

人工智能赋能“程序设计与问题求解” 课程的教学改革与实践研究

樊庭平¹ 谢绍敏¹ 付华亮^{2*}

1. 桂林电子科技大学, 计算机工程学院, 中国·广西 北海 536000;

2. 桂林电子科技大学, 基础教学部, 中国·广西 北海 536000

【摘要】针对传统“知识传递”型课堂难以有效提升程序设计类课程教学效果, 学生的问题求解能力没有得到系统培养的问题。本文在基于人工智能的教学改革策略, 构建以AI为导向的C++课程体系, 阐述人工智能技术如AI编程助手在的编程学习的作用, 提出AI赋能的混合式教学模式, 并结合实际教学展示其改革成果与面临的挑战, 为相关课程的教学改革提供参考。

【关键词】人工智能; 程序设计与问题求解; AI编程助手; 混合式教学模式

【基金项目】教育部产学研合作协同育人项目“《Python程序设计》课程建设与教学实践改革”(项目编号: 202102211103)

1 现有“程序设计与问题求解”课程教学的局限性和不足

1.1 教学方法单一, 缺乏灵活性与互动性

统教学模式以教师为中心, 采用“满堂灌”方式传授知识。像在编程教学中, 教师照本宣科讲解基础语法等理论, 单方面演示算法公式推导和代码, 忽略学生主观能动性, 致使学学生难以理解算法实际应用场景与内在逻辑。同时, 统一的教学进度无法兼顾不同学生的学习速度和理解能力, 学习能力强的觉得进度慢, 基础薄弱的跟不上, 最终拖累整体教学效果。

1.2 教学内容更新缓慢, 无法跟上技术发展的步伐

目前的“程序设计与问题求解”课程的教学内容还是侧重基于基础语法的学习。教材中的部分知识点陈旧, 无法反映当前行业的最新动态和发展趋势。例如, 近年来人工智能、大数据、区块链等新兴技术蓬勃发展, 对编程人员的技能要求也发生了很大变化。但传统课程中可能很少涉及这些领域的相关知识和技术, 导致学生毕业后所学知识与市场需求不匹配, 缺乏竞争力。

2 人工智能赋能程序设计

2.1 人工智能时代的软件编程

人工智能技术的快速发展, 现有的大语言模型的发展, 尤其是AI编程助手(GitHub Copilot, DeepCode等)的出现推动了智能编程领域的发展。很多编程助手可以集成到C++的主流IDE中, 其功能主要以下三个方面: (1) 研发知识问答: 程序员不管是技术还是语言问题, 都可以向AI编程助手

寻求解答。编程助手不仅能提供高准确性的答案, 还会附上具有针对性的示范代码。(2) 自动代码的生成: 程序员可以通过自然语言或注释, 就能借助AI编程助手实现代码的自动生成。此外, 编程助手还具备依据代码上下文智能补全代码的能力。(3) 代码分析和优化: AI编程助手能够深入剖析代码, 精准识别潜在问题和性能瓶颈, 并基于丰富的算法和数据模型, 提供专业的优化建议和方案。

2.2 基于AI辅助编程的学生程序设计能力培养

对学生而言, 在《程序设计与问题求解》的课程中, 要提高自己的编程能力, 一方面, 需要学生自身不断的学习和实践, 尤其编程是一门实践性极强的学科, 学生需要排查和解决编程中遇到的问题。另一方面, 还需要有丰富的编程经验开发的人员帮忙答疑解惑, 引导学生思考解决问题的思路。AI编程助手这类大模型的出现, 为学生的编程学习提供了新的可能。AI编程可以作为智能辅助工具, 为学生提供了一种全新的学习和实践方式。它能够弥补传统学习方式的不足, 为学生提供更加便捷、高效、个性化的学习支持, 帮助学生更快地提高编程能力。

3 基于人工智能的教学改革策略

构建以AI为导向的C++课程体系。构建以AI为导向的C++课程体系, 要遵循由浅入深的学习节奏, 同时也要理论与实践相结合。首先, 在C++基础课程中, 除了传统的语法、数据结构和算法教学外, 增加与AI应用紧密相关的内容。例如, 在讲解数据类型时, 详细介绍 AI 领域常用的数据表示方法, 如向量、矩阵等; 在算法教学中, 引入AI算法中的基

