Theory and Practice of Science and Technology
2025, VOL. 6, NO. 7, 110-113
DOI: 10.47297/taposatWSP2633-456922.20250607

Exploration and Practice of Computer Basic Subject Teaching with Digital-Intelligent Integration: Taking the Operating System Course as An Example

Yayi Huang, Guijuan Guo*

Modern Information Industry College, Guangzhou College of Commerce, Guangzhou, Guangdong, 511363, China

ABSTRACT

This paper focuses on the teaching reform of fundamental computer science courses in the context of digital-intelligent integration. Using the Operating system course as a case study, it explores the application pathways of digital-intelligent technologies throughout the entire teaching process. By analyzing issues in traditional teaching methods, such as abstract theories, rapid technological updates, and low student engagement, this paper proposes a practical approach to optimize teaching through digital-intelligent tools, including artificial intelligence large-model platforms, digital human tools, and knowledge graphs. It deeply integrates teaching and learning, constructing a closed-loop digital-intelligent teaching model that spans the pre-class, in-class, and post-class stages, and provides an empirical analysis of the teaching outcomes. The results show that the digital-intelligent teaching model significantly improves students' overall performance and autonomous learning ability, offering feasible reference and practical support for the digital-intelligent teaching reform of foundational computer science courses and related engineering disciplines.

KEYWORDS

Digital-intelligence integration; Operating system; Teaching exploration; Practical research

1 Introduction

Digital-intelligent technology is the product of the deep integration of digitalization and intelligence, encompassing fields such as big data, artificial intelligence, and cloud computing. With the development of technology, digital-intelligent technologies are profoundly reshaping the way society functions and produces. At the same time, they have impacted traditional educational models. In this context, how to empower education through digital-intelligent technologies has become an unstoppable trend of the times. This is especially significant for the computer science discipline, where one of the core educational goals is not only to help students build a strong foundation of professional knowledge but also to cultivate their spirit of inquiry into cutting-edge technologies and innovation. This holds significant practical implications. This paper takes the Operating System course as a case study to explore the teaching practices and applications of digital-intelligent technologies in foundational computer science education, aiming to provide a reference for the innovation of teaching models in other courses.

2 Current situation analysis

Operating System course is a fundamental required course in the computer science discipline, positioned as a core subject in the curriculum. The Operating System is the most basic and core software in a computer system. It controls and manages various resources within the system and serves as a crucial interface between other software and computer hardware, providing essential support for user services. This course primarily introduces the design methodologies, technical principles, and implementation algorithms of operating systems. It also expands on the latest technologies by discussing the practical applications of major operating systems such as Linux, Windows, and Unix. Studying this course helps students master fundamental knowledge and skills in computer science, enhances their overall understanding of computer systems, and provides a solid foundation for further studies in other computer science courses.

However, this course is extensive, highly interdisciplinary, and covers a broad range of topics. The theoretical knowledge involved is relatively abstract and difficult for students to understand. At the same time, the technologies used in operating systems are continuously evolving, and the related application scenarios and experimental designs must keep pace with these developments. In this context, leveraging digital-intelligent technologies to empower the course teaching becomes particularly important. On one hand, by utilizing information technologies such as big data and artificial intelligence, teachers can optimize the course content and present it to students in more intuitive formats, such as images and videos. This helps overcome the bottleneck in traditional Operating System course teaching, where abstract concepts are hard to grasp, and aids students in building a systematic knowledge framework. On the other hand, through the networked search capabilities of generative models, teachers can stay updated on the latest application scenarios of operating systems and integrate them into the course. Additionally, experimental activity designs can be optimized based on real-world needs, enhancing students' ability to connect theory with practice.

