DOI: 10.66069/ojspub.1137260615

"Co-Pilot" Rather Than "Autopilot": The Boundaries of Generative AI's Integration and Instructional Design in University Information Technology Courses at Chinese Liberal Arts and Law Universities

Liang Ma

Shanghai University of Political Science and Law, Shanghai, 201701, China

Abstract: ***The widespread adoption of generative artificial intelligence (GenAI) presents an opportunity for liberal arts and law universities in China to overcome traditional challenges in university information technology courses, but it also carries the risk of AI "taking the wheel" and replacing students' cognitive processes. Based on the academic characteristics of liberal arts students in Chinese law and political science universities--namely, weak foundational skills in mathematics and science and severe tech anxiety--this paper proposes positioning GenAI as a "co-pilot" in classroom instruction. By defining the boundaries of GenAI's intervention in terms of cognitive load, decision-making authority, and ethical norms, this paper constructs a three-tier instructional design model comprising "AI scaffolding for dimensional reduction," "Socratic dialogue," and "human-machine dialectical debate." It further elaborates on the specific applications of this model in curriculum restructuring and classroom implementation. Teaching practice has demonstrated that this design can transform students' cognitive roles from passive "code recipients" to active "AI reviewers" and "logical architects." While breaking down technological barriers, it effectively safeguards the educational objectives of cultivating computational and critical thinking, thereby providing a theoretical foundation and practical paradigm for the intelligent transformation of computer fundamentals education for humanities students in law and political science universities in the AI era.***

Keywords: Generative Artificial Intelligence, Co-pilot, Computer Science Education for Liberal Arts Students, Intervention Boundary, Instructional Design, Human-Computer Collaboration.

1. Introduction: The Dilemmas of Computer Science Classes in Liberal Arts and the "Double-Edged Sword" Effect of GenAI

University information technology courses in liberal arts and law schools have long faced a threefold dilemma: "difficult for teachers to teach, arduous for students to learn, and of no practical use after completion." This dilemma is rooted in the unique academic context: students primarily major in fields such as law, sociology, and political science, and generally lack prior knowledge in mathematics and science. Approximately 70% of students admitted in entrance surveys that they "feel intimidated by mathematics," and some even deliberately avoided technology-related learning content during high school. Traditional course content, however, relies heavily on a code- and syntax-centric approach, using abstract data types, loop structures, and function calls as entry points. Furthermore, long-standing adherence to the framework of the Basic Requirements for Teaching University Computer Fundamentals (Ministry of Education 2016). This approach easily leads to "cognitive overload"– working memory must simultaneously process unfamiliar English keywords, strict syntax rules, and implicit computational logic. The lack of prior knowledge further prevents students from utilizing existing mental schemas for chunking, often resulting in a cycle of rote memorization and rapid forgetting. At the same time, most textbook case studies revolve around business scenarios in technology companies, creating a severe disconnect from the knowledge system of law and political science majors. Students commonly feel alienated, asking, "How is learning this useful for my future career as a judge or lawyer?" This further amplifies tech anxiety and learning fatigue.

The explosive proliferation of Generative Artificial Intelligence (GenAI) has opened an unprecedented window of opportunity to break this deadlock. Large language models, represented by ChatGPT, Wenxin Yiyan, and Tongyi Qianwen, leverage natural language interaction to shift the human-machine interface for technical operations--such as programming and data analysis--from precise syntactic commands to ambiguous semantic descriptions. Humanities students no longer need to memorize whether the fourth parameter of the "VLOOKUP" function is 0 or 1; they simply need to use a natural language command like "Please extract the date of birth from this column of ID numbers" to obtain a functional result. This effectively eliminates the coding barrier that has long plagued humanities students, allowing computational thinking to be expressed through the natural language logic they are more adept at constructing. However, scholars such as Wu Di have pointed out in their discussions on the educational applications of general-purpose AI large models that while these models offer convenience, they may also pose the risk of insufficient deep cognitive engagement among learners (Wu et al. 2023). When AI can generate seemingly "perfect" code, reports, or even data analysis conclusions with a single click, introducing it directly into the classroom as an "answer generator" easily lures learners into the "autopilot" trap: students issue a vague prompt, copy the AI's output without understanding it, and skip all processes of logical construction, trial and error, and verification. Behind this apparent leap in efficiency lies a dissolution of deep cognitive engagement, which may lead to skill atrophy, the withering of critical thinking, and even the formation of a new type of “algorithm dependency.”

