-19: commentary with a focus on 8 knowledge synthesis steps. *Journal of clinical epidemiology*, 126:177–183, 2020.

[42] Jonathan Uesato, Ananya Kumar, Csaba Szepesvari, Tom Erez, Avraham Ruderman, Keith Anderson, Nicolas Heess, Pushmeet Kohli, et al. Rigorous agent evaluation: An adversarial approach to uncover catastrophic failures. *arXiv preprint arXiv:1812.01647*, 2018.

[43] Guillaume Vidot, Christophe Gabreau, Ileana Ober, and Iulian Ober. Certification of embedded systems based on machine learning: A survey. *arXiv preprint arXiv:2106.07221*, 2021.

[44] Andreas Vogelsang and Markus Borg. Requirements Engineering for Machine Learning: Perspectives from Data Scientists. In *Proc. of the 27th Int'l Requirements Engineering Conf. Workshops*, pages 245–251, 2019.

[45] Jie Zhang, Earl Barr, Benjamin Guedj, Mark Harman, and John Shawe-Taylor. Perturbed model validation: A new framework to validate model relevance. 2019.

[46] Jie M Zhang, Mark Harman, Lei Ma, and Yang Liu. Machine learning testing: Survey, landscapes and horizons. *IEEE Transactions on Software Engineering*, 2020.

[47] Mengshi Zhang, Yuqun Zhang, Lingming Zhang, Cong Liu, and Sarfraz Khurshid. Deeproad: Gan-based metamorphic testing and input validation framework for autonomous driving systems. In *2018 33rd IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pages 132–142. IEEE, 2018.

PAPER VI

EXPERIENCES FROM CONDUCTING INTERACTIVE RAPID REVIEWS – TWO INDUSTRIAL CASES

Sergio Rico, Nauman bin Ali, Emelie Engström and Martin Höst. Under review in a journal. 2023.

Abstract

Context: Evidence-based software engineering (EBSE) aims to improve research utilization in practice. It relies on systematic methods to identify, appraise, and synthesize existing research findings to answer questions of interest for practice. However, the lack of practitioners' involvement in these studies' design, execution, and reporting indicates a lack of appreciation for the need for knowledge exchange between researchers and practitioners. Furthermore, the results of such systematic literature studies often lack relevance for practice.

Objective: Previously, we proposed interactive rapid reviews as a tool to foster knowledge exchange between industry and academia. In this study, we report the experience of using this proposal in two cases.

Method: We analyzed the conduct of two interactive rapid reviews by two different groups of researchers and practitioners. We collected data through interviews, and the documents produced during the review (like review protocols, search results, and presentations). The interviews were analyzed using thematic analysis.

Results: We report how the two groups of researchers and practitioners performed the interactive rapid reviews. We observed some benefits, like promoting dialogue and paving the way for future collaborations. We also found that practitioners entrusted the researchers to develop and follow a rigorous approach and

were more interested in the applicability of the findings in their context. The problems investigated in these two cases were relevant but not the most immediate ones. Therefore, rapidness was not a priority for the practitioners.

Conclusion: The study illustrates that interactive rapid reviews can support researcher-practitioner communication and industry-academia collaboration. Furthermore, the recommendations based on the experienced challenges and benefits from the two cases complement the detailed guidelines researchers and practitioners may follow to increase interaction and knowledge exchange.

1 Introduction

As an applied research area, software engineering research relies on a deep understanding of industrial software engineering practices to produce relevant and applicable knowledge. Without such understanding, researchers risk focusing on irrelevant aspects of existing problems [18, 22], missing necessary information [1], providing solutions that do not apply nor are generalizable to other contexts, or presenting results in a complicated way that is difficult for practitioners to access and interpret [14].

In many cases, a deep understanding of industrial practices requires close collaboration with industry. Garousi et al. identified industry collaboration [20] and the use of appropriate research approaches [35] as two of the most frequent improvement suggestions for increasing the relevance of software engineering research [18]. To motivate such collaborations, both parties need to benefit from them.

However, secondary studies aiming to inform practice are often conducted without any participation of practitioners. At best, researchers start with a need from practice, convert it into an answerable research question, identify, critically appraise and aggregate available evidence to help answer the question, and document the approach and findings in research papers [24].

The lack of practitioners' involvement in systematic literature studies suggests that researchers under-appreciate the contextual nature of software engineering findings. The underlying assumption is that knowledge can be transferred or communicated to practice at the end of the literature studies. One consequence of this approach is that the relevance of the findings of a literature review is relatively low for practitioners [8].

We believe that software engineering knowledge is socially constructed [21] and context bound [13]. Therefore, to overcome this limitation, we propose to extend the rapid review guidelines [9] to also include practitioners in the process [30]. Rapid reviews are systematic reviews intended to support decision-making under time constraints [9, 16], which is often a requirement in practice.

We refer to our proposed method as an interactive rapid review, (IRR) [30]. It extends the rapid review guidelines by 1) prioritizing *knowledge exchange* between

researchers and practitioners, 2) identifying and focusing on *practitioners-relevant* needs, and 3) identifying opportunities for *industry-academia collaboration* during the review [30]. Thus, in an IRR, researchers and practitioners work together to answer practical problems relevant to practitioners based on research results.

In this study, we investigate the practical application of IRRs in two independent cases of industry-academia collaboration. We studied how the IRR guidelines were applied and perceived and collected information about expectations, the usefulness of our proposal, results, and the experiences of researchers and practitioners involved. Based on the two cases, this paper presents the following contributions:

- A description of how the teams conducted the IRRs.
- Identification of the benefits and the challenges when using IRRs.
- Further recommendations for researchers and practitioners when conducting IRRs.

The remainder of this paper is structured as follows. Section 2 presents background and related work. Section 3 presents the steps of an IRR. Section 4 describes the method followed in this study. In Section 5, we present the study's results. After that, we present some recommendations for conducting future IRRs in Section 6. Section 7 discusses the results, and Section 8 concludes the paper.

2 Background and Related Work

2.1 Secondary studies in software engineering

Researchers in software engineering have widely adopted the use of secondary studies [4] as a means to synthesize software engineering knowledge. However, these studies have mainly been used in academic environments, and to identify gaps in research [17], and are criticized for the lack of industry-relevant results [12]. There is a need to connect secondary studies with practice. A few improvements have been suggested to make the presentation of the results more meaningful for teachers and practitioners [7]

Some voices in the software engineering research community have claimed that secondary studies need to connect more with practice [25]. There are a few examples of secondary studies that have involved practitioners (see e.g., [2]) however, none of the existing guidelines sufficiently incorporate interaction with practitioners. For this reason, we propose to conduct rapid reviews interactively with practitioners.

2.2 Rapid reviews in software engineering

Rapid Reviews are a well-known approach in medicine for synthesizing research findings under time restrictions. In software engineering, some researchers have also proposed using rapid reviews. For example, Cartaxo et al. [9] proposed using rapid reviews to provide decision-makers information quickly.

Some examples of recent rapid reviews conducted by researchers in software engineering include a rapid review on migrating from monoliths to microservices architecture [28], and a rapid review on testing of context-aware software systems [26].

IRRs follow similar steps as rapid reviews [10]. However, the overarching goal and, consequently, the role of practitioners and researchers in the different steps are different. In rapid reviews, the goal is to provide a quick answer to a specific question, while in IRRs, the additional goal is to maximize the opportunities for knowledge exchange between researchers and practitioners, acknowledging the context-dependant nature of software engineering.

3 The steps of an interactive rapid review (IRR)

In this section, we briefly revisit the main steps involved in an IRR. The steps of an IRR are shown in Figure 1.

1. *Prepare the review:* The first step involves forming a team of researchers and practitioners participating in the review. The researcher leading an IRR presents the general aim of an IRR, and the typical process, timeline, and expected time commitments. Next, the team needs to agree on the expectations, the extent of involvement, and the responsibilities of researchers and practitioners. The IRR topic emerges from the industry's specific needs and relevant aspects of their context, e.g., current software engineering practices. At the end of this step, a team is formed, and they have identified preliminary information needs.

2. *Identify research questions and develop the IRR protocol:* The second step involves more detailed planning, where the initial research questions and a protocol are described. Researchers and practitioners refine the research questions as the understanding of the practitioners' context improves. This is an iterative process during which the review team develops an understanding of the terminology and domain jargon. It may take time to develop a consensus on the research questions and the scope of the review. Once the research questions are sufficiently clear, the review team further articulates decisions like the search strategy, inclusion and exclusion criteria, and approach to conducting the analysis. The protocol is a living document that is revised and updated throughout an IRR.

