FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Assignment of bachelor's thesis

Title:	Project adaptation in code completion via in-context learning
Student:	Maksim Sapronov
Supervisor:	Evgenii Glukhov, M.Sc.
Study program:	Informatics
Branch / specialization:	Artificial Intelligence 2021
Department:	Department of Applied Mathematics
Validity:	until the end of summer semester 2025/2026

Instructions

In recent years, advancements in the fields of code and natural language processing have significantly transformed code completion systems, making them more accurate and capable of assisting with software development tasks. Some recent Large Language Models (LLMs) and Code LLMs can utilize an entire software project. However, most open source pre-trained models can process less than 5% of the project's code files at once. Such a restricted context length of Code LLMs limits their ability to comprehend the structure of a real-world software project fully. As a result, models often struggle to integrate information spread across multiple files, such as dependencies, class hierarchies, and external library usage, leading to outputs that lack project-wide coherence.

It is known that LLMs have an ability for in-context learning, i.e. it can adapt to novel tasks if there are some examples in the context. In particular, for the code completion task, some of the project files are included in the context as examples of completion. It is common practice now to include a repository-level training step in the pre-training of Code LLMs. However, some open questions in this domain are the main focus of this research project.

Research Questions:

For pre-trained Code LLMs with no context window extension:

1. Does the improvement of the quality of code completion depend on a context

---

Electronically approved by Ing. Magda Friedjungová, Ph.D. on 28 January 2025 in Prague.

composition approach for in-context learning? Estimate the influence.
2. Does fine-tuning help to improve the quality of code completion for a particular context composer?
For pre-trained Code LLMs with context window extension:
1. Are there any context composition techniques that don't impact in-context learning abilities?
2. Does the improvement of the quality of code completion depend on a context composition approach for the repository-level training step? Estimate the influence.

Guidelines for Elaboration:
1. Conduct a survey of the existing papers on the following topics: in-context learning, code completion models, repository-level code completion.
2. Developing a Context Composition Framework for Repository-level Code Completion Task
- Design and implement a framework for extracting a context from a repository, include language-specific processing for code files.
3. Investigating the Impact of Context Composition on Code Completion
- Analyze how variations in context composition (from 2) and context length influence the quality of one-line code completion for a pre-trained Code LLM.
4. Developing a Fine-Tuning Pipeline
- Design and implement a pipeline for the complete fine-tuning of a pre-trained Code LLM.
5. Evaluate Context Composition Methods on Fine-tuned Model
- Base model is https://huggingface.co/deepseek-ai/deepseek-coder-1.3b-base
- The dataset for the fine-tuning is available to company staff at JetBrains through internal resources. It contains data from open-source repositories on GitHub.
6. Extend Model's Context Length with Repository-level Training
- Base model is https://opencoder-llm.github.io
7. Compare Context Extensions for Different Context Composers

- Dataset for the fine-tuning is available to company staff at JetBrains through internal resources. It contains data from open-source repositories on GitHub.
- For evaluation use the approach that is suggested in https://arxiv.org/abs/2406.11612