

大模型赋能下计算机省一流课程教学内容更新机制研究

申玉鹏

山东现代学院 山东济南 250104

摘要: 在大模型技术迅速发展的背景下, 高校计算机课程面临内容体系滞后、技术更新跟进不足等现实问题。省级一流课程作为课程改革的重要载体, 亟需建立与大模型技术相匹配的内容更新机制。本文以省级一流课程建设为背景, 分析大模型技术对教学内容组织与更新路径的深层影响, 从 Web 开发、后端逻辑、数据存储、前端交互等模块出发, 构建以“技术演化—知识映射—课程重构”为核心的教学内容更新机制。通过引入 AI 辅助、动态知识图谱与真实案例系统, 实现课程内容与技术前沿的有机对接, 为高校计算机类课程的高质量发展提供实践路径。

关键词: 大模型技术; 计算机课程; 课程改革; AI 辅助

引言

近年, 大模型在编程辅助、智能问答和语言生成方面表现出强大能力, 开始深刻影响高校计算机类课程的教学内容与方法^[1]。传统的教学体系多以稳定性为主^[2], 教学内容依托教材和静态资源, 更新周期缓慢, 已难以满足大模型快速演化背景下的课程建设需求。对于省级一流课程, 如 Java 语言程序设计、Python 开发、MySQL 数据库、前端技术和 Java Web 系统开发等课程, 其教学内容已暴露出与当前技术现实不匹配的结构性问题, 表现为知识覆盖不足、案例过时、能力结构与岗位要求脱节等。当前, 教育部门鼓励将课程建设与新一代信息技术融合, 推动大模型、人工智能与课程内容的深度联动^[3]。然而, 如何实现教学内容的动态调整与结构优化, 仍缺乏一套可推广、可落地的机制路径。本文立足课程一线教学实际, 探索教学内容如何借助大模型技术驱动更新重构, 提升教学内容的时效性、适用性与系统性, 推进课程高质量发展。

1 大模型技术发展背景与教育融合现状

1.1 大模型发展与编程智能化趋势

当前的大模型在自然语言理解、语义生成与代码编写等方面表现出强大能力^[4]。GPT、GLM 等模型不仅能自动生成结构化程序段, 还可进行跨语言转换与代码优化, 具备支持 Java、Python、SQL 等主流语言开发的基础能力。在 IDE 插件、Web 开发环境、调试工具等环节, 大模型技术已逐步融入编程实践中。

这类模型具备语义上下文理解能力, 能够完成从接口

设计到类结构生成的全过程建议。在 Web 开发中, 已可辅助完成 Servlet 编写、API 封装、前端页面布局等任务, 降低入门门槛, 提升开发效率。这对教学内容提出新要求: 传统讲授语法与框架的教学模式, 必须引入“模型理解—人机协同—工具辅助”的新教学视角。

1.2 大模型在课程教学中的融合现状

目前, 多所高校开始探索将大模型引入教学实践^[5], 涵盖自动批改编程作业、生成教学脚本、分析数据库查询逻辑等场景。以 Python 语言课程为例, 教师可利用大模型提供初始代码骨架, 学生在此基础上优化逻辑; 在数据库 MySQL 教学中, 大模型可辅助解释查询优化方案与索引设计思路, 提升学生理解。

但实际应用仍以工具级嵌入为主, 未能在课程内容结构、更新机制和能力评估体系上形成完整的协同关系。以 Java Web 开发为例, 课程往往仍局限于 MVC 三层结构与 Servlet 处理流程的教学, 未能融入微服务拆分、接口文档自动生成等新兴技术。课程内容与大模型赋能后的工程现实之间存在明显落差。

1.3 教学内容更新机制的缺位与挑战

当前课程内容更新依赖教师个人, 缺乏系统支持。技术变化快, 传统教学落后, 学生学后仍需重新培训。数据库教学仅教手工 SQL, 未涉及 AI 辅助生成等应用; 前端教学未跟上响应式布局和组件化趋势, 降低知识转化。课程与技术脱节是主要瓶颈。需构建技术演进更新机制, 推动内容模块化、案例结构化与资源动态化。