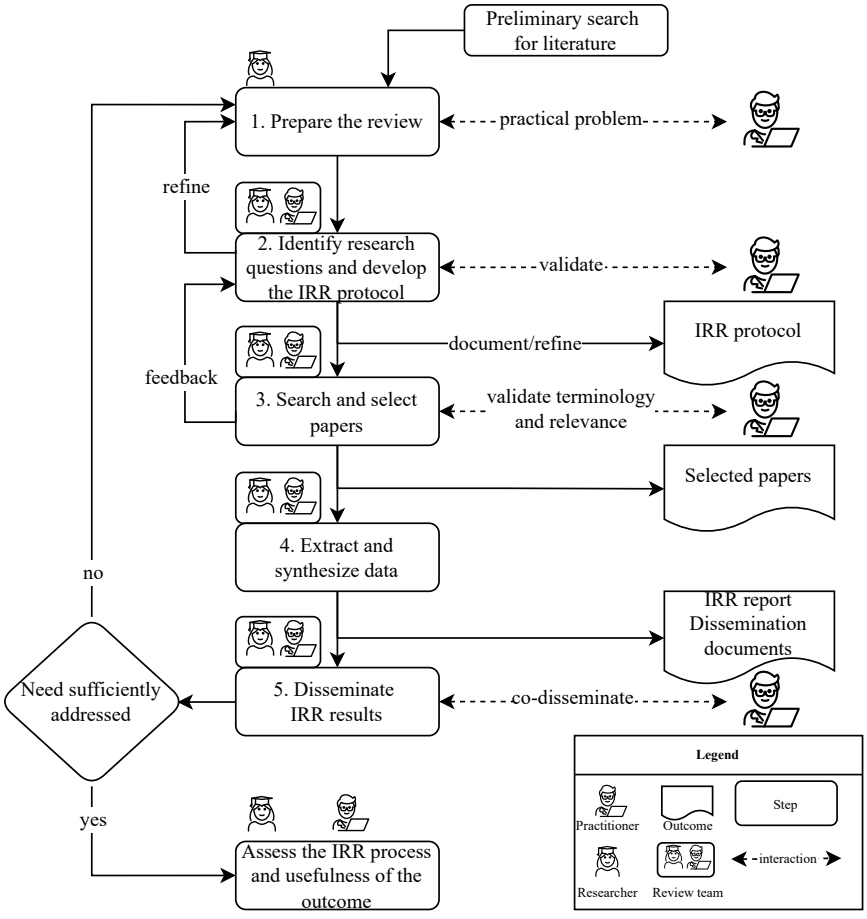


Figure 1: Steps for conducting an IRR

3. *Search and select papers:* In the third step, the search and selection of papers are performed. Several decisions to ensure a rapid literature review are taken in this step at the cost of completeness of coverage. Furthermore, the team develops criteria and a shared understanding of what papers are considered relevant, in particular, taking the practitioner's perspective and context into consideration.

4. *Extract and synthesize data:* Step four is about extracting data from the included papers and synthesizing the results and findings. Preparing the reports and templates to be filled with the results may help to save time and focus on the synthesis. These reports may include summaries, slides, infographics, etc. To make the results more accessible for practitioners, the review team can consider creating narrative synthesis [29] and provide summaries [7].

5. *Disseminate IRR results:* The final step in the IRR is to disseminate the IRR results. The dissemination actions are designed to communicate the results to practitioners and researchers. When sharing the results with the practitioners, an active role of the practitioners involved in the team may add more context and increase the interest in the findings.

4 Research methodology

In this study, we aim to collect the experience of using the proposed IRR approach in practice and identify further improvements in the guidelines. For this purpose, we posed the following research questions:

1. **How did the teams comprising researchers and practitioners conduct the IRRs?**

   To answer this question, we collect information about how two teams comprising both researchers and practitioners followed the steps proposed to conduct an IRR [30]. The approach is briefly summarized in Section 3. We describe how each team applied the guidelines.

2. **What are the benefits and challenges when conducting IRRs?**

   With this research question, our goal is to collect the benefits and challenges observed during the conduct of the reviews.

We attempted to answer the above research questions in a two-case exploratory case study. In the two cases, the IRRs were conducted by two independent groups of researchers and practitioners.

There were three main types of participants in the reviews. Practitioners ($P_{i,j}$, i.e., j:th practitioner in review i) participate from outside the university for the

purpose of the review, researchers ($R_{i,j}$) participate from the university for the purpose of the review, and meta-level researchers (M_j) participate from the university to conduct the research presented in this paper.

This IRR was conducted as a collaboration between one researcher and one practitioner from case company 1. The company is a multinational company developing software and hardware in the area of networking and communications.

The review initially focused on exploring the criteria for selecting open-source and closed-source software tools in software development. The goal of the IRR was to identify important factors and challenges in selecting software tools and to provide recommendations for improving the selection process at the company.

The following participants were involved in the review:

- Practitioner $P_{1,1}$: An experienced practitioner working with "technology studies", which involves understanding current research
- Researcher $R_{1,1}$: A senior researcher from Lund University, active in the area of Requirements Engineering, with experience from conducting secondary studies
- Meta-level researcher M_1: A researcher aiming to facilitate and support the review, and collect data for the purpose of the research presented in this paper. M_1 is the first author of this paper. M_1:s main research interest in industry-academia collaboration in software engineering research.

4.1 Case-MLTest: Machine learning testing

This IRR was conducted as a collaboration between researchers and practitioners from case company 2, a manufacturer of network cameras for physical security and video surveillance industries. The review's objective was to understand more about ML testing, and identify research results of interest to the case company.

The following participants were involved in the review:

- Practitioner $P_{2,1}$: A developer with a background in mechatronics and mathematics, who has worked at the company for about five years with machine learning applications.
- Practitioner $P_{2,2}$: A researcher employed by the case company, with a Ph.D. in mathematics, who is currently continuing research in the same area as their Ph.D. topic and applying their research in product development.
- Practitioners $P_{2,3}$ and $P_{2,4}$ also participated in the review from the case company. However, they were not interviewed for this study since they only attended the meetings and were less actively involved in the steps.
- Researcher $R_{2,1}$: An experienced researcher in the area of machine learning in software engineering

- Researcher $R_{2,2}$: A Ph.D. student in the area of software testing and machine learning. The review is relevant to the researcher's thesis work.
- Researcher $R_{2,3}$: Experienced researcher in the area of software testing. The third author of this paper.
- Meta-level researcher M_1 with the same role as in Case-SoftSelection.

Researchers $R_{2,1}, \ldots R_{2,3}$ have been involved in traditional literature reviews mainly with academic participants prior to this IRR. This means that the concept of a systematic literature review is not new to them. The conducted IRR was presented in a conference publication [32].

4.2 Data collection

In this study, we used interviews for data collection. The interviews were semi-structured [31], i.e., we identified the main themes to cover in the interviews and prepared several questions to guide the exploration of these themes. However, we adapted to the conversation during the interview, which meant adapting specific questions and their order, and asking additional questions. For example, if the interviewee jumped forward to an interesting topic, they were not interrupted.

The complete list of questions can be found as additional material.[1]

In Case-SoftSelection, interviews were conducted both before and after the review. In Case-MLTest, interviews were conducted only after the review.

Researcher M_1 participated in both reviews to be able to follow the process from beginning to end. Data collection before the review in Case-MLTest was mainly conducted by M_1 participating in and observing the planning of the review.

In both reviews, researcher M_1 supported the review team by answering questions concerning the IRR method.

After the reviews, researchers and practitioners from both review teams were interviewed to understand their experiences. We interviewed them about how they conducted the reviews, their experiences collaborating with practitioners/researchers in this work, and their thoughts about the obtained results. We asked them about the expected vs. actual results, contributions to the company, and the IRR's contribution to the long-term objective of learning how to collaborate. Additionally, we inquired about their views on research evidence and what makes a research paper good in their opinion.

Besides the interviews, we had access to the project files, where the researchers kept track of the steps and stored the files related to their project. These files provided additional information that we used to confirm and compare the answers given in the interview. For example, we looked at the details of the IRR, such as the search string, the inclusion/exclusion criteria, the number of papers found, and the presentation slides.

[1] LINK

4.3 Coding and analysis

The eight interviews were transcribed and coded using QSR International's NVivo qualitative data analysis software for coding and analysis. Below, we describe the codes used in the analysis and the steps followed to analyze the data.

Coding levels

We used three levels of coding to organize and analyze the data. The first level (level-1) acted as an index to map the chunk of the interview to one or more of the aspects of interest, such as benefits, challenges, and steps. The second-level (level-2) grouped findings, and the third level (level-3) codes described or qualified the second-level codes. For example, suppose the level-1 code was about benefits. In that case, the level-2 code could be about a specific benefit (e.g., mutual understanding), and the level-3 codes could describe unique aspects of this benefit or capture relevant findings (e.g., views alignment, define common terminology, other's perspective). The initial level-1 codes were:

- Case Description: We used this code to index chunks of the interview that we could use to describe the case. Some level-2 codes we were interested in include background, profiles of researchers and practitioners, initial views of each other, experiences working with industry/academia, and participation in secondary studies.

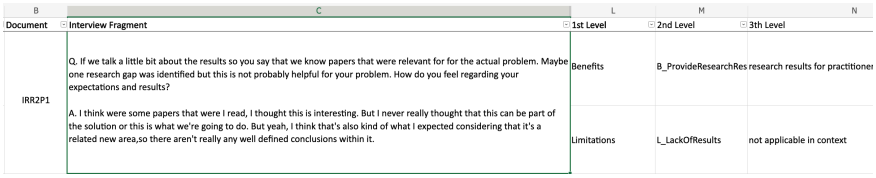
- Expectations: We captured with this code the interviewee's expectations. With level-2 codes, we classified the expectations according to who had the expectations (researchers or practitioners) and the type of expectation (e.g., empirical validation, exchange, new knowledge).

- Steps (code for research question 1, i.e., about the conduct of an IRR): With this code, we indexed the chunks of the interview where the interviewees described how they conducted the IRR. The level-2 codes are the steps of the IRR (see Section 3).

- Benefits and Challenges (related to research question 2, i.e., benefits and challenges of IRRs): With this code, we indexed the chunks of the interview where we identified positive aspects brought by the IRR (i.e., benefits), or challenges in conducting an IRR. Each of the benefits was coded under a level-2 code. The initial set of level-2 codes were the expected benefits and challenges based on the experiences from rapid reviews in medicine [23] and our experience conducting secondary studies in software engineering.

The initial set of codes evolved during the coding and analysis process [11]. For instance, we added one more level-1 code related to the outcomes of the IRR.

