Majumdar, R. et al. (Eds.) (2025). Proceedings of the 1st International Conference on Learning Evidence and Analytics. Asia-Pacific Society for Computers in Education

An Exploratory Study on the Impact of a Self-Explanation-Based Adaptive Chatbot on Learning Strategies

Hui-Chun HUNG[a*], Tzu-Pau LIN[a], Shu-Ming WANG[b]
[a]*Graduate Institute of Network Learning Technology, National Central University, Taiwan*
[b]*Department of Information Management, Chinese Culture University, Taiwan*
*hch@cl.ncu.edu.tw

Abstract: This study developed a programming-learning support system that blends the self-explanation strategy with generative artificial intelligence (Gen AI). The system automatically generates adaptive questions from students' submitted code and provides instant feedback through a chatbot to guide self-explanation and reflection. A 16-week experiment conducted in a Python course showed that students with high chatbot interaction significantly outperformed their low-interaction peers in learning-strategy gains, especially in the use of rehearsal strategies. Correlational and regression results further confirmed a positive influence of interaction frequency on strategy improvement. The findings show that Gen AI assistance can enhance programming learning and promote strategic learning behaviors, highlighting its practical value for adaptive instruction.

Keywords: Self-explanation, Programming Learning, Generative Artificial Intelligence, Chatbot

1. Introduction

With the rapid advancement of technology, students' ability to leverage emerging tools has become a central concern in education. Programming practice nurtures logical reasoning and problem-solving skills while familiarizing learners with cutting-edge algorithms, data science, and machine-learning techniques (Ogegbo & Ramnarain, 2021).

However, effectively assessing students' grasp of programming concepts and providing deep feedback in class often faces constraints of manpower and time (Hao et al., 2022). With the recent advances and wide adoption of generative-AI technologies such as ChatGPT (Popovici, 2023), researchers have begun exploring how these tools can be integrated into programming education. In particular, the self-explanation strategy—prompting learners to reflect on and articulate the code they have written—has been shown to promote deep understanding and knowledge internalization (Lehtinen et al., 2021). Therefore, developing a system that can automatically generate questions tailored to students' code and engage them in adaptive interaction via generative AI opens new avenues for programming instruction.

This study designed such a system, combining generative AI with self-explanation prompts. Through experimental teaching and learning-behavior analysis, we investigate its effects on students' programming comprehension, learning-strategy use, and self-explanation behaviors, offering practical guidance for educators seeking to employ generative AI for adaptive tutoring.

2. The Self-Explanatory Adaptive Chemistry Companion System

The system delivers two core functions: (1) adaptive question generation and (2) real-time self-explanation feedback. Early in the semester, the instructor provides foundational theory and hands-on guidance. After each instructional segment, students complete a programming assignment. Once an assignment is submitted, the AI question-generation module analyzes the actual code and produces individualized questions. During the following class meeting, students interact with the chatbot, explaining their answers and revising them based on instant feedback—thereby strengthening their grasp of programming principles.

3. Research Method

3.1 Instructional context and participants

The study was embedded in a 16-week course titled “Python Data-Mining Practice.” Topics ranged from basic Python syntax to data analysis and machine-learning models. Thirty graduate students from a northern Taiwanese university constituted the final valid sample.

3.2 Procedures

In the preparation phase we finalized course content and assignments and adopted three learning-strategy subscales—rehearsal, critical thinking, and metacognitive self-regulation—from Pintrich (1991) Motivated Strategies for Learning Questionnaire (MSLQ). We then built the system’s front-end, back-end, and database to support classroom interaction and log collection.

During the instructional phase, Week 1 covered course orientation and pre-test questionnaires (five-point Likert scale). From Weeks 2–15, each lesson ended with a take-home programming task; the next week, students tackled self-explanation questions generated from their own code, using chatbot feedback to refine their explanations. In Week 16, the same questionnaires were administered to capture changes in learning strategies and students’ perceptions of the system.

