Software Engineering Assignment

Signe Sidwall Thygesen

1 My research topic in the field of scientific visualization

I work with scientific visualization, an interdisciplinary field where we develop techniques, methods and algorithms for visualizing scientific data with the purpose of improving the analysis process and providing a better understanding of the underlying scientific phenomenon that the data represents. The field contains a mix of theories and tools from areas such as computer graphics, interaction design, human-computer interaction, and mathematics. Further, since it involves the visual human perception system, it is linked to cognition and psychology (not in focus in my research).

For a bit of an historical background, the field of scientific visualization has mainly emerged from the ability to use computers to visualize scientific data (although visualizations have existed for a much longer time). Therefore, it stems from a technical background, centred around algorithms and implementation. Further, since there is a strong connection to the fields of natural sciences, where the data to visualize usually comes from, I would say that the field is influenced by the natural sciences views on epistemology.

A general approach within the field is to implement the ideas in software prototypes, illustrating techniques for analysis, visualization, and exploration of the scientific data. For prototyping, we use and develop open source software (Inviwo[1] is mainly developed within my research group) in an iterative manner based on feedback from collaborators.

The data I focus on comes from theoretical material science, aimed specifically towards simulations describing electronic structures on a very detailed level, involving collaborations with domain scientists from theoretical chemistry and theoretical physics. These domains are heavily supported by the software systems running the simulations of electronic structures, and the amount of data is continuously increasing. Therefore, new tools for exploration and analysis of the data is needed. In my work, we develop visual analysis methods that support exploration and understanding of the electronic structures.

I would say that my work is a mix of *problem-driven* and *technique-driven* research. It is motivated by a problem from the domain of the data. To have a sufficient understanding of the domain is crucial, and collaboration with people that work in the domain is important. We also focus on the development or improvement of a technique within the field of data analysis and visualization. For example, the development of topological methods for scientific data (building on the field of Topological Data Analysis) is a topic that my research group is working on.

2 Selected aspects of Behavioural Software Engineering

Since I work with software and I collaborate with people in multiple settings, I would say that the lecture content is highly related and relevant to my research. I also recognize much of the course content from previous experience of developing software in industry.

[1] inviwo.org

Instead of selecting separate concepts, I've here chosen to elaborate on the broader concept of Behavioural Software Engineering [LFW15] (mainly based on how it was presented during the lecture), relating it to two aspects in my research: the behavioural aspects *within* my research studies, and the behavioural aspects *in the process* of conducting my research.

2.1 Behavioural aspects within my research studies

The researchers I collaborate with are using and developing software to run data simulations. One aim with my research (as mentioned in the introduction) is to develop visual analysis methods and techniques that could help them in their process to understand the data. Based on the concept of Behavioural Software Engineering, one could argue that focusing solely on the methods and techniques is not enough, that we need to take the more complex, interrelated factors of psychology into account in order to make a real improvement in their workflow. One could also argue that the purpose of our research might not be to improve their workflow, but rather to show possible new ideas and techniques. We of course try to understand what type of visualization would be fruitful in their specific environment by gathering feedback along the way, but I would still say that our focus is mainly on the development of techniques and methods.

Could we focus *more* on the behavioural aspects in how they analyse and visualize the data instead of focusing on novel visualization techniques? This would require a different approach in the research methodology. A study I think would be interesting is to look at how a visualization affects the way scientists work over a *longer* period of time. How is it adopted? How does it affect the way they communicate about the data? And how they understand the data? These aspects are probably much more difficult to study. Also, the question is to what extent our research is driven by what is possible to publish (here my impression is that the techniques and algorithms are still the main focus).

2.2 Behavioural aspects in my research process

From what I have seen so far of the academic environment, my impression is that the working process is surprisingly little discussed. We focus a lot on the results, in our case the prototypes, techniques and methods as output, finally presented in papers. But how to develop the way we collaborate, and how to develop the working environment for the individuals and groups that are producing the research, is in my opinion rarely discussed.

Since we work with software and people, both within the research group and together with collaborators, I believe that many of the thoughts from Behavioural Software engineering applies here as well. However, I also see some differences between the software engineering in industry and in our research context, which could affect to what extent it makes sense to draw parallels between our research and the Behavioural Software Engineering concepts. In the following, I describe two examples of differences.

A first example is the different sizes of the teams that does the code development. According to my experience, the software engineering in industry is more focused on the team effort, in larger companies having groups of developers working together in teams, whereas in the academic context, the actual software development in a project might be done by a single PhD student. This could be due to how research projects are being set up, and the fact that the doctoral title in the end is personal (and the academic world encourages self going individuals). In the academic world, the work might be tighter connected to individuals than in the industry (where there is a higher need for redundancy). However, I still believe that the possibility to have software developed in teams would give better quality (e.g. fewer bugs) and end result.

