ISSN 2522-6398 Vol. 8, Issue 10: 53-60, DOI: 10.25236/FER.2025.081008

Research on the Application of Large Language Models in Embedded Systems Education

Chenrui Wu[a,*], Yao Huang[b], Zhenzhong Chu[c]

School of Mechanical Engineering, University of Shanghai for Science and Technology, Shanghai, China
[a]wuchenrui@usst.edu.cn, [b]huangyao@usst.edu.cn, [c]zhenzhongchu@usst.edu.cn
**Corresponding author*

Abstract: *Large language models in artificial intelligence are profoundly transforming work methodologies across various domains. Undergraduate education must accelerate pedagogical reform to adapt to this technological transformation and cultivate high-quality professionals capable of proficiently utilizing artificial intelligence tools. This paper examines the application potential of large language models in three key aspects: knowledge graph construction, instructional implementation, and engineering practice, using embedded systems courses as a case study. Through the integration of large model technology, more intelligent knowledge point recommendations and personalized learning pathway planning can be achieved. The introduction of real-time interaction and intelligent question-answering systems in classroom instruction facilitates enhanced student engagement and comprehension capabilities. In engineering practice components, large models can be leveraged to generate rich case studies and simulation scenarios, thereby strengthening students' hands-on competencies. This paper further discusses the implementation pathways and developmental trends of AI large language models in embedded systems education.*

Keywords: *Large Language Model, Teaching Application, Embedded System, Knowledge Graph*

1. Introduction

The rapid advancement of artificial intelligence technology has brought unprecedented opportunities and challenges to the educational sector. The introduction of large model technology has not only enhanced teaching efficiency but also provided new possibilities for personalized learning[1,2]. In the teaching process, large models can leverage deep learning and natural language processing technologies to precisely analyze students' learning behaviors and knowledge mastery, assisting teachers in dynamically adjusting instructional strategies. Simultaneously, artificial intelligence can optimize educational resource allocation and promote the sharing of high-quality educational resources. As AI technology continues to evolve, its educational applications will expand continuously, driving the transformation of teaching models from “teacher-centered” to “student-centered” approaches, making the teaching process more intelligent and humanized[3]. This transformation not only places higher demands on students' innovative capabilities but also injects new momentum into higher education pedagogical reform.

Embedded systems, as a core course in robotics engineering programs, possesses both high theoretical and practical characteristics. This major, actively promoted as a new engineering discipline by the state, encompasses multidisciplinary content including mechanical engineering, electrical engineering, mechanics, electronic information, and artificial intelligence. Compared to electrical engineering and computer science programs, the embedded systems course in robotics programs features compact scheduling and intensive workload. The course content covers multiple layers including hardware architecture, operating systems, and driver development, characterized by substantial information content, high entry barriers, and strong practical orientation. Traditional teaching models exhibit obvious deficiencies in content updating, practical resource allocation, and responding to students' individualized needs. The introduction of AI large model technology can effectively address these challenges. This paper systematically explores the application pathways of large models in embedded systems courses from three aspects: instructional design, teaching implementation, and engineering practice.

ISSN 2522-6398 Vol. 8, Issue 10: 53-60, DOI: 10.25236/FER.2025.081008

2. Knowledge Graph Construction Assisted by Large Models

2.1 Traditional Knowledge Graph Construction

Currently, knowledge graphs for embedded systems courses predominantly adopt hierarchical tree structures, where associations between child nodes often rely on manual design and filtering by teachers. This approach is not only inefficient in construction but also struggles to flexibly reflect the complex intersections and mapping relationships between knowledge points. This construction method appears particularly inadequate when facing embedded systems courses that integrate multidimensional content spanning hardware, operating systems, and driver development. Traditional graphs are slow to update and poorly scalable, unable to adapt to rapidly iterating technological developments and personalized teaching needs, thereby limiting the efficient organization and intelligent delivery of educational resources.