2 省级一流课程内容体系的更新需求与问题分析

2.1 教学内容滞后于主流技术栈变化

计算机类课程核心内容滞后于产业实践。Java Web 教学停留在 JSP、Servlet 等基础，未覆盖 Spring Boot、Restful API 等现代框架；后端转向微服务架构，学生缺乏理解，导致技能断层。数据库课程如 MySQL 聚焦数据定义、基本查询和索引优化，忽略事务隔离、多表一致性和分库分表策略；实际需应对大规模并发、高频调用和缓存策略，课程能力不足。前端教学 HTML/CSS 以静态页面为主，缺乏组件化、响应式布局和交互设计；JavaScript 围绕 DOM 操作，未深入 Vue、React 框架，学生掌握原始技能而非可迁移技能。Python 教学注重基础语法、数据结构和面向对象，未涵盖大模型调用、接口处理和任务调度；企业用于接口构建、自动化脚本和模型部署，教学内容不支撑这些目标。

2.2 教学资源组织结构与内容分布脱节

一流课程在内容分布与结构组织方面，往往仍采用线性结构推进模式，教学模块割裂，缺乏统一的能力导向与跨模块关联。例如，Java 语言教学中函数结构与面向对象设计与 Java Web 应用开发脱节，学生虽掌握语法，却无法在实际项目中合理应用所学；Python 教学与数据库操作教学未形成协同，缺乏基于数据库驱动的数据处理案例，使学生不能完成从数据获取到处理再到可视化呈现的完整流程。

前端教学结构也存在割裂现象。HTML 与 CSS 讲授中缺少与 JavaScript 的整合设计，三者之间缺乏统一项目任务支撑。常见教学内容如页面登录设计、注册验证或图文排版示例，内容老旧、无动态性，缺乏实战性训练。

课程内容更新依赖教材改版与教师个人经验积累，资源来源单一。学生普遍依赖教师 PPT 与代码片段进行学习，案例层次较浅，重复度高，对技术变化响应不足。缺乏教学资源共享平台，课程内容缺少版本管理与更新机制，教学组之间协作程度低，阻碍内容迭代。

2.3 教学目标体系与技术能力要求不匹配

当前多数计算机类课程仍以“知识讲授”为目标，评价标准集中在期末考试与课堂答题的准确率，缺乏对实际开发能力、问题解决能力、模型调用能力的考察。以 Java 课程为例，课程目标是掌握类与对象、继承与多态、异常处理等语法特性，但未设置“能构建基本服务端业务逻辑”“能处理多模块协同开发”等能力导向型目标。

数据库课程中的考核多集中在“写出正确 SQL 语句”，忽略实际工程中需要考虑的执行效率、数据完整性与事务控制问题。Python 课程往往未设计真实数据驱动任务，如从 API 获取数据、进行预处理、保存至数据库并返回结果，导致学生缺乏任务视角。

当前 Java Web 开发课程虽然设置了实验内容，但多为静态示例或模板填空型任务，缺乏真实交互性任务与多模块综合实践。教学任务未与现实岗位能力相对接，学生完成课程后仍需重新接受培训。

随着大模型在自动编码、自动补全与语义理解方面的应用深入，学生未来需掌握的是“与模型协同开发”的复合型能力，如能设计模型提示语、能解读模型生成代码意图、能将模型输出嵌入具体模块。当前教学目标体系中尚未体现这类能力指标，也未形成完整的课程联动路径支持此类能力的生成。

2.4 内容更新机制缺失与路径不清

省级一流课程设计强调结构完整，但执行中缺乏内容更新制度化机制。教师依赖个人经验更新，缺乏明确标准，难统一教学目标与质量。例如，计算机体系结构课程多年未更新案例，仍讲 8086、MIPS，未涉及 GPU、分布式计算等新知识；前端开发未响应 HTML5、Web3.0、React 等趋势。课程大纲修订周期长，更新难度大，案例滞后。缺少行业合作反馈机制，未引入企业案例、真实数据集，学生能力与工作场景脱节。大模型背景下，需动态机制支撑教师工具和平台支持，否则难自主转型。

3 面向大模型技术的计算机课程内容更新机制设计

3.1 教学内容动态更新机制的构建路径

针对当前 Java Web 应用开发、Python 语言程序设计等课程中存在的内容滞后问题，需构建以技术演进为导向的教学内容更新机制。该机制应以“教学内容抽象层→行业前沿抽取层→内容映射与替换层”三层逻辑为基础，确保课程知识体系能够与技术发展保持同步。

以 Java Web 课程为例，可在课程设计中引入“年度内容审查机制”，定期对比主流企业岗位中使用的框架（如 Spring Boot、Thymeleaf、JWT 等）与课程教学覆盖范围，进行内容筛查和补充。对于 Python 课程，则应建立围绕“模型调用—数据处理—接口部署”链条的内容更新单元，将 requests 库的实际 API 应用、FastAPI 的接口发布能力与