Similarly, level-2 codes were added when we identified new benefits or challenges. The final set of codes is available as additional material.[2]



B	C	L	M	N
Document	Interview Fragment	1st Level	2nd Level	3th Level
IRR2P1	Q. If we talk a little bit about the results so you say that we know papers that were relevant for for the actual problem. Maybe one research gap was identified but this is not probably helpful for your problem. How do you feel regarding your expectations and results?	Benefits	B_ProvideResearchRes	research results for practitioner
	A. I think were some papers that were I read, I thought this is interesting. But I never really thought that this can be part of the solution or this is what we're going to do. But yeah, I think that's also kind of what I expected considering that it's a related new area,so there aren't really any well defined conclusions within it.	Limitations	L_LackOfResults	not applicable in context

Figure 2: Example of part of the spreadsheet row

Coding steps and analysis

We have followed the steps described below to code the interviews:

1. In NVIVO, the first author coded the chunks of the interview (i.e., questions-answers) with the level-1 codes.

2. To facilitate the exchange within the research team, the chunks of the interview were exported to a spreadsheet document (Fig. 2) with the following columns:

    - Check validity: Used to indicate whether other researchers reviewed the chunk.
    - Document: An identifier that represents the IRR and the interview. For example, IRR1P1_Before means the first interview with the practitioner in Case-SoftSelection.
    - Interview fragment: The verbatim text of the interview. Each cell contained a question-answer pair.
    - First-level code: The level-1 code was used to code the interview fragment. One interview fragment can be coded under multiple level-1 codes.
    - Second level code: As described above, the level-2 code groups the findings. One first-level code can be coded under multiple second-level codes.
    - Third level Code: The description/qualification of the 2nd level code. One second-level code can be coded under multiple 3rd level codes.
    - One column for recommendations
    - One column for comments from each author

[2]LINK

3. The first author coded the interviews.
4. The third and fourth authors reviewed the coding in the spreadsheet. Disagreements were discussed and resolved to reach a consensus.
5. The second author reviewed the coding.
6. The first author addressed the comments and suggestions from the reviews.

While the coding was being done, the research team met weekly to discuss the coding process and the evolving codes. Once the interviews were coded, the codes related to the steps were used to develop the narrative about how the IRRs were conducted (as reported in Section 5.1). Similarly, the codes related to the benefits and challenges (2nd-level) were used as a basis for synthesizing and summarizing the findings reported in Section5.2.

The set of recommendations was developed incrementally. The first set of recommendations was derived from explicit coding (i.e., recommendations expressed by the interviewees). Furthermore, some recommendations are responses to identified challenges. Finally, all authors reviewed and discussed the final set of recommendations in a meeting and revised them while writing the manuscript.

5 Results and analysis

In this section, we present the results of the coding and analysis. The section has three subsections, outlined, in accordance with the research questions. Section 5.1 describes how the IRRs were conducted. Section 5.2 includes the benefits of conducting IRRs as a tool for researchers-practitioners knowledge exchange. Section 5.3 describes aspects found challenging for the review teams when conducting IRRs.

5.1 Conducting IRRs

In this section, we describe step by step how the two review teams conducted the IRRs based on the guidelines. Table 1 shows a summary of how the IRRs were conducted. We used two styles to format the quoted text in the subsections. Text quoted from the guidelines is marked by a sidebar and takes an entire paragraph, while text from the interviews is included in the text within quotation marks.

Preparation

> "*The first step involves forming a team of researchers and practitioners participating in the review. The researchers who lead the IRR need to present the aims, process, timeline and expected commitment to the group.*"

Table 1: Summary of how IRRs were conducted

Case	Case-SoftSelection	Case-MLTest
General Topic	Software Component Selection	Testing Machine-Learning Systems
Practitioners' expectations	Pilot industry-academia communication, explore research findings on criteria for software selection	Explore research on machine learning testing
Researchers' expectations	Pilot industry-academia communication, develop a collaborative network	Explore machine-learning testing in practice, networking
Team	1 Researcher ($R_{1,1}$), 1 Practitioner ($P_{1,1}$)	4 Practitioners ($P_{2,1}$, $P_{2,2}$, $P_{2,3}$, $P_{2,4}$), 4 Researchers ($R_{2,1}$, $R_{2,2}$, $R_{2,3}$, M_1)
Research questions	What criteria are relevant for the company to consider when selecting a SE tool or component?	General question: How to test the dataset?
Search	Key-words search in Scopus	Based on 3 Systematic Literature Reviews
Papers found	147 primary studies. 27 papers coded	180 primary studies mapped to 10 challenges. 5 of the papers mapped to the general question
Analysis	Extracting criteria from papers, and exchanging with practitioners	Developing a taxonomy of ML-testing based on literature and one case context
IRRs outcomes	Preliminary model for component selection	Examples of problem-solution matches in terms of technological rules [33] (non-extensive)
Dissemination	Share preliminary results with practitioners, working sessions to build a model, research paper	Share papers with practitioners, share findings with company representatives
Post-IRRs	Second iteration to extend the search and validate the model	Master thesis proposal, new projects

Since the level of prior knowledge about IRR differed in the two cases, the need for an introduction to the method varied. In Case-SoftSelection, the researcher leading the review was new to the concept of IRRs, while the researcher leading Case-MLTest had been involved in developing the IRR guidelines. Thus, we provided the IRR guidelines to the researcher $R_{1,1}$ to lead Case-SoftSelection. Moreover, we provided some material to support the presentation of the ideas, initial planning, and a document to develop the protocol. In Case-MLTest, the third author of this paper $R_{2,3}$ was part of the team conducting the IRR.

In both cases, the review teams were formed after the initial discussions, when the topic was agreed on. In Case-SoftSelection, there were no changes on the practitioners' side, while the research team was formed based on the emerging topic. Initially, with only one researcher, $R_{1,1}$, one more researcher (second author in this paper) joined the team after one iteration of search and selection. In Case-MLTest, three more practitioners with different but relevant roles were added to the team, while the researchers' team remained the same.

> "*Next, the team needs to agree on the expectations, the degree of involvement, and the responsibilities of researchers and practitioners. Ideally, the IRR topic*

> *emerges from the industry's specific needs, and relevant aspects of the context and current practices are introduced to the team.*"

In both cases, the motivation for conducting the IRRs was to explore possibilities for industry-academia collaborations on new topics. Thus the industry's specific needs were not of the highest priority.

The idea to conduct Case-SoftSelection came when a practitioner $P_{1,1}$ reached out to a group of researchers wanting to explore ways to work together. The third author of this paper took part in these initial discussions and suggested conducting an IRR as a first step. At this point, the main interest of conducting the review was to explore potential topics of mutual interest and, to some extent, test the experience of working together. In Case-SoftSelection, Practitioner $P_{1,1}$ presented an initial list of topic ideas: "*I just said that I have a lot of ideas like 10 or 15. I said pick something that is interesting like this, or that*" - [$P_{1,1}$].

Case-MLTest was, on the other hand, initiated by researchers interested in extending their network of industry contacts working with machine learning applications. Previously, the researchers $R_{2,1}$, $R_{2,3}$ developed a platform for machine learning testing approaches [5]. Researcher $R_{2,1}$ approached an industry contact $P_{2,2}$ within that network, who became the primary company contact for this review. The researchers' initial idea was to explore aspects of testing machine learning in practice. The practitioners were also interested in joint efforts with researchers on the same topic. Here the broad topic was suggested by researchers and found to be relevant enough for the practitioners, as described by $R_{2,1}$: "*They had the application and you always want to test your applications. It's not like they have immediate problems in this area.*" - [$R_{2,1}$]

> "*At the end of this step (preparation), it is expected to have an IRR team, and an initial description of the information needs, including a topic and context variables.*"

The two review teams that were formed were of different sizes. Case-SoftSelection initially involved one practitioner and one researcher, while Case-MLTest involved four researchers and four practitioners. In both cases, the information need was described at a very high level of abstraction, "How to test ML applications" and "How to select tools and software components". Even though the information needs were general at this point, and one was more specific than the other, they described the topics of interest at that time. These needs were the starting point for starting the search and specifying the research questions. The particular elements of the context that influence the review were not identified at this stage but evolved through interaction between researchers and practitioners during the reviews in both cases.

Identify research questions and develop the IRR protocol

> "*The second step consists of more formal planning where the IRR research questions are defined and an initial protocol to conduct the IRR is started.*"

After two meetings in Case-SoftSelection, the researcher and practitioner agreed on a more precise idea about the topic to explore and the next steps in the IRR. Then, the review team comprised the researcher $R_{1,1}$ and practitioner $P_{1,1}$. The review's main topic was the selection of software components.

In Case-MLTest, during a meeting with the practitioners, the researchers shared the overview and the list of topics. Then, the practitioners presented their work and products supported by machine learning and challenges. After the meeting, the two groups agreed on the broad topic of the IRR, testing of ML applications.

> "*Defining research questions with practitioners is an iterative process that requires understanding the practitioners' context, practices, challenges, and terminology.*"

Since, in none of the cases, any specific research questions were decided upfront, some initial effort was spent identifying questions of high relevance for everyone involved. In Case-SoftSelection, after a couple of initial conversations between $P_{1,1}$ and $R_{1,1}$, they came up with a preliminary idea about exploring the selection of tools and software components. The researcher reviewed papers related to the first research question about the criteria for selecting software components. Meanwhile, the practitioner also identified criteria, not from research papers but by reflecting on their own experience and consulting their colleagues. In Case-MLTest, an initial brainstorming meeting was held. Before this meeting, the researchers developed a preliminary taxonomy of state-of-the-art ML-testing using the SERP-taxonomy architecture [15]. The taxonomy served two purposes, to present a general overview of the published literature to the practitioners and guide the discussions about the practitioners' context and needs. This meeting resulted in a list of potential research questions, which were then ranked independently by everyone involved in the meeting (four researchers and four practitioners) to select the most relevant questions. Based on the ranking, the first research question was formulated about data and input testing, i.e., how to test the data.