With the introduction of AI large model technology, the construction methodology of knowledge graphs undergoes fundamental transformation. Large models can perform deep semantic analysis and contextual understanding of multimodal educational resources including textbooks, courseware, academic papers, and experimental manuals, automatically extracting core concepts, key technologies, and practical points to form structured knowledge nodes. Furthermore, models can autonomously establish associative relationships between concepts based on curriculum outlines and knowledge logic, dynamically constructing and optimizing knowledge topology. For example, by analyzing semantic correlations and co-occurrence frequencies between “real-time operating systems” and “task scheduling” models can automatically establish multiple associative edges with different weights, thereby forming a dynamic and scalable knowledge network. This network not only encompasses static knowledge structures but can also integrate student learning behavioral data to achieve knowledge relationship optimization and personalized presentation based on authentic feedback.

2.2 AI-based Embedded System Knowledge Graph Construction

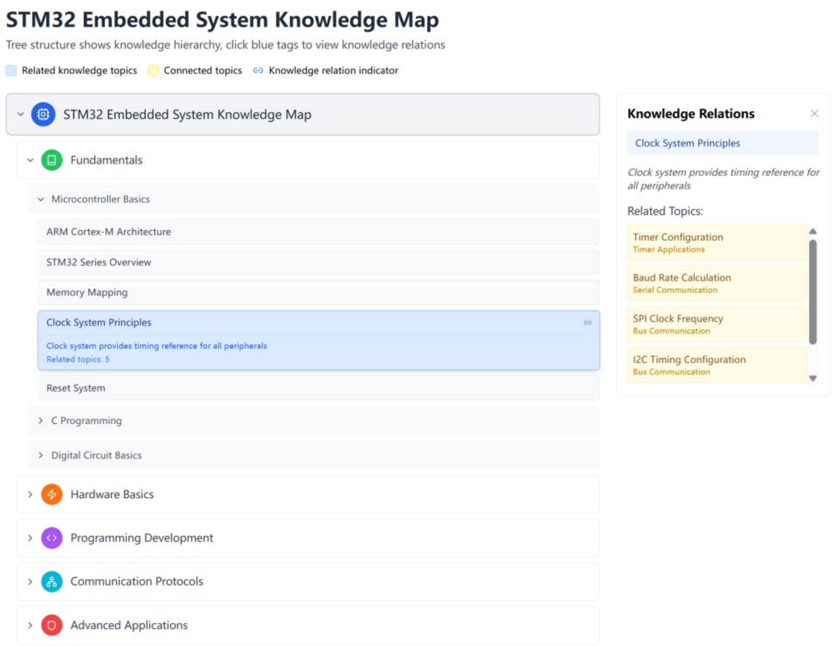

Figure 1 Knowledge graph structure centered on embedded systems

This intelligent knowledge organization approach provides powerful support for both teaching and learning. For students, the system can extract the most suitable knowledge points for their current stage from the graph based on real-time learning status (such as knowledge mastery level, error types, and retrieval habits), and generate pathway planning that aligns with individual advancement needs. For teachers, visualization analysis tools can help identify knowledge weakness areas for the entire class or individual students, providing data evidence for adjusting teaching strategies. For instance, for knowledge points like “interrupt mechanisms” and “concurrency control” where most students experience comprehension difficulties, teachers can focus on designing case explanations and experimental sessions, thereby enhancing the targeting and effectiveness of instruction. This data-driven,

ISSN 2522-6398 Vol. 8, Issue 10: 53-60, DOI: 10.25236/FER.2025.081008

dynamically adjusting knowledge graph is gradually becoming the core infrastructure for achieving large-scale differentiated instruction.

