

Reform and Practice of General-Education Artificial Intelligence Course for Software Engineering Majors in Application-oriented Undergraduate Colleges Oriented Toward “AI+X” Integrated Competence

Huiying Zhang, Chaoyong Rong

Pujiang Institute of Nanjing Tech University, Nanjing 211200, Jiangsu, China

Copyright: © 2026 Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY 4.0), permitting distribution and reproduction in any medium, provided the original work is cited.

Abstract: Against the background of educational digital transformation and the “AI-plus” action strategy, general-education artificial-intelligence education for computer majors in application-oriented undergraduate colleges is confronted with dual challenges of “content homogenization” and “competence mismatch”. Taking the software engineering major as the research object and based on the “1+N” curriculum-system framework, this paper puts forward a reform scheme for general-education courses featuring hierarchical progression, tool empowerment, and industry-education integration. By constructing the “1+N+AI” teaching mode of “core general education + major-specific customization + large-model empowerment”, traditional “knowledge imparting” is transformed into “application-competence building”. Practice shows that this mode can effectively improve students’ AI literacy, interdisciplinary innovative thinking, and capabilities of solving complex engineering problems with AI tools. It provides an operable path for private application-oriented undergraduate colleges to cultivate compound-type “software engineering + AI” talents.

Keywords: Application-oriented undergraduate colleges; Software engineering; General-education artificial intelligence course; “1+N” mode; Large-model empowerment; Industry-education integration

Online publication: October 27, 2026

1. Introduction

With the explosive development of generative-AI technologies represented by large language models from 2024 to 2026, artificial intelligence has evolved from a single technical domain into a general-technology foundation driving industrial upgrading ^[1]. The *Opinions on Accelerating the Advancement of Educational Digitalization* (2025) issued by the Ministry of Education clearly requires universities to construct “general-purpose plus characteristic” general-education artificial-intelligence courses ^[2-3]. This policy

orientation indicates that general-education AI education in universities has shifted from an “optional assignment” to a “mandatory task”, and its curriculum quality bears a direct relation to the adaptability and competitiveness of application-oriented talents in the new round of scientific-and-technological competition. For private application-oriented undergraduate colleges, the talent-training objectives for software engineering majors differ from both “algorithmic theoretical research” in research-oriented universities and AI-tool operation in vocational colleges. The core competitiveness of application-oriented-undergraduate software-engineering talents lies in their capabilities of optimizing the software-development life cycle with AI technologies, constructing intelligent software systems, as well as systematic thinking for landing AI solutions in specific vertical industries such as smart agriculture and intelligent construction ^[4-5].

Nevertheless, current general-education AI courses in application-oriented undergraduate colleges commonly suffer from the pain point of “too profound theories for comprehension and too superficial practice for real-world usage”. Specifically, some colleges overemphasize structural deduction of neural networks and mathematical principles of back-propagation in teaching, and neglect students’ understanding of practical application paths of AI technologies in software-engineering scenarios. Though other courses introduce AI-tool operations, they merely stay at the invocation level and fail to guide students to embed AI capabilities into the whole software-development workflow. This dilemma of “reaching neither end” urgently calls for a middle-of-the-road path in curriculum design that balances cognitive rules and major-specific demands. Combining disciplinary characteristics of software engineering majors, this paper aims to explore a scientific and implementable reform path for general-education AI courses.

2. Current status and pain points of general-education AI courses for software engineering majors in application-oriented undergraduate colleges

In-depth analysis reveals that the causes for these contradictions include both objective-condition constraints and subjective-cognition deviations. Objectively, private colleges enroll students with diverse backgrounds and uneven mathematical foundations, making it hard for teaching content of uniform difficulty to cater to students at both ends. Subjectively, some curriculum designers fail to accurately grasp talent-positioning for application-oriented undergraduate education. They tend to follow curriculum systems of research-oriented universities or simply transplant practical training from vocational colleges, resulting in wavering curriculum positioning between “academicism” and “instrumentalism”. Combined with learning situations in private colleges, four major dimensional contradictions exist as follows.

