

基于建构主义理论的Linux课程大语言模型辅助教学改革与实践

王 锋, 王 彬, 金海燕, 李 薇

西安理工大学计算机科学与工程学院, 陕西 西安

收稿日期: 2026年7月13日; 录用日期: 2026年8月15日; 发布日期: 2026年8月24日

摘 要

针对Linux基础及应用课程中Linux体系庞杂入门难和学生过度依赖大语言模型的教学难点, 提出了以建构主义学习理论为指导, 引入大语言模型作为认知脚手架, 构建了智能脚手架与情境化学习相结合的教改方案。该方案不仅提供知识问答, 更扮演促进深度理解的认知伙伴, 以及动态规划学习内容的路径生成器, 同时记录学习过程的学生建构轨迹库。阐述了教改方案的实施过程, 并总结了教学效果。

关键词

建构主义, 大语言模型, 教学方案, 知脚手架

Reform and Practice of Large Language Model-Assisted Teaching in Linux Course Based on Constructivist Theory

Feng Wang, Bin Wang, Haiyan Jin, Wei Li

School of Computer Science and Engineering, Xi'an University of Technology, Xi'an Shaanxi

Received: July 13, 2026; accepted: August 15, 2026; published: August 24, 2026

Abstract

In response to the teaching difficulties in the Linux Foundation and Application course, such as the complexity of the Linux system and the difficulty in getting started, as well as students' excessive reliance on AI, this paper, guided by constructivist learning theory, introduces large language models as cognitive scaffolding and constructs a teaching reform plan that combines intelligent scaffolding with contextual learning. This solution not only offers knowledge Q & A but also acts as a cogni-

tive partner promoting in-depth understanding, a path generator for dynamically planning learning content, and a student-constructed trajectory library recording the learning process. This article elaborates on the implementation process of the educational reform plan and summarizes the teaching effects.

Keywords

Constructivism, Large Language Model (LLM), Teaching Plan, Cognitive Scaffolding

Copyright © 2026 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

2025 年 4 月, 教育部等九部门在《关于加快推进教育数字化的意见¹》明确提出“加快建设人工智能教育大模型”, 完善语料库, 推动垂直应用, 是当前行动的重要依据。

目前, 随着人工智能技术的发展, 大语言模型(MML) [1]-[3]为计算机专业学生的学习带来了诸多便利, 但同时也产生了明显的负面影响: 部分学生在完成作业时缺乏独立思考, 过度依赖大语言模型直接获取答案, 导致相关知识未能内化, 难以培养具备扎实专业能力的人才。建构主义学习理论[4]强调知识是由学习者基于自身经验主动建构的, 而非被动接收, 这为解决学生过度依赖大语言模型的问题提供了理论依据。

提出了基于建构主义学习理论, 结合大语言模型, 通过记录学生的学习过程, 初步探索如何缓解这一依赖现象。

2. Linux 基础及应用课程教学中存在的问题

目前 Linux 基础与应用教学中存在的主要问题有: (1) Linux 体系庞杂, 入门困难。由于 Linux 知识体系庞大且分散, 初学者往往难以构建系统性的知识框架和思维模式, 陷入无所适从的困境, 导致简单问题重复出现, 复杂问题多种多样, 而教师没有充足的时间对每位学生进行及时指导, 从而影响学生掌握相关知识。(2) 学生过度依赖 MML。当学生面对复杂的、系统性、连贯性的实验项目时, 常常感到无从下手。为完成作业任务, 部分学生过度依赖 AI 搜索答案, 导致平时作业完成情况良好, 但最终考核结果不尽人意。

针对上述问题, 可运用建构主义学习理论进行指导。该理论源于瑞士心理学家让·皮亚杰系统阐述了认识是一种以主体已有知识和经验为基础的“主动建构”过程。心理学家利维·维果茨基提出支架式教学[5], 重视互动与合作学习, 关注学习过程而非结果, 对现代教育尤其是建构主义教学实践产生了革命性影响。建构主义学习理论的核心原则[6][7]包括: 主动建构性——学习是学习者基于已有经验主动加工信息、形成个人理解的过程, 而非被动接收; 情境性——知识需在真实或仿真的情境中建构, 以促进向复杂现实问题的有效迁移; 社会互动性——学习通过协作、讨论等社会互动得以深化与发展; 个人独特性——尊重学习者由于经验与认知差异形成的多元化理解, 鼓励多元视角。