Figure 1 demonstrates a knowledge graph structure centered on core embedded systems knowledge, radiating outward to multiple dimensions including hardware architecture, operating systems, driver development, and others, clearly presenting the multi-layered and multi-directional expansion characteristics of course content. This structure not only embodies systematic and logic but can also dynamically adjust knowledge presentation according to students' learning progress. For example, for students with weaker hardware foundations, large models can prioritize recommending content related to microprocessor principles and interface technologies; for students who have mastered fundamentals, the system can guide them toward advanced content such as real-time operating systems and embedded software development. Such personalized recommendations help enhance students' depth of understanding of complex knowledge and learning initiative, assisting teachers in identifying critical competency nodes and achieving differentiated instruction.

3. Teaching Implementation Based on Intelligent Question-Answering

3.1 Current Challenges in Traditional Classroom Teaching

Currently, embedded systems courses in universities commonly adopt medium-sized class instruction, with student populations typically ranging from 30 to 40 individuals. In this educational environment, teachers face a persistent contradiction: the tension between limited classroom time and students' individualized tutoring needs[5]. Beyond theoretical explanations and experimental demonstrations, teachers struggle to find sufficient time for in-depth responses to each student's questions. Post-class tutoring is further constrained by temporal and spatial limitations, making it impossible to provide timely and comprehensive learning support. This dilemma is particularly pronounced in practice-intensive courses with abstract concepts like embedded systems, directly affecting students' learning experiences and knowledge mastery outcomes.

3.2 AI-Powered Intelligent Question-Answering Systems

Intelligent question-answering systems based on AI large models provide innovative solutions to address this instructional challenge. These systems, leveraging advanced large language models[4] (such as DeepSeek), construct intelligent question-answering frameworks covering the entire teaching process. Architecturally, the system deeply integrates natural language processing, knowledge graphs, and machine learning technologies, enabling accurate understanding of students' natural language questions and retrieving and generating precise answers from specialized knowledge bases in the embedded systems domain. Unlike traditional keyword-matching question-answering systems, large model-driven systems possess deep semantic understanding capabilities, allowing them to identify the authentic intent behind questions. Even when facing incomplete or ambiguous expressions, they can gradually clarify student confusion through multi-turn dialogues.

3.3 Core Advantages and Capabilities

The system's core advantages manifest across several dimensions: First, it provides 24/7 uninterrupted tutoring services, enabling students to ask questions anytime and anywhere, breaking the temporal and spatial limitations of traditional tutoring. Second, the system supports multi-turn dialogue interactions, simulating human teachers' heuristic teaching approaches by helping students autonomously construct knowledge systems through progressive guidance. For instance, when a student asks "why do we need real-time operating systems," the system does not provide direct answers but guides students to independently consider relationships between concepts such as real-time requirements and task scheduling mechanisms through a series of guiding questions. This Socratic questioning approach significantly enhances students' critical thinking and problem-solving abilities.

More importantly, intelligent question-answering systems possess continuous evolution capabilities. The system automatically records and analyzes all question-answering interaction data, continuously optimizing question-answering strategies and knowledge base organization through machine learning algorithms. When discovering that multiple students have common questions about the same knowledge point, the system automatically marks the issue as a teaching priority and generates specialized explanatory materials and typical exercises. Simultaneously, the system dynamically adjusts response

ISSN 2522-6398 Vol. 8, Issue 10: 53-60, DOI: 10.25236/FER.2025.081008

depth and detail based on students' cognitive levels and learning progress, achieving genuine personalized teaching support.

3.4 Data-Driven Pedagogical Improvement

From a data-driven pedagogical improvement perspective, intelligent question-answering systems generate tremendous value. Student question-answering data recorded by the system becomes a valuable resource for understanding student learning conditions. Through big data analysis, teachers can accurately identify difficulties and blind spots in instruction, discovering common misconceptions and confusion among students. For example, analysis might reveal that many students have biased understanding of "interrupt nesting" concepts or lack intuitive recognition of "memory management unit" operating principles. These insights enable teachers to adjust teaching content and methods in a targeted manner, promptly addressing instructional deficiencies.

