Journal of Electronic Research and Application, 2025, Volume 9, Issue 5
http://ojs.bbwpublisher.com/index.php/JERA
ISSN Online: 2208-3510
ISSN Print: 2208-3502

Research on Teaching Reform Strategies of Python Programming Course Based on Artificial Intelligence Technology

Hong Chen*

Changji Vocational and Technical College, Changji 831100, Xinjiang, China

**Author to whom correspondence should be addressed.*

Abstract: As one of the core courses for computer-related majors, the Python programming course has become increasingly important in the era of artificial intelligence. It aims to help students develop good computer thinking and improve their abilities in programming and data analysis. The application of artificial intelligence technology in the teaching of Python programming courses is of great significance for optimizing the allocation of teaching resources, enriching students' learning experience, and significantly improving teaching quality. Based on this, this paper first briefly expounds on the importance of applying artificial intelligence technology in the teaching of Python programming courses. On this basis, it focuses on exploring effective strategies for the teaching reform of Python programming courses based on artificial intelligence technology, hoping to provide new ideas for the teaching of Python programming courses and contribute to cultivating more Python programming talents with artificial intelligence literacy.

Keywords: Artificial intelligence technology; Python programming course; Teaching reform; Effective strategies

Online publication: October 15, 2025

1. Introduction

In the era of artificial intelligence (AI), AI technology has become a key driver for promoting reform and innovation in education and teaching. For Python programming courses, traditional teaching models have the following drawbacks: First, teachers cannot accurately and comprehensively grasp each student's learning situation to provide personalized guidance. Second, teaching content fails to keep pace with the rapid development of computer technology, leading to a potential disconnect between what students learn and practical applications. The application of AI technology, however, not only helps teachers accurately identify students' individual needs but also facilitates dynamic adjustments to teaching content, ultimately significantly improving teaching effectiveness. Thus, the research in this paper holds significant practical relevance.

2. The importance of applying AI technology in Python programming course teaching

2.1. Facilitating optimal resource allocation and personalized learning experiences

In traditional teaching models, teachers often rely on textbooks and teaching aids to prepare lessons and design teaching processes, mostly adopting a "one-size-fits-all" approach that struggles to meet students' diverse and personalized learning needs. Students remain in a passive position of receiving knowledge, resulting in low learning quality [1]. The application of AI technology, however, helps optimize resource allocation while providing students with personalized learning experiences—achieving two goals at once. Specifically, first, AI-driven learning platforms enable teachers to offer personalized learning resources and tutoring plans to students at different levels, realizing intelligent resource allocation and recommendation to improve resource utilization efficiency [2,3]. Second, AI technology can intelligently analyze students' learning behaviors, homework completion rates, and exam scores to generate learner profiles. Based on these profiles, intelligent platforms can develop personalized learning paths, accurately identify weak areas, maximize learning effectiveness, and ultimately achieve personalized teaching.

2.2. Helping teachers identify teaching issues and adjust strategies in a timely manner

Python programming courses place great emphasis on monitoring students' real-time progress, but traditional teaching models may struggle to achieve this comprehensively. AI-based intelligent tools, such as smart learning platforms and classroom interaction systems, can promptly capture key indicators like students' response speed, accuracy, and review frequency, thereby truly reflecting their mastery of knowledge points, learning progress, and issues in the teaching process. This allows teachers to accurately grasp students' learning status and enhance the timeliness and relevance of feedback. For common problems faced by most students, teachers can provide centralized explanations; for individual issues, they can offer one-on-one targeted tutoring—greatly improving the relevance and effectiveness of teaching [4,5].

2.3. Stimulating students' interest in learning Python programming

For beginners, learning Python programming can be challenging. When faced with tedious grammar rules, abstract logical concepts, and monotonous code writing, some students may develop fear or resistance, affecting their final learning outcomes [6]. The integration of AI technology, however, can provide students with real-world cases and vivid projects for practice. It can also leverage AR, VR, and other technologies to create immersive programming environments, fully stimulating their desire to learn and explore—yielding twice the result with half the effort. For example, students can write and debug Python code in virtual scenarios and obtain visual results, transforming tedious code learning into an engaging exploration journey and infusing their learning path with sustained vitality [7,8].