基于上述理论, 提出了结合 LLM 构建“智能脚手架 + 情境化学习”的解决方案: 在教学中提供真实或接近真实的项目情境, 鼓励学生通过探索、试错、协作来构建自己的知识体系, 教师则扮演引导者

¹http://www.moe.gov.cn/srcsite/A01/s7048/202504/t20250416_1187476.html

而非灌输者。具体设计遵循建构主义原则(如脚手架、情境学习、认知冲突、社会协商)来引导、约束和塑造 LLM 的交互逻辑与输出内容,将其融入系统设计,引导 LLM 超越“信息提供者”的角色,转变为“认知协作者”。通过 LLM 实现:个性化即时指导——随时解答学生问题,弥补教师精力不足;动态脚手架——根据学生水平提供分层提示,引导其自主思考;情境生成与模拟——生成多样化的实验场景,帮助学生迁移知识。同时,通过过程性评价(如学习者建构轨迹库)更准确地反映学生能力,从而帮助学生从被动接收信息转向主动建构知识。

3. Linux 课程教学改革方案系统设计

利用 LLM 搭建认知脚手架[8],构建支持主动探索与意义建构的 Linux 学习环境,将教学重心从“教”转向“学”,确立学生的主体地位,教师负责教学设计和学生疑难指导。如图 1 所示,该方案由学生、教师、和 LLM 三个角色组成。

3.1. 学生是知识建构的主体

主动探索:通过情境化任务开展学习,在协作界面中与 LLM、其他同学互动;

记录反思:利用学习日志固化知识,在反思工具中审视自己的认知过程;