3.5 Full-Process Teaching Integration

In actual teaching implementation, intelligent question-answering systems are deeply integrated into the entire embedded systems course process. During pre-class preview phases, the system proactively pushes relevant background knowledge and preview materials based on teaching progress, guiding student thinking about key difficulties through preset questions. Students can ask questions at any time during preview, receiving immediate feedback that helps clear knowledge obstacles in advance. During classroom instruction, the system serves as an intelligent teaching assistant, responding in real-time to questions submitted by students through mobile terminals, enabling teachers to quickly understand student comprehension and adjust instruction pace and emphasis accordingly. During experimental practice phases, the system particularly strengthens support for programming and debugging issues, helping students analyze code errors, explain hardware phenomena, and provide practical solution recommendations. During post-class review phases, the system generates personalized review plans and exercise sets based on each student's question-answering history and individual knowledge mastery, focusing on reinforcing knowledge consolidation in weak areas.

3.6 Transformative Impact on Teaching Models

Notably, intelligent question-answering system implementation also brings profound transformation to teaching models. Traditional unidirectional knowledge transmission gradually transforms into bidirectional, interactive knowledge construction processes. Students evolve from passive knowledge receivers to active knowledge explorers, while teachers transition from simple knowledge transmitters to learning process guides and organizers. This transformation significantly improves student learning initiative and engagement, cultivating autonomous learning and lifelong learning capabilities.

4. Large Model-Assisted Engineering Practice

4.1 Transforming Traditional Development Challenges

Since ChatGPT's launch at the end of 2022 garnered global attention, generative artificial intelligence technology, particularly large language models (LLMs), has demonstrated powerful capabilities in code generation and comprehension, profoundly transforming software engineering education and practice models. Embedded systems development, as a crucial field connecting hardware and software, has long faced challenges in teaching including complex development environments, difficult debugging, and strong hardware dependencies. The introduction of large model technology provides innovative solutions to these challenges, significantly enhancing development efficiency and teaching quality.

In traditional embedded systems teaching practice, students often spend considerable time addressing fundamental issues such as development environment configuration, syntax error troubleshooting, and low-level driver debugging. Taking commonly used traditional IDEs like Keil MDK as examples, their code editing and debugging functions are relatively basic, lacking intelligent development assistance capabilities. This leads students to invest excessive energy in syntactic details and environment debugging, making it difficult to focus on cultivating core competencies such as system architecture design and algorithmic innovation. The emergence of large model-assisted programming technology has fundamentally improved this situation.

ISSN 2522-6398 Vol. 8, Issue 10: 53-60, DOI: 10.25236/FER.2025.081008

4.2 Intelligent Development Support and Implementation

By migrating development environments to modern IDEs supporting large model assistance (such as VS Code) and combining plugin systems specifically optimized for embedded development (such as Embedded IDE and Trae plugins as shown in Figure 2), students can obtain comprehensive intelligent programming support. In actual teaching applications, these tools demonstrate value across multiple dimensions: First, during code writing phases, systems can intelligently generate code suggestions conforming to embedded C language specifications based on natural language descriptions or code snippets, including common patterns such as peripheral initialization, interrupt service routines, and driver program frameworks. For example, when students input "initialize STM32's USART1," the system automatically recommends complete initialization code templates including key steps such as baud rate settings, data bit configuration, and interrupt enabling.