> "*Then, it may take time to develop agreements about the research questions and related terminology.*"

As stated above, in both cases, it required several interactions, in terms of meetings, workshops, and offline communication, to define the research questions for the review. $R_{1,1}$ pointed out that provided presentations, templates, and checklists were a help to communicate expectations within the team:

"*Since we all know what a literature review is. Having the presentation slides and the template for having different things to fill in made it very clear. These are the things we need to agree, and it was good to present them to the practitioner to*

get them to understand what the method was about. And to try and get the scope nailed down. I think that was our biggest challenge at the beginning, to have something that was a reasonable scope that was clear and sort of not all over the place. So I would say that having these templates helped, but then, of course, the discussion still has to be there and you have to get the practitioner into the little box that is easier for us to handle" - [$R_{1,1}$]. Furthermore, the background section in the template (part of the material initially to $R_{1,1}$) showed to be helpful in validating the problem understanding with the case company. $R_{1,1}$ filled it out during the initial discussions to develop a problem understanding. This was then sent to $P_{1,1}$ to confirm the view and fill in the gaps. In both cases, the final set of review questions was exploratory. Furthermore, time limitations prevented the exploration of the initial broad topics, but questions were refined during the reviews.

> "*Once the research questions are defined, the review team may develop an initial version of the review protocol. The review protocol contains information about the search strategy, inclusion and exclusion criteria, approach to conducting the analysis, and the decisions made along with the review. Besides, the protocol is updated with the progress and result of the following steps.*"

Researcher $R_{1,1}$ did not use the template we provided for the IRR protocol. In their view, the document did not help accomplish the IRR faster because it included many details to fill in. Instead, they kept track of the steps, search strings, inclusion criteria, decision, and search details in auxiliary files according to their own preferences. These files were stored in a project repository where the researcher kept track of review steps, meetings, advances in the review, search results, bibliography files, document drafts, and reflections and suggestions about the process. One of the reasons for researcher $R_{1,1}$ to keep track of the IRR in detail was to be prepared if the results would be used for an academic publication. The review team agreed that they would work iteratively, and the initial goal was to build a model that could summarize and synthesize their findings. In Case-MLTest, the researchers started developing the research protocol based on the template we provided. The protocol specified the research questions, search strategy, and inclusion criteria. An additional document kept track of the work plan, activities, roles, and responsibilities.

Search and select papers

> "*In the third step, the search and selection of papers are performed. Several decisions to ensure a rapid literature review are taken in this step at the cost of completeness of coverage. Furthermore, the team develops criteria and a shared understanding of what papers are considered relevant for the IRR.*"

In both cases, the researchers mainly searched and selected papers. They had previous experience conducting systematic literature reviews and followed similar principles. They kept track of the process and documented the decisions that were

made. The practitioners did not have any opinions about the process for finding the papers and were more interested in finding something applicable in their context than ensuring extensiveness in the search: "*I don't really know how much time they spent looking for it, so it's hard to know if it would be possible to look more or not. I'm quite confident that they spent more than we could do from our side. So I still think it is very valuable, and I trust their opinion enough not to spend more time myself on it if they come to some conclusion. I would say.*" - [$P_{2,1}$].

In Case-SoftSelection, Scopus[3] was used to search for literature. Search and selection were made in three iterations while defining the final scope within the review team. In Case-MLTest, the search step was skipped since the researchers were aware of three recent literature reviews on the topic and used them as a starting point. Although they were confident in the rigor of the searches in those secondary studies, they were also aware that the field is active and it is possible to miss something. However, completeness was not the main priority " *Q. How important is being systematic vs. finding something applicable for them? A. Being extensive wasn't our priority. So we wanted to really have something applicable. Once you have something applicable, it is easier to start from that. Q. And for them? Do you think it is the same? A. Yes, they do not care about completeness.* " - [$R_{2,1}$]

In Case-SoftSelection, the review topic was not the main topic of the researcher's expertise. However, they were confident in the systematic review process.

In both cases, selecting the relevant set of papers was iterative and involved feedback from the practitioners. In Case-SoftSelection, the review team had meetings discussing their findings. In Case-MLTest, practitioners were involved in reading and commenting on papers of potential interest, which helped refine the inclusion/exclusion criteria. " *then they started doing the review, finding a bunch of articles. And then they sent us a few of them, and we looked at them. They were not very relevant. They talked about the training data and stuff that we've talked about. But not really for image training data. It was more general for other kinds of media. And I found after reading those first papers I couldn't really see how to apply those general techniques to image data. So then there was a second round where they added this criterion that we wanted to work with images or videos. And then, I found more relevant work.*" - [$P_{2,2}$]. After the first iteration in Case-SoftSelection, the review team realized that many of the papers were quite old, so they adapted the search strategy to find more recent research.

In both instances, there were an equal number of research publications that were screened (hundreds) and reviewed (30–40) studies. However, since the goal of Case-MLTest was to find an applicable technique rather than develop a more general theory, as in Case-SoftSelection, the procedure of excluding papers continued until only a handful of papers remained. The practitioners then evaluated the papers and provided feedback to the researchers about their relevance to their current problems. As a result, the inclusion and exclusion criteria were updated.

[3]Scopus is one of the largest abstract and citation databases of peer-reviewed literature: scientific journals, books and conference proceedings. (Retrieved from https://www.scopus.com, [3])

Extract and synthesize data

> "*Step four is about synthesizing the results and findings from the included papers. An idea to better communicate the findings is to design the reports and documents that will be used to share the results in advance. It is vital to ensure that the findings will be easy to follow for the practitioners. For that reason, it is suggested to use narrative synthesis and practitioner-friendly summaries. A recommended practice is to hold reaction meetings where the IRR team presents preliminary results to the team or an extended group of practitioners. The reaction meetings give feedback to the team and may inspire them on how to communicate the results.*"

The data analysis approaches differed between the two cases due to the somewhat different goals. Case-SoftSelection had a higher ambition of synthesizing results of a larger share of included papers. On the other hand, in Case-MLTest more effort was spent selecting relevant and applicable approaches for their case company's context.

Case-SoftSelection applied thematic coding to find answers to their research questions. The researcher derived an initial set of codes that evolved along with the coding based on the research question. The papers were coded using Nvivo. After clustering the codes, the outcome of the IRR was a list of criteria for selecting software components. On the other hand, the main focus of the coding in Case-MLTest was to improve the inclusion and exclusion criteria. As a result, the initial SERP-taxonomy [15] was extended with information retrieved from the included papers and the practitioners' feedback regarding their context. Based on the common taxonomy, researchers were forced to be explicit about the relevant details of the proposals.

Disseminate results

> "*The final step in the IRR is disseminating the IRR results. The dissemination actions are designed to communicate the results to practitioners and researchers. When sharing the results with the practitioners, the practitioners involved in the review team have an active role, e.g., when presenting or discussing the results, they may add more context and thus increasing the interest in the findings. In addition, even though it is not the primary goal, the researchers involved may be interested in communicating the findings to academic audiences through research papers.*"

In neither case did the researchers conduct specific actions to disseminate the results beyond the review team, e.g., within the organizations. However, in Case-MLTest, preliminary results were shared with practitioners from another business unit where the results were relevant. "*And I read maybe three of those articles. And one of them I passed on to another team that actually [~~they~~] evaluated [~~it~~]*

and compared it with a few other techniques that they were aware of." - [$P_{2,2}$]. After the reviews, both review teams also reported their results and experiences in scientific publications.

In both cases, preliminary results were presented by the researchers at different stages during the review. The manner in which these were presented was also influenced by the expectations of the practitioners. In Case-SoftSelection, the topic was more general, and the identified papers were more divergent. There the researcher extracted and synthesized contributions and presented a preliminary model of component selection (a taxonomy of criteria) to the practitioner. Researcher $R_{1,1}$ shared the list of criteria with practitioner $P_{1,1}$. They also shared some of the papers (actual pdf files) that were the most relevant to the topic. Then, jointly throughout a series of meetings and discussions (at least three meetings including a working session with a whiteboard), they integrated the criteria found in the research papers with the ones collected by the practitioner to produce a model for selecting software components. The model and research findings were not communicated to a larger group of practitioners within the company. Instead, the model was an input for a research study where the review team planned to complete and evaluate the model. In Case-MLTest, the practitioners were up to date with current research and were used to reading research papers. The preliminary results were presented in terms of selected papers and the researcher's reflections on potential inclusion and exclusion criteria.

The final results of Case-SoftSelection were reported as a short Powerpoint presentation showing the taxonomy and explanations of the identified criteria. In Case-MLTest visual abstracts [33] were created, summarizing the contributions of the five best problem-solution matches, and presented at the concluding review meeting.

5.2 Benefits

By benefits, we mean the positive impact of the IRRs on the researchers, practitioners, their relationships, and their organizations. A benefit could be experienced in different ways by different stakeholders. For example, obtaining results from literature in a structured way can be seen as a benefit for the practitioners if we assume that they usually do not conduct systematic reviews. On the other hand, for the researchers who typically perform systematic reviews, the benefit is the possibility of involving practitioners in the process and increasing the industry relevance of the reviews.

Before analyzing the benefits of conducting IRRs, let us discuss their context and preconditions. When analyzing the advantages of Case-SoftSelection for developing networks for collaboration, it is fair to say that from the beginning, an important motivation for $P_{1,1}$ was to find ways to collaborate with academia. The IRR approach seemed to be a way to start working on something concrete to find common topics of interest and build a relationship. According to $R_{1,1}$, $P_{1,1}$ was

more interested in the meta-level, i.e., finding ways to collaborate. So the overarching goal in Case-SoftSelection was to explore ways to collaborate. The researcher, $R_{1,1}$, was also interested in collaboration with industry, although not as a research topic explicitly finding practical means to work smoothly with industry. Thus, the Case-SoftSelection may be seen as a way to pilot and assess the feasibility of collaboration with academia.

Case-MLTest was motivated by researchers $P_{2,1}$ $P_{2,3}$ since they were working on a project about testing machine learning. They were interested in seeing things from industry practice and networking with practitioners in the field. According to $P_{2,1}$, the IRR allowed the participants to work on a specific problem and look at the horizon for future collaborative work. The topic interested them, and then they allocated resources and got involved in the review. In the long term, the Case-MLTest contributed to identifying common interesting topics, meeting potential new collaborators, and determining how they can complement each other to work together. As one of the results, the review team got an overview of the field that facilitates identifying opportunities for new studies.