础算法，如搜索算法在路径规划中的应用，为后续的AI学习奠定坚实的编程基础。另一方面，将AI相关的核心课程融入C++教学中，如机器学习、深度学习、计算机视觉、自然语言处理等，与C++编程相结合。通过项目实践的方式，让学生使用C++实现AI算法和模型，如使用C++和OpenCV库进行图像识别项目，使用C++和TensorFlow库进行深度学习模型的训练和部署，使学生在实践中掌握AI技术与C++编程的融合应用。以下是在部分《程序设计与问题求解》课程教学中加入的AI库的教学大纲。

表3-1 基于C++的机器学习基础库

课时	学习目标	学习内容	实验案例
4	掌握C++核心语法与OpenCV基础	(1) C++语法回顾(类、模板、STL) (2) OpenCV安装与环境配置 (3) OpenCV图像读写与显示	实验1: 用OpenCV读取图像并显示灰度图
4	理解内存管理与AI库集成	(1) C++指针与内存管理 (2) OpenCV矩阵操作(Mat类) (3) 图像像素遍历与修改	实验2: 实现图像颜色反转(RGB通道操作)
4	掌握文件操作与数据预处理	(1) C++的文件流类 (2) OpenCV图像滤波(高斯、中值滤波)	实验3: 对图像添加噪声并用滤波降噪
4	学习AI项目开发流程	(1) CMake构建项目 (2) OpenCV与TensorFlow库链接 (3) 简单命令行工具开发	实验4: 用CMake编译含OpenCV的C++程序

4 AI赋能混合式教学在《程序设计与问题求解》课程的应用

4.1 教学方案与流程

以“模板和异常处理”教学为例，借助AI编程助手开展混合式教学，通过“课前-课中-课后”三阶段联动，实现智能化与个性化学习闭环。

课前，学生在对分易平台观看教师录制的教学视频，教师分享优质学习链接拓展学生知识。学生借助CodeGeeX、ChatGPT等AI工具，输入编程需求生成基础代码。如学习异常处理时，输入“生成一个处理除零异常的代码示例”，获取代码框架，为课堂学习预热。课堂上，教师实时编码演示，运用模板和异常处理知识解决实际问题。此时，Visual Studio Code的智能代码补全插件等AI助手实时提供代码补全建议，提升编码效率，像输入try关键字时，自动提示catch块结构。随后，教师引导学生分组讨论代码优化方案，针对函数模板讨论如何提升通用性和性能，对异常处理代码探讨如何全面捕获和处理

异常，激发思维碰撞，加深知识理解。

课后，学生利用专门的测试用例生成AI工具，输入要测试的函数或类，生成涵盖边界条件和正常情况的测试用例，对自己编写的代码进行验证，检查漏洞错误后提交作业。教师批改作业，掌握学生知识掌握情况，以便后续针对性辅导。

项目协作中，涉及模板和异常处理的项目，团队借助AI工具进行任务拆解。例如处理用户输入并计算数据的项目，AI识别用户意图，将其拆分为输入验证、数据处理、异常处理等模块。针对各模块，AI生成模块化代码，如输入验证模块生成格式验证代码，异常处理模块生成应对各类异常的代码。教师在旁指导学生进行代码整合与调试，确保项目顺利运行，培养学生团队协作与实际项目开发能力，如下表所示。

表4-1 AI赋能混合式教学模式

阶段	具体实施
课前预习	学生通过平台观看教学视频，使用AI生成基础代码(如类定义、函数模板)。
课堂互动	教师演示实时编码，AI助手提供代码补全建议；学生分组讨论代码优化方案。
课后实践	学生使用AI生成测试用例，完成作业并提交给教师
实验协作	分组团队通过AI拆解实验任务(如用户意图识别)，生成模块化代码，教师指导整合与调试。

4.2 教学实施与学生学习成效评价

4.2.1 教学实施

(1) 教师培训：在实施AI赋能混合式教学前，教师应参加相关AI辅助教学的培训。提升新兴技术与教学相结合的能力。

(2) 课程内容重构：将AI相关知识理论融入实际案例C++课程，构建基于C++的包含深度学习、机器学习的实践项目。

(3) 资源整合与管理：教师需要对教学资源进行整合，包括录制高质量的教学视频、筛选优质的学习链接等。同时，利用对分易平台等教学管理系统，合理安排教学任务，发布学习资源。