This highlights the core issue: How can we fully leverage GenAI to lower technical barriers while maintaining the fundamental goal of cultivating students’ computational and critical thinking skills? To answer this question, we must move beyond the binary debate of “to use or not to use” and delve into the role of AI in classroom instruction, the boundaries of its intervention, and the restructuring of instructional design to accommodate it. This article unfolds along this logic. Based on a two-semester, 90-class-hour teaching reform practice in the “University Information Technology” course at a liberal arts university specializing in law and politics, it proposes and validates the positioning of GenAI as a learning “co-pilot” and its systematic instructional design model.

2. Core Concepts and Theoretical Framework: A Paradigm Shift from “Autopilot” to “Co-Pilot”

2.1 Characteristics and Causes of “Autopilot”

The “autopilot” behavior observed in teaching typically manifests as follows: after a task is assigned, students quickly open an AI chat window, enter minimal prompts such as “Write a legal analysis report on online privacy rights” or “Help me create a student grade ranking table,” and then copy and paste the generated results in their entirety for submission, remaining completely oblivious to the formulas, principles, logical structures, or even obvious conceptual errors in the content. In a classroom exercise for the “Python Data Analysis” module, a law student used the prompt “Analyze the data in this table” to have the AI generate a complete report containing regression analysis and significance tests. However, when asked “What does the p-value represent?”, the student admitted, “I hadn’t thought about it; I assumed whatever the AI produced was correct.”

The emergence of “autopilot learning” does not stem from student laziness, but rather from the convergence of two forces. On the one hand, the evaluation system in traditional computer science education is overly focused on “correct results”–whether a program runs or a report follows standard formatting–and pays little attention to the thought processes students undergo. On the other hand, GenAI’s “one-click generation” feature perfectly aligns with this evaluation orientation, offering learners a shortcut to the finish line without having to endure cognitive struggle. According to path dependence theory, when a tool can eliminate task barriers at an extremely low immediate cost, and the long-term benefits of trial and error and reflection are not captured by the evaluation system, learners will naturally choose the strategy with the highest apparent efficiency. Thus, “autopilot” becomes a hidden crisis born of rational choice.

2.2 The Essence and Role of the “Co-Pilot”

The “co-pilot” metaphor borrows the concept of human-machine collaboration from aviation and driving, emphasizing AI as an assistant, guide, and collaborator rather than a controller. In humanities and computer science classrooms, GenAI’s “co-pilot” role is assigned three concrete functions: First, it serves as a “scaffold” to reduce cognitive load, providing necessary external support during the early stages of student learning to prevent them from being overwhelmed by technical details; Second, it serves as an “interpreter” of multimodal explanations, capable of translating abstract mathematical and scientific concepts into narrative logic familiar to students of law, politics, and life sciences; Third, it acts as a “sparring partner” that exposes logical flaws, stimulating students’ critical thinking and scrutiny through its generation of “seemingly reasonable yet flawed” responses.

However, regardless of which of these roles AI assumes, the ultimate authority for decision-making, verification, and value judgment must remain firmly in the hands of the student, who serves as the “main driver.” This means that while AI can generate code frameworks, students must be able to explain the function of every line of code; while AI can provide analytical approaches, students must determine whether those approaches are applicable to their specific problem contexts; and while AI can assist in writing reports, students must take responsibility for the accuracy of their arguments and facts. This fundamental principle is the dividing line between a “co-pilot” and a “substitute driver.”

2.3 Theoretical Foundation

The “co-pilot” model has a solid foundation in educational psychology. Vygotsky’s Zone of Proximal Development (ZPD) theory posits that learning occurs most effectively at the point between a learner’s independent capabilities and the level they can reach with external support (Vygotsky 1978). GenAI serves precisely as the provider of this dynamic, adaptive support: through natural language dialogue, AI can perceive the student’s current level of understanding in real time and provide prompts, explanations, or partial solutions in a manner that most closely aligns with their “reachable edge.” As the student’s abilities improve, prompts can be gradually reduced, ultimately removing the scaffolding to enable independent operation.

