

生成式人工智能在程序设计课程教学中的应用研究

莫洁安¹, 卢丽金^{2*}

¹广西民族师范学院数学与计算机科学学院, 广西 崇左

²广西民族师范学院物理与电子信息工程学院, 广西 崇左

收稿日期: 2026年9月25日; 录用日期: 2026年10月21日; 发布日期: 2026年10月28日

摘要

针对《Python程序设计》课程学生基础差异明显的问题, 本文探讨生成式人工智能融入课程教学的路径。在方法上, 综合采用文献研究、案例分析与设计研究方法, 构建由自适应学习路径、实时互动问答、智能项目评估与协作编程支持四个模块构成的教学模型, 进而提出实施路径、风险治理与准实验评估方案。该设计从支撑高阶能力培养、个性化学习路径、产业案例库与师资平台四个层面推进课程改革, 并借助学术诚信、技术依赖、数字鸿沟与数据安全四类机制实施风险控制。预期成效通过项目完成质量、代码纠错能力、学习参与度、创新思维与课程满意度五项指标进行观测, 可为本科院校《Python程序设计》课程改革提供参考。

关键词

生成式人工智能, Python程序设计, 教学改革, 个性化学习

Research on the Application of Generative Artificial Intelligence in Programming Course Teaching

Jie'an Mo¹, Lijin Lu^{2*}

¹School of Mathematics and Computer Science, Guangxi Minzu Normal University, Chongzuo Guangxi

²School of Physics and Electronic Information Engineering, Guangxi Minzu Normal University, Chongzuo Guangxi

Received: September 25, 2026; accepted: October 21, 2026; published: October 28, 2026

*通讯作者。

文章引用: 莫洁安, 卢丽金. 生成式人工智能在程序设计课程教学中的应用研究[J]. 教育进展, 2026, 16(10): 2043-2049. DOI: 10.12677/ae.2026.16102317

Abstract

Aiming at the problem of obvious differences in students' foundations in the "Python Programming" course, this paper discusses the path of integrating generative artificial intelligence into course teaching. In terms of methods, this paper comprehensively adopts literature research, case analysis and design research methods to construct a teaching model composed of four modules: adaptive learning path, real-time interactive Q&A, intelligent project evaluation and collaborative programming support, and then propose implementation paths, risk governance and quasi-experimental evaluation schemes. The design promotes curriculum reform from four levels: supporting high-order ability cultivation, personalized learning paths, industrial case databases and teacher platforms, and implements risk control through four mechanisms: academic integrity, technical dependence, digital divide and data security. The expected results are observed through five indicators: project completion quality, code error correction ability, learning participation, innovative thinking and course satisfaction, which can provide reference for the reform of "Python Programming" courses in undergraduate colleges.

Keywords

Generative Artificial Intelligence, Python Programming, Teaching Reform, Personalized Learning

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

2022 年 OpenAI 发布 ChatGPT, 生成式文本技术出现跨越式发展, 通常被视为生成式人工智能走向实用的分界点。其影响很快越出技术圈: 自动化内容生成、个性化学习路径设计与智能辅导系统开始改写课程开发与教学组织的逻辑。以 ChatGPT 为代表的技术形态是生成式人工智能在教育领域机遇与挑战同时放大[1]: 教师可获得课程设计与实时反馈支持, 学生批判性思维与创新能力有了新训练载体; 挑战则集中在技术依赖、数字鸿沟扩大与既有评估体系的适应性上。Python 凭借语法简洁与生态完备, 成为高校程序设计类课程的主力语言[2], 也是生成式人工智能最容易介入的教学场景之一。

