HICSS-58 Minitrack: The Impact of AI on Software Engineering

Stefan Wittek
Clausthal
University of Technology
switt@tu-clausthal.de

Peter Salhofer
University of Applied Sciences
Graz
peter.salhofer@fh-joanneum.at

Sandra Gesing
US Research Software Engineer
Association
sandra@us-rse.org

1. Introduction

The software engineering community has recognized that numerous software engineering tasks throughout the entire software process can greatly benefit from the use of AI-based methods and algorithms. This realization has fueled the emergence of the research field known as "AI for Software Engineering." This field has gained momentum due to the escalating size and complexity of software systems and, consequently, software engineering tasks.

Researchers and industry experts have harnessed the potential of this approach across various use cases. These applications range from predicting the effort required for a specific software feature based on previous coding tasks (Alsaadi, 2022) to generating test cases from real-time observations of system behavior (Aniculaesei, 2018).

The recent advancements in AI models for natural language processing, particularly the utilization of the transformer architecture (Vaswani, 2017) and generative pre-training of these architectures through language modeling tasks (Radford, 2018), have accelerated progress in this field. Consequently, tasks like generating complete code for methods and classes and automatically documenting existing code have reached a level where they can be incorporated into industrial-level products. A notable example of this progress is the impressive code generation capability seen in tools like GitHub Copilot (Dakhel, 2023).

However, the application of AI in software development extends far beyond code-centric tasks. It empowers the use of unstructured data, including requirements, software architecture descriptions, commit messages, bug reports, and more, to enhance the practice of software engineering.

With this minitrack, we aim to provide insights into this rapidly evolving and diverse research field.

The paper *The Impact of Generative AI-Powered Code Generation Tools on Software Engineer Hiring: Recruiters' Experiences, Perceptions, and Strategies* provides a first qualitative analysis of the impact of these tools on hiring and education. While these tools boost developer productivity, they complicate the evaluation of software engineering candidates' abilities. This study examines recruiters' experiences with GenAI-powered tools and their challenges in assessing candidates. A survey of 32 industry professionals shows that although most are familiar with the tools, few organizations have adapted their hiring processes to account for them.

Additionally, the paper *Machine Learning-assisted Test Records Analysis during the Real-Time Validation of Automotive Software Systems based on HIL Simulation* provides an application example of AI support in the development process of a safety critical Software System. In the automotive industry, ISO 26262 mandates rigorous testing to ensure software system quality, particularly during system integration and testing. However, analyzing time series data manually to identify system failures is time-consuming and complex. This paper introduces a AI-assisted approach for real-time validation of automotive software systems (ASSs).

2. References

Alsaadi, B., & Saeedi, K. (2022). Data-driven effort estimation techniques of agile user stories: a systematic literature review. Artificial Intelligence Review, 55(7), 5485-5516.

Aniculaesei, A., Howar, F., Denecke, P., Rausch, A. (2018) "Automated generation of requirements-based test cases for an adaptive cruise control system," 2018 IEEE Workshop on Validation, Analysis and Evolution of Software Tests (VST), Campobasso, Italy, 2018, pp. 11-15, doi: 10.1109/VST.2018.8327150.

URI: https://hdl.handle.net/10125/109715
978-0-9981331-8-8

Dakhel, A. M., Majdinasab, V., Nikanjam, A., Khomh, F., Desmarais, M. C., & Jiang, Z. M. J. (2023). Github copilot ai pair programmer: Asset or liability?. Journal of Systems and Software, 203, 111734.

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving language understanding by generative pre-training. https://www.mikecaptain.com/resources/pdf/GPT-1.pdf

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.