From the interview material, several benefits were identified. They can be classified into four categories: (1) mutual learning, (2) overview of the field, (3) usage of research, and (4) future collaboration.

Mutual understanding

The advantages of researchers and practitioners working together are related to the parties gaining mutual understanding, the researchers learning from the industrial context, and the co-creation of knowledge. There were a few advantages identified linked to working together and thereby would not as easily be obtained in a researcher-only literature review.

In the Case-SoftSelection, the researcher and practitioner started the collaboration in this review and had different views of the topic. The researcher perceived that the practitioner was more interested in new findings, while the researcher put more emphasis on published findings, which may not be so new. On the other hand, the practitioner perceived that his involvement helped the researcher focus on relevant findings. "*So we spent a lot of time on that. I think that helps, and I think that's good because ~~then,~~ when you start to talk about actual issues, it's much easier.*" - [$P_{1,1}$]

Then, there is some indication that performing the IRR helped to focus on topics with industry relevance. As the researcher states "*Now we have our model and I sort of introduced the thought that we could sort of redo the literature review and then he was a bit more interested in yes that might be a good idea. And he had actually found an article that he was referring to and I read that was relevant.*" - [$R_{1,1}$]

Then, we can see indications that the joint work contributed to understanding each other's perspectives and advancing a common model that synthesizes the

practitioner's and researcher's perspectives and findings.

Overview of the field

The researchers thought that they obtained not only knowledge from the literature but also from the company. For them, one contribution is that they were able to understand the industrial perspective of the problem that they investigated: "*Well, I think all the way through the study, we have learned a lot about the topic that we are looking at. But I think the most significant for me from this rapid review with [Company] was that we actually knew what was interesting from the industry perspective*" - [$R_{2,2}$]

Even if the researchers did not know the review's topic in detail before the review starts, conducting the IRR was an opportunity to gain knowledge on the industrial perspective in context.

Usage of the results of the review

When it comes to the usage of the results of the review, it can be seen in different ways. Industry practitioners find it positive to get a general understanding of the research front. Even if they normally do not conduct systematic literature reviews, they find it positive to see confirmation of ideas and problems in the literature. "*...maybe you can get confirmation on your own ideas that it is like basic stuff, right? or do we totally diverge*" - [$P_{1,1}$] By contrasting the actual practices with the outcomes from research papers, practitioners get indications if their practices are roughly in line with other companies, or if they are behind other companies in a specific topic.

However, $P_{1,1}$ reflected that this contrast is not easy to do. Besides, it depends on the area. Overall, the participants found value in becoming aware of the research state of the art and having some clues to what scientists and other industries are working on. It should also be noted that the participants identified research literature before, but this way of identifying literature was good compared to other approaches (e.g., Twitter and ad-hoc search were mentioned). The industry representatives did not express high requirements on empirical evidence. Instead, they said, for example, that if they find one single relevant paper with a relevant solution for them, that would be valuable. On the other hand, they would be more cautious if the paper only present theoretical results and see that as a risk. They stated that it is positive if the paper includes evaluations, but if it does not require too much effort, they may try it out themselves in their context. It is also mentioned that when they find new solutions, they want to compare them quantitatively with metrics.

Develop Networks for future collaborations

The fact that new networks were formed was seen as a benefit for practitioners and researchers. It is seen as positive that they have learned how to collaborate and

found a process with meetings to manage the work. The researchers also emphasized the positive impact of gaining an understanding of the industrial context and problems. This makes it easier to formulate collaboration projects in the future. Actually, this IRR spawned a M.Sc. collaboration projects and ideas for future applications. One practitioner highlighted the usefulness of gaining insights into the interests of researchers from reading papers, as it made it easier to collaborate with researchers in the future."*I think so because it gives me much more insight into what they are interested in. So if I'm sitting with a problem in the future, then I feel that I have more knowledge of when would this actually be interesting to research, and then I could reach out to them and ask [if] there is something that you want to cooperate with.*" - [$P_{2,1}$]

5.3 Challenges

In the two cases analyzed in this study, we identified some challenges when conducting IRRs. By challenge, we refer to something that poses difficulties when performing the review. The challenges and limitations can be categorized into the following main classes:

- challenges related to roles i.e, researchers and practitioners not being aware of their responsibilities
- challenges related to the lack of results matching the needs and expectations of the review team
- challenges related to the timeliness of the reviews

Different roles

Conducting a literature review is demanding, and having researchers and practitioners collaborating in the loop poses even more challenges. Practitioners have their objectives related to the current challenge and directions in their companies, and researchers are interested in finding more generalized results, which is also reflected in the way primary studies are written.

We noticed in the two cases that the practitioners had little awareness of their role in the IRR. Even though the term and steps were introduced at the beginning of the IRR, and they were actively involved in the activities, they were unaware that they were participating in a different type of literature review and the steps to conduct it. This has a positive side, in that it does not burden the practitioners with research aspects they are unfamiliar with. However, it also has a negative side, since if the practitioners were more aware of the steps, they could be more involved in the process and relate the results to their own context. Additionally, if their experiences were positive, they could be more motivated to conduct IRRs in the future.

Another challenge observed was different expectations and goals, which affected how the team conducted the review.For researchers, conducting literature reviews is a regular research task, but practitioners can have other expectations. In Case-SoftSelection, there was a mismatch between the researchers' and the practitioners' expectations and goals. The researchers thought that the practitioners were not as interested in the generalized knowledge as the researchers were. Instead, the researchers believed that the practitioners were more interested in single. To overcome this challenge, the researchers followed the review protocol, which made it more relevant to conduct the review. "*I think it was very good to have something concrete to do to produce some output to talk about. Otherwise, I think my collaborator [...] has a lot of things to say for [their]self, so it has been good to have [the process] to get some information and some knowledge from the review.*" - [$R_{1,1}$]

Lack of results matching needs and expectations

In both cases, we noted that the papers found during the review did not precisely match the practitioners' needs and expectations. This challenge can also be seen as a result of different expectations between researchers and pratitioners. The identified articles were not genuinely meeting the expectations of the practitioners. There were three main types of mismatch between the papers found during the review and the practitioners' needs and expectations. One type of mismatch was that the practitioners had specific questions from their specialized field, and the papers found were not directly related to these questions. This can be challenging for the review team, as they may not be able to find the information they are looking for and may be disappointed with the results of the review.

The second type of mismatch was that the identified papers were older than expected by the practitioners. The practitioners may be looking for the most up-to-date information on a topic, and older papers may not be perceived as relevant or useful. However, it is important to recognize that research and practice have different paces, and what may be considered an up-to-date problem in practice could be considered a problem solved in research.

Finally, the papers can also be considered to be too "long-term" or too theoretical for application in the short term. Practitioners may be looking for more practical or applied information that they can use in their work immeditely. Theoretical or long-term papers may not be considered useful in these cases. One of the practitioners commented on this challenge: "*I think the things I found there [were] probably a lot more long-term than what I'm looking for. For example, one of the papers was interesting and perhaps we can use this at some point, but it's not something that I'm going to spend more time now because I think it will take [a] too long time to get payback for it.*" - [$P_{1,1}$]

That is, the identified papers were older, more long-term, and not in the practitioners' specific fields than they would have wanted.

Timeliness

We investigated the extent to which the participants saw the reviews as "rapid". One researcher ($R_{1,1}$) stated that the review was relatively rapid compared to previous reviews they had conducted. However, the researcher also had experiences from other secondary studies that were conducted in a shorter time frame. The practitioners (Case-MLTest) did not discuss the lead time as much as they discussed the effort involved them. They noticed that they had many other tasks in parallel, and they did not have to spend a lot of effort on this review. Regular meetings were seen as a good way to keep the work going.

In both cases, the researchers commented on the need to use tools that could support activities e.g, on the search, selection, managing references, and analysis. However, these tools were not used in any of the reviews.

6 Recommendations for conducting IRR

Based on the research findings, we suggest the following recommendations when researchers and practitioners conduct IRRs. These practices aim to make the IRR valuable for researchers and practitioners, increase the benefits, and address the challenges previously described. Table 2 lists the recommendations for each step proposed to conduct IRRs. Below we describe the recommendations in more detail following the steps of the IRR process.

6.1 Prepare

The team in charge of the IRR can **add expertise to the review team when needed**. Although the IRRs guidelines suggest forming a team of researchers and practitioners in the first step, the teams do not necessarily maintain the same composition throughout the process. For example, in the Case-SoftSelection, the team was formed by one researcher and one practitioner. However, the team was expanded for a second iteration to include another researcher when they realized that they would like to consider the views of a researcher with a different background and more specific experience.

Some characteristics of the practitioners or their company can make working with them easier. For instance, in Case-SoftSelection, the company was a large telecom company, and the practitioners were from a part of the organization that was in charge of the frontier of cutting-edge technologies. Consequently, as part of their work assignments, the practitioners focused on both advances in academic research and conducting applied research. In Case-MLTest, the company started as a startup incubated in a university environment and later was acquired by the company. These facts indicate, to some extent, an openness and willingness to work with researchers. Thus, when planning to conduct IRRs with industry partners, we suggest **identify cultural aspects that may influence a positive environment**

Table 2: Recommendations for conducting IRRs

Step	Recommendation
Prepare	Add expertise to the review team when needed Identify cultural aspects that can result in a positive environment for the IRR Involve practitioners with a research background or appreciation for research Identify the expectations and motivations of the practitioners to conduct the IRR Develop consensus on a goal that brings value to both sides. Select topics interesting for both researchers and practitioners.
Define RQ	Plan for a lot of initial interaction Focus on the practitioners' context Hold the input meeting where practitioners present the problem and context
Search and selection	Be prepared to handle results that could be considered, in principle, negative or incomplete Define small concrete outcomes Get feedback on the preliminary results
Extract and synthesize data	Thematic analysis may help to overcome terminology and context gaps Adapt the analysis to expectations and available literature
Disseminate results	Find means and ways to have practitioners' friendly communication Be aware of different terminologies Translate the results
IRR management	Remember that the guidelines suggest a flexible approach that can be adapted to the needs of each IRR. Keep the IRRs focused, rapid, and interactive. Have a shared repository and keep track of the decisions made while conducting the review Take the opportunity to learn to work together Meet, talk, and develop joint work sessions.

for the IRR. Identifying these aspects is not straightforward. However, researchers can be aware of some signals that suggest a willingness to collaborate, e.g., attitude during the meetings, openness to discuss current problems, and dedication of time. In summary, if the researchers understand the practitioners' cultural aspects and context, they can promote actions to develop a positive environment for the IRR, which can increase the chances of success and the potential benefits.