Cognitive load theory further provides a basis for defining the boundaries of the “co-pilot” role. Sweller et al. point out that cognitive load during the learning process includes intrinsic cognitive load (determined by the complexity of the task itself), extrinsic cognitive load (caused by the way information is presented), and related cognitive load (used for schema construction and automation) (Sweller et al. 1998). The overload experienced by liberal arts students when facing programming tasks stems largely from non-logical extrinsic loads–such as syntax memorization, punctuation and spelling, and function signatures–which consume their extremely limited working memory resources. The integration of GenAI can effectively eliminate this irrelevant mathematical redundancy through semantic conversion – students express logic in natural language, and the AI converts it into precise code–thereby redirecting the freed-up working memory capacity toward core logical construction, process design, and error diagnosis, i.e., increasing effective relevant cognitive load. Research by Zhang Xiaoli, Huang Ronghuai, and others also emphasizes that cognitive load theory provides direct guidance for instructional design in multimedia and intelligent technology environments; the integration of technological tools should be guided by the principles of reducing extrinsic load and promoting schema construction (Zhang and Huang 2010).

3. Defining the "Intervention Boundaries" of GenAI in Humanities Computer Science Classrooms

To ensure that AI remains in the "passenger seat" rather than slipping into the "driver's seat," instructional design must establish clear and inviolable triple boundaries for it.

3.1 Cognitive Boundary: Separation of Generation and Verification

In computer science education for the humanities, the principle of separating "generation" from "verification" must be strictly adhered to. AI can handle "generation"–providing initial code frameworks, summarizing literature perspectives, and constructing alternative solutions–-but students must independently take responsibility for "verification" and "interpretation." This means that any content generated by AI is explicitly designated in the classroom as a "work-in-progress awaiting review," rather than a "deliverable final product."

For example, in the "legal case data cleaning" task, the AI generated several lines of Python code based on the student's natural language description. The instructional requirements are then set: students must add comments line by line, verbally explain the purpose of each step to peers or instructors, and proactively design test data to verify the code's boundary conditions. If students discover flaws in the AI's code when handling missing values and correct them, this is precisely the moment when deep learning occurs; conversely, if students declare the task complete without reviewing the generated code, it is considered an incomplete learning assignment. Consequently, the endpoint of AI intervention is defined as "comprehensible input," while the student's cognitive starting point is positioned as a "critical reviewer," forming a cognitive feedback loop of "generation-review-iteration."

3.2 Sovereignty Boundaries: The Demarcation Between Logical Architecture and Execution Details

The boundary of sovereignty concerns the attribution of high-level decision-making authority within learning tasks. When faced with complex, comprehensive tasks–such as designing an intelligent legal case retrieval system or constructing a spatio-temporal analysis framework for crime data–the "right to architectural design" must remain entirely in the hands of the student. This includes: determining the analytical indicator system, designing the business workflow of the information system, and selecting the framework and dimensions for argumentation. AI provides assistance only at the level of explicitly authorized implementation details, such as writing specific data aggregation functions based on the student's established metric system, automatically generating a segment of code according to the student's flowchart, or polishing the syntax of text that has already been fully articulated.

The rationale behind this separation is that the cultivation of higher-order thinking skills–such as systems analysis, logical structuring, and value judgment–relies precisely on learners personally managing the macro-structure of tasks and making continuous decisions throughout the process. If AI were to "pre-build the entire framework" for students from the outset, it would effectively deprive the course of its most valuable component: critical thinking training. In daily teaching, this boundary is translated into a practical guideline: "AI handles only the detailed tasks you explicitly instruct it to do; you must draw the big-picture flowchart yourself."

3.3 Ethical Boundaries: Balancing Technological Efficiency and Academic Integrity

In AI-enabled classrooms, the definition of academic integrity faces new gray areas. A simple "ban on use" is both out of step with current trends and difficult to enforce effectively. The instructional design in this paper instead establishes a proactive and actionable ethical boundary: process transparency and mandatory documentation. This approach draws on strategies proposed by scholars such as Mollick for assigning AI-based assignments to students, with the core principle being to transform the AI usage process from a "black box" into a visible, assessable academic practice (Mollick and Mollick 2023).