双向互动:学生向 LLM 引擎提问、求助,接收来自引擎的脚手架支持、反思引导和知识关联提示。

3.2. LLM 由三层构成,本系统通过 DeepSeek API 提供大语言模型对话与推理能力[9]

第一层:交互与建构层

提供了访问界面,所有底层的资源和上层支持,最终都服务于这个层面的学习活动。

情境化问题呈现:将学习任务以真实的方式展示给学生。

协作式探索界面:支持学生与 LLM、与其他学生进行互动、讨论和协作。

反思工具与学习日志:提供结构化工具,帮助学生固化、提炼和审视自己建构的知识。

核心作用:提供以活动为中心的交互界面,将学习设计为一系列可操作的、需要主动参与和反思的建构过程。

第二层:LLM 认知引擎层

情境建构器:基于底层情境库,动态生成或适配贴近真实的学习任务和场景。

认知脚手架:在学生探索过程中,提供恰到好处的提示、示范或资源支持,帮助他们跨越“最近发展区”。

反思促进器:通过提问、对比等方式,引导学生回顾与思考自己的学习过程,培养元认知能力。

概念连接器:帮助学生识别并理解新旧知识间的联系,主动编织个人知识网络。

核心作用:将 LLM 的通用能力转化为教育学功能,成为一个理解学习过程、能提供动态、个性化认知支持的“智能学伴”。

第三层:知识生态与情境库

该层是教学方案的知识地基和情境来源。

情境案例库:存储真实的 Linux 运维场景,确保学习始于真实需求。

概念关系网(知识图谱):组织 Linux 知识概念及其内在联系。

认知策略库:存放问题解决的思维模式与方法论,引导学生“如何思考”而非仅仅“知道什么”。

学习者建构轨迹库:记录每位学习者的探索路径和认知发展过程,是实现个性化支持的数据基础。

核心作用:确保学习内容情境化、结构化、可追溯,将知识确定在真实应用中,并为上层提供丰富的、可供调用的资源。

3.3. 教师是教学设计的引导者和兜底保障

顶层设计：审核和优化情境案例库、知识图谱的质量，确保学习内容的方向正确。

疑难介入：当 LLM 无法处理复杂问题时，教师介入进行个性化指导。

学情分析：通过查询学生建构轨迹库，了解整体学习进展和个体差异，进行精准干预。

策略调优：根据学情反馈调整 LLM 的脚手架策略。

整个过程支持并记录学生的知识建构。

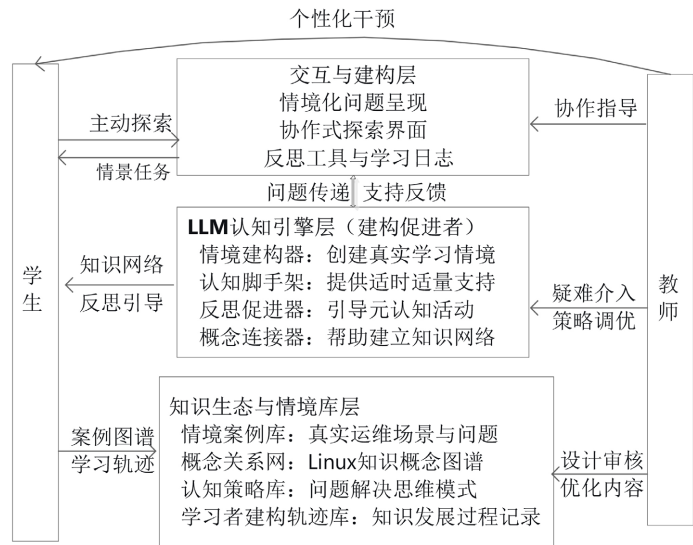


Figure 1. System design of Linux course teaching reform scheme
图 1. Linux 课程教改方案系统设计

4. Linux 课程教改方案实施过程

4.1. 教学方案在教学实践由五个阶段组成

- 1) 情境导入阶段：教师基于真实运维场景设计驱动性问题，将学生带入具有挑战性的问题情境。LLM 在此阶段充当“情境感知助手”，根据学生初始能力推送差异化案例背景，并通过启发式提问激活学生已有经验，引导其进入学习状态。
- 2) 探索与建构阶段：学生以问题为导向，自主与 LLM 展开对话探究。LLM 承担“知识提供者”、“思维引导者”和作为“过程记录者”。
- 3) 反思与深化阶段：学生在完成初步探索后，与 LLM 展开反思性对话，引导学生复盘思维过程。教师组织小组交流，LLM 同步生成各组讨论要点对比。
- 4) 评估与迁移阶段：教师设置变式问题或相似情境的新任务，检验学生的知识迁移能力。LLM 作为“评估助手”，不仅判断答案正确性，更进一步分析学生的解决路径与思维特征，生成个性化评估报告，指出优势与改进方向。
- 5) 个性化延续阶段：根据前四阶段的过程数据，LLM 为每位学生生成定制化的后续学习建议，推送针对性的实践任务与拓展资源。教学内容：学习成果延伸、个性化学习路径规划。

4.2. Shell 脚本实现自动化备份举例

教学目标：综合运用变量、条件判断、循环和命令替换，编写一个健壮的目录备份脚本。

1) 情境导入

教师发布项目需求：为 Web 服务器编写一个备份脚本 `backup.sh`，具体要求包括备份 `/var/www/html` 目录、以“日期_时间”命名备份文件、备份失败时发送告警邮件、保留最近 7 天的备份。学生在收到需求后，开始分析并拆解为技术要点，如日期生成、条件判断、循环删除和错误处理等。在此阶段，MML 通过情境库提供“运维自动化之备份策略”的标准模板与约束条件，交互层则为学生提供在线脚本编辑器、测试环境入口和需求检查清单。

2) 探索与建构

教师扮演顾问角色，仅回答“元问题”，如指导学生如何查找资料，而不直接提供具体语法答案。学生开始自主编写脚本，过程中会遇到诸如如何获取当前日期、如何判断命令是否执行成功、如何循环删除 7 天前的文件等典型问题。此时，LLM 引擎层作为“代码教练”发挥作用：一方面提供认知脚手架，当学生写出如 `if [$? -eq 0]; then` 这样的代码时，LLM 会提示是否可以直接将命令放在 `if` 条件中；另一方面作为概念连接器，当学生使用 `find` 命令时，系统会高亮与定时任务 `cron` 的关联，为后续学习做好铺垫。

3) 反思与深化

在学生首次运行脚本并认为“成功”后，MML 会自动注入一个故障场景，例如模拟磁盘空间不足导致 `tar` 命令失败。学生发现脚本虽“成功”运行但未生成备份文件，于是开始排查问题并增加错误处理逻辑。此时，LLM 作为反思促进器，会发起引导性问题，如“你的脚本现在真的健壮吗？如果备份目录不存在怎么办？如果 `mail` 命令没安装怎么办？”同时，策略库提供“防御性 Shell 编程”的检查清单，帮助学生进一步深化对脚本健壮性的理解。