In the analysis phase, questionnaire data and system usage logs were examined to test our hypotheses and inform instructional improvements.

4. Results

4.1 Pre-test equivalence

Students were split into high- and low-interaction groups by the median number of chatbot turns. Mann-Whitney U tests showed no significant pre-test differences in overall learning strategies ($U = 102$, $z = -0.44$, $p = .663$), rehearsal ($U = 100.5$, $z = -0.50$, $p = .614$), critical thinking ($U = 89.5$, $z = -0.97$, $p = .333$), or metacognitive self-regulation ($U = 100$, $z = -0.52$, $p = .603$), indicating baseline equivalence.

4.2 Post-test comparison

Dunn–Bonferroni pairwise tests assessed post-test differences. The high-interaction group scored significantly higher on rehearsal strategies ($Md = 4.25$) than the low-interaction group ($Md = 3.75$), adjusted $p = .006 < .01$. Although the high-interaction group showed modest advantages in overall strategy use, critical thinking, and metacognitive regulation, these did not reach significance (adjusted $p > .10$). Thus, frequent chatbot engagement particularly encouraged active rehearsal behaviors, whereas higher-order strategies may require longer cultivation.

5. Conclusion

Integrating self-explanation techniques with a cutting-edge generative-AI chatbot within programming education environments offers a multifaceted approach to enhance student learning. This integration facilitates a deeper understanding of core programming concepts by encouraging students to articulate their reasoning processes. Through an iterative dialogue format, students are prompted to engage in self-monitoring—actively assessing their understanding—and rehearsal—practicing problem-solving strategies repeatedly. Such interactive engagement has been shown to significantly increase the use of rehearsal strategies, especially among students who participate in high levels of interaction with the AI system. This effect supports the efficacy of the system's ability to initiate a comprehensive and active learning cycle. This study also observes that there were no statistically significant improvements in students' critical thinking skills and metacognitive regulation measures.

Overall, this research presents a promising and scalable framework that empirically demonstrates the potential of generative AI to serve as an adaptive tutor. It underscores the importance of customized dialogue scripts tailored to diverse learner profiles and suggests extending observational periods in future studies. Longer-term investigations could provide deeper insights into how higher-order cognitive and metacognitive strategies evolve over time, ultimately refining the AI's role in fostering autonomous, strategic learners across varied programming contexts.

Acknowledgements

This study was supported by the National Science and Technology Council, R.O.C. under the research project grant NSTC113-2628-H-008-001-MY3, and the Ministry of Education "Teaching Practice Research" Subsidies Program under project number PSK1136863. The authors would also like to thank the Science and Technology for Learning Center at National Central University for its support.

References

Hao, Q., Smith IV, D. H., Ding, L., Ko, A., Ottaway, C., Wilson, J., Arakawa, K. H., Turcan, A., Poehlman, T., & Greer, T. (2022). Towards understanding the effective design of automated formative feedback for programming assignments. Computer Science Education, 32(1), 105-127. https://doi.org/10.1080/08993408.2020.1860408

Lehtinen, T., Santos, A. L., & Sorva, J. (2021). Let's ask students about their programs, automatically. 2021 IEEE/ACM 29th International Conference on Program Comprehension (ICPC), 467-475. https://doi.org/10.1109/ICPC52881.2021.00054

Ogegbo, A. A., & Ramnarain, U. (2021). A systematic review of computational thinking in science classrooms. Studies in Science Education, 58(2), 203-230. https://doi.org/10.21585/ijcses.v3i1.53

Pintrich, P. R. (1991). A manual for the use of the Motivated Strategies for Learning Questionnaire (MSLQ). https://eric.ed.gov/?id=ED338122

Popovici, M.-D. (2023). ChatGPT in the Classroom. Exploring Its Potential and Limitations in a Functional Programming Course. International Journal of Human–Computer Interaction, 1-12. https://doi.org/10.1080/10447318.2023.2269006