2. 生成式人工智能在程序设计课程教学中的应用现状

就渗透范围而言, 生成式人工智能已进入 C 语言、Python、Matlab 等多门程序设计课程的教学实践。其典型做法为: 依据学习数据定制课前推送内容, 课中借助虚拟实验场景进行互动讲解, 进而形成适配不同节奏的学习路径。在效果层面, 已有证据表明: 相关调查显示, 超过三分之一的 C 语言学习者已将生成式人工智能作为学习工具[3]。因此, 技术并非教学的外部选项, 而是学生行为中的既成事实, 课程需回应的是如何引导而非是否引入。

在本科层次, 《Python 程序设计》课程的针对性研究仍显薄弱。既有研究多聚焦于工具功能验证与单一环节效果检验, 较少探讨产业需求与本科培养目标如何在同一门课程中统筹安排。课堂观察亦提示, 生成式人工智能的作用发挥高度依赖教师的问题设计与互动组织, 工具本身并不自动带来教学质量的提升。本科院校课程改革不宜停留于通用方案移植。本研究据此定位: 以学生基础差异与产业环境为约束条件, 提出涵盖实施路径、风险治理与成效评估的完整设计, 而非局限于单项技术应用的效果报告。

3. 面向 Python 课程的教学改革实施路径

本研究构建的 Python 教学系统总体架构系由基础支撑、学情感知、教学组织、资源生成与评估反馈等层次构成的分层体系, 后续教学改革路径均据此逐层展开。各层次在功能上相互衔接: 基础支撑为系统运行提供必要保障, 学情感知获取学习者状态信息, 教学组织据此统筹教学安排, 资源生成层进一步形成适配的教学内容, 评估反馈层则对整体运行效果进行持续观测。在系统实际运转中, 学情数据汇入教学组织层后, 经该层分析与处理, 流转至资源生成层输出相应教学内容; 评估反馈层则贯穿教学全程, 对相关过程信息进行动态采集与回传, 为后续路径的动态调整提供依据。

3.1. 深化人工智能对高阶能力的支撑

倘若仅将生成式人工智能定位为答疑工具或代码生成器, 其应用便止步于浅表层面。教学改革的着力点应当落实于编程思维、算法设计、项目实战与问题解决等高阶目标。围绕 Python 课程知识结构, 可从四个维度推进: 以阶梯式问题探究夯实算法思维, 借助项目实战促进落地能力提升, 通过逆向纠错训练培育批判性意识, 依托跨学科场景拓展创新边界。上述四个维度分别对应本研究四个教学模块——自适应学习路径、实时互动问答、智能项目评估与协作编程支持, 并分别锚定至相应教学环节: 自适应路径支撑课前诊断, 实时问答贯穿课中答疑, 智能评估衔接课后作业, 协作编程支持覆盖小组实训。

3.2. 个性化学习路径适配

路径适配的逻辑起点在于学情数据的转化。生成式人工智能能够汇集入学基础测试成绩、课堂互动行为等数据, 并据此对基础薄弱点、语言理解难点与学习偏好进行标注。学情画像的价值并不在于数据量的简单堆积, 而在于能否被转译为具体的学习活动安排。从多源数据采集、学情画像构建到分层模块生成, 构成了学情画像向分层模块转化的适配链路, 并经由平台实时追踪形成动态调整回路。

就基础差异而言, 针对基础薄弱的学生, 教学讲解应侧重语法可视化, 将抽象代码转化为贴近生活经验的场景图像, 以促进语言符号与经验形象之间的认知联结; 针对进阶学生, 则可推送与产业需求相衔接的项目实训, 如经济数据分析或 Web 应用开发。合作学习任务的智能拆分, 亦是对学生偏好群体学习特征的回应。若适配机制缺乏反馈回路, 极易退化为一次性分层。因此, 需借助平台对学习状态进行实时追踪, 当识别出高频知识疏漏或学习效率下降时, 即触发补偿性学习模块, 动态调整学习节奏与难度; 推荐算法亦应接受学生反馈校正, 在知识掌握度、学习效率与兴趣留存之间维持平衡。

