

BUILDING LANGUAGE-AGNOSTIC EDUCATIONAL MATERIALS FOR SOFTWARE ENGINEERING COURSES WITH LLM

Nikola Valchanov, Todorka Terzieva

Abstract. *Large Language Models (LLMs) can be integrated in almost every aspect of a modern enterprise regardless of its field of operations. LLMs are increasingly integrated into software development where their potential to support multilingual code comprehension is used to build features and/or migrate existing codebases across versions of a given library, language or framework.*

While highly valuable in software engineering this LLM capability provides means to assist with building implementational examples showcasing general principles for solving a given set of problems across different languages, libraries and frameworks.

This study investigates the use of LLMs to create code examples in Java and Node.js from a single instructor-maintained C# codebase, with the goal of enhancing students' understanding of cross-language software engineering concepts. The paper presents a workflow for migrating examples of object-oriented design and architectural patterns, written in C# to Java and NodeJS through automated translation using LLMs. The resulting codebases acknowledge specific code structure and best practices for each respective language and framework.

Findings indicate that LLMs can reliably preserve architectural intent while adapting implementations to language-specific idioms, enabling instructors to maintain one codebase while offering students a multilingual perspective on core engineering principles.

Key words: Large Language Model (LLM), AI, Education, Code Generation

Nikola Valchanov^{1,*}, Todorka Terzieva¹

¹ Paisii Hilendarski University of Plovdiv,
Faculty of Mathematics and Informatics,
236 Bulgaria Blvd., 4027 Plovdiv, Bulgaria

Corresponding author: nvalchanov@uni-plovdiv.bg