

WASP Course Software Engineering Assignment

Name: Yufei Zhu

1 Research Introduction

My research topic is within the area of mapping of dynamics (MoDs). Perceiving dynamics is a key requirement for autonomous robot operations in environments shared with other agents. Because dynamic changes typically follow spatio-temporal patterns, it is possible to build maps of dynamics. MoDs can be beneficial for mobile robots in several areas such as motion planning and semantic understanding. In my research I explore the new application area of human motion prediction. I exploit MoDs that encode human dynamics as a feature of the environment for long-term human motion prediction. By using MoDs, motion prediction can utilize previously observed spatial motion patterns that encode important information about motion flow in the environment. In contrast to prior planning-based approaches, MoDs can be exploited to infer goal locations, constraints and preferences implicitly, without requiring them as input, which is a key feature to predict long-term human motion. My research project will also address challenges in constructing MoDs in a more flexible way. One of these challenges is to learn MoDs online for life-long learning enabling efficient updates based on live motion observations. Current MoD representations are typically built offline based on past observations. Another challenge is predicting MoDs based on semantic information of the environment, without having to observe motion data for a long time. Aside from the general benefits of data sparse methods, this line of work will facilitate MoDs that can adapt to the changes in the environment. I also research ways to build event-based MoDs rather than considering only long-term statistical trends which are done in current MoD approaches. Building MoDs in a more flexible way helps to achieve accurate human motion prediction. To address the aforementioned challenges, my research questions include: First, how to use MoDs to predict human motion, particularly over extended periods of time? Second, how to efficiently update MoDs online? Third, how to predict MoDs based on semantic information of the environment? Last, how to learn event-based flow components of MoDs? In summary, my research project aims for a more powerful representation of human dynamics and a wider usage of MoDs for mobile robots collaborating with humans. My research is situated in the field of robotics, specifically aimed at enabling robots to better comprehend their surrounding environments, which often include human beings.

2 2 principles/ideas/concepts/techniques from lecture

2.1 Version Control

In software engineering, version control is important. It helps us keep track of changes in code and is one fundamental step in the iterative process of software engineering. Version control system (VCS) allows multiple people to work on a project simultaneously. Git is one popular and widely used VCS. Git offers features such as branching, merging, and conflict resolution to manage changes. A good usage of VCS will benefit a lot to implementation of methods in AI/ML. During the development of softwares, multiple features of VCS will be used. When the software project involves a lot of experimentation, VCS allows developers and researchers to keep track of different versions of models and data, so it is easy to go back to a previous state if needed. AI/ML projects often require input from data scientists, machine learning engineers and software developers. In this case, VCS enables collaboration among these roles, helping resolve conflicts. When dealing with data management, VCS enables versioning of large datasets, and improves

reproducibility of training models. During the deployment step, VCS helps in tracking changes both in code and in model parameters, which benefit for safe deployment of models to production environments. Also, VCS allows users to annotate the data, helping to understand the previous changes. This characteristic is good for debugging and further development in complex AI/ML projects. Besides git, Nano version control (NanoVC) is proposed to be the next step in VCS in computer science and AI applications. NanoVC emerges from Git, and it combines in memory version control and nano scale modeling [1]. In my research project, I develop AI/ML models for human motion prediction and will deal with large human trajectory data. Even when some part of the research project only involves myself in the development, by using git, I can track my history changes and make good annotations. If I want to apply my code to different scenarios like predict human motion in another environment, VCS allows me to create a new branch and make it isolated from the main branch.