* **Corresponding Author:** Guijuan Guo, guoguijuan727@163.com

In recent years, many scholars have conducted related explorations and research on empowering education through digital-intelligent technologies, achieving certain results. For example, Fang et al [1]. explored the connotations and value of using knowledge graphs to empower ideological and political education in university courses under the context of digital-intelligent education, and designed a corresponding teaching model. Yu et al [2]. focused on how to apply artificial intelligence to empower the entire teaching process, as well as exploring the transformation of educational models. Li et al [3]. discussed the application of modern intelligent technologies in teaching activities, with a focus on programming project experiments. Based on the achievements of previous research, this paper focuses on the overall closed-loop design of the Operating System course, deeply integrating domestic tools such as DeepSeek and Huawei Cloud, addressing the challenges faced by traditional course teaching. The paper also conducts an empirical evaluation of teaching practices to verify the improvement in learning outcomes through digital-intelligent empowerment.

3 Teaching exploration and practice

3.1 Optimizing pre-class preparation with digital-intelligent tools

The Operating System course involves a large amount of theoretical content, with many concepts that are difficult to understand, and the rapid pace of technological updates. Therefore, leveraging digital-intelligent tools can optimize the course content, analyze key and difficult points, and design teaching activities, significantly improving the teacher's preparation efficiency.

Knowledge is actively constructed through social interaction within real-life contexts. During the course design phase, instructors can use digital-intelligent tools to introduce enterprise-level case studies and relevant application scenarios, allowing students to combine theory with practice. For example, when explaining processor privilege levels in operating systems, instructors can use artificial intelligence platforms such as DeepSeek and Doubao to design case study backgrounds, linking cloud computing trends with operating system privilege levels and interrupt mechanisms. This prompts students to consider why Docker containers might increase escape risks in privileged modes and how system attackers can exploit interrupts to trigger security vulnerabilities. Students can then be encouraged to propose corresponding solutions. By creating real-world scenarios with digital-intelligent tools to guide students in analyzing and solving problems, their ability to apply knowledge comprehensively is enhanced, and their awareness of information security is strengthened.

Additionally, by using digital tools such as Mofa Youyan and Huawei Cloud MetaStudio, teachers can create video-based teaching materials by writing scripts, with 3D digital humans explaining the content to students. This increases both the fun and intuitiveness of the lessons. Combining video explanations and demonstrations by digital humans, instructors can assign preview tasks to students, allowing them to gain a preliminary understanding of the course content before class. During this process, teachers can also use teaching platforms like Chaoxing Fanya for learning analytics to assess students' pre-class preparation. This enables instructors to focus on the difficult parts of the content during the lesson, achieving a blended teaching model that combines both online and offline components. This approach breaks the time and space limitations of traditional teaching methods, making education no longer confined to the classroom, while also helping students better understand the course material and improving their ability to learn independently.

3.2 Empowering classroom teaching with digital-intelligent tools

In traditional Operating System course teaching, many concepts are relatively abstract and difficult to explain, making them hard for students to understand. By using digital-intelligent tools, instructors can present the teaching content in a visual format, helping students better understand and retain the material. For example, animations can be used to simulate the process of process synchronization, presenting it in an intuitive way to students, which also increases the engagement and enjoyment of the class. Moreover, traditional classroom board writing often relies on the teacher's experience. However, with the help of digital-intelligent tools for analysis and image generation, instructors can organize clearer blackboard-writing structures and frameworks, thus aiding students in understanding abstract concepts.

During the classroom teaching phase, instructors can guide students to actively use digital-intelligent tools for learning, exploring ways to cultivate students' digital-intelligent learning abilities, thereby creating a dynamic where the teacher leads and the students are the main participants[4]. Taking the Operating System course as an example, teachers can leverage large-model APIs to deeply integrate specialized course content with carefully designed prompts, building a course intelligent agent to assist in teaching and guiding students in using prompt engineering to retrieve key knowledge points. Students, in turn, can use the course intelligent agent to conduct the questions and answers, quickly associating knowledge points and achieving knowledge transfer and expansion, enabling them to apply what they have learned to other contexts. This interactive learning mode can stimulate students' interest in learning and improve teaching efficiency.