Specifically, for any assignment or project involving AI assistance, students must retain and submit a complete "Prompt Iteration Log," truthfully documenting every round of dialogue, every modification to the prompt, and the rationale behind each change–from the initial vague requirements to the final satisfactory result. Additionally, the final report must include a statement in a specific format declaring the scope of AI involvement, for example: "The code generation in Section 3 of this report was assisted by Tongyi Qianwen; see the appendix log for the prompts. The interpretation and analysis of the data conclusions were completed by me." This approach transforms AI from an invisible "ghostwriter" into a tool that must be cited, shifting the boundaries of academic integrity from "whether AI was used" to "whether AI's contributions are presented truthfully and transparently." This is highly consistent with the legal and political disciplines' own emphasis on procedural justice and evidence traceability, and is also easily understood and accepted by students in the humanities.

4. Construction of a Teaching Design Model Based on the "Co-Pilot" Role

Within the constraints of the aforementioned three-fold boundaries and in light of the cognitive characteristics of liberal arts students, this paper proposes the "3C" three-stage instructional design model, namely Construct (scaffolding and dimensional reduction), Collaborate (Socratic human-computer dialogue), and Confront (human-computer critical thinking and debate).

4.1 First Stage: Construct–AI-Assisted Scaffolding and Dimension Reduction

Objective: To eliminate the "blank-page anxiety" humanities students experience when faced with technical tasks, lowering the entry barrier from "creating something out of nothing" to "evidence-based review and revision."

Design: In the practical application phase of each new skill module, we completely abandon the traditional practice of "requiring students to independently write code or functions from a blank page." The replacement process is as follows: Students first describe the task requirements to the AI using natural language, for example, "I have a table containing case numbers, filing dates, and closure dates. I want to calculate the number of days each case took to process and calculate the average number of days by year." The AI then generates an initial Excel formula or Python code framework. Subsequently, the students' core learning tasks are redefined as: 1) Understanding the AI-generated code line by line and using comments to explain its meaning; 2) Experimenting with different input parameters or data samples to observe changes in the output; 3) Actively "spot errors" to check whether the AI makes mistakes when handling edge cases (such as cases spanning multiple years or inconsistent date formats); 4) Optimizing the initial solution based on their understanding and explaining the rationale for the optimizations.

Practical Case: Second-year undergraduate students majoring in AI-driven law were generally confused by date functions such as "DATEDIF" during an Excel classroom assignment on "statute of limitations analysis." After introducing the "Construct" strategy, student Xiaoli articulated her requirement in natural language, and the AI generated the formula "=DATEDIF(C2,D2,\"d")". Instead of simply pasting the formula, Xiaoli was asked to "translate it into plain language"–she wrote: "Calculate the number of days between the case filing date in C2 and the case closure date in D2." Subsequently, during testing, she discovered that the formula returned an error when the case closure date was blank. With the AI's assistance, she proactively added an IF statement to handle this exception. This process transformed the student's cognitive role from a "function memorizer" to a "logic reviewer," effectively alleviating technical anxiety.

4.2 Level Two: Collaborate–Socratic Human-Machine Dialogue

Objective: To overcome the comprehension barriers that humanities students face with abstract concepts due to weak mathematical and scientific foundations, and to achieve "meaning construction" of knowledge by leveraging GenAI's semantic translation capabilities.

Design: A customized AI teaching assistant is introduced, with its response style constrained by a pre-set System Prompt: "You are a teaching assistant skilled at explaining computer science concepts using examples from law and sociology. You never provide dry definitions directly, but instead use analogies based on scenarios familiar to students of law and political science." When students encounter difficult-to-understand abstract concepts during their studies–such as loops, conditional statements, variables, or databases–they are encouraged to actively ask the AI questions. Students are guided to use typical question formats such as: "Please explain what an if-else structure is using the process of a court trial," or "Using the example of population information management by public security authorities, tell me what 'primary keys' and 'foreign keys' mean in a database." The AI generates personalized, context-based explanations, while students are required to paraphrase them and verify their own understanding.