LangChain 调用逻辑纳入教学资源, 解决内容覆盖不足问题。

这一机制的关键在于设立技术前沿监测节点。可基于开源社区(如 GitHub 热门项目)、开发者文档更新频率、岗位技能关键词等数据源, 定期整理技术趋势报告, 供教学团队参考, 实现内容更新的前置预警与主动调整。

3.2 跨模块融合式内容组织框架

针对课程结构割裂问题, 应转向基于任务驱动的“模块融合式”教学内容设计框架。该框架不以课程章节为单位, 而是以开发任务为核心, 将多个知识点在同一项目任务中打通整合。

例如, 在“学生信息管理系统”这一教学任务中, 前端模块可覆盖 HTML 结构设计、CSS 样式与 JavaScript 表单验证, 后端使用 Java 或 Python 实现业务逻辑, 数据库部分涉及 MySQL 表设计与增删查改操作。通过这种项目型教学, 将前端开发、Java Web 编程、数据库原理自然融合, 打破课程割裂, 增强内容之间的逻辑联系。

此外, 针对计算机体系结构课程内容封闭、更新慢的问题, 可引入“结构解剖+模型关联”的设计模式, 将经典 CPU 结构讲解与当前 AI 芯片架构做对比, 引导学生从底层结构逻辑延伸理解大模型的算力调度基础, 为后续模型调用课程奠定理解基础。

3.3 目标导向型教学能力结构重构

现行教学目标以知识点掌握为导向, 难以衡量学生岗位实际开发能力, 因此应以岗位能力要求重构目标, 将任务型能力纳入评价体系。

例如, Java 课程目标从“掌握继承与多态”转为“能基于类结构设计通用业务逻辑”, Python 课程从“会用 for 循环处理列表”转为考查“封装数据清洗函数处理 API 结果”。

考核形式需转向开放式任务, 如 Java Web 课程设计“实现带用户验证的简易留言板”, 要求学生完成注册、会话管理、数据存储等任务, 将知识转化为系统能力。

大模型可辅助目标对齐, 如教师用它生成考题模板、调试案例和评分标准, 提升评价效率与一致性。

3.4 教学资源智能推荐与案例更新机制

为解决教学资源更新慢、内容旧的问题, 建设智能教学内容推荐平台。该平台基于大模型, 输入教学目标、课程

模块与关键词, 自动生成代码片段、项目示例或教学任务建议, 并提供参考答案供教师筛选。例如, 输入“Python 接口调用+数据可视化”, 系统推荐 requests+matplotlib 综合示例, 并生成教学步骤与拓展任务, 减少教师时间负担。

在案例更新方面, 建立项目案例库与学生反馈机制。案例通过版本控制管理, 定期由教师组与助教复审更新, 结合学生反馈调整难度、逻辑与功能设置。例如, 数据库课程引入学生真实项目的表结构与查询逻辑, 前端课程使用企业官网或开源界面的 UI 设计稿作为开发任务, 增强案例现实性与典型性。

4 结束语

大模型技术快速发展, 传统计算机省级一流课程面临内容体系滞后、教学资源陈旧和能力结构不匹配挑战。本文构建了以“技术—知识—课程”为主线的更新机制, 提出 AI 驱动动态调整框架、真实案例融合机制和目标协同路径, 实现课程内容与前沿技术同步演化。研究表明, 该机制提升教学质量和产业响应能力。未来需实证检验、完善资源管理平台 and 教师协同机制, 推动广泛应用, 支持高等教育改革。

参考文献:

- [1] 米洪, 郑莹. 基于能力框架与 GROW 模型融合的智能化职业场景现场工程师培养路径研究 [J]. 电脑知识与技术, 2024, 20(34): 141-143.
- [2] 刘铭, 赵妍妍, 秦兵. 大模型时代下的知识工程课程再思考 [J]. 黑龙江教育 (高教研究与评估), 2025, (06): 43-48.
- [3] 赵竞, 郑捷. 生成式人工智能在高职教学改革中的应用研究——以通识课人工智能基础为例 [J]. 现代职业教育, 2025, (20): 93-96.
- [4] 张娟. 大模型交互功能助力计算机学科教学的实证研究与实践路径优化 [C]// 中国高校校办产业协会终身学习专业委员会. 第四届教育信息技术创新与发展学术研讨会论文集. 哈尔滨信息工程学院, 2025: 759-762.
- [5] 谢蔚泓, 刘莲, 祝烈煌. 国产自主大模型引领高校党建创新发展的路径探索 [J]. 北京教育 (高教), 2025, (07): 82-84.

作者简介: 申玉鹏 (1998—), 男, 汉族, 山东省德州市齐河县, 助教, 硕士, 研究方向为目标检测。