This course reconstructs the experimental teaching system with a focus on progressively advancing competencies. By relying on course intelligent agent technology and integrating mainstream operating systems such as Ubuntu and openEuler, a three-tiered experimental design is implemented according to students' cognitive levels. For example, in the operating system process experiment, the basic competency task requires students to write a C program to simulate process control, including generating child processes, outputting their PIDs, and terminating specified processes. The intermediate competency task asks students to design a producer-consumer model using multi-processing (C fork()). The advanced competency task, using the example of the multi-sensor data acquisition system in the Xiaomi car SU7 Ultra Pro, involves optimizing the producer-consumer model using the semaphore mechanism. Throughout the activities, students can use course intelligent agents to assist with programming and complete the project through group discussions. This design deeply integrates digital-intelligent tools throughout the entire experimental process, enhancing students' system design thinking and engineering practical skills.

In addition, within the context of digital-intelligent integration, instructors can guide students to consider how traditional operating systems should address issues such as intelligent scheduling, resource allocation, security isolation, and ecosystem compatibility in AI-integrated environments. In doing so, they can encourage students to explore innovative practices in optimizing operating system kernel design, fostering students' creative thinking. For example, students can use intelligent agents to analyze the strengths and weaknesses of existing scheduling algorithms and propose improvements tailored to AI workloads.

According to observations, compared to traditional teaching methods, after incorporating digital-intelligent tools into the classroom teaching phase, students show higher levels of participation and engagement. They are able to think more deeply about classroom issues and present their own insights. This also indicates that using digital-intelligent tools in classroom teaching can significantly enhance students' motivation to learn.

3.3 Using digital-intelligent tools to assist in after-school reinforcement

The internalization and mastery of knowledge require continuous reinforcement. In the after-school reinforcement phase, instructors can use digital-intelligent tools to automatically generate tiered exercises based on the knowledge content of each lesson and students' classroom performance, and push them through learning platforms according to students' different levels. For students with weaker foundations, exercises can focus on conceptual understanding, helping them grasp the core knowledge points of the course. For intermediate-level students, the after-school exercises can incorporate code practice, such as asking students to write a simple time-slice round-robin scheduling program to deepen their understanding of course concepts through programming. For advanced students, more open-ended and exploratory learning tasks can be designed, such as asking them to design a processor scheduling strategy for a new multi-task embedded system, in order to enhance their system design thinking. Using digital-intelligent tools, instructors can set graded exercises and dynamically optimize the structure of the questions based on students' responses, thus distinguishing between students of different ability levels and achieving personalized learning outcomes.

Using the knowledge graph engine based on digital-intelligent tools, instructors can organize the core knowledge points of the course and their interrelationships [5]. For example, in the chapter on processes and process management, instructors can generate mind maps through platforms like DeepSeek and ChatGPT to logically connect knowledge points such as process control, thread management, and synchronization and mutual exclusion. This helps students form a systematic knowledge framework and explore intersecting knowledge points with cutting-edge fields like AI, big data, and the Internet of Things, thereby building a comprehensive knowledge base.

In addition, regarding the enrichment of online and extracurricular resources, instructors can utilize platforms such as Huawei Talent Online to establish online training courses for the openEuler system. On one hand, students can access cutting-edge operating system technologies, expand their skill set, and meet their diverse learning needs. On the other hand, after completing the online courses, students can earn exam vouchers for professional certifications such as HCIA and HCIP. These certifications help students deepen their understanding of the knowledge in the certification process and enhance their core competitiveness for future employment. At the same time, instructors can encourage students to participate in the Huawei ICT Competition, where they can showcase their knowledge, promoting both teaching and learning through competition.

4 Teaching practice results and analysis

In order to assess the varying levels of knowledge mastery among different student groups, we adopted a comprehensive evaluation method for the students' overall performance, which includes both formative and summative assessments. The summative assessment, which accounts for 50% of the final grade, is the final exam. It is conducted in the form of a closed-book written test to evaluate students' mastery of the fundamental course knowledge, with a total of 100 points. The formative assessment, also accounting for 50% of the final grade, evaluates students' daily performance in four areas: attendance, homework, experimental operation, and comprehensive performance. The full score for the formative assessment is 100 points, and the grading criteria are outlined in Table 1. Attendance includes class arrival, absences, tardiness, early departure, and leave requests. Homework consists of group assignments, paper-based assignments, computer-based assignments, and question selection on the Chaoxing Fanya platform. Experimental operation includes, but is not limited to, system configuration, user and group management, process management, thread management, job management, memory management, device management, file operations, and more. Comprehensive performance includes question-answer process in class, discussions, and hands-on tasks. By utilizing digital-intelligent technologies, instructors can dynamically manage students' formative assessments and analyze learning progress, which allows them to better understand students' grasp of the course content and adjust the teaching material and pace accordingly.