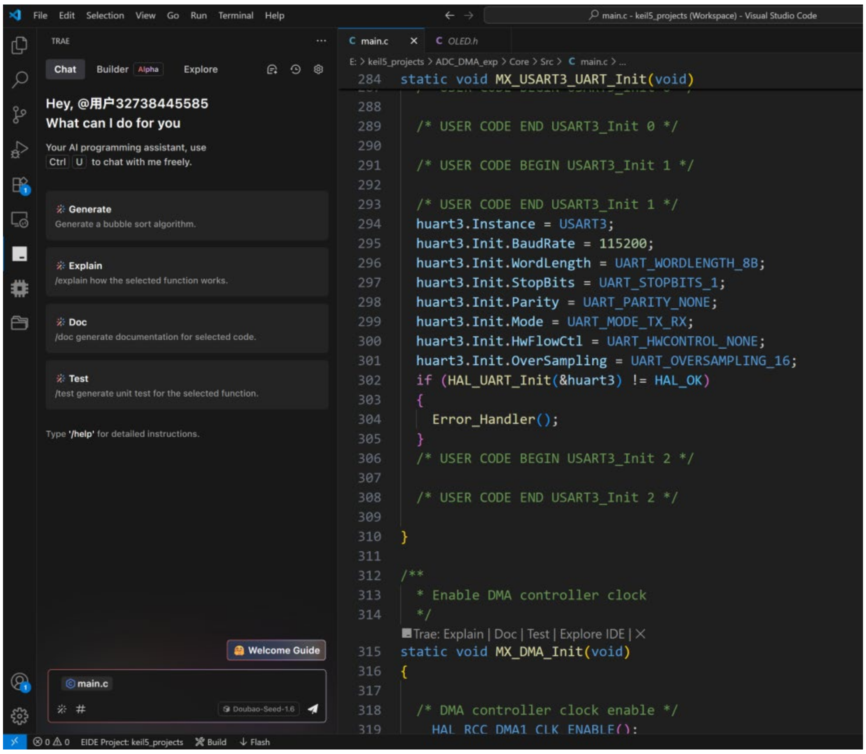

Figure 2 Embedded Development Environment Based on Trae and Embedded IDE

Second, in debugging and optimization phases, large models can analyze code logic, identifying potential error patterns and performance bottlenecks. Systems can not only detect syntax errors but also discover problems unique to embedded development, such as register configuration conflicts, improper interrupt priority settings, and memory access violations. More importantly, systems can provide explanatory error analysis and modification suggestions, helping students understand problem sources rather than simply correcting errors. This understanding-based debugging process significantly enhances students' troubleshooting capabilities and systems thinking abilities.

Additionally, large model-assisted tools greatly improve the learning experience for Hardware Abstraction Layer (HAL) and low-level drivers. In embedded development, hardware register configuration often involves extensive technical documentation consultation and bit-operation programming. Large models, by encapsulating common hardware operation patterns and providing intuitive API suggestions and configuration examples, lower barriers to low-level hardware programming. Students can more rapidly implement hardware function verification, dedicating more time to system-level functionality and innovative application development.

From a teaching management perspective, large model-based development platforms provide valuable data collection and analysis capabilities. Systems automatically record students' code writing processes, error distributions, debugging trajectories, and other behavioral data, providing teachers with quantified pedagogical insights. Through analyzing this data, teachers can accurately identify students' common difficulties with specific knowledge points, such as understanding levels of critical concepts like interrupt handling, memory management, or real-time performance assurance. These data-driven

ISSN 2522-6398 Vol. 8, Issue 10: 53-60, DOI: 10.25236/FER.2025.081008

insights enable teachers to adjust teaching emphasis timely, strengthening targeted explanations and practice for relevant content.

4.3 Pedagogical Innovation and Balanced Implementation

Notably, large model-assisted tool applications also bring innovation to teaching methods. Teachers can design more open-ended and challenging project tasks because students can leverage AI tools to resolve some fundamental technical issues, thereby investing cognitive resources in higher-level design and innovation activities. For example, students can focus more on iot device communication protocol design, energy consumption optimization strategies, or real-time performance tuning and other system-level issues rather than struggling with individual peripheral driver implementation.

However, attention must also be paid to potential dependency issues that large model-assisted programming might bring. Teaching must emphasize students' understanding and mastery of underlying principles, avoiding excessive tool dependence that leads to fundamental knowledge deficits. A reasonable approach positions large models as “intelligent assistants” rather than “code generators”. It guides students to critically use AI suggestions, understand underlying principles and considerations, and cultivate independent thinking and problem-solving abilities.