A second example is the difference in hierarchies within the teams. The collaborations in research might consist of a diverse constellation when it comes to academic experience. It's a mix of PhD students, Post Docs and professors, and therefore also an inherently strong hierarchy to take into consideration. According to my experience, this hierarchical mix is not as strong within the industrial context (but there other hierarchical patterns can take place).

The mentioned differences might be hard to change, because of traditions within the academic environment, but I still think that my working process could be improved by taking inspiration from, for example, the agile way of working in industry. (I have made a quick search for tips and tricks on how to implement agile ways of thinking in an academic context, and found some blog posts[2], online lessons[3] and papers [Bie24] [SH18].)

I would be very interested in a study that took a Behavioural Software Engineering perspective and applied it to an academic context (probably there are already such studies?), targeting questions such as: How does the hierarchical structures in the academic environment affect the individuals in the way they work and experience work? How is the organizational structure of universities influencing the scientific process (and scientific output)? How does the stages of group development play out in an academic context?

3 Selected ideas from guest lectures

Research vs "reality". In both guest lectures, it was acknowledged how things in practice ("reality") can differ from a research setting. For example, when Per Lenberg from Saab reported how his colleagues had answered to *"How do you foresee software engineering relevance changing with the increased integration of AI/ML in your operation?"* the results were mainly "I don't know", "nothing will change" and "work will be more boring". He discussed the gap between what could be anticipated in research (like "our solution is the best, surely people will adopt it"), and what happens in industry ("people are conservative"). Changes take time. On a somewhat similar theme, Parthasarathy Dhasarathy from Volvo group pointed out the differences between a research/testing environment and a practical setting when saying that in the practical setting extremes, are rare but there will be inconsistencies. This concerns how we should verify the software to meet a real world scenario.

Relating this to my work, I meet challenges in how our techniques could be integrated into an already existing analysis pipeline within the domain. Our novel ideas of how to visualize data might not be in line with the domain scientist's current workflow, and new ideas take time to understand and adopt. Moreover, if we would develop software that would be maintained over time (and not only software prototypes that later might be forgotten), there are so many new and challenging factors to take into consideration, not the least verification.

The organizational + social relations chart. Per Lenberg's organizational chart with the social relations put on top put an emphasis on that there is more to an organization than the titles. To some extent this feels rather obvious but can be easy to forget. When having collaborations with researchers in other domains, I am reminded about how different both the organizational levels and the social relations can play out in different settings. This concerns for example who should be authors on a paper (definitely a political question), or who should be invited to a research meeting.

[2] nature.com/articles/d41586-019-01184-9, cloudinary.com/blog/agile-research-is-it-possible

[3] nlesc.github.io/teamwork-for-research-software-development

4 Papers from the CAIN conference

In this section, I describe two papers I have read from the CAIN conference and answer the given questions (a - d), one paragraph for each question. Since I do not work directly with AI engineering, I will answer question (d) related to *software* or *data* engineering, how I could potentially make this engineering better/easier in my research, based on ideas from the paper.

4.1 *What About the Data? A Mapping Study on Data Engineering for AI Systems*

The paper *What About the Data? A Mapping Study on Data Engineering for AI Systems* by Petra Heck [Hec24] was presented at CAIN 2024. Here, some challenges concerning data management within AI engineering from the existing literature are discussed. The author tries to answer the question *"How to do data engineering for AI systems?"* by studying recent papers that deal with data engineering tools, activities, frameworks or architectures. The paper itself does not provide a single core idea, but maps out some solutions suggested in the studied papers. The data engineering is crucial for the engineering of AI systems, since the AI systems are centred around data. There is also a need to map out data engineering practice separate from software engineering practice, since the skills required to do data engineering might be different from the ones required for software engineering.

In my research, the data plays a central role. We work with scientific data in collaboration with domain scientists. I do not work with AI systems specifically, but definitely software engineering (and AI/ML could potentially be a tool used in the analysis pipeline.) I would say that the paper relates to my research in the way it emphasizes the importance of data. So far, I have worked with a few smaller data sets and in tight collaboration with the scientists that generated the data, so the data engineering aspects to take into consideration within the projects have not brought that much of trouble, but could of course be the case in the future. However, we do struggle with data infrastructure after a finished project. How should we store the data to make it accessible for the future? The data is contextual, and requires additional information to understand, which could be hard to incorporate in the data storage.

I believe that a large AI-intensive software project could definitely benefit from some of the suggestions in the paper. As already mentioned, the paper does not bring up one single core idea, but rather points to several ideas from the studied papers. For example, DataOps - the process to automate and manage data life cycle stages, or Data-Oriented Architecture - a software architecture that focuses on the data (both from Section *6.4 Implications for Researchers*), could be useful concepts for creating a solid ground for the data management in the project. The paper also brings up some suggestions for practitioners (Section *6.3 Implications for Practitioners*), for example the increase of available open source tools for data engineering, which might be of use. My work, which from a general perspective is about visualizing data, could help the project to understand the data and/or the AI-models better, by extracting features and characteristics (maybe using Topological Data Analysis) and present them visually with the possibility of exploration.