Practical Example: When explaining "loop structures," the traditional description–"repeatedly executing a block of code for each element in a sequence"–confuses many liberal arts students. A sociology major asked the AI: "Can you explain 'loops' using the data entry process for a social survey questionnaire?" The AI responded: "Imagine you have 500 paper questionnaires, each with 20 questions. A 'loop' is like repeating the process of reading through each questionnaire 20 times in sequence–starting with the first questionnaire from question 1 to 20, then picking up the next one and continuing from question 1... until all 500 questionnaires have been processed." The student later remarked, "I suddenly understood: a loop means I don't have to write the same action 500 times." In this process, AI did not replace the student's understanding but acted as a 'translator,' converting unfamiliar programming language into the student's existing cognitive experience, thereby promoting deep semantic processing.

4.3 Stage Three: Confront—Human-Machine Competition and Critical Thinking

Objective: To transfer the strong logical reasoning and evidence-review skills of liberal arts students in law and politics to the evaluation of AI output quality, and to strengthen critical thinking and information verification literacy through "human-machine confrontation."

Design: Actively leverage the inherent "hallucination" tendency of large language models as a teaching resource. Instructors pre-design or generate in real-time AI-generated materials containing hidden logical errors, factual inaccuracies, or misattributed legal provisions. These are assigned to students with the instruction: "This is an answer generated by an AI system. Please identify all errors within 20 minutes and provide the basis for your corrections." Error types can be designed from easy to difficult: Beginner – obvious errors in dates or article numbers; Intermediate – reversed causality, conceptual confusion; Advanced – arguments that appear logically consistent but contain deviations in professional legal interpretation. Students must draw upon their professional knowledge and traditional information retrieval skills (such as logging into Beida Law Treasure, the China Judgment Document Network, or CNKI for verification) to complete the full process of "identifying errors-proving them-and correcting them."

Practical Case: In a specialized course on "AI and Legal Information Retrieval," the AI was tasked with generating a review on "Key Points of the Application of the Personal Information Protection Law." The instructor had pre-embedded fictitious or misleading information, such as "Article 25 of the Personal Information Protection Law stipulates the right to be forgotten" and "A court in Shanghai issued the nation's first judgment on a copyright case involving AI-generated content in 2024." Law students were divided into groups to conduct an "adversarial review," tasked with verifying each claim one by one within a limited timeframe using professional legal databases. Ultimately, the third group successfully pointed out that the "right to be forgotten" is actually a concept from Article 17 of the EU's GDPR, while the "right to erasure" stipulated in Article 47 of China's Personal Information Protection Law, though similar, differs in its legal basis. They provided the original text of the law and academic interpretations as evidence. This process transformed "checking the AI's work" into a highly specialized legal evidence review exercise. While students experienced the sense of accomplishment from "defeating the AI," they also deepened their understanding that "AI cannot be trusted unconditionally."

5. Implementation Path: Deep Integration of Course Content and the "Co-Pilot" Model

The aforementioned 3C model is not an isolated classroom activity but is deeply embedded within a 90-class-hour curriculum spanning both the spring and fall semesters, functioning through systematic integration with traditional teaching content.

5.1 Integration of Information Retrieval and AI-Generative Retrieval

In the first-semester "Information Literacy" module (approximately 12 class hours), traditional information retrieval instruction is conducted in parallel with AI-generated retrieval, with a deliberate focus on creating cognitive conflict between the two. The specific implementation approach is as follows: the same retrieval task is designed to be completed via two parallel tracks. Track One requires students to conduct literature searches and fact-checking using traditional professional databases such as CNKI, Peking University Law Database, and the China Judgment Document Network; Track 2 allows students to ask AI questions on the same topic to obtain AI-generated summaries or answers. Subsequently, the course enters the "Confront" phase, requiring students to compare the two sets of results item by item, identify "hallucinations" and "information gaps" in the AI-generated content, and use the reliable sources from Track 1 to correct them. For example, students might discover that when summarizing the historical development of a legal system, the AI has fabricated a non-existent judicial interpretation document number. This discovery itself becomes a highly impactful learning moment. After completing this module, typical student feedback includes: "I didn't realize AI could fabricate legal provisions; from now on, whenever I use it for searches, I must verify the results by checking the databases." This habit of "prudent adoption" in AI usage is precisely the goal the course aims to achieve.

