Journal of Artificial Intelligence Practice (2025)
Clausius Scientific Press, Canada

DOI: 10.23977/jaip.2025.080119
ISSN 2371-8412 Vol. 8 Num. 1

Research on the Reform of Programming Courses Driven by Generative Artificial Intelligence

He Wang

School of Information & Intelligence Engineering, University of Sanya, Sanya, Hainan, 572022, China

Keywords: Generative Artificial Intelligence (GAI); DeepSeek; Programming Education; Curriculum Reform

Abstract: Exploring the multi-dimensional impact of generative artificial intelligence (GAI) on programming courses in higher education, analyzing the opportunities and challenges it brings in areas such as teaching, academic integrity, and future talent demands. Through an in-depth analysis of the technical characteristics of generative AI, its code generation capabilities, and its current applications in the educational field, it is found that generative AI offers significant advantages in enhancing programming learning efficiency and personalized teaching. However, it also poses new challenges to academic integrity and student motivation. In light of the trends in AI development, the study highlights the necessity of curriculum reform and proposes measures such as constructing a AI-assisted teaching model, optimizing course design, and innovating assessment systems to promote the reform of programming education, thereby meeting the new demands of the AI era.

1. Introduction

Since the beginning of 2025, the large language artificial intelligence model DeepSeek has rapidly become a focal point in the global technology field due to its revolutionary technological breakthroughs and its notable characteristics of low cost and high efficiency. Its open-source models, DeepSeek-R1 and DeepSeek-V3, not only rival top international closed-source models (such as GPT-4o) in performance metrics but have also reshaped the ecosystem with their extremely low training costs[1-2].

The influence of DeepSeek has transcended the purely technical domain. Its open-source nature and high-efficiency models have significantly lowered the barrier to entry for artificial intelligence technology, driving the intelligent transformation of numerous industries. Against this backdrop, an increasing number of fields are actively participating in this wave of change. Notably, the innovation in generative artificial intelligence technology has had a significant impact on the education field, especially posing new challenges and opportunities for the teaching models and content design of programming courses in higher education. This necessitates in-depth academic exploration to address the transformations brought about by generative artificial intelligence.

2. Overview of Generative Artificial Intelligence

Generative Artificial Intelligence (GAI) is an advanced AI technology capable of autonomously generating multimodal content such as text, images, audio, video, and code. From the perspective of current technological developments, typical representatives of generative AI systems include the GPT series (such as GPT-4) released by OpenAI, which have demonstrated exceptional performance in fields such as natural language processing and code generation[3].

2.1 Technical Characteristics of Generative Artificial Intelligence

The technical implementation of generative artificial intelligence primarily relies on three core elements: large models (algorithmic architecture), big data, and high computing power. Models such as GPT-4 and Sora, through innovative mechanisms like "brute-force computing" and "meaningless combination," have broken through the traditional boundaries of creativity, enabling the generation of novel content that exceeds human expectations and redefining existing senses of creativity[4]. At its core, generative AI uses deep learning techniques and pre-trained language models to learn patterns from massive datasets and generate new content[5].

The development path of generative AI shows a "general-to-specific" characteristic, where models first acquire general capabilities through large-scale data training and are subsequently optimized and fine-tuned for specific application scenarios[6-7]. This technological evolution contrasts sharply with the gradual learning process of humans. Currently, generative AI technology has demonstrated significant application potential across various fields, including business, education, and healthcare[8-9]. Its powerful capabilities provide robust technical support for the digital transformation of related industries.

2.2 Applications of Generative Artificial Intelligence in Code Generation

The application scenarios of generative artificial intelligence in code generation are extensive. Firstly, in the area of automated code generation, generative AI can automatically produce complete code implementations based on natural language descriptions. Developers can describe the desired functionality using simple natural language instructions, and the AI model can generate the corresponding code based on these descriptions. Secondly, in terms of code error detection and repair, generative AI can automatically identify potential issues in the code and provide suggestions. By analyzing the context of the code, AI models can accurately detect syntax errors, logical errors, and other issues, and generate corresponding solutions. Additionally, generative AI can offer many other features such as code completion suggestions and cross-programming language translation.

3. The Effects of Generative Artificial Intelligence on Programming Education

3.1 The Positive Effects

Generative artificial intelligence has certain positive effects on traditional programming courses in higher education. Generative AI can assist students in learning code. When students encounter difficult problems during their programming studies, they can directly ask the AI system and receive immediate feedback. Using its powerful natural language processing capabilities, generative AI can translate complex programming concepts into easy-to-understand explanations, significantly lowering the cognitive barrier for beginners. For lower-grade students or non-computer science majors, the AI system can provide annotated code examples and syntax error correction guidance, thereby aiding in the understanding of abstract concepts[10]. By reading and understanding the code

and comments generated by AI, students can gradually grasp programming thinking and code implementation skills while cultivating the ability for self-directed learning.

Additionally, generative AI technology offers intelligent, flexible, and personalized learning methods for programming education. It eliminates language barriers, allowing students to learn anytime and anywhere, transcending physical and time constraints, and making the learning process more equitable and accessible[10]. Under this new learning model, students' learning activities are no longer confined to traditional classroom environments. This flexibility not only enhances learning efficiency but also provides more equal learning opportunities for students with diverse backgrounds and learning abilities.