Involve practitioners with a research background or appreciation for research. In Case-SoftSelection and Case-MLTest, the practitioners involved had research experience and, therefore, some appreciation for research work and working with researchers. For instance, in Case-MLTest, the researchers signaled that the communication was much more straightforward since one of the practitioners had a Ph.D. and his work included contact with research. However, research background or appreciation for research includes not only Ph.D. holders. Practitioners who have co-supervised master's theses or participated in research studies can also be suitable candidates. For instance, as seen in Case-SoftSelection, the practitioner had a positive attitude toward working with researchers making it easier to start talking.

In our two cases, the researchers were in charge of planning the IRRs. Based on their experiences, we highlight some aspects to consider when planning the IRRs. At the very beginning of the IRR, the researcher should **identify the expectations and motivations of the practitioners to conduct the IRR.** These motivations differ slightly from the information needs explicitly related to the IRR topic. By the expectations and motivations, we mean the implicit reasons that encourage practitioners to work with researchers. These reasons may vary. Some examples are: getting feedback from different perspectives, hiring people, building a brand and reputation, getting help with a particular problem, or fulfilling a requirement from managers and staff. If the researchers know what motivates the practitioners to participate, then they can **develop consensus on a goal that brings value to both sides.**

IRRs are supposed to be a joint effort between researchers and practitioners. However, once the project starts, there is an inherent risk of losing commitment and willingness to work. Previous studies about industry-academia collaboration have pointed to the lack of relevance as one of the causes [19]. Moreover, in the case of secondary studies, it takes a long time between the problem formulation and the publishing of the results. Therefore, it is important that the **selected topics have to be interesting for both researchers and practitioners.** This can help to maintain engagement and commitment to the collaboration. Furthermore, one way to maintain the interaction and provide value to both sides along the review process is to plan to deliver small concrete outcomes. Examples of these small concrete outcomes include a summary of papers, a short list of papers, or a summary of the main findings. Providing regular, small deliverables can help to keep the IRR on track and ensure that both researchers and practitioners see the value of working together.

6.2 Define RQ

Getting information about the practitioners' context is critical when scoping the problem and formulating research questions. For this purpose, the team should **plan for a lot of initial interaction** to get a deep and common understanding of the review questions. In our two cases, it took several iterations to define the questions. The researchers' efforts in this stage should **focus on the practitioners' context**. In Case-MLTest **holding the input meeting where the practitioners presented the problem and contexts to the researchers** was an opportunity to discuss with the practitioners. These types of meetings allow the researchers to ask specific questions about the context and identify other variables that may be relevant to the problem. Furthermore, in our two cases, the teams used tools to support this step, like developing SERP-taxonomy [15] and ranking the topics according to the participant's interests.

6.3 Search and select papers

One of the critical aspects of our proposal for IRRs is to be in continuous contact with the practitioners. During the search and selection of papers, the practitioners' feedback is key to ensuring that the papers are relevant to the practitioners' context. The practitioners can be involved in refining inclusion/exclusion criteria by providing them with preliminary results as in Case-SoftSelection or by sharing papers to read and react to as in Case-MLTest. In the cases studied in this paper, we saw how defining small concrete outcomes was beneficial to promoting discussions about the findings. Instead of a single big outcome after a search and selection performed only for the researchers, the researcher had several meetings that promoted discussions about the findings. Similarly, we noticed the value of sharing and getting feedback on preliminary results in terms of example papers. In this case, the discussions between the researchers and practitioners contributed to refining the inclusion/exclusion criteria and identifying research gaps. Thus, the recommendations are to **get feedback on the preliminary results** and to **define small concrete outcomes** to ensure that the papers are relevant to the practitioners' context and keep the interaction on.

Researchers need to **be prepared to handle results that could be considered, in principle, negative or incomplete.** In Case-SoftSelection, when sharing the research papers with the practitioner, the practitioner felt that the papers were old and did not represent how the company worked. To address this, instead of sharing the papers directly with the practitioner, the researcher analyzed and summarized the results in other formats that the practitioner recognized as valuable. In Case-MLTest, among the papers the researchers found, none of them seem to apply to the practitioner's context. The reason for this lack of results was that the topic was recent both in research and academia. Therefore there were no techniques to apply directly to the practitioner's context, i.e., computer vision. However, some papers brought ideas and insights that could be useful for the practitioner's context.

Additionally, a very positive result of this Case-MLTest is that the researchers and practitioners formulated a master thesis directly on the topic, a Ph.D. project related, and plan to keep doing joint work around the specific topic. Thus, the results were not negative, but the researchers need to be prepared to handle similar situations and find ways to communicate results in a way that is useful and relevant to the practitioners.

6.4 Extract and synthesize data

Once the review team has selected a set of papers, the next step is to extract the information from the papers and synthesize it. As the review is supposed to be conducted in a short time, the guidelines suggest thematic analysis as a suitable method for this step. **Thematic analysis may help to overcome terminology and context gaps**. We have seen in Case-SoftSelection that the researchers adapted the extraction and synthesis to their expectations and available literature i.e., building a model for software component selection. On the other hand, in Case-MLTest, synthesis was unnecessary since the practitioners were interested in finding specific papers that matched their problem. Then, the synthesis was not their top need but papers they could implement in their context. Thus, the review team needs to **adapt analysis to expectations and available literature** to ensure that the results of the IRR are relevant and useful to the practitioners.

6.5 Disseminate results

The overall message when disseminating results is to be aware of different terminologies and contexts. It is also important to care about sharing valuable results for the practitioners' context. Note that the guidelines include one specific step named dissemination, although researchers often disseminate results to practitioners throughout the IRR process, e.g., when sharing preliminary results. In Case-SoftSelection, some terms only used inside the company were unfamiliar to the researchers. While in Case-MLTest, the topic of testing machine learning was relatively recent in industry and academia and lacked standard terminology. To disseminate results under these scenarios, the review team needs to **be aware of different terminologies** and use language that is understandable and relevant to both parties.

Another aspect to highlight is the need to translate results. By this, we mean to make the results understandable in connection with the practitioners' context. For example, in Case-SoftSelection, the key information was extracted, shared in joint work sessions, and summarized in presentations. The **translation of the results** is essential to ensure that the results are valuable and useful to the practitioners.

6.6 IRR management

The following recommendations are based on what we observed worked well in Case-SoftSelection and Case-MLTest related to the IRR process and working together. First, regarding the steps and activities, it is essential to **remember that the guidelines suggest a flexible approach that can be adapted to the needs of each IRR.** Therefore, as we saw in Case-SoftSelection and Case-MLTest, the review team can adapt to keep a flexible approach and adapt the steps to the needs of the IRR. Second, the questions that best fit the IRRs are narrow, specific, and related to current problems faced by the practitioners. Selecting narrow questions relevant to both parties is essential to **keep the IRRs focused, rapid, and interactive.** Third, we observed the advantages of **having a shared repository and keeping track of the decisions made while conducting the review**. The shared repository facilitated communication within the review team, and the memories supported the researchers when sharing the results and writing the academic papers. Fourth, researchers and practitioners should **take the opportunity to learn to work together**. This requires mutual understanding and respect and may not happen immediately. Finally, the researchers and practitioners in Case-SoftSelection and Case-MLTest recognized the importance of **meeting, talking, and developing joint working sessions** to foster knowledge exchange.

7 Discussion

This section discusses the study's results and the implications of the findings for future research and practice. The discussion is organized as follows. First, we discuss the study's results, specifically regarding the benefits and challenges of IRRs(7.1). Second, we discuss the implications of the findings for future research and practice(7.2). Third, we discuss the IRRs proposal in terms of interaction, rigor, relevance, and flexibility, which are the key aspects of the IRRs(7.3). Finally, we discuss the study's limitations and the threats to validity(7.4).

7.1 Benefits and challenges of IRRs

We found that the IRRs provided benefits individually for researchers and practitioners. Additionally, it contributed to building a relationship between them. On the individual level, researchers got chances to learn about the practitioners' context and problems, besides getting an overview of the field from the practitioners' perspective. On the other hand, practitioners got chances to take advantage of the research results and the researchers' expertise, allowing them to get a broader view of the research area and develop an awareness of the state of the research.

Building long-term relationships is a key factor for successful collaborations [34]. In this sense, the IRRs provided opportunities for researchers and practitioners to meet, discuss, and exchange knowledge. These interactions allowed the

researchers and practitioners to develop a shared understanding of the problem and the research area (Case-SoftSelection) and to build a shared vision of the problem-solution match (Case-MLTest).

The main challenge in the IRRs was related to the availability of results, and some other minor challenges were how to organize the roles and the time and effort required. The availability of results was challenging in both cases, but with different representations. In Case-SoftSelection, the results got the initial impression of being outdated, and in Case-MLTest, the results were not directly applicable in the practitioners' context. In both cases, the researchers and practitioners overcame the challenge by adapting the search and selection strategies and discussing the results. The early feedback from the practitioners helped the researchers to adapt the results to the practitioners' context and therefore increase the relevance of the results. Since lack of relevance for practice is a critique of traditional literature reviews [7], researchers conducting systematic literature reviews could incorporate similar strategies to overcome this challenge.

