SHOW: Determining Code Proficiency Levels from Python Textbooks

Ruksit Rojpaisarnkit[1], Gregorio Robles[2], Jesus M. Gonzalez-Barahona[2], Kenichi Matsumoto[1] and Raula Gaikovina Kula[3]

[1]*Nara Institute of Science and Technology, Japan*
[2]*Universidad Rey Juan Carlos, Madrid, Spain*
[3]*The University of Osaka, Japan*

Abstract

The ability to measure developer proficiency is crucial, as it reflects an individual's capability to understand and interpret efficient, effective, and well-structured code. It is an essential aspect of ensuring software quality. Although prior work has proposed different approaches for measuring proficiency through code, the process of learning various coding concepts remains non-trivial and highly debated. This paper proposes a framework for determining code proficiency through learning aids that serve as ground-truth—namely, textbooks—and two automated methods: Übersequence and Clustering. We conducted an empirical study to identify Python code proficiency levels. We then assessed the effectiveness of the framework by addressing two key research questions. Using a dataset of 22 introductory Python textbooks and code constructs referenced from Python *AST*, we achieved a high coverage of 85.51% for Python code constructs. Our analysis reveals a significantly high similarity in the sequence of code construct introductions across the textbooks, supporting our methodology of using textbooks to assess code proficiency. The resulting Übersequence demonstrates the feasibility of assigning proficiency levels to individual code constructs, while clustering further enables a structured grouping perspective. To demonstrate practical applicability, we present examples and initiate discussions on future research directions, particularly software maintenance tasks such as bug assignment and code reviews.

Keywords

Software Maintenance, Software Evolution, Mining Software Repositories, Code Proficiency

1. Summary

In this paper –accepted and to be published at TOSEM [1]– we introduce a novel, comprehensive framework for assessing code proficiency, explicitly designed to be programming-language agnostic. We validate its feasibility by applying it to the Python language, yielding promising results that successfully delineate distinct and meaningful groupings of proficiency levels. This achievement demonstrates that the complex challenge of accurately determining coding proficiency is both achievable and highly practical, validating the framework's core utility and establishing a new avenue for research with far-reaching implications. For researchers, the framework offers a structured methodology for studying coding skills at scale. In industry, practitioners gain a foundation for tools that can evaluate team capabilities, identify specific skill gaps, or optimize software maintenance tasks by aligning them with developer expertise. Furthermore, educators and students can leverage it to tailor learning curricula and track individual progress. The integration of clustering and the Übersequence provides a novel methodology for systematically mapping language constructs to proficiency levels, inspiring future research into automated or AI-driven systems for skill assessment. We recognize that most constructs identified in this initial study belong to Python's standard library. Therefore, we hypothesize that by adjusting the ground-truth sources (e.g., using specific textbooks for domains or specialized PyPI libraries), the framework can be customized to create personalized proficiency profiles appropriate for specific roles or technologies. Future work will focus on exploring the generalizability of this framework across diverse programming languages and specialized domain applications.

The 24th Belgium-Netherlands Software Evolution Workshop, 17–18 November 2025, Enschede, The Netherlands
rojpaisarnkit.ruksit.rn1@is.naist.jp (R. Rojpaisarnkit); grex@gsyc.urjc.es (G. Robles); jesus.gonzalez.barahona@urjc.es (J. M. Gonzalez-Barahona); matumoto@is.naist.jp (K. Matsumoto); raula-k@ist.osaka-u.ac.jp (R. G. Kula)
© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

Acknowledgments

This work is supported by the JSPS KAKENHI Grant Number JP20H05706, JP24H00692 and the Spanish Ministry of Science, Innovation, and Universities under the Excellence Network AI4Software (Red2022-134647-T) and through the Dependentium project (PID2022-139551NB-I00).

Declaration on Generative AI

During the preparation of this work, the authorss used generative AI in order to: Grammar and spelling check. After using these services, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] R. Rojpaisarnkit, G. Robles, J. M. Gonzalez-Barahona, K. Matsumoto, R. G. Kula, Determining code proficiency levels from python textbooks, Transactions on Software Engineering and Methodology Accepted; pending publication (2026). URL: https://arxiv.org/abs/2408.02262.