5.2 Integration of Data Processing and AI Natural Language Analysis

In the "Fundamentals of Data Analysis" module (approximately 18 class hours in the first semester), the first-order Construct dimension-reduction strategy is comprehensively applied to address the complex Excel functions and PivotTable operations that have long posed the greatest challenge for liberal arts students. Instructors no longer require students to memorize the complete syntax of any complex functions; instead, the starting point for in-class exercises is uniformly set as "natural language requirement descriptions." Before performing hands-on exercises, students first write their requirements on a task sheet, for example: "I want to categorize and summarize cases by province based on the 'Defendant's Location' column, and calculate the number of cases and the average case value for each province." Students then input this natural language description into the AI, which returns the corresponding function combinations (such as SUMIFS, COUNTIFS, etc.) or PivotTable creation steps, along with explanations of key parameters. Students' focus shifts to "paste-verify-fine-tune-explain": they enter the AI-generated formulas into Excel, test their accuracy with different datasets, investigate the causes of unexpected results, and document the meaning of each step in plain language in their task logs. In the data analysis project for the final exam, all students were able to complete highly complex multi-condition summary analyses with the help of AI. During the subsequent oral defense, their ability to explain data logic and business implications far exceeded their previous performance when they had merely memorized functions.

5.3 Integration of Programming and AI Agent Development

In the second-semester module "Computational Thinking and Intelligent Applications" (approximately 24 class hours), the course undertook a more radical integration: it completely eliminated the requirement for liberal arts students to write Python code by hand, instead introducing no-code AI agent development platforms such as Coze and Dify. The rationale behind this shift is that, for liberal arts students in law and politics, the core value of computational thinking lies not in mastering the syntax of a specific programming language, but in the ability to decompose real-world problems into computable steps, define inputs and outputs, and design processing workflows. This is precisely the role of the "main driver"–students act as product managers and architects, responsible for conceptualizing the agent's functions, designing dialogue workflows, and constructing the logical structure of the knowledge base; while the AI serves as the "co-pilot," responsible for generating specific processing code at designated points and writing structured descriptions of knowledge base entries.

Typical student final projects include the "Campus Legal Consultation Agent," the "Labor Rights Q&A Bot," and the "Sentiment Analysis Tool Based on Judicial Rulings." Taking the "Campus Legal Consultation Agent" group as an example, the workflow designed by four law students is as follows: 1) User inputs a legal question → 2) Intent recognition node (AI-assisted prompt generation to determine whether the issue pertains to contract law, tort law, or intellectual property) → 3) Routing to the corresponding knowledge base for retrieval → 4) AI generating a preliminary response based on the search results → 5) Attaching risk warnings and links to relevant legal provisions. Throughout the entire workflow, the logical architecture, classification of knowledge base entries, and wording of risk warnings were all designed by the students, while the AI primarily handled code block generation and language refinement. During the pitch, the team not only demonstrated a functional AI agent but also submitted a detailed "prompt iteration log" documenting the evolution of over forty rounds of dialogue–from the initial vague idea of "building a legal robot" to the final refined workflow–fully illustrating the "student-led" process of logical construction.

6. Reconstructing the Evaluation System: Shifting from "Correct Results" to "Process and Critical Thinking"

In alignment with the "Co-Pilot" instructional design, the course's evaluation system must undergo a fundamental shift–from result-oriented "project assessment" to process-oriented "cognitive trajectory assessment."

6.1 The "Prompt Iteration Log" as a Vehicle for Formative Assessment

In traditional exams, rote memorization metrics–such as "whether a student has memorized the correct parameter order for a specific function"–are completely eliminated. Instead, the interaction process between the student and the AI itself serves as the core basis for evaluation. For each project assignment, students must submit a complete "prompt iteration log." When evaluating, teachers no longer focus on the perfection of the final product, but rather on the problem-decomposition skills, structured thinking demonstrated through the gradual refinement of requirements, and the iterative mindset of making corrections based on feedback that students exhibit throughout this series of dialogues. For example, a log that begins with "Help me process this table" and gradually refines to "Please extract the year from the 'Case Filing Date' column in the 'Case Information Table,' count the number of cases by year and 'Case Type,' and output the results to a new worksheet" would be considered a high-quality process record. This is because it demonstrates the student's ability to clarify and multidimensionalize a vague problem–which is the very essence of computational thinking.