Challenges related to the organization and roles could be overcome by clearly understanding each participant's roles and responsibilities. The IRRs consumed little time in our two cases, but the time frame was distributed over a relatively long period. The IRRs were not part of the main priority among the researchers' and practitioners' responsibilities. Besides, the IRRs were not formally attached to a funded research project or a specific product/service in the industry. This lack of priority had a double effect. On the one hand, in the IRRs, the participants were free to follow an exploratory approach; on the other, the IRRs took longer than expected.

7.2 Implications for research and practice

As we mentioned in the introduction, our study implies a hypothesis that the relevance of literature studies will improve by involving practitioners in the process. We found that conducting the IRRs provided several occasions for the participants to meet, talk, and discuss. By focusing on topics directly related to the practitioners' context, we believe that the researcher's efforts were more aligned with the practitioners' needs compared to when the researchers work on their own.

Our proposed approach for IRRs suggests how and when, in the process, researchers and practitioners can interact. By identifying explicit roles and tasks for practitioners, we expected that IRRs would offer a higher degree of engagement from practitioners. This approach aligns with previous research in software engineering which emphasizes the importance of industry collaboration and the use of appropriate research approaches to increase the relevance of research [19] and the importance of the context in the research process [6].

Based on our findings, we recommend conducting IRRs to test the feasibility of extended collaboration between researchers and practitioners. Our study suggests that IRRs can serve as a practical starting point for joint projects, as they provide

opportunities for both parties to gain a deeper understanding of the problem at hand and the research area in question. Additionally, conducting an IRR early in the collaboration process can help determine whether the partnership is viable and whether both parties are willing to invest the necessary time and effort. For IRRs to be successful, there should be a practical problem that practitioners are facing that researchers can help solve. Furthermore, the practitioners should be willing and interested in participating in the review process.

When starting a research project e.g. a Ph.D. project, a common practice is to conduct a systematic literature review to get an overview of the state of the art in the research area. We see in the IRRs a way to complement this practice by involving practitioners to capture their perspectives and insights. It is important to note that IRRs should not be viewed as a substitute for systematic literature reviews, but rather as a complementary approach. As described in the initial proposal [30] IRRs prioritize knowledge exchange, and context-awareness, over covering the entire research area.

In future research, we plan to collaborate with researchers and practitioners to conduct more IRRs and study the impact of IRRs on the research and practitioners' context. Overall, we believe that fostering an exchange between researchers and practitioners is a promising way to increase the relevance and applicability of software engineering research.

7.3 Interaction, Rigor, Relevance, and Flexibility

The guidelines for conducting IRRs are rooted in three principles, interaction, rigor, relevance, and flexibility. The interactions between researchers and practitioners aim to increase the IRRs relevance. The flexibility allows the IRRs to be adapted to the needs of the participants and the context. Finally, rigor is essential to ensure the quality of the results. Rigor is the scientific aspect. The IRR should be systematic. It means that the search is not an ad-hoc search but follows the steps described in a protocol that aims to cover the literature in the area. We discuss these aspects in this section.

Interaction

Our proposed approach for IRRs suggests how and when, in the process, researchers and practitioners can interact. By identifying explicit roles and tasks for practitioners, we expected that IRRs will offer a higher degree of engagement from practitioners.

We achieved a high degree of engagement in these two cases. We found that conducting the IRRs provided several occasions for the participants to meet, talk, and discuss the topic in the two cases. How the interaction occurred varied between the cases, and it may be hard to anticipate how the interaction would happen. Focusing the IRRs on topics directly related to the practitioners' context probably

contributed to the interaction since they were familiar with the topic and were directly interested in the results.

Beyond promoting interaction, in both cases, researchers and practitioners exchanged knowledge. Conducting the reviews was helpful for the researchers to learn from the practitioner's particular scenarios and problems. The gain from the exchange on the practitioners' side was an overview of the research area and some research results that could be useful for the practitioner's context. As seen in both cases, the knowledge exchange happens as a consequence of working together oriented to the specific problem and with a tailored route.

The interaction allowed knowledge exchange, getting to know each other, and learning to work together. Being a relatively short process, the IRRs required less time and effort from the participants compared with other collaborations where the practitioners have an active role. Then, it makes the IRRs a good opportunity to test the feasibility of more extended collaboration. We noticed that the researchers and practitioners in both cases were open to the idea of continuing the collaboration after the IRRs.

Rigor

Concerning the systematic nature of the IRRs, we found that being systematic was a concern mainly for the researchers while the practitioners were more concerned with the relevance of the results. In both cases, the researchers involved had experience conducting secondary studies and recognized the importance of following a protocol and keeping track of the steps and decisions. When we asked the practitioners about the importance of being systematic and the risk of not covering all the relevant papers, they said they were interested in finding relevant or nearly relevant findings in their context. They also expressed trust in the researchers and their methods of searching and finding papers. Therefore, details about the search and selection steps were not a big concern for them. Nevertheless, we still think the IRRs should be systematic and follow a protocol.

The research community in software engineering values the systematic character of literature studies, and secondary studies could be criticized if they are not systematic. Besides, researchers are often interested in the general state of the art in a research area, and being systematic is a way to address this concern. Overall, we consider that the systematic aspect of the IRRs may be necessary for conducting IRRs in other scenarios, which makes the IRRs different from other ad-hoc literature reviews. This tension between rigor and relevance may be addressed in future studies.

Relevance

We expected that IRRs would improve the relevance of literature reviews by providing research results relevant to practitioners' problems and their context. We found that practitioners' involvement in the IRRs helped achieve this to a large

extent. Formulating research questions that capture practitioners' interest sets the study on the right course. Furthermore, practitioners' involvement provides insights into their problems and context, which helps operational decisions during the IRR about identifying, selecting, and synthesis of relevant research results.

Flexibility

The IRRs need to be flexible to adapt to the particular needs of each case. We noticed, for example, that the teams were formed and research questions specified together in the early stages of the IRR. Similarly, both cases' search and selection strategies were updated after presenting intermediate results. However, in other steps of the IRRs, the steps were conducted in different ways. While in Case-SoftSelection, the search was traditional, i.e., search engine by keywords, in Case-MLTest, the set of papers was based on previous secondary studies. Moreover, the analysis followed different approaches in Case-SoftSelection, the analysis aimed at getting a general understanding of the topic, in Case-MLTest, the analysis aimed at finding the best problem-solution match. Also, the presentations of research results differed in the two cases, adapted to the expectations of the practitioners. In Case-SoftSelection syntheses of research, contributions were presented. In Case-MLTest, contributions were presented per article, preliminary in terms of full research articles, and finally, as visual abstracts of the research contributions. In summary, in these two experiences, the IRR teams followed a flexible approach and balanced between following the guidelines and the need to adapt to the particularities of each case.

7.4 Limitations of this study

In this study, we are observing and reflecting on the application of IRRs in two cases of industry-academia collaboration. Conclusions are drawn based on interviews, observations, and the experiences of two researchers (also co-authors of this paper) as part of the review teams. We present no quantitative results and do not propose any causal models. Therefore, to reason about the validity of our conclusions, we apply the framework by Maxwell [27] comprising descriptive validity, interpretive validity, theoretical validity, generalizability, and evaluative validity.

Descriptive validity refers to the factual accuracy of the collected data. To achieve as accurate and complete data from interviews as possible, all researchers were involved in designing the interview protocol, two researchers conducted each interview, and interviews were recorded and automatically transcribed. In addition to data from interviews, observations were made by the first author in the initiation of the studies as well as by the second and third authors as participants in the studies. These observations may be biased by our different roles and pre-understanding of the IRRs guidelines.

Interpretive validity refers to the researcher's interpretation of the situation. In our case, it regards the interview situations. Not every nuance is captured in the interviews. To avoid misinterpretations, we let the interviewees read this manuscript. Regarding participatory observation, this threat is mitigated by the actual involvement in the cases.

Theoretical validity relates to interpretation or theorizing at a higher abstraction level. Our theoretical conclusions evolved through thematic coding, analysis, and writing this manuscript. All five authors were involved in both these activities, ensuring agreement among the researchers. To help the reader assess the theoretical validity, all steps of coding and interpretation have been transparently reported in this manuscript.

Generalizeablility A threat to our conclusions' general validity is that we had IRR experts (or at least access to them) in both cases. This means that we still do not know how feasible it is to implement the approach guided by the protocols alone. However, we provide examples to follow by describing how the IRRs were conducted in those two cases. Furthermore, the relationships between industry and academia vary from place to place and between domains. It also depends on individual relationships between researchers and practitioners. Thus the application of our findings may require adaptation in other situations. We still contend that the report's general conclusions and recommendations can support other industry-academia collaborations, especially in the initiating stages.

Evaluative validity relates to our underlying values. Our recommendations are not neutral but based on assumptions about any envisioned stakeholder's preferences. Although subjective, these assumptions are non-controversial (e.g., effective communication is good, producing relevant knowledge is desirable, and meeting the expectations of involved participants is good).

8 Conclusions

This paper reports two independent IRRs performed by academic researchers and industry practitioners. Conducting the IRRs favored a positive environment for interaction and knowledge exchange. The motivation for conducting an IRR may vary and affect the interpretation of our findings. In both cases, the participants' motivation included exploring ways for collaboration between researchers and practitioners. The teams did not set the detailed research questions upfront. Instead, a common general interest in the studied topics was the starting point.

In both cases, starting the IRRs included a lot of interaction, i.e., formulating research questions, identifying inclusion/exclusion criteria, and agreeing on an expected outcome. Then, it took a couple of meetings to form the review team. This step goes hand in hand with deriving research questions. Similarly, it requires several exchanges to align the academic and industrial problem formulation. Although this is a creative process that can be carried out in different ways, both cases were

helped by following predefined protocols. Such protocols facilitated collaborative work by including clear goals, steps, and responsibilities.

