# Automatic Translation of Code Repositories Using Large Language Models

Technical University of Munich

Department of Informatics

Chair of Robotics, Artificial Intelligence and Real-time Systems

## Background

Translating code between programming languages and preserving the original functionality (transpilation) is often an avoided task in software engineering, as manual code migration is expensive and resource-intensive. However, translating the software to a modern language reduces maintenance effort, reliability, security, and performance [10, 6]. For instance, legacy systems written in COBOL remain critical in sectors like banking and government but are costly to maintain and difficult to integrate with modern software architectures [8]. Traditional rule-based transpilers have long been used for automated translation but struggle with large and complex codebases and language-specific structures. Due to these limitations, machine learning approaches are an active field of research, especially large language models (LLMs) [5, 7].

Recent advances in LLMs demonstrate that these models can capture the syntactic structure and semantic intent of code across multiple programming languages. Unlike traditional transpilers that require manual mappings for each language feature, LLMs can learn code relationships from large-scale datasets. LLMs can internally model similarities between language constructs of a trainee correctly. This enables them to generalize translation patterns across programming languages [7]. But LLMs can also struggle in these tasks without sufficient context and carefully engineered prompts. They then act as next-token predictors rather than understanding the broader translation task [5]. So, despite promising results, code translation with LLMs remains a complex and not well-researched challenge [7, 10, 5].

## Description

This thesis aims to explore the automatic translation of code repositories using large language models, focusing primarily on translating the CORA toolbox [1] from MATLAB to Python. CORA is a large library of approximately 170,000 lines of code, making it a challenging but valuable case study for evaluating the capabilities of modern LLMs in large-scale code translation tasks.

We will set up an automated translation process using integrated development environments (IDEs) with an included LLM framework like Cursor[1]. As part of an iterative refinement cycle, we aim to translate a portion of the CORA codebase. Major parts of this process include determining the optimal content to include in the context window for each translation step [10] and deciding how to split and organize the codebase, as well as implementing edge case routines. We also include few-shot prompting [3, 2], chain of thought reasoning [9], and self-refinement techniques [4] to improve the translation. In addition, the process involves automatic testing and verification of the translated code to ensure correctness.

The goal of the thesis is to identify and discuss the challenges encountered and provide an evaluation of the future potential for achieving a full translation of a codebase like CORA using LLMs.

## Tasks

- Literature research on large language models
- Familiarize with the toolbox CORA
- Design and iterative implementation of an automatic translation process
- Evaluate the quality of the translation and the techniques used
- Optional: Fully translate the CORA toolbox from MATLAB to Python

[1] https://www.cursor.com/

**Supervisor:**
Prof. Dr.-Ing. Matthias Althoff

**Advisor:**
Tobias Ladner, M.Sc.

**Research project:**
FAI

**Type:**
BT

**Research area:**
Large language models

**Programming language:**
MATLAB, Python

**Required skills:**
Knowledge in formal methods and machine learning, good mathematical background

**Language:**
English

**Date of submission:**
20. Mai 2025

**For more information please contact us:**

Phone: +49 (89) 289 - 18140

E-Mail: tobias.ladner@tum.de

Website: ce.cit.tum.de/cps/

## References


[1] Matthias Althoff. An introduction to CORA 2015. In *ARCH@ CPSWeek*, pages 120–151, 2015.

[2] Manish Bhattarai, Javier E. Santos, Shawn Jones, Ayan Biswas, Boian Alexandrov, and Daniel O'Malley. Enhancing code translation in language models with few-shot learning via retrieval-augmented generation. In *2024 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–8, 2024.

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.

[4] Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. SELF-REFINE: iterative refinement with self-feedback. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23. Curran Associates Inc., 2023.

[5] Rangeet Pan, Ali Reza Ibrahimzada, Rahul Krishna, Divya Sankar, Lambert Pougeum Wassi, Michele Merler, Boris Sobolev, Raju Pavuluri, Saurabh Sinha, and Reyhaneh Jabbarvand. Lost in translation: A study of bugs introduced by large language models while translating code. In *2024 IEEE/ACM 46th International Conference on Software Engineering (ICSE)*, pages 995–1007, 2024.

[6] Soumit Kanti Saha, Fazle Rabbi, Song Wang, and Jinqiu Yang. Specification-driven code translation powered by large language models: How far are we?, 2024.

[7] Qingxiao Tao, Tingrui Yu, Xiaodong Gu, and Beijun Shen. Unraveling the potential of large language models in code translation: How far are we? In *2024 31st Asia-Pacific Software Engineering Conference (APSEC)*, pages 353–362, 2024.

[8] Ashish Upadhaya. Understanding legacy software: The current relevance of cobol. Master's thesis, VU University Amsterdam, 2023.

[9] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22. Curran Associates Inc., 2022.

[10] Hanliang Zhang, Cristina David, Meng Wang, Brandon Paulsen, and Daniel Kroening. Scalable, validated code translation of entire projects using large language models, 2024.



Technical University of Munich



Department of Informatics

Chair of Robotics, Artificial Intelligence and Real-time Systems