Overall, large model technology brings paradigmatic transformation to embedded systems engineering practice teaching. Through intelligent code assistance, debugging support, and data analysis, it not only significantly improves development efficiency and teaching quality but more importantly liberates students' cognitive load, enabling them to focus on system-level design and innovative implementation. This transformation aligns with new engineering education development directions, providing powerful technical support for cultivating embedded engineering talents with systems thinking and innovation capabilities.

5. In-Depth Discussion of Large Model-Assisted Teaching

5.1 Paradigmatic Transformation in Educational Models

The introduction of large model technology marks profound transformation occurring in teaching paradigms. Its application not only reconstructs traditional knowledge transmission models but more importantly injects new vitality into teaching interaction and collaboration mechanisms. By establishing real-time, dynamic feedback loops, large models can provide highly personalized learning guidance for students while granting teachers unprecedented capabilities to track and analyze student learning states and developmental trajectories. This transformation evolves teaching from teacher-centered unidirectional indoctrination to student-teacher collaborative, bidirectional interactive knowledge construction processes, promoting the formation of more open, inquiry-oriented learning environments.

5.2 Technical Challenges and Limitations

However, large model technology applications in educational instruction still face several important challenges that must receive full attention during advancement. The foremost issue is "hallucination." Since large models are essentially statistical models based on probabilistic generation, lacking deep understanding of the real world and fact-checking mechanisms, they may generate seemingly reasonable but actually incorrect or misleading information when producing output. In disciplines like embedded systems that require extremely high accuracy and reliability, such errors may cause serious cognitive bias and practical risks. For example, models might recommend inappropriate register configuration methods or generate code structures with potential security vulnerabilities. To address this issue, multi-layered protection mechanisms must be established in teaching: first, technical-level verification, such as integrating professional knowledge bases and rule engines for real-time validation of model outputs; second, teaching-level guidance, cultivating students' habits of critically using AI tools and emphasizing cross-verification and principle-based investigation of key knowledge points; third, institutional-level standards, clarifying large models' auxiliary positioning in teaching and establishing usage boundaries and review processes.

5.3 Student Dependency Risks and Mitigation Strategies

On the other hand, students' dependency risks regarding large models cannot be ignored. The

ISSN 2522-6398 Vol. 8, Issue 10: 53-60, DOI: 10.25236/FER.2025.081008

convenience of technology may lead students to over-rely on tool outputs, reducing opportunities for autonomous thinking and deep exploration, potentially weakening problem-solving abilities and innovative thinking formation in the long term. Particularly in programming and practice phases, simply copying and pasting model-generated code without understanding underlying mechanisms will seriously affect learning effectiveness. This "tool dependency" phenomenon involves not only cognitive dimensions but may also impact students' learning motivation and exploratory spirit.

Addressing this issue requires systematic thinking from educators at the instructional design level. First, the positioning of large models in teaching should be clearly defined as "assistance" rather than "replacement," emphasizing their value in reducing cognitive load and improving efficiency while warning against potential mental laziness. Second, teaching activities must be carefully designed to balance technology use with capability development. For example, "phased usage strategies" can be adopted, limiting large model use during fundamental concept learning phases and encouraging student autonomous exploration, while encouraging reasonable utilization of large models for technical implementation during complex project development phases. Additionally, metacognitive capability development should be strengthened, enabling students to consciously monitor and regulate their learning processes and reasonably assess when and how to use large model assistance tools.

5.4 Ethical, Equity, and Cultural Considerations

Large model applications in education involve multidimensional considerations including ethics, equity, and culture. Training data bias in models may lead to output content containing certain degrees of cultural or gender bias; disparities in students' conditions for accessing and using technological tools may exacerbate educational inequality; furthermore, excessive reliance on technological solutions may weaken emotional connections and humanistic care between teachers and students. These factors all require comprehensive consideration while advancing technological applications, through establishing multidimensional evaluation systems and continuous improvement mechanisms to ensure technological applications genuinely serve educational goal achievement.