Furthermore, we found that practitioners trust the researchers to navigate the research results and find relevant articles. In our two cases, they did not see a need to take part in developing the details of the search and selection protocol directly. Instead, they were interested in commenting on the output of the procedure i.e., papers and theory, which in turn was helpful in protocol development. Depending on the communication gap, several such iterations were needed.

How to prioritize analysis effort was guided by the needs of the practitioners rather than scientific standards. Timeliness, relevance and applicability of output were more important than finding the best evidence or covering all the related literature.

Results may have value even if not disseminated beyond the review team. In our two cases, spontaneous knowledge sharing within the company took place in one case and plans for future studies emerged in both reviews. However, both reviews were reported as scientific publications after concluding.

This study presents two successful cases of using IRRs to support researchers-practitioners communication. The recommendations presented in this paper are based on the experiences of these two cases. The recommendations complement the steps for performing IRRs.

Acknowledgment

This work has been supported by ELLIIT, a Strategic Area within IT and Mobile Communications, funded by the Swedish Government. We would like to thank the researchers and practitioners who participated in the interviews for their time and valuable insights.

References

[1] Nauman Bin Ali. Is effectiveness sufficient to choose an intervention?: Considering resource use in empirical software engineering. In *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, ESEM, Ciudad Real, Spain, September 8-9, 2016*, pages 54:1–54:6, 2016.

[2] Nauman Bin Ali, Emelie Engström, Masoumeh Taromirad, Mohammad Reza Mousavi, Nasir Mehmood Minhas, Daniel Helgesson, Sebastian Kunze, and Mahsa Varshosaz. On the search for industry-relevant regression testing research. *Empir. Softw. Eng.*, 24(4):2020–2055, 2019.

[3] Nauman Bin Ali and Binish Tanveer. A comparison of citation sources for reference and citation-based search in systematic literature reviews. *e Informatica Softw. Eng. J.*, 16(1):220106, 2022.

[4] Nauman Bin Ali and Muhammad Usman. A critical appraisal tool for systematic literature reviews in software engineering. *Information and Software Technology*, 112:48–50, 2019.

[5] Markus Borg. The AIQ meta-testbed: Pragmatically bridging academic AI testing and industrial Q needs. In *International Conference on Software Quality*, pages 66–77, 2021.

[6] Lionel Briand, Domenico Bianculli, Shiva Nejati, Fabrizio Pastore, and Mehrdad Sabetzadeh. The case for context-driven software engineering research: generalizability is overrated. *IEEE Software*, 34(5):72–75, 2017.

[7] David Budgen, Pearl Brereton, Sarah Drummond, and Nikki Williams. Reporting systematic reviews: Some lessons from a tertiary study. *Information and Software Technology*, 95:62–74, 2018.

[8] David Budgen, Pearl Brereton, Nikki Williams, and Sarah Drummond. What support do systematic reviews provide for evidence-informed teaching about software engineering practice? *e-informatica software engineering journal.*, 14(1):7–60, 2020.

[9] Bruno Cartaxo, Gustavo Pinto, and Sergio Soares. The role of rapid reviews in supporting decision-making in software engineering practice. In *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018*, pages 24–34, 2018.

[10] Bruno Cartaxo, Gustavo Pinto, and Sergio Soares. Rapid Reviews in Software Engineering. In Michael Felderer and Guilherme Horta Travassos, editors, *Contemporary Empirical Methods in Software Engineering*, pages 357–384. Springer International Publishing, 2020.

[11] Daniela S Cruzes and Tore Dybå. Recommended steps for thematic synthesis in software engineering. In *2011 international symposium on empirical software engineering and measurement*, pages 275–284. IEEE, 2011.

[12] Vinicius dos Santos, Anderson Yoshiaki Iwazaki, Katia Romero Felizardo, Erica Ferreira de Souza, and Elisa Yumi Nakagawa. Towards Sustainability of Systematic Literature Reviews. In *Proceedings of the 15th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, Bari Italy, October 2021. ACM.

[13] Tore Dybå, Dag I.K. Sjøberg, and Daniela S. Cruzes. What works for whom, where, when, and why? on the role of context in empirical software engineering. In *Proceedings of the ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, ESEM '12, page 19–28, New York, NY, USA, 2012. Association for Computing Machinery.

[14] Emelie Engström, Robert Feldt, and Richard Torkar. Indirect effects in evidential assessment: A case study on regression test technology adoption. In *Proceedings of the 2nd International Workshop on Evidential Assessment of Software Technologies*, EAST '12, page 15–20, New York, NY, USA, 2012. ACM.

[15] Emelie Engström, Kai Petersen, Nauman Bin Ali, and Elizabeth Bjarnason. Serp-test: A taxonomy for supporting industry—academia communication. *Software Quality Journal*, 25(4):1269–1305, dec 2017.

[16] Robin M Featherstone, Donna M Dryden, Michelle Foisy, Jeanne-Marie Guise, Matthew D Mitchell, Robin A Paynter, Karen A Robinson, Craig A Umscheid, and Lisa Hartling. Advancing knowledge of rapid reviews: an analysis of results, conclusions and recommendations from published review articles examining rapid reviews. *Systematic reviews*, 4(1):1–8, 2015.

[17] Katia Romero Felizardo, Érica Ferreira de Souza, Bianca Minetto Napoleão, Nandamudi Lankalapalli Vijaykumar, and Maria Teresa Baldassarre. Secondary studies in the academic context: A systematic mapping and survey. *J. Syst. Softw.*, 170, 2020.

[18] Vahid Garousi, Markus Borg, and Markku Oivo. Practical relevance of software engineering research: synthesizing the community's voice. *Empirical Software Engineering*, 25(3):1687–1754, May 2020.

[19] Vahid Garousi, Kai Petersen, and Baris Ozkan. Challenges and best practices in industry-academia collaborations in software engineering: A systematic literature review. *Information and Software Technology*, 79:106–127, 2016.

[20] Vahid Garousi, Dietmar Pfahl, João M. Fernandes, Michael Felderer, Mika V. Mäntylä, David Shepherd, Andrea Arcuri, Ahmet Coşkunçay, and Bedir Tekinerdogan. Characterizing industry-academia collaborations in software engineering: evidence from 101 projects. *Empirical Software Engineering*, 24(4):2540–2602, August 2019.

[21] Donald Hislop, Rachelle Bosua, and Remko Helms. *Knowledge management in organizations: A critical introduction*. Oxford University Press, 2018.

[22] Vladimir Ivanov, Alan Rogers, Giancarlo Succi, Jooyong Yi, and Vasilii Zorin. What do software engineers care about? gaps between research and practice. In *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering*, pages 890–895, Paderborn Germany, August 2017. ACM.

[23] Valerie J King, Adrienne Stevens, Barbara Nussbaumer-Streit, Chris Kamel, and Chantelle Garritty. Paper 2: Performing rapid reviews. *Systematic Reviews*, 11(1), 2022.

[24] Barbara A. Kitchenham, Tore Dybå, and Magne Jørgensen. Evidence-based software engineering. In *Proceedings of the 26th International Conference on Software Engineering (ICSE)*, pages 273–281, 2004.

[25] Claire Le Goues, Ciera Jaspan, Ipek Ozkaya, Mary Shaw, and Kathryn T Stolee. Bridging the gap: From research to practical advice. *IEEE Software*, 35(5):50–57, 2018.

[26] Santiago Matalonga, Domenico Amalfitano, Andrea Doreste, Anna Rita Fasolino, and Guilherme Horta Travassos. Alternatives for testing of context-aware software systems in non-academic settings: results from a rapid review. *Information and Software Technology*, 149:106937, 2022.

[27] Joseph Maxwell. Understanding and validity in qualitative research. *Harvard educational review*, 62(3):279–301, 1992.

[28] Francisco Ponce, Gastón Márquez, and Hernán Astudillo. Migrating from monolithic architecture to microservices: A rapid review. In *38th International Conference of the Chilean Computer Science Society, SCCC 2019, Concepcion, Chile, November 4-9, 2019*. IEEE, 2019.

[29] Jennie Popay, Helen Roberts, Amanda Sowden, Mark Petticrew, Lisa Arai, Mark Rodgers, Nicky Britten, Katrina Roen, Steven Duffy, et al. Guidance on the conduct of narrative synthesis in systematic reviews. *A product from the ESRC methods programme Version*, 1(1), 2006.

[30] Sergio Rico, Nauman Bin Ali, Emelie Engström, and Martin Höst. Guidelines for conducting interactive rapid reviews in software engineering – from

a focus on technology transfer to knowledge exchange. Technical report, 2020.

[31] P. Runeson, M. Höst, A. Rainer, and B. Regnell. *Case Study Research in Software Engineering: Guidelines and Examples*. Wiley Publishing, 2012.

[32] Qunying Song, Markus Borg, Emelie Engström, Håkan Ardö, and Sergio Rico. Exploring ML testing in practice – lessons learned from an interactive rapid review with Axis Communications. In *Proceedings, CAIN'22, 1st Conference on AI Engineering – Software Engineering for AI*, 2022.

[33] Margaret-Anne Storey, Emelie Engström, Martin Höst, Per Runeson, and Elizabeth Bjarnason. Using a visual abstract as a lens for communicating and promoting design science research in software engineering. In *International Symposium on Empirical Software Engineering and Measurement (ESEM)*, pages 181–186, 2017.

[34] Claes Wohlin, Aybuke Aurum, Lefteris Angelis, Laura Phillips, Yvonne Dittrich, Tony Gorschek, Hakan Grahn, Kennet Henningsson, Simon Kagstrom, Graham Low, et al. The success factors powering industry-academia collaboration. *IEEE software*, 29(2):67–73, 2011.

[35] Claes Wohlin and Per Runeson. Guiding the selection of research methodology in industry–academia collaboration in software engineering. *Information and Software Technology*, 140, 2021.