Table 1 Criteria for calculating formative assessment scores

Project	Formative assessment				Full marks
	Attendance	Homework	Experimental operation	Comprehensive performance	
Weight	10%	35%	35%	20%	100
Fraction	100	100	100	100	

Taking the Operating System course as an example, a comparison of the comprehensive scores for the two semesters shows that, in the second semester of the 2024-2025 academic year, students' overall performance was better than in the second semester of the 2023-2024 academic year. Specifically, both the average score and the pass rate increased, while the failure rate decreased, as shown in Table 2. This reflects, to some extent, the significant improvement in teaching outcomes after integrating intelligent technologies into the course. On one hand, the increase in the average score and good quality rate indicates that students were more engaged in class, with better completion of assignments and experiments. At the same time, they demonstrated a deeper understanding of the knowledge points in the final exam, with higher accuracy in their answers. On the other hand, the personalized guidance enabled by intelligent technologies helped prevent students from falling behind, leading to a decrease in the failure rate.

Table 2 Comparison of overall performance

Semester	Number of students	Average score	Good quality rate	Failure rate
2023-2024-2	111	72.2	23%	3%
2024-2025-2	109	75.9	36%	1%

Based on feedback from teaching evaluations and student interviews, 93% of students believe that digital intelligence tools significantly enhanced their learning outcomes in the course. For example, the visual presentation of digital humans, animations, and knowledge graphs effectively reduced the difficulty of understanding course content. Meanwhile, AI-powered intelligent agents facilitated task analysis and explanation, stimulating students' thinking about experimental design. These advantages not only encouraged students to take a more proactive approach to exploring knowledge but also deepened their understanding of the course content.

5 Conclusion

Against the backdrop of the rapid development of digital tools and artificial intelligence, the deep integration of digital intelligence technologies into courses has become an inevitable requirement for the future transformation and development of education. This paper explores the application of digital intelligence technology in the Operating System course, focusing on the pre-class preparation, in-class teaching, and after-school consolidation stages. It also analyzes and discusses the results of the teaching practice.

Indeed, empowering teaching with digital intelligence technologies still carries certain risks. For instance, students may become overly reliant on the prompts provided by AI platforms, lacking the spirit of independent exploration. Additionally, digital intelligence platforms are vulnerable to issues like network fluctuations or server failures, which may disrupt timely responses to teaching needs. However, we should not let these challenges prevent us from embracing the opportunities brought by digital intelligence in education. In the face of both opportunities and challenges, we should maintain a positive and open mindset, continuously innovate teaching methods, and improve our ability to apply digital tools and design effective lessons. At the same time, as educators, we must guide students in the proper use of digital intelligence tools, provide value-based leadership, and cultivate their digital literacy and independent learning capabilities.

References

[1] Fang Hongjun, Cai Hong, Wang Xinyu. Research and design of ideological and political teaching empowered by knowledge graph in the context of digital education[J]. Journal of Beijing Union University, 2025, 39(2): 7-14.

[2] Yu Guoming, Li Fan, Teng Wenqiang. AI+Education: Upgrading and Transforming Teaching Model in the Era of Artificial Intelligence[J]. Ningxia Social Sciences, 2024, (2): 191-198.

[3] Li Xue, Fan Qinggang, Wang Zhong, et al. Exploration of project-based teaching mode for AI-enabled programming courses[J]. Computer Education, 2025, (5): 33-38.

[4] Yuan Ying, Meng Fengjuan. Digital intelligence empowerment, cross-integration - Promoting the reform of university public mathematics teaching under the background of "four new"[J]. Science and Technology Wind, 2025, (11): 74-76.

[5] Liu Shuang. Application of artificial intelligence (AI) + knowledge graph in blended teaching[J]. Office Automation, 2024, 29(7): 42-44.