- (1) Contradiction between weak mathematical foundations and profound algorithm theories. Though software-engineering undergraduates in private colleges possess programming basics, their foundations in higher mathematics and linear algebra are relatively weak ^[6-7]. If general-education AI courses follow paths of formula deduction and theoretical proof, students are highly likely to develop fear-of-learning sentiment and end up “listening to incomprehensible lectures”.
- (2) Ambiguous positioning between tool users and system constructors. Vocational colleges focus on enabling students to operate AI tools such as ChatGPT for copywriting. If software-engineering students in application-oriented undergraduate colleges stop at this stage, they will lose their professional advantages. How to advance from API invocation to designing AI-empowered software architectures constitutes a blind spot in current teaching-content design ^[8].
- (3) Disconnection between general-purpose cases and industrial scenarios. Existing general-education-course

cases are mostly general-purpose image classification or text processing, lacking practical business scenarios relevant to software-engineering majors such as code generation, automatic test-case generation and intelligent operation-and-maintenance. Consequently, students cannot perceive concrete values of AI within their own major^[9].

- (4) Contradiction between limited computing-power resources and high experimental demands. Unlike prestigious public universities, private colleges can hardly build large-scale GPU computing-power centers. Traditional deep-learning-training experiments impose extremely high hardware requirements, leading to the phenomenon of prioritizing theory over practice^[10]. Even when some colleges adopt cloud-laboratory solutions, large-scale concurrent experiments are often restricted by network bandwidth and platform stability. In limited practical sessions, students can only finish simple preset-parameter adjustments instead of experiencing the complete workflow from data preparation and model training to deployment and launch. As a result, general-education courses degenerate into theoretical lectures “on paper”, deviating from the original intention of cultivating application-oriented talents.

3. Construction of the “1+N+AI” curriculum system

Targeting the above-mentioned pain points, this paper puts forward the “1+N+AI” curriculum-reform model, aiming to lower thresholds via large-model technologies and realize hierarchical teaching through modular design^[11].

“1” — One core general-education module: AI Fundamentals and Thinking Restructuring. Its objective is to establish cognition and understand principles rather than deduce formulas. The content of general-education AI courses for software-engineering majors is restructured as follows.

- (1) Brief history of AI development and AI ethics, focusing on discussions about AI’s influence on the software industry, such as the rise of low-code platforms and the role transformation of programmers.
- (2) Popularized principles of large models: visualize core ideas of the Transformer architecture (Attention mechanism) with visualization tools and avoid complicated mathematical deductions.
- (3) Prompt Engineering is listed as a compulsory skill. Students are taught how to accurately express demands to large models and leverage AI to assist with code writing and debugging.

This module is designed to make AI act as a “collaborator” in students’ programming workflows. While improving coding efficiency, it cultivates their capabilities of problem decomposition and precise expression — the core competencies distinguishing software engineers from ordinary tool users.

“N” — N vertical-scenario modules for software engineering: Instead of teaching “general-purpose AI”, it focuses on “AI within software engineering”. Three optional sub-modules are set.

- (1) AI-empowered software development: use tools such as GitHub Copilot for automatic code generation, code-comment generation and code review.
- (2) Intelligent software testing and operation-and-maintenance: deploy AI for test-case generation, log analysis and anomaly detection.
- (3) Industry-intelligent-solution development: combined with institutional characteristics such as intelligent manufacturing and modern agriculture in the Jiangsu region, teach the development of AI-plus-industrial-control software systems.

“+AI” — Large-model-empowered teaching measures.

- (1) AI teaching assistants and AI learning companions: adopt Retrieval-Augmented Generation technology to

build a course knowledge base. Students can raise course-related questions anytime and receive real-time replies from AI, which eases teachers' Q&A burden ^[12].

- (2) Light-weight cloud-based experiments: adopt the virtual-real integration mode of cloud-based GPU plus local light-weight models to resolve insufficient computing power. Students invoke cloud-based large-model capabilities locally through APIs to complete practical projects such as intelligent chat-robot development.

The design of "light-weight entry plus cloud-based computing power" effectively lowers students' entry barriers. Even students with weak programming foundations can experience the complete closed-loop of AI-project development in general-education courses, enhancing their sense of learning gain and professional identity. Meanwhile, automatic Q&A and learning-path-recommendation functions of the post-class AI-assistant system provide data support and accurate portraits for hierarchical teaching in subsequent N modules.