3. Effective strategies for teaching reform in Python programming courses based on artificial intelligence technology

3.1. Enriching teaching content

Enriching teaching content requires teachers to promptly integrate AI cases into Python programming courses. Meanwhile, attention should be paid to the organic combination of compulsory and elective content. This ensures that teaching content remains aligned with the times, laying a solid foundation for students' future careers. First, teachers should timely incorporate AI cases into Python programming teaching to fully stimulate students' interest

in learning and help them gain a deeper and more comprehensive understanding of AI. Python boasts rich AI libraries such as NumPy, Pandas, Matplotlib, Scikit-learn, TensorFlow, and PyTorch, which provide strong support for AI development, covering data processing, visual presentation, and algorithm implementation [9]. Therefore, in actual teaching, teachers can flexibly introduce AI cases based on these AI libraries. For example, using the Scikit-learn library to complete iris classification or implementing handwritten digit recognition through the TensorFlow library. These two cases help guide students to gradually explore the fields of machine learning and deep learning, enabling them to establish an overall understanding of core concepts, basic processes, and typical algorithms in AI. This, in turn, fully stimulates students' interest in learning and exploration, guiding them to transition from superficial learning to deep learning. Second, emphasis should be placed on organically combining compulsory and elective content to build a hierarchical and systematic learning system that meets students' personalized learning needs. In addition to compulsory content such as basic Python syntax, web crawlers, data analysis and processing, and data visualization, teachers should also push personalized learning resources to students through intelligent learning platforms, such as game development, network programming, natural language processing, and computer vision [10,11]. Students can freely choose learning content based on their personal interests, ensuring the cutting-edge nature of teaching content while providing solid support for their future career development. Teachers can provide students with diverse learning resources such as videos, tutorials, literature, software, and code examples through online learning platforms. They can also introduce real enterprise project tasks into the classroom to deepen personalized talent cultivation.

3.2. Optimizing teaching processes

To enrich students' learning experience and enhance the effectiveness of classroom teaching, teachers can actively adopt a diversified blended teaching model that integrates "in-class and after-class learning, online and offline engagement." This model divides the classroom into three stages—pre-class, in-class, and post-class—to optimize and restructure the teaching process. In the pre-class preview stage, teachers can release a list of preview tasks in advance. Students need to complete pre-class tasks according to the list, such as watching learning videos on the Classroom platform, finishing thinking questions and test questions, and forming a preliminary understanding of key and difficult knowledge through independent learning. During this process, teachers can use online teaching platforms to monitor students' preview progress in real time, accurately identify their weak points, and adjust subsequent teaching strategies accordingly. Taking the aforementioned "iris classification case" as an example, teachers can guide students to watch videos related to machine learning, require them to install Anaconda3 and learn about the Scikit-learn library, and set test questions and thinking questions, such as: What are the concepts of machine learning and supervised learning? What is the relationship between machine learning, artificial intelligence, and deep learning? At the same time, teachers can ask students to list specific scenarios of supervised learning in daily life and explain the reasons [12]. In the in-class learning stage, teachers should focus on explaining and practicing common problems among students and vividly demonstrating teaching cases. During this process, teachers should find appropriate opportunities to integrate ideological and political education content to cultivate more programming talents with both professional knowledge and excellent ideological qualities. Furthermore, teachers should flexibly use AI assistants to support students' learning and practice, giving full play to the enabling role of AI technology in Python programming teaching. For example, teachers can use tools like ChatGPT, Doubao, and DeepSeek to generate questions for students to answer, or encourage students to use tools like CodeGeek to compare the semantic similarity between their code and code generated by ChatGPT,

preventing code plagiarism and educating students on professional ethics [13]. In the post-class consolidation stage, teachers can assign homework based on teaching content and students' overall preview and learning performance, and use AI technology for intelligent homework correction. Students can review knowledge points and practice programming skills through intelligent teaching platforms to achieve timely consolidation and internalization of knowledge and skills, significantly improving learning quality.