6.2 Rubric Design for the "Human-Machine Collaboration Project"

The end-of-term demo assessment employs a new rubric, reducing the weight of "Technical Implementation Difficulty" from the traditional 60%–70% to 20%, while increasing the weight of "Professional Empowerment" (i.e., the extent to which the project addresses real-world legal and political science issues, 30%), "Critical Thinking and Reflection" (i.e., the extent to which AI outputs are effectively questioned, verified, and corrected, 30%), and "Presentation of the Collaborative Process" (the quality of prompt logs and the clarity of human-machine division of labor, 20%) as the primary evaluation dimensions. The rubric explicitly stipulates: Students who can identify specific errors in AI-generated content, explain the causes of those errors, and make effective corrections will receive explicit bonus points; Conversely, if students accept the AI's default output wholesale without demonstrating any evidence of scrutiny, even if the final work is technically "perfect," they will lose points in the "Critique and Reflection" dimension, and the evaluation comments will note a "lack of driver awareness." This adjustment establishes "collaborating with AI while prudently transcending it" as the core educational objective of the course, fundamentally shifting students' motivations for using AI.

7. Conclusion and Outlook

In the era of generative AI, the mission of introductory computer science courses at liberal arts colleges specializing in law and politics is no longer to train "low-spec" programmers capable of writing code independently, but rather to cultivate digital citizens equipped with "human-machine collaborative leadership." The core proposition of being a "co-pilot" rather than a "stand-in driver" serves as both an empathetic deconstruction of the technological anxiety long experienced by liberal arts students–acknowledging that code syntax is not a cross they must bear–and a resolute defense of their higher-order critical thinking abilities. The cultivation of logical frameworks, evidence review, and value judgments can never be outsourced to algorithms.

The "boundary of intervention" delineation and the "3C" three-stage instructional design model proposed in this paper represent the systematic implementation of this philosophy in concrete teaching practice. After two semesters of iteration, the course has achieved preliminary success in reducing students' apprehension, enhancing their ability to transfer computational thinking, and cultivating a culture of prudent AI use. The professional depth and level of critical reflection in students' final projects significantly surpassed the outcomes of traditional teaching models. Methodologically, this study represents an exploration of educational reform grounded in real classroom contexts. Future research could draw upon the qualitative research approach advocated by Chen Xiangming, using in-depth interviews and classroom ethnography to more deeply reveal the micro-cognitive processes underlying student-AI interactions (Chen 2000). At the same time, further attention should be paid to the differing cognitive patterns and questioning strategies exhibited by students from different disciplinary backgrounds–such as law and sociology–-when interacting with AI. This involves exploring more personalized mechanisms for adjusting the "co-pilot" intervention level and continuously tracking whether students can maintain the "driver" role in real-world scenarios after graduation, when no longer subject to the constraints of course evaluations. This will serve as the ultimate yardstick for assessing the true effectiveness of this pedagogical reform.

References

[1] Ministry of Education. 2016. Basic Requirements for Teaching University Computer Fundamentals Courses. Beijing: Higher Education Press.
[2] Wu, Di, Li Huan, and Chen Xu. 2023. "An Analysis of the Impact of General-Purpose Large Language Models in Education." Research on Open Education 29(2): 19-25.
[3] Vygotsky, L. S. 1978. Mind in Society: The Development of Higher Psychological Processes. Cambridge: Harvard University Press.
[4] Sweller, J., J. J. G. van Merriënboer, and F. Paas. 1998. "Cognitive architecture and instructional design." Educational Psychology Review 10(3): 251-296.
[5] Zhang, Xiaoli, and Ronghuai Huang. 2010. "Cognitive load theory and its implications for multimedia

instructional design.” Chinese Journal of Educational Technology (9): 23-28.
[6] Mollick, Ethan, and Lilach Mollick. 2023. “Assigning AI: Seven approaches for students, with prompts.” SSRN. https://ssrn.com/abstract=4475995.
[7] Chen, Xiangming. 2000. Qualitative Research Methods and Social Science Research. Beijing: Education Science Press.

Author Profile

Liang Ma Male, 1982.3, Lecture of Shanghai University of Political Science and Law. Main research interests: Intelligent Education and Personalized Teaching.