(4) 课堂组织与引导：在课堂互动环节，教师要精心组织小组讨论，合理分组，确保每个小组都有不同层次的学生，以促进学生之间的互帮互助和共同进步。在学生讨论过程中，教师要积极巡视，及时给予指导和反馈，引导

学生深入思考问题，避免讨论偏离主题。

4.2.2 学生学习成效评价

在实施AI赋能混合式教学模式后，通过课堂、作业和考试反馈，对学生的学习成效进行深入分析，有以下几个显著变化：（1）学习成绩的提升：AI赋能的混合式教学模式通过个性化学习路径、智能推荐资源和实时反馈机制，帮助学生更好地理解和掌握理论知识。如表4-1, 2023-2024学年第一学期的学生通过利用AI编程助手辅助编程，比2023-2024学年第一学期的学生比较，理论考试成绩平均提高了19.6%，实验考核成绩平均提高了30.6%，这充分反映了AI赋能混合式教学模式可以提高整体学习效果。（2）编程能力的提高：在编程类课程中，AI赋能的混合式教学模式通过自动化的代码评估、实时错误提示和个性化练习推荐，显著提升了学生的编程能力。学生可以通过AI工具进行反复练习，系统能够根据学生的编程水平动态调整难度，帮助学生在实践中逐步提升编程技能。此外，AI还能提供即时的代码优化建议，帮助学生理解最佳实践，从而提高编程效率和代码质量。（3）自主学习能力的增强：AI赋能的混合式教学模式强调学生的自主学习能力。通过智能学习平台，学生可以根据自己的学习进度和兴趣选择学习内容，AI系统会根据学生的学习行为数据提供个性化的学习建议。这种模式培养了学生的自主学习能力，使他们能够更好地管理学习时间，提升学习效率。

表4-1 《程序设计与问题求解》成绩统计(任课教师 樊庭平)

年级	理论考试平均成绩	实验考核平均成绩
2023-2024学年 第一学期	75.3	67.2
2024-2025学年 第一学期	82.4	73.8
增长率	19.6%	30.6%

5 人工智能赋能教学改革面临的挑战

尽管AI编程助手为“程序设计与问题求解”教学带来了效率提升与个性化支持，但其在具体课程落地中仍面临多重挑战，需从技术适配性、教育目标契合度及学生认知规律等维度综合应对。

5.1 AI编程助手集成开发环境（IDE）的复杂性

学生在使用集成开发环境（IDE）时可能遇到的问题，尤其是在结合AI编程助手的情况下。首先，IDE的复杂性本身可能是一个障碍，特别是对于大一学生来说，配置和操作已经有一定难度。如果再加上AI助手的集成，可能会增加学习曲线。

5.2 过度依赖导致底层理解缺失

AI工具的即时反馈特性可能使学生陷入“快餐式学习”。比如，学生的调试能力退化，会产生依赖AI自动修复错误，学生失去手动调试的机会，难以培养问题定位能力；而且，直接生成完整代码框架可能阻碍学生从需求分析到模块设计的系统性训练。

AI编程助手与“程序设计与问题求解”课程的深度融合，本质上是技术工具与教育规律的博弈过程，唯有在效率与深度、便利性与严谨性之间找到平衡，方能真正实现“赋能”而非“替代”的教育价值。

6 结论

AI编程助手与“程序设计与问题求解”课程的深度融合，本质上是技术工具与教育规律的博弈过程，唯有在效率与深度、便利性与严谨性之间找到平衡，方能真正实现“赋能”而非“替代”的教育价值。

参考文献：

[1] 尹良泽, 徐建军, 李姗姗, 等. 人工智能时代下的计算机程序设计课程教学探探[J]. 计算机教育, 2025, (02): 123-127.

[2] 于永涛. 人工智能赋能C++程序设计课程的教学改革与实践研究[J]. 电脑知识与技术, 2024, 20(35): 178-180.

作者简介：

樊庭平（1989.10-），女，汉族，河北省邢台市，硕士研究生，讲师，研究方向：信息技术教育学。

谢绍敏（1984.08-），女，汉族，广西省北海市，硕士研究生，高级工程师，研究方向：图像处理。

通讯作者：

付华亮（1987.04-），男，汉族，山西省临汾市，硕士研究生，讲师，研究方向：微分差分方程。