3.3. Reforming teaching methods

Given the differences in students' programming foundations and abilities, as well as the various problems they may encounter in programming practice, teachers can flexibly use generative AI to empower heuristic teaching, allowing students to become the masters of the classroom, meeting their personalized learning needs, and promoting the transformation of Python programming teaching toward digitalization and personalization. Specifically, teachers can use the AI assistant on the Classroom platform to guide students' learning and exploration [14,15]. This platform's AI assistant offers multiple functions, such as heuristic question answering, heuristic knowledge point learning, and code analysis, helping students solve problems in a timely manner and significantly improving their learning efficiency. The AI assistant can actively generate programming exercises for students to practice. If students encounter difficulties during practice, they can seek help from the AI assistant, which will then automatically enter the heuristic answering interface and generate prompts for reference, effectively inspiring students. Instead of directly providing answers, the AI assistant guides students step by step to establish problem-solving logic, which helps cultivate good thinking habits [16]. In addition to the above methods, teachers can adopt project-driven learning by introducing real enterprise programming cases into the classroom and encouraging students to complete projects in groups. This cultivates students' teamwork spirit and improves their comprehensive programming abilities. After completing tasks, teachers should guide students to participate in interactive discussions and share their experiences, including new insights during programming, problems encountered, and solutions. This deepens students' understanding of knowledge and mastery of skills, creates a positive learning atmosphere, and promotes the collaborative development of teaching and learning [17].

4. Conclusion

In conclusion, the application of artificial intelligence technology in the teaching of Python programming courses is gradually reshaping the classroom teaching ecology, injecting strong vitality and impetus into the reform and innovation of curriculum teaching, and helping to comprehensively improve the teaching and learning effects. In the future, teachers should continue to deepen the research on the application of artificial intelligence technology in the teaching of Python programming courses, properly handle the relationship between technical teaching and manual teaching, continuously optimize teaching methods, and contribute to the cultivation of more high-quality talents who firmly master the professional knowledge and skills of Python programming.

Disclosure statement

The author declares no conflict of interest.

References

[1] Su T, 2024, Analysis of Teaching Reform of Python Programming Course for Artificial Intelligence. Office Automation, 29(9): 47–49.

[2] Min Y, Xu C, 2022, Teaching Reform and Practice of "Python Programming" Course in the Era of Artificial Intelligence. Science and Technology Guide - Electronic Edition (Second Half), (12): 187–189.

[3] Liang S, Du R, Hang W, 2023, Exploration on the Practice of Intelligent Education of Python Programming from the Perspective of Artificial Intelligence. Software, 44(2): 42–45, 65.

[4] Li L, Li D, Yang H, et al., 2023, Curriculum Reform of "Python Programming" under the Background of Artificial Intelligence. Modern Information Technology, 7(17): 178–182, 188.

[5] Sheng G, Zhao M, Liu Z, et al., 2022, Exploration on the Curriculum Reform of "Python Language Programming" under the Background of Artificial Intelligence. Technology Wind, (30): 83–85.

[6] Li F, 2019, Preliminary Exploration on the Construction of Python Language Programming Course under the Background of Artificial Intelligence. Education Informatization Forum, (10): 27–28.

[7] Wang Y, Wang Y, Zheng Y, 2024, Development of a New Form of Python Programming Textbook Based on Artificial Intelligence and Multimedia. Knowledge Economy, 697(33): 214–216.

[8] Yang H, Wu G, Tian Z, et al., 2024, Analysis of Problems and Countermeasures in Python Programming Teaching. Industrial and Technological Forum, 23(17): 145–147.

[9] Song S, 2023, Teaching Practice of Python Programming Course. Application of Integrated Circuits, 40(9): 150–151.

[10] Wang Y, Fan D, Jiang J, 2022, Teaching Reform and Practice of Python Programming Course. Technology Wind, (34): 95–97.

[11] Wang Y, Liu W, Wang K, et al., 2025, Development and Application of a Teaching Tool for Python Language Programming Based on Knowledge Graph. China Information Technology Education, (1): 99–102.

[12] Chen J, 2018, Teaching Research on Python Programming Course. Southern Agricultural Machinery, 49(24): 9.

[13] Hao G, Yang J, Wu D, et al., 2019, Design and Implementation of an Intelligent Q&A System for the "Python Programming" Course. Computer Knowledge and Technology, 15(7): 86–88.

[14] Luo J, 2024, Reform of Higher Vocational Python Programming Course Based on Intelligent Robot Design with AI Assistance. Great Wuhan, (19): 129–130.

[15] Zeng X, Wang M, Li G, et al., 2024, Discussion on Problem-Solving-Oriented Python Programming Teaching Method. Computer Knowledge and Technology, 20(25): 117–119, 122.

[16] Lei L, Li P, 2025, Innovation of Teaching Method for Python Programming Language Based on Reinforcement Learning. Computer Education, (1): 110–115.

[17] Duan Y, Xu C, Wei L, et al., 2021, Exploration on Teaching Method of Python Language Programming Course. Technology Wind, (17): 40–42.