3.2 The Negative Effects

Although generative artificial intelligence has brought some positive effects to computer programming education, its negative effects cannot be overlooked. Among these, the most prominent issue is the serious challenge to academic integrity. Some students tend to overly rely on AI-generated code, directly copying and submitting it as their own work. This behavior not only violates the fundamental principles of academic integrity but may also lead to poor performance in exam situations, as students fail to truly master core programming skills.

Another noteworthy concern is that generative AI may lead to a decline in students' motivation to learn. When students realize they can easily obtain AI-generated code, they may become overly dependent on such tools, thereby losing the ability to think independently and solve problems on their own[10-11]. This dependency may reduce students' intrinsic motivation for in-depth learning and exploration, ultimately affecting their long-term learning outcomes and career development potential. Research from Hangzhou Normal University found that some students tend to frequently use generative AI tools throughout the programming process, obtaining immediate solutions by directly copying code or debugging information, which may hinder their ability to understand and master coding[11].

4. Analysis of the Necessity for Reforming Programming Courses

4.1 Adaptability to Technological Development Trends

In response to the effects of generative artificial intelligence on programming education, experts in related fields have conducted in-depth research and discussions in recent years. Research shows that current generative AI technology still has many limitations: while it performs excellently in solving basic programming problems, its accuracy significantly declines as problem complexity increases. Its code-writing capabilities have not yet had a disruptive impact on the computer industry, and traditional teaching models remain irreplaceable[12]. However, the development speed of generative AI is remarkable. GPT-4 has demonstrated performance close to human levels in handling complex tasks across multiple domains, significantly outperforming earlier ChatGPT models[13]. Additionally, in terms of code generation capabilities, the DeepSeek-R1 version not only surpasses DeepSeek-V3 and GPT-4o-0513 but also close to the OpenAI-o1-1217 version[1]. This suggests that the code-writing capabilities of generative AI are rapidly improving and will continue to be optimized in the future. Therefore, universities should initiate explorations and practices of related curriculum reforms as early as possible.

4.2 Balancing Academic Integrity and Learning Outcomes

To address the issue of academic integrity arising from students directly submitting AI-generated

code, many perspectives suggest restricting students' use of AI and propose solutions such as developing anti-AI detection tools to prevent over-reliance on AI-generated content. However, unlike traditional plagiarism detection mechanisms, which can be judged using quantitative metrics such as text repetition rates, detecting AI-generated content lacks clear rules, making it difficult to accurately identify the source of the content. When using AI tools for detection, there is a high likelihood of misjudgment, where non-AI-generated content may be incorrectly flagged as AI-generated, potentially leading to unfair accusations against students who have not violated integrity rules[14]. Furthermore, solutions to basic programming exercises often exhibit high similarity, further complicating the distinction between AI-generated and human-written code. Educators should shift their mindset: rather than expending significant resources to prevent students from using generative AI, they should actively explore how to effectively integrate AI technology into the teaching process to enhance students' core competencies.

4.3 Transformation of Future Talent Demands

With the rapid development of artificial intelligence technology, various industries are actively promoting AI-enabled digital transformation. In this context, individuals proficient in using AI tools will have a competitive edge in the future job market. AI-assisted programming will gradually become the industry norm, while the core responsibilities of programmers will shift toward higher-level design and analysis tasks. Therefore, mastering the efficient use of AI tools may become one of the core competencies of future programmers. Reforms in university programming courses should focus on cultivating students' ability to use AI tools while strengthening their programming design thinking and innovation capabilities to meet the new demands of the AI era.

5. Implementation Recommendations for Curriculum Reform

In response to the opportunities and challenges brought by generative artificial intelligence to programming courses, educators need to adopt multi-dimensional strategies.

5.1 Construction and Application of Generative AI-Assisted Teaching Models

The introduction of AI technology will transform the teaching model from a "teacher-student" binary structure to a "teacher-AI-student" ternary structure. This not only expands human thinking but also complements human capabilities, stimulating learning motivation, and enhancing self-directed learning and innovative thinking abilities[15][16]. In university programming courses, generative AI can provide strong technical support for teaching innovation. By implementing strategies such as personalized learning, project-based learning, and multi-source generative learning materials, while emphasizing the cultivation of programming thinking, students' learning outcomes can be effectively improved, meeting their individual needs and fostering practical skills, programming thinking, and innovation capabilities[16].

5.2 Optimization and Reconstruction of Course Content for the AI Era

Course design should incorporate content on human-AI collaborative programming, systematically guiding students on how to efficiently utilize AI tools and fully use the advantages of human-AI collaboration. The core objective of the course should shift from traditional code-writing skill development to higher-level competencies such as problem analysis, algorithm design, and system architecture. Although mastering basic coding knowledge remains essential, cultivating the ability to collaborate with AI to complete complex tasks is particularly important in the context of

rapid AI development. Therefore, courses should reduce the time spent on basic code writing and instead focus on training students in the efficient and appropriate use of generative AI tools, helping them better adapt to future technological demands.

