Master’s Thesis

Comparative Analysis of LLMs for Automated Code Refactoring

Code refactoring involves restructuring existing source code to enhance its internal structure while preserving its external behavior [1]. As software systems evolve, they can become increasingly complex over time. This complexity can lead to technical debt, which in turn affects development efficiency and increases the risk of defects. The integration of **Generative AI** into software engineering workflows has triggered a revolution in automating coding tasks, including code refactoring [2]. However, the selection of the most suitable **Large Language Model (LLM)** for refactoring tasks presents significant challenges due to varying capabilities, accuracy, and performance among different models. This thesis is designed to conduct a comparative analysis of several leading LLMs like GPT-4, Codex, and LLaMA in the context of automated code refactoring.

[1] Martin Fowler, Kent, John Brant, William Opdyke, Don Roberts, D. B. “Refactoring: Improving the Design of Existing Code”, Addison-Wesley, New York, (1999).

[2] Shirafuji, Atsushi et al. “Refactoring Programs Using Large Language Models with Few-Shot Examples.” _2023 30th Asia-Pacific Software Engineering Conference (APSEC)_ (2023): 151-160.

Feel free to contact me directly if you are interested in this topic!

Please include your current **CV** and **grade report**, as well as a short **motivation letter** and **when** you intend to start your thesis.

Sivajeet Chand, M.Sc.
sivajeet.chand@tum.de
Tel. +49 89 289 17314

Chair of Software and
Systems Engineering (i4)
TUM School of Computation,
Information and Technology