2.2 Software Testing

Testing is an important part in software engineering. It involves the execution of a program and to see whether there are errors or bugs. Software testing aims to validate the software behavior and discover faults. There are multiple methods of testing, including unit tests, integration tests, system tests and acceptance tests. Having testing helps the software to meet all of its requirements and works as intended. The reliability of the software is enhanced by using testing techniques. Testing can help the software perform consistently under various, even extreme conditions, such as a spike of users. Also, in the agile way of working, testing is not a phase that occurs only at the end of the development cycle, but rather is an ongoing activity that takes place concurrently with development. Testing allows for continuous feedback, adjustments, quality assurance and risk mitigation. There is an approach named test-driven development (TDD), which emphasizes writing tests before writing the actual code. When developing software in AI/ML, testing is also important and brings some challenges as well. The system behavior can be very complex and hard to have thorough and meaningful test cases. The testing data can be a large size and evaluation is hard to be standardized. But on the other hand, AI techniques have been increasingly employed in software testing. Natural language processing helps to understand codes and requirements and AI techniques can be used to automate testing tasks, analyze results and even predict future software issues. In the robotics research field, software testing can have challenges in practice. [2] conducted a qualitative study of the testing practices used by the robotics community and identified the challenges faced by practitioners when testing their systems. The challenges are grouped into real-world complexities, community and standards and component integration. In my research project, testing can help as well. During the development of human motion prediction approaches, unit tests can be applied to ensure the correctness of functions. Testing can be used to validate the efficiency and accuracy of MoDs in predicting human motion. Historical human motion data can be used to create a test suite. When deployed to real robots, integration tests can be applied to ensure the robot is running normally.

3 2 principles/ideas/concepts/techniques from guest lecture

3.1 Sentiment Analysis

In software engineering, sentiment analysis means the task of identifying the polarity of the source materials, such as code. The polarity means the valence, which is the positive or the negative semantic orientation. Natural language processing and computational linguistics can be used for sentiment analysis. Sentiment analysis aims to determine the attitude, opinions, or emotions. The output is generally a score or a category like "positive," "neutral," or "negative." Traditional tasks of sentiment analysis can be word or document polarity, which is identifying the semantic orientation of a word or input document. Besides traditional tasks, sentiment analysis can also be used in software engineering. In the task of sentiment analysis, AI/ML algorithms can be the main

approaches. The possible approaches can include supervised machine learning algorithms, unsupervised algorithms like lexical approach, and other text preprocessing approaches like tokenization. There are still issues and challenges within sentiment analysis. When processing context in different languages whose meaning change depending on the domain in which they are employed, there are not many tools and resources available for all the languages. Sarcasm and irony are also critical challenges for sentiment analysis [3]. The concept of sentiment analysis could be extended to the emotional mapping of dynamic human-robot interactions. My research project is about mapping human motions, which include position and velocity of humans, but without the emotions of humans. Emotions can carry their own form of dynamics, which will affect human motion. By integrating sentiment analysis with MoDs, the emotional state can be a new feature in predicting human motion. Sentiment analysis can also help in robotics. For example, a robot can learn to adapt its interactions based on the detected mood of the customer, based on the conversation between them. When integrating sentiment analysis with MoD components and then used in downstream tasks such as motion planning for robots, robust and modular software frameworks are needed to provide real-time data collection, software testing, and accurate analysis.

3.2 Boundary Value Testing

Boundary value testing is part of testing techniques in software engineering. It focuses on testing values that lie at the boundary edge of the input domain. Boundary value testing helps to find errors or bugs in code in the extreme cases. For example, a program may handle the normal cases but fail to process a large number of inputs. In software engineering, errors can occur at boundaries, which may bring trouble in some cases. Boundary testing has been mainly used in traditional software, but it could be extended to test the robustness of AI/ML models. By testing the limits of machine learning models, the robustness of software can be improved. Boundary value analysis is an important topic in boundary value testing. [4] addresses the core problem of how to automate black-box boundary value analysis, and proposes a generic method and implementation of automated boundary value analysis. My research on MoDs could benefit significantly from the application of boundary value testing. I can apply boundary value testing in my research area, because knowing the boundary conditions is also crucial for MoD. When predicting human motion, the boundary value can be the large initial velocity of a human being, or predict for an extended time period like 60 seconds, and the number of people to predict. Understanding boundary conditions under which MoD models operate could offer insights into their limitations and areas for improvement.