4. Innovative practice of teaching content and methods

Transformation of teaching content. De-emphasize mathematics and stress logical reasoning. Remove complicated gradient-descent deduction and adopt Python code to demonstrate "how loss functions decrease intuitively". Place stress on tool-orientation and engineering-orientation. Adopt zero-threshold development platforms including EasyDL and Hugging Face, enabling students to train self-developed image-recognition models without GPU devices and integrate them into Java/Python-developed software systems. Adopt real-enterprise project cases. For instance, live-stream sales-data-analysis cases can be adapted into intelligent-upgrade projects for e-commerce background-management systems for software-engineering majors ^[13]. Similarly, in the field of intelligent manufacturing, cases of real-time analysis and anomaly early-warning for industrial-IoT equipment data can be introduced. Students build AI prediction models based on real production data and embed model outputs into existing enterprise management-information systems, accomplishing full-chain training ranging from data analysis to system integration.

Teaching methods: adopt mixed teaching based on "project-driven plus human-AI collaboration". AI generates preview reports and mind maps. Students watch micro-lectures to grasp basic concepts for pre-class AI-oriented learning. Case teaching demonstrates applications of AI across the whole software-development life cycle. A "human-AI programming contest" is held: one group writes code purely manually while the other adopts AI-assisted coding. Development efficiency and code quality are compared to trigger in-depth reflections on "human-AI collaboration". For example, in software-testing teaching, a comparative-experiment scenario is set: one group writes test cases purely manually; the other leverages AI tools to automatically generate test scripts and execute regression tests. Experimental results reveal not only AI's efficiency advantages in repetitive labor but also limitations of AI-generated test cases in boundary-condition coverage, prompting students to examine boundaries and strategies for human-AI collaboration rationally, and truly realize learning-promotion through contests and thinking-promotion through practice. Students are encouraged to participate in contests such as "Internet-plus" China College Students' Innovation and Entrepreneurship Competition and China Computer Design Competition, and convert course projects into contest works as post-class innovative expansion.

5. Reconstruction of assessment and evaluation system

Break the traditional mode where a single paper determines final grades, and establish a multi-dimensional evaluation system of “process data plus deliverable works plus comprehensive literacy”^[14].

Process-oriented assessment accounts for 40 % of total scores, including platform data such as interaction frequencies with AI teaching assistants and completion rates of online experiments; usual performance, including class-discussion participation and contributions to group collaboration.

Project-practical assessment accounts for 40 % of total scores, including work acceptance: students are required to submit a runnable “AI-plus-software” system (e.g., a library-management system with face-recognition login). Defense focuses on examining design logic and integration capabilities of AI modules.

Basic knowledge accounts for 20 % of total scores, mainly assessing students’ understanding of AI ethics, basic principles and prompt-writing skills.

In addition, students are encouraged to organize course deliverables into technical blogs or project documents, publish them on open-source communities or campus platforms, and receive peer reviews and public feedback.

6. Implementation guarantees and expected outcomes

Build a “dual-competency teacher” team: on-campus software-engineering teachers take charge of system-architecture teaching while enterprise AI engineers participate in joint teaching focusing on algorithm implementation^[15]. Improve teachers’ AI literacy: organize regular large-model-technology training for teachers and encourage them to leverage AI tools for lesson-preparation and scientific research. Furthermore, the college plans to hold one AI-teaching salon every semester, inviting enterprise technical experts and on-campus teachers to jointly discuss new requirements for software-engineering talent cultivation brought by AI-technology evolution, and continuously update curriculum-case libraries and project-resource packages.

Expected outcomes cover student-level and major-level aspects. Students upgrade from “code-writers” to “high-quality-code-writers empowered by AI”, acquiring preliminary capabilities for developing intelligent software products and achieving markedly enhanced employment competitiveness. The major forms a software-engineering-plus-AI curriculum group with characteristics of private application-oriented undergraduate colleges, and produces high-quality teaching cases and teaching-reform papers.

7. Conclusion

The reform of general-education artificial-intelligence courses for software-engineering majors in application-oriented undergraduate colleges should neither be a “compressed version” of computer-science major courses nor a “reproduced version” of higher-vocational skill courses. Oriented toward application-oriented talent cultivation, the “1+N+AI” mode proposed in this paper leverages large-model technologies to resolve teaching bottlenecks of “difficult mathematics and expensive computing power”, and realizes in-depth “AI-plus-major” interdisciplinary integration via modular design. In the future, we will further explore personalized teaching-path recommendation enabled by AI technologies, so as to provide sustained impetus for cultivating high-quality software-engineering talents adapting to the development of new-quality productive forces. The essence of education is to ignite rather than to instill. The significance of general-education AI courses lies not merely in enabling students to master a technical tool, but also

in opening a new window for them to observe the world. When future software engineers can restructure problems with AI thinking and respond to complex demands through human-AI-collaborative solutions, this course will truly accomplish its mission.