5.3 Innovation and Reform of the Course Assessment and Evaluation System

Assessment methods should place greater emphasis on open-ended projects and the evaluation of practical abilities. For example, comprehensive project tasks can be designed, requiring students to complete project designs and deliver presentations, while reducing the weight of traditional paper-based exams. Additionally, the proportion of traditional programming assignments should be decreased, with basic programming learning and practice integrated into classroom activities. This approach helps prevent students from directly submitting AI-generated content due to laziness after class and ensures better supervising of their foundational knowledge acquisition process.

Furthermore, it is recommended to incorporate tasks in the assessment system that require the use of AI tools, evaluating students' ability to solve problems with AI. This ensures that the assessment methods comprehensively reflect students' overall competencies, including technical application skills, problem-solving abilities, and human-AI collaboration capabilities. Such reforms will better align with the new demands of the AI era for programming talent.

6. Conclusion

Generative AI not only provides intelligent and personalized learning tools for programming education but also poses new requirements for traditional teaching models and assessment methods. To address these changes, universities urgently need to implement systematic reforms in curriculum design, teaching methods, and evaluation systems. In the future, as generative AI technology continues to advance, programming education should place greater emphasis on fostering human-AI collaboration and higher-order thinking skills. Educators should actively explore the deep integration of AI and education, laying the foundation for cultivating talents in the era of artificial intelligence.

References

[1] DeepSeek-AI et al. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning[J/OL]. ArXiv abs/2501.12948, 2025. [Accessed: February 25, 2025]. Available: https://arxiv.org/abs/2501.12948.

[2] DeepSeek-AI et al. DeepSeek-V3 Technical Report[J/OL]. ArXiv abs/2412.19437, 2024. [Accessed: February 25, 2025]. Available: https://arxiv.org/abs/2412.19437.

[3] Achiam, OpenAI Josh et al. GPT-4 Technical Report.[J/OL]. ArXiv abs/2303.08774, 2023. [Accessed: March 11, 2025]. Available: https://arxiv.org/abs/2303.08774.

[4] Duan Weiwen, Yu Meng. Lifelong Creativity Cultivation in the Era of Generative Artificial Intelligence[J]. Studies on Science Popularization, 2024, (02): 13-22+38+101-102.

[5] Zhang Xi, Yang Xiaoshan, Xu Changsheng. Current State and Future Development Directions of ChatGPT and Generative Artificial Intelligence[J]. Bulletin of National Natural Science Foundation of China, 2023, (05): 743-750.

[6] Xiao Yanghua. Generative Language Model and Artificial General Intelligence: Connotation, Approach and Implications[J]. Renming Luntan Xueshu Qianyan, 2023, (14): 49-57.

[7] Tu Liangchuan. The Philosophical Narrative of"Generative Artificial Intelligence" Approaching General Intelligenc[J]. Journal of Northeast Normal University, 2023, (04): 40-47+93.

[8] Liu Bangqi, Nie Xiaolin, Wang Shijin, et al. Generative Artificial Intelligence and the Reshaping of Future Education:Technical Framework,Capability Characteristics and Application Trends[J]. E-education Research, 2024, (01): 13-20.

[9] Faisal Kalota. A Primer on Generative Artificial Intelligence[J/OL]. Education Sciences, 2024, 14(2): 172. [Accessed: February 26, 2025]. Available: https://doi.org/10.3390/educsci14020172.

[10] Bo Junge, Qiao Yanan, Qi Qi, et al. Exploring Application Potential and Challenges of AIGC Technology in

University Programming Courses[J]. Computer Technology and Development, 2024, (06): 214-220.
[11] Sun Dan, Zhu Chengcong, Xu Zuodong, et al. A Study on Analysis of College Students' Programming Learning Behavior Based on Generative Artificial Intelligence[J]. E-Education Research, 2024, (03): 113-120.
[12] Wang Yuxuan, Xu Wenhao, Yu Haomiao, et al. Challenges and Opportunities Brought by Generative AI to C Language Programming Teaching[J]. Computer Education, 2024, (08): 133-141+145.
[13] Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, et al. Sparks of Artificial General Intelligence: Early Experiments with GPT-4[J/OL]. ArXiv abs/2303.12712, 2023. [Accessed: February 26, 2025]. Available: https://arxiv.org/abs/2303.12712.
[14] Tom Farrelly, Nick Baker. Generative Artificial Intelligence: Implications and Considerations for Higher Education Practice[J/OL]. Education Sciences, 2023, 13(11): 1109. [Accessed: February 26, 2025]. Available: https://doi.org/10.3390/educsci13111109.
[15] Yang Zongkai, Wang Jun, Wu Di, et al. Exploring the Impact of ChatGPT/AIGC on Education and Strategies for Response[J]. Journal of East China Normal University (Educational Sciences), 2023, (07): 26-35.
[16] Zhang Hongzhuo, Zhou Xiaobao, Xu Yuhuan, et al. Generative Artificial Intelligence Empowering Innovation in Computer Programming Course Teaching[J]. Computer Education, 2024, (07): 44-48.