3.3. 契合产业需求的 AI 教学案例库开发

(1) 锚定区域核心产业场景构建案例模块

案例库的第一层级为产业场景。智能制造、智慧物流与空天信息数据处理构成了三个可直接转化的领域。以入选工业和信息化部 2024 年先进计算典型案例的工业 AI 大模型为参照, 可对其数据采集与关键指标预测流程加以模拟; 面向典型业务场景, 则可开发贸易数据校验系统, 以适配合规筛查与物流跟踪需求。文化类场景亦被纳入案例序列: 传统音乐特征提取服务于文化传承。

(2) 深化校企协同共建机制

案例的行业属性源自企业实践。通过与本地企业合作, 可将真实项目分解为阶梯式教学任务, 其内容涵盖从基础数据清洗到 AI 模型调优与部署的完整过程, 覆盖 Python 全技能层级。企业技术标准与操作规范应内嵌于案例之中, 如与 Python 的协同应用及多语言系统适配, 使练习环节转化为工程实践。学院依托访企拓岗行动拓展合作范围, 与区域外企业联合开发智能物流管理系统、智能家居控制软件等教学题目。

(3) 案例的能力进阶设计

案例库若仅以产业类别作为划分依据，则每个案例均须标注其所对应的能力台阶。其中，智慧农业类案例侧重于农业大数据建模与预测算法设计，智慧物流类案例则着力强化跨平台业务联动与全链路整合思维。各案例内部均设置智能工具应用环节，用以辅助需求拆解并提供代码优化建议；然而，工具介入深度随能力台阶的上升而递减，学生自主决策的比重相应提高。跨学科方向同样具有拓展空间：医学数据分析领域的应用研究表明，跨学科案例的能力台阶与评价口径均须结合学科特点进行重新校准[4]，以避免案例难度与本科培养目标之间出现错位。

3.4. 师资数字素养提升与 AI 教学平台搭建

(1) 师资数字素养的分级提升

教学改革最终落脚点在于教师。构建分级培训、场景实践与长效激励有机衔接的机制。培训内容依据年龄与技术基础进行分层：青年教师侧重 AI 编程教学工具的实际操作。北京交通大学所采用的“三化”路径在组织方式上提供了可资借鉴的经验。激励环节不可或缺：设立 AI 教学创新专项，并将融合教学成效纳入教师考核与职称评聘。工程教育领域的研究表明，教师能力培养需与学生能力测评认证体系同步设计[5]。

(2) 校本平台的功能设计与安全边界

平台建设目标可归纳为：聚焦教学、贴合学情、安全可控。高校“一库一表一平台”的建设思路具有借鉴意义，即整合 Python 课程教学资源，并嵌入贴合产业需求的案例库与学情适配模块。核心功能包括：AI 学情动态分析，用于捕捉基础差异并追踪进度；个性化资源推送，针对薄弱知识点生成定制化练习；实时代码校验与 AI 答疑系统，支持多语言交互。技术支撑可联合本地企业协同共建，但数据安全保障与隐私保护机制须由学校主导设定，学生学习行为数据不应作为企业可自由使用的资源。平台贯通教学、实训与评价的数据链路后，“教-学-评-改”闭环方具备运行条件。

4. 应用风险与保障机制

随着生成式人工智能逐步引入课程教学，学术诚信、技术依赖、数字鸿沟与数据安全四类风险并非孤立存在，而是在实际教学过程中相互交织、相互影响，共同形成复合性的风险格局。若要确保相关应用在课程教学中顺利落地，就必须针对上述交织并存的各类风险，配置系统化、相配套的保障机制，使风险防控与条件保障之间形成对应关系。其整体结构关系如图 1 所示。