Funding

2026 General Project of Philosophy and Social Sciences for Universities in Jiangsu Province (Project No.: 2026SJYB189); Young Backbone Teacher Training Project of Pujiang Institute of Nanjing Tech University (Project No.: 2025PJYQ04); 2025 Fund Project of China Federation of Computer-Basic-Education in Colleges and Universities (Project No.: 2025-AFCEC-113); 2023 University-level Scientific-Research Innovation Team of Pujiang Institute of Nanjing Tech University (Project No.: NJPJ20230303)

Disclosure statement

The authors declare no conflict of interest.

References

- [1] Xia N, Hong R, Wang M, et al., 2026, Innovation and Practice of the “1+N” Mode for General-Education Artificial-Intelligence Education. *Computer Education*, 4: 192–198.
- [2] Sun Y, Zhou Q, Yu Q, et al., 2026, Construction Strategies for General-Education Artificial-Intelligence Curriculum System With Local-University Characteristics — A Case Study of Shandong Agricultural University. *Software Guide*, 1–7.
- [3] Liu S, 2026, A Comparative Analysis of General-Education AI Course Teaching Between Higher-Vocational Colleges and General Universities. *Journal of Zhangjiakou Vocational and Technical College*, 39(1): 111–113.
- [4] Wan Y, Zhang S, Xia X, 2026, Research on Interdisciplinary-Integration Practice of General-Education Artificial-Intelligence Courses in Higher-Vocational Colleges — Taking the Module of Data Analysis and Visualization as an Example. *Journal of Nanjing Open University*, 1: 71–78.
- [5] Bian X, Sheng Y, Wang Q, et al., 2026, Task-Oriented Blended Teaching for General-Education Fundamental-AI Courses. *Computer Education*, 2: 242–248.
- [6] Lu B, Chen Y, Zhang J, 2022, Teaching-Design and Practical-Exploration of General-Education Artificial-Intelligence Courses — Taking Henan Polytechnic University as an Example. *Science-Technology Vision*, 32: 79–82.
- [7] Li Q, 2026, Research on Content Construction of General-Education Artificial-Intelligence Courses in Higher-Vocational Colleges. *Science-Economy-Market*, 2: 119–121.
- [8] Zou C, Sun F, Liu X, 2021, Practical Exploration of General-Education Artificial-Intelligence Courses in Higher-Vocational Colleges. *Modern Vocational Education*, 17: 34–35.
- [9] Liu G, 2026, Design and System Implementation of Intelligent Practical Platform for General-Education Artificial-Intelligence Courses. *Information System Engineering*, 2: 38–41.
- [10] Zhou H, Dong F, 2026, Four-Dimensional Innovation of Industry Intellectualization Empowered by Artificial Intelligence — Taking the Textbook General Knowledge of Artificial Intelligence as an Example. *Vocational and Technical Education*, 2: 55–59.

- [11] Yang F, Zhang L, Wang J, 2026, Construction of General-Education Artificial-Intelligence Curriculum System in Application-Oriented Universities — A Case Study of the “1+4+N” Mode in Lanzhou City University. *Journal of Gansu Teachers College*, 1: 87–91.
- [12] Guo Y, Lou L, 2026, Teaching-Reform and Innovative Practice for Software-Engineering Majors Under the Background of Educational Digitalization. *Computer Education*, 2: 196–201.
- [13] Fu Y, Zhang J, 2026, Construction and Empirical Research of AI-Empowered Software-Engineering Teaching Mode Based on “Case-Driven plus Project-Orientation”. *Journal of Shijiazhuang University*, 28(3): 152–158.
- [14] Liang T, Huang Z, Lu S, 2026, Exploration on Teaching Reform of General-Education Artificial-Intelligence Courses in Application-Oriented Undergraduate Universities. *Journal of Guangxi Open University*, 37(2): 31–38.
- [15] Cai X, Hu Z, 2026, Research and Practice of General-Education Artificial-Intelligence Courses in Application-Oriented Undergraduate Colleges. *Innovation Education Research*, 14(2): 170–174.

Publisher’s note

Bio-Byword Scientific Publishing remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.