4) 评估与迁移

这一阶段以小组间的代码评审形式展开。教师布置任务，要求学生评审另一组的脚本，重点评估其错误处理的完备性和代码的可读性。学生在阅读他人代码的过程中提出改进建议，并在讨论区进行辩论。社会互动在交互层的协作区发生，LLM 可在此过程中总结常见的问题模式，例如“很多脚本忽略了 `set-e`”，并推荐策略库中的代码规范文档，帮助学生在迁移应用中提升代码质量。

5) 个性化延续

MML 根据学生在学习过程中的建构轨迹进行个性化任务推送。对于代码逻辑清晰的学生，推送“将脚本改造为支持命令行参数和配置文件”的进阶任务；对于错误处理能力较弱的学生，则推送“Shell 脚本异常处理专题”互动练习。学生由此进入下一轮建构循环。轨迹库中记录的错误处理模式、代码结构复杂度等数据，成为个性化推荐的直接依据，使学习路径更具针对性。

五个阶段将建构主义理念的知识建构、最近发展区和情境学习转化为可操作的教学流程：教师负责情境设计与关键点拨，LLM 负责过程性支持与数据分析，学生负责自主探究与反思建构。

5. Linux 课程教改方案教学效果分析与评价

为验证 Linux 课程教学改革在真实学习场景中的有效性，本方案设计了学生过程评价标准，教学效果分析与评价将从学业表现、学习过程深度分析、认知能力发展评估三个方面展开，旨在构建一个多视角、全过程、可量化的综合评价体系，为教师后续改进教学内容和方法提供参考。

5.1. 实验基本情况

实验周期：2025 年 9 月~2026 年 1 月(一学期)

实验对象：某高校计算机专业 4 个平行班，共 160 名学生

分组设计：

实验组(2 个班, 80 人): 使用三层模型教学
控制组(2 个班, 80 人): 传统讲授 + 实验室练习模式
教学时长: 32 学时《Linux 基础及应用》
评价指标: 知识掌握度、技能应用能力、学习体验

5.2. 学生学习过程轨迹库评分指标

为了防止学生过度依赖 MML, 依据轨迹库中记录的学生学习过程数据, 从学生学习活动参与、认知建构、元认知发展及问题解决四个维度, 评价学生学习过程, 自动给每位学生学习过程进行评分。

Table 1. Learning process scoring criteria

表 1. 学习过程评分标准

评分维度	具体指标	评分标准(分值范围)	数据来源(轨迹库)
学习活动参与度	主动探索次数	10 分: ≥ 10 次	操作行为日志中的“主动发起探索”事件(如新建任务、自由提问、扩展查询)
		6 分: 5~9 次	
		2 分: < 5 次	
	协作交互频率	5 分: ≥ 5 次有效互动 3 分: 2~4 次 1 分: 0~1 次	与 LLM 对话轮次、协作标记
认知建构深度	概念关联数量	5 分: 主动建立 ≥ 5 个新概念连接	概念连接器使用记录、知识图谱中新增的边、学生绘制的关系图
		3 分: 2~4 个	
		1 分: 0~1 个	
	问题解决策略质量	5 分: 采用 ≥ 2 种不同策略, 且逻辑正确 3 分: 使用 1 种策略, 基本合理 1 分: 策略混乱或无效	认知策略库调用记录、反思日志中描述的问题解决过程
元认知能力	知识应用迁移	5 分: 能将所学应用于未讲授的新情境	情境案例库中的迁移任务表现、LLM 对开放性回答的语义分析
		3 分: 能复现练习过的问题	
		1 分: 仅能机械记忆	
	反思日志质量	10 分: 包含具体困惑、原因分析及改进方向 6 分: 简单描述做了什么 2 分: 无反思或敷衍	反思工具提交内容, 经 LLM 评分类似度打分
问题解决成效	学习路径调整次数	5 分: 主动调整学习计划 ≥ 3 次	轨迹库中学习路径分支记录、与 LLM 计划变更的对话
		3 分: 调整 1~2 次	
		1 分: 从未调整	
	最终正确率	5 分: $\geq 90\%$ 3 分: 60%~89% 1 分: $< 60\%$	任务提交答案与标准解(或 LLM 评判)的匹配度

表 1 上述各项分值求和满分 100 分, 可对应以下学习过程等级:

优秀(90~100 分): 学习过程高度自主, 认知建构深刻, 元认知能力突出。

良好(80~89 分): 能完成主要学习目标, 但深度或策略性有待加强。

中等(70~79 分): 学习目标完成较好, 但知识应用迁移有待提高。

合格(60~69 分): 基本完成任务, 但主动性和反思意识不足。

待改进(< 60 分): 学习过程碎片化, 需要更多支持与引导。

评分表由 LLM 根据轨迹库的数据自动计算, 为教师和学生提供即时的过程性反馈。

5.3. 学业表现

图 2 数据表明实验组在期末考试成绩均值、项目实战完成度、复杂问题解决率、错误重复率和知识

迁移能力五个维度都比控制组显著提升，提升幅度最少 16.3%，最多 64.2%，差异极显著($p < 0.001$)。证明教学方案在解决课程教学难题方面有一定成效。

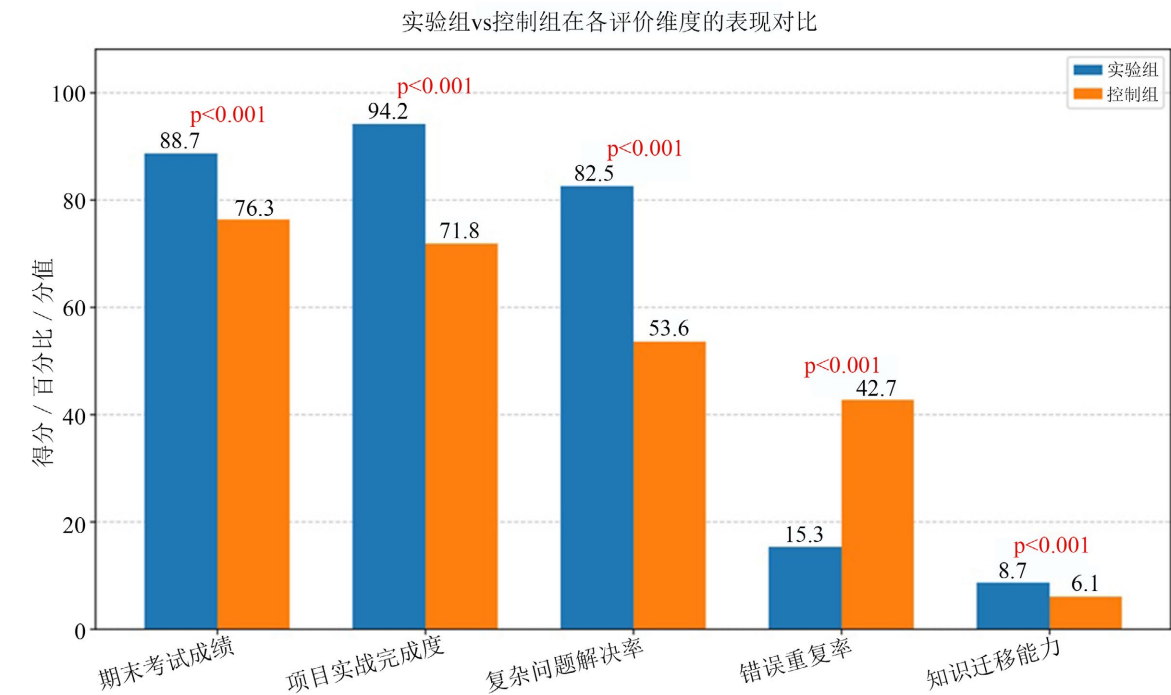


Figure 2. Academic performance
图 2. 学业表现

5.4. 学习过程分析

Table 2. Construction trajectory data analysis
表 2. 建构轨迹数据分析

学习行为指标	均值	与传统模式对比	教学意义
主动探索时长占比	68.4%	+285%	从被动接收转向主动建构
反思活动参与度	73.2 次/人	+412%	元认知能力显著提升
概念网络连接数	14.3 个/知识点	+189%	知识结构化程度更高
个性化路径匹配度	86.7%	N/A	系统精准识别学习需求
社会互动频次	12.5 次/周	+220%	协作学习常态化

表 2 数据表明，学生主动探索时长占比达到 68.4%，较传统模式提升近 3 倍，反思活动参与度高达 73.2 次/人，较传统模式增长超过 4 倍，概念网络连接数达到 14.3 个/知识点，较传统模式提升近 2 倍，个性化路径匹配度达到 86.7%，社会互动频次达到 12.5 次/周，较传统模式提升 2 倍以上。五个维度的行为指标表明学生从 Linux 知识的被动接受者，转变为自己知识体系的主动建构者，实验组共 80 人。