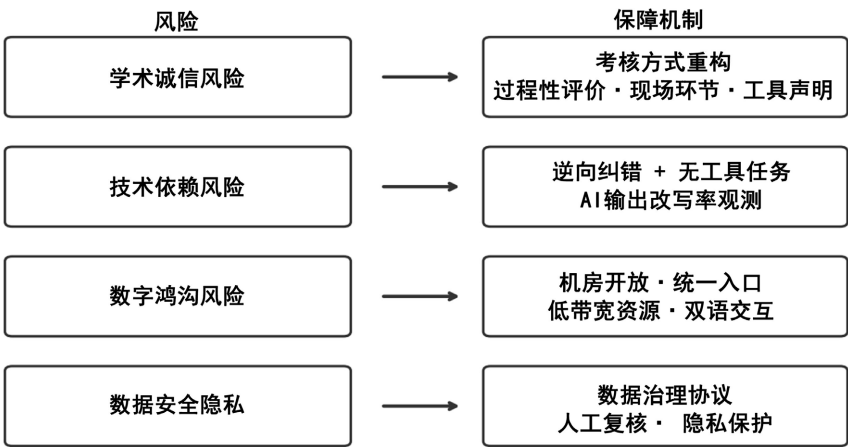


Figure 1. Application risk and safeguard mechanism framework
图 1. 应用风险与保障机制框架

图 1 将四类风险纵向排列, 每一类均配置预防、监测与处置三道保障环节, 形成“风险 - 保障”对应框架。

4.1. 学术诚信风险与课程考核方式重构

生成式人工智能介入课程教学后, 作业与考核的有效性首先受到冲击。学生借助工具生成完整代码并提交, 教师难以从结果层面加以辨识, 以最终代码为主要依据的评分方式因而丧失区分度。李显已在职业教育情境中指出教育诚信隐患[6]。在本科课程中, 此类风险更为集中, 课程设计与综合实训环节的成绩权重较高。

可行的应对措施并非禁用工具, 而是对考核结构进行改造。一种做法是将评价重心由结果前移至过程: 代码提交须附带需求拆解记录、技术选型理由与调试日志, 评分依据由“代码正确性”拓展至“过程可解释性”。另一种做法是在考核中保留一定比例的现场环节, 例如要求学生就所提交代码接受追问, 并在限定时间内完成局部修改与重构。课程还须就工具使用确立明确边界, 即明确哪些环节鼓励使用、哪些环节禁止以及如何进行引用标注, 相关内容应在大纲中予以规定。

4.2. 技术依赖与批判性思维弱化防范

技术依赖与批判性思维弱化构成一组相互关联的风险。对照实验结果表明, 生成式人工智能在复杂编程题目上的准确率显著下降[2]。若学生缺乏对工具使用边界的明确认知, 生成结果易被直接视为正确答案, 独立调试训练的机会亦随之流失。

辅以阶段性任务: 在算法设计与调试环节设置若干禁用生成式工具的练习, 用以检验学生的实际掌握程度。在评价方面, 可增设“AI 输出改写率”观测点, 评估学生对生成内容的修改幅度与质量——改动过少表明理解不足, 改动虽多而质量不高则说明判断力尚未形成。

4.3. 数字鸿沟与学生可及性保障

数字鸿沟在院校情境中呈现为具体差异, 而非抽象概念。学生自有设备与家庭网络条件的差异, 直接影响课外练习的时间投入与学习深度; 若默认所有学生均能在课外自由使用大模型服务, 改革非但未能弥合既有差距, 反而可能加剧原有不平等。

保障机制的建构应以可及性为起点。机房与实训室的开放时段应充分覆盖课外练习需求, 校内应提供统一的工具访问入口, 防止学生因付费能力差异而获取不同质量的服务; 课程资源同时应保留低带宽可用的形式, 确保网络条件较差的学生不被排除在外。语言层面的可及性同样具有关键意义: 平台交互功能不应仅停留在术语翻译层面, 非母语学习者在表述习惯上的差异亦应被工具系统有效识别。

