SHOW: A Method for Inferring Python Proficiency from 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 accurate measurement of developer proficiency is paramount for ensuring software quality, as it directly reflects an individual's capacity to comprehend and produce efficient, effective, and well-structured code. While various code-based approaches for proficiency assessment have been proposed, the underlying process of learning coding concepts remains complex and widely debated. This paper introduces a novel framework for determining code proficiency by leveraging textbooks as ground-truth learning aids. The framework employs two automated methods, Übersequence and Clustering, to achieve this goal. We conducted an empirical study using a dataset of 22 introductory Python textbooks and Python AST code constructs. This analysis covered a high 85.51% of Python code constructs. Our findings demonstrate a remarkably high similarity in the sequential introduction of these constructs across the textbooks, validating the use of textbooks for proficiency assessment. The resulting Übersequence successfully assigns proficiency levels to individual code constructs, while the Clustering method provides a complementary, structured grouping perspective. We conclude by illustrating the framework's practical utility and discussing future applications in software maintenance tasks like bug assignment and code reviews.

Keywords

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

1. Summary

In this talk –based on a research that has been accepted 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.

BENEVOL'25: Proceedings of 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)

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.