4 2 topics from list

4.1 Security and Privacy

Security and privacy are important in software engineering. Software engineering can be attacked by unauthorized access and can result in data alteration or device broken. The privacy data which contains sensitive information can be exposed and misused. Therefore, ensuring the security and protection of privacy is important. Security and privacy in the Internet of Things (IoT) are critical concerns. IoT software operates in a distributed environment where multiple devices communicate with each other and centralized servers [5]. In my research area, it is essential to ensure that MoDs are securely stored and processed to prevent misuse. MoDs can be encrypted for safety. In the human motion prediction area, data privacy can become an ethical problem. To learn models of human flow patterns, we need to collect data on human motion trajectories. Mostly, the trajectories are anonymous and people are labeled with numbers. But during the research process, the problem of data privacy can still occur. We can use the mobile robot to collect human trajectory data. When working with mobile robots, we collect data including environmental data and human data. The environment data can be the distance to nearby obstacles, the temperature and humidity of the environment, and the image of the environment. The human data can be the image and voice of nearby humans. After the data collection, to conceal the sensitive data that can be used to identify

people, we blur the faces of the recorded people and mark the participants with just numbers. We aim to make the collected human data anonymous. But even when the faces are blurred, with machine learning technology, the participants can be identified to some degree. Personal information can be learned from the human trajectory and the behavior of interaction with robots. If we build a model for human walking behavior and search for all the students in Örebro University to find who this person is, we may identify this person. So when we publish the dataset and make it public, and when we publish our learned model of human motion, we need to keep an eye on how the dataset and model are used. And in our own devices, we need to remove the original videos with human faces, and human identity information as well. The collected and generated data are owned by the authors.

4.2 Project management

In software engineering, project management involves the scope definition, planning, team management, requirements gathering, design and architecture, development, quality assurance, monitoring and control, documentation, deployment and maintenance and support. In project management, an agile way of working can be good to promote flexibility and collaboration among team members. Effective project management is crucial in software engineering, as it can help teams to deal with complexities and uncertainties, ensuring that the project can be completed successfully and have sustainability. Project management in AI/ML projects can be different from traditional ones. The outcome of the project is not only a software product, and can be a complex, interacting system. New project management paradigms is a future work for AI or robotics research. [6] addresses how project management pull and AI technology push are likely to result in incremental and disruptive evolution of project management capabilities and practices. In my research project, I also use project management techniques. Visualization tools are important in project management. I use a tool named Jira to manage my own research project. With a kanban board, I can optimize my workflow and prioritize my tasks. When collaborating with other researchers, a shared visualization board can improve the flexibility and transparency.

5 Discussion

Software engineering plays an important role in my research area. With techniques including version control, testing, continuous integration and project management, I can develop my model in a more flexible and sustainable way, and improve the robustness of my project. Developing automated testing in AI/ML softwares can be a potential future area of software engineering. Automating the process of validating and verifying AI models can ensure reliability, robustness and fairness. In my research work in using MoDs for human motion prediction, security assurance and privacy protection is important. Software engineering will develop specialized protocols for securing AI-driven systems against attacks. The future of software engineering is closely connected with AI/ML. For my research field, this integration can be helpful to incorporating data-driven development and provide possibilities of predicting human motion in real-time systems. A future trend can also be focused on developing frameworks that can efficiently process and analyze data on robots.

References

- [1] L. Machowski and T. Marwala, “Nano version control and the repo as the next data structure in computer science and artificial intelligence,” in *2021 Ethics and Explainability for Responsible Data Science (EE-RDS)*, pp. 1–7, 2021.
- [2] A. Afzal, C. L. Goues, M. Hilton, and C. S. Timperley, “A study on challenges of testing robotic systems,” in *2020 IEEE 13th International Conference on Software Testing, Validation and Verification (ICST)*, pp. 96–107, 2020.

- [3] M. Wankhade, A. C. S. Rao, and C. Kulkarni, "A survey on sentiment analysis methods, applications, and challenges," *Artif. Intell. Rev.*, vol. 55, p. 5731–5780, oct 2022.
- [4] F. Dobslaw, R. Feldt, and F. de Oliveira Neto, "Automated black-box boundary value detection," 2022.
- [5] F. Al-Turjman, H. Zahmatkesh, and R. Shahroze, "An overview of security and privacy in smart cities' iot communications," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 3, p. e3677, 2022. e3677 ETT-19-0133.R1.
- [6] F. Niederman, "Project management: openings for disruption from ai and advanced analytics," *Information Technology & People*, vol. 34, no. 6, pp. 1570–1599, 2021.