4.4. 数据安全、隐私保护与体系化治理

数据与伦理层面的治理须进行系统性建构。校本平台汇集学情画像、代码提交记录与行为日志, 数据采集范围、存储期限、访问权限与脱敏规则均须事先明确约定; 在与企业共建运维过程中, 应明确数据使用范围限于教学目的。工具输出的可靠性问题亦应纳入教学规范, 并由人工复核对事实性偏差、偏见与漏洞予以拦截。

技术滥用所产生的社会成本已有先例, 提示风险治理不能停留于课堂内部的技术约束层面。根据南都大数据研究院的统计, 2024 年搜索热度较高的 50 个国内 AI 风险舆情案例中, 超过五分之一与 AI 造谣相关。上述案例表明, 生成式人工智能的风险兼具现实性与外溢性, 仅凭工具禁用难以有效应对。在教学形态层面, 从混合教学转向人智协同教学是更为根本的转型: 教师角色由知识传递者转变为学习过程的组织者, 风险治理亦唯有在这一新形态中方可切实落实[6]。

5. 预期成效与评估设计

5.1. 教学改革预期成效

教学改革的预期成效可从不同层面加以界定。最为直接的成效体现于学习效果：生成式工具介入后，学生在语法与基础算法环节所遇障碍减少，编程任务完成质量应获得显著提升。参与度的变化亦随之显现，课堂教学研究表明，生成式人工智能借助互动形式能够增强课堂沉浸感，使抽象概念趋于具象化，且更契合偏好实践与合作学习的学生群体。另一项常被忽视的成效涉及就业竞争力：案例库所对接的贸易、智慧农业等场景与区域岗位需求具有同构性，学生积累的项目经验因而具备可迁移性。

5.2. 五项评估指标体系的构建

改革成效的迭代优化有赖于可观测性；若成效无法观测，改革将难以实现有效迭代。本研究设置了由编程项目完成质量、代码错误修复能力、学习参与度、创新思维水平与课程反馈满意度构成的五项一级指标，各指标的观测点、数据来源与采集方式见表 1。指标设计既借鉴了学习、能力与评价三个向度的分析框架[7]，亦吸纳了工程教育能力测评与认证体系的思路[5]。

Table 1. Evaluation index system for the effectiveness of teaching reform in python courses
表 1. Python 课程教学改革成效的评估指标体系

一级指标	主要观测点	数据来源	采集方式
编程项目完成质量	功能完整性、代码规范性、可扩展性、文档完备程度	项目提交物、课程设计成果	依据评分量表评定，双人交叉复核
代码错误修复能力	缺陷定位准确率、修复用时、修复方案质量、AI 输出改写质量	逆向纠错任务、限时调试任务	平台自动记录与教师评分结合
学习参与度	课前测试完成率、课堂互动次数、课后练习提交及时率、平台活跃时长	校本平台过程性数据	平台自动采集，按周汇总
创新思维水平	方案多样性、跨学科迁移表现、开放性问题的解法质量	开放性项目、跨学科选题作品	教师与企业导师联合评议
课程反馈满意度	教学方式认可度、工具使用体验、能力自评变化、文化适配感受	结构化问卷、访谈记录	学期中与学期末两次施测

五项指标在结构上呈现层次性关联：前两项指向硬技能，可依托客观量表与平台数据加以测度；第三项反映学习投入，主要以过程性数据为依据；后两项涉及主观判断，须借助多主体评议与自评互证予以综合判定。

5.3. 评估方法与实施安排

评估实施拟采用准实验设计。拟选取同一年级两个平行班级分别设定为实验组与对照组，由同一教师承担教学任务以控制教师变量；实验组依据第四章所述路径组织教学，对照组维持原有教学安排。通过学期初前测与学期末后测测度编程能力变化，主要因变量包括期末项目成绩、平台过程性数据与问卷得分。数据分析拟采用独立样本 t 检验对组间差异进行比较，并计算效应量。数据采集按三条路径并行开展：编程任务表现以平台过程性数据为支撑；结构化问卷于学期中与学期末各施测一次，以捕捉态度与能力自评的变化。以“准实验设计-教学实施-数据采集三路径-五项指标-结果回流/改”五个环节刻画评估运行机制：异常指标触发问题链与案例库修订，学生反馈纳入推荐算法参数调整；以同教师任课作为控制变量，各组数据用于检验改革是否缩小组间差距。