6. Conclusion and Future Prospects

This paper systematically analyzes large model applications in embedded systems teaching practice, deeply exploring how artificial intelligence technology empowers higher engineering education reform from three dimensions: knowledge graph construction, teaching implementation optimization, and engineering practice improvement. Research indicates that large model technology significantly enhances embedded systems course teaching effectiveness and learning experiences through intelligent knowledge organization methods, personalized learning support, and efficient development assistance. At the knowledge management level, dynamic knowledge graphs constructed based on large models achieve systematized knowledge organization and adaptive presentation; at the teaching interaction level, intelligent question-answering systems provide round-the-clock, personalized learning support; at the engineering practice level, AI-assisted programming tools substantially lower technical barriers, enabling students to focus on system-level design and innovative implementation.

However, research also reveals important challenges facing large model applications in teaching. Technical-level "hallucination" problems require addressing through multi-layered verification mechanisms and critical thinking cultivation; educational-level tool dependency risks demand rethinking instructional design principles to seek balance between technological assistance and capability development; furthermore, issues such as ethical equity and cultural adaptability require consideration within broader contexts.

Looking toward the future, research on large model applications in education should continue deepening from several directions: first, domain-specific models more suitable for educational scenarios need development, enhancing output accuracy and pedagogical appropriateness through integration of educational theory and disciplinary knowledge; second, exploration of new human-machine collaborative teaching models should clarify role positioning and interaction mechanisms for teachers, students, and AI systems in teaching processes; third, comprehensive teaching effectiveness evaluation systems need establishment to quantitatively analyze large model application impacts on learning outcomes; finally, attention should be paid to ethical dimensions of technological applications, developing responsible usage guidelines and regulatory standards.

Future educational intelligentization development should not merely pursue superficial efficiency in

ISSN 2522-6398 Vol. 8, Issue 10: 53-60, DOI: 10.25236/FER.2025.081008

technological applications but should deeply consider how to reshape learning experiences and developmental pathways through deep integration of technology and education. Large models, as important achievements in artificial intelligence technology development, derive their genuine value from enhancing rather than replacing human teachers' roles, from expanding rather than limiting students' thinking spaces. Only through coordinated evolution of technological applications and educational concepts can we fully leverage large model technology's empowering effects and cultivate innovative engineering talents capable of adapting to future societal development.

References

[1] Xu Kenan. Exploration of the Application of Artificial Intelligence in Computer Science Teaching in Higher Vocational Education. Frontiers in Educational Research (2025), Vol. 8, Issue 7: 202-206. https://doi.org/10.25236/FER.2025.080729.

[2] Liu Tiezhu. Intelligent Technology-Enabled Development Path for Vocational Education Teachers in the Construction of an Education Powerhouse: A Study Based on DeepSeek Large Model. Frontiers in Educational Research (2025), Vol. 8, Issue 7: 165-173. https://doi.org/10.25236/FER.2025.080724.

[3] Jiao Xiaocong. Modernization Construction of Chinese-style Vocational Education "Dual-Qualified" Teachers: Era Significance, Logical Principles, and Action Strategies [J]. Vocational Education Research, 2024, (04): 12-18.

[4] Deng Jianpeng, Zhao Zhisong. DeepSeek's Breakthrough and Transformation: On the Regulatory Direction of Generative Artificial Intelligence [J/OL]. Journal of Xinjiang Normal University (Philosophy and Social Sciences Edition), 2025, 1-10.

[5] Grodotzki J, Upadhya S, Tekkaya A E. Engineering Education amid A Global Pandemic[J]. Advances in Industrial and Manufacturing Engineering, 2021, 3: 100058.