Although many ideas in the paper are related to data engineering for AI specifically, I believe there are some takeaways from the paper that could be translated into my domain. One is the already mentioned DataOps, a framework for dealing with the data management through its whole life cycle. This could be a relevant when it comes to managing the data after a finished project (probably using it on a larger scale than within a specific project). Another takeaway is how one could/should view the data engineering within data-centric practices, not as a single activity in the engineering of a system, but as a choice of perspective that affects the whole approach. Even though I believe that, within my research, we already have a data-centric approach when it comes to how the specific scientific data strongly guides our choices of technical methods, there might still be more aspects from the data-centric perspective that we could take inspiration from.

4.2 *Developer Experiences with a Contextualized AI Coding Assistant: Usability, Expectations, and Outcomes*

The paper *Developer Experiences with a Contextualized AI Coding Assistant: Usability, Expectations, and Outcomes* by Pinto et al. [PDSR+24] was presented ad CAIN 2024. A *contextualized* coding AI assistant can be more beneficial in software engineering practices than a *general-purpose* AI assistants, since the incorporation of domain knowledge makes it tailored to the specific environment where the software is being developed. In this paper, the result from studying developer's experiences when using a contextualized coding AI assistant is described. The focus is on two aspects: the user experience and the correctness of the generated solutions. The study found that the contextualized coding AI assistant resulted in time savings and increased productivity, but also found challenges related to what knowledge sources were necessary, and limitations when applied to complex code. The study presented in this paper can help distinguish when and how such assistants are feasible to use and what limitations these assistants have. Further, since the coding AI assistants are AI systems themselves, this study reports things to work on in the future, to develop even better contextualized coding AI assistants.

The paper is related to my work in the sense that I also develop software in a specific domain where a contextualized coding assistant might be useful.

Within a large software project, such as an AI-intensive software project, the use of a coding assistant could help make the software development more efficient, and utilizing the domain knowledge through a *contextualized* AI coding assistant might be a good choice. This study provides some insights into what benefits, challenges and limitations a contextualized AI coding assistant would bring, which could help identify where it would make most sense to use it. Further, conducting a similar study within the environment of the large scale AI-intensive software project could give even better knowledge of what role the coding assistant plays within this context, and how it is perceived by the developers. My work in the context of this AI-intensive software project could be related to visualizing the output from such a contextualized AI coding assistant (if I again view my work from a more general perspective, to visualize data). For example, to summarize an aggregated collection of answers with visual representations. These representations could incorporate parameters such as "correctness", uncertainty and domain knowledge. Although not related to my research directly, I also think it would be interesting to follow what is suggested in the future work section, to explore how a coding assistant impacts the collaborative coding practices and team dynamics. By studying the impact over a longer time could give insight into how coding assistants might change the software developer role.

A contextualized coding assistant could possibly be beneficial within my prototype development as well. I have been quite sceptical to the use of code assistants in the past since we work with such specific code, but the possibility to incorporate domain knowledge into the assistant might make it more useful. A challenge could be, however, that such knowledge is hard to obtain since I am not working *within* this domain but rather collaborating with people in the domain. This links back to one of the observed challenges in the paper, what type of knowledge sources are necessary for the coding assistant. Another use case is instead to make the code assistant specific for the open source project we develop in my group (Inviwo), with the purpose of developing that code more efficient and possibly serve as some documentation and answering questions about the code.

References

[Bie24] Katharina Biely. Agile by accident: how to apply Agile principles in academic research projects. *SN Social Sciences*, 4(1):12, January 2024.

[Hec24] Petra Heck. What About the Data? A Mapping Study on Data Engineering for AI Systems. In *Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering - Software Engineering for AI*, CAIN '24, page 43–52, New York, NY, USA, 2024. Association for Computing Machinery.

[LFW15] Per Lenberg, Robert Feldt, and Lars Göran Wallgren. Behavioral software engineering: A definition and systematic literature review. *Journal of Systems and software*, 107:15–37, 2015.

[PDSR+24] Gustavo Pinto, Cleidson De Souza, Thayssa Rocha, Igor Steinmacher, Alberto Souza, and Edward Monteiro. Developer Experiences with a Contextualized AI Coding Assistant: Usability, Expectations, and Outcomes. In *Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering - Software Engineering for AI*, CAIN '24, page 81–91, New York, NY, USA, 2024. Association for Computing Machinery.

[SH18] Enric Senabre Hidalgo. Management of a multidisciplinary research project: A case study on adopting agile methods. *Journal of Research Practice*, 14(1):P1, 2018.