在指标分析环节, 有两点需加以注意: 其一, 应控制教师因素, 实验组与对照组宜由同一教师承担教学; 其二, 应对各组学生数据分别进行统计, 据此判断改革是否切实缩小组间差距。评估结果最终回流至教学环节: 指标异常知识点触发问题链与案例库修订, 学生反馈进入平台推荐算法参数调整, 由此形成“教-学-评-改”闭环的一轮运行。

6. 结语

将生成式人工智能引入《Python 程序设计》课程, 所引发的变革并不仅限于教学工具层面的更替。课程目标、教学组织、考核方式与教师角色均需进行相应调整, 改革的核心难点正在于其整体性——若仅在教学环节中单点植入智能化工具而保持其余结构不变, 改革成效往往较为有限, 甚至可能诱发学生规避思维训练的反向结果。本文所构建的改革路径包含四个层面: 面向高阶能力培养的教学设计、针对差异化学情的个性化适配、对接产业需求的案例库建设, 以及师资与平台支撑条件; 同时配套风险治理机制与五项指标评估设计, 旨在回应改革如何安全落地、成效如何得以观测两个现实问题。相较于已有研究, 本文的主要贡献在于将语言适配与产业需求作为硬性约束条件纳入课程设计, 进而形成面向地方本科院校的在地化改革方案, 而非对通用智能化方案的简单移植。

对地方应用型本科院校而言, 该项改革还具有另一层现实意义。为学习基础相对薄弱的学生提供可视化学习资源, 将贴近地方生活的文化元素纳入 Python 编程教学内容, 开发文化数字化保护等特色模块, 有助于学生接续自身文化脉络; 模拟贸易数据分析、电商运营等真实场景的训练, 则将课程教学与区域经济发展的实际需求相衔接。技术条件的改善固然为缩小区域与群体间教育差距提供了可能性, 但此种可能性能否转化为现实, 取决于设备可及性、语言适配与教师能力等条件能否得到系统性回应。需要说明的是, 本文现阶段所呈现的是课程改革的设计框架与评估方案, 相关成效指标尚未经过完整教学周期的实证检验, 有待后续准实验数据的进一步验证。

基金项目

本文为广西高等教育本科教学改革工程项目“地方院校数据科学与大数据技术专业人才培养的政校企协同育人模式探索与实践”(项目编号: 2023JGA337)的阶段性研究成果。

参考文献

- [1] 张红卓, 周小宝, 许玉焕, 等. 生成式人工智能赋能计算机程序设计类课程教学创新[J]. 计算机教育, 2024(7): 44-48.
- [2] 谢如石. 基于生成式人工智能技术的 Python 编程课程教学探索[J]. 电脑知识与技术, 2025, 21(30): 166-168.
- [3] 李清丽, 邵一萌. 生成式人工智能在 C 语言程序设计教学中的应用探索[J]. 计算机教育, 2025(12): 103-108.
- [4] 查昶玮. 生成式人工智能赋能 Python 编程在医学数据分析中的应用研究[J]. 科技创新与应用, 2025, 15(6): 1-5.
- [5] 徐晓飞, 张策. 生成式人工智能赋能工程教育及学生能力的培养、测评与认证体系[J]. 高等工程教育研究, 2025(4): 1-9.
- [6] 易凯谕, 韩锡斌. 从混合教学到人智协同教学: 生成式人工智能技术变革下的教学新形态[J]. 中国远程教育, 2025, 45(4): 85-98.
- [7] 白雪梅, 郭日发. 生成式人工智能何以赋能学习、能力与评价? [J]. 现代教育技术, 2024, 34(1): 55-63.