5.5. LLM 脚手架有效性

表 3 数据显示，四类干预措施的接受采纳率均高于 76%，其中部分干预类型的采纳率超过 94%。LLM

作为认知脚手架，在学习连续性、促进知识结构化建构、激发元认知反思以及提升知识迁移能力等方面有一定效能。

Table 3. Matrix of LLM intervention types and their effects
表 3. LLM 干预类型及效果矩阵

干预类型	应用频率	受采纳率	续错误率降低
认知脚手架提示	5.3 次/学时	91.2%	67.3%
概念连接展示	2.1 次/学时	88.7%	58.4%
反思促进提问	1.8 次/学时	76.5%	72.1%
情境建构支持	1.2 次/学时	94.3%	81.6%

5.6. 认知能力发展评估

表 4 实验组最高是 50 分，数据显示实验组五项能力的提升均超过 17 分，表明教学干预对认知发展的促进作用具有全面性而非局限于某一特定能力。LLM 认知脚手架促进了学生认知能力的发展，实现了从“学会知识”到“发展思维”的教育跃升。

Table 4. Cognitive ability scale
表 4. 认知能力量

能力维度	实验组前测	实验组后测	提升值	控制组提升值
问题分解能力	18.3	36.7	+18.4	+8.2
系统思维	16.2	34.8	+18.6	+6.5
元认知调控	15.8	33.2	+17.4	+5.1
知识迁移	14.6	32.5	+17.9	+4.8
协作解决问题	17.1	35.4	+18.3	+6.7

上述数据表明学生学业成绩得到了提升，同时也有助于动手能力的提高。

6. 结论

目前，在 Linux 下进行应用开发顺应了整个技术产业的发展趋势。教改方案使学生在交互与建构层通过真实活动进行探索；LLM 认知引擎层实时理解学生状态，从知识生态与情境库层调用适配的资源与策略，为学生搭建动态的“认知脚手架”；整个过程形成闭环，持续支持并记录学生的知识建构。实现了以认知发展为中心，从标准化生产转向个性化成长。最终，学生在知识与理论层面能深刻理解 Linux 的哲学，在编程与开发层面能尽快成长为一位合格的专业软件开发者。后续研究计划进一步探索该模型在职业教育、企业培训等不同场景中的应用拓展。

基金项目

西安理工大学教学改革项目“基于大模型的 Linux 基础与应用智能辅助系统研究”(xjy2549)；

陕西省本科教育教学改革研究项目(重点项目)(No. 25BZ062)“师生机”三元协同：基于多模态 AI 课堂行为分析的教师能力提升精准赋能研究与实践。

参考文献

[1] 高汉威, 林潇潇. 生成式人工智能技术驱动下基于建构主义学习理论的高校数字艺术教育新路径——以专业实

- 践课程为例[J]. 中国教育装备技术, 2026(2): 42-48.
- [2] 张玮, 王珊珊, 李筱璨, 等. 基于大语言模型的智能教学内容提炼与自适应学习平台设计研究[J]. 高等工程教育研究, 2026(1): 174-178, 195.
- [3] 赖兆林, 李广原, 丁红, 等. 基于大语言模型的数据结构课程自适应教学模式构建[J]. 中国现代教育装备, 2026(3): 35-37, 41.
- [4] 詹晓萍, 陈道华, 蒋海蓉. 建构主义理论视域下的初中地理跨学科实验教学设计——以“模拟云的形成”为例[J]. 地理教学 2026(3): 50-54.
- [5] 崔凯欣. 建构主义理论在博物馆教育中的应用: 从理论到实践的探索[J]. 文物鉴定与鉴赏, 2026(02): 101-104.
- [6] 林远科, 张望, 袁刚, 等. 基于建构主义理论的体育专业实习实践平台模式探析[J]. 体育世界, 2026(1): 96-98.
- [7] 吴伟, 杨双奇, 周福辉, 等. 知识图谱驱动大语言模型在课程体系建设中的应用探索[J]. 大学教育, 2025(17): 39-43.
- [8] 钟益华, 陈继力, 陈逸璇. 面向大语言模型系统能力培养的“三段式”实验教学设计——以 ChatGLM-6B 本地微调为例[J]. 计算机教育, 2026(2): 23-28.
- [9] 毛新军, 董威, 尹良泽, 等. 大语言模型辅助软件开发实践的探索[J]. 高等工程教育研究, 2025(6): 33-38.