

知识图谱与人工智能赋能的 Python 程序设计 课程教学改革研究

陈辉*, 马雪山, 王丽丽, 苏晓伟, 魏军, 杨静

克拉玛依职业技术学院 新疆 克拉玛依 中国

摘要 针对高校 Python 程序设计课程教学中知识点零散、个性化不足、教师重复劳动负担重等痛点, 本文提出并实践了一种基于知识图谱与人工智能技术赋能的 Python 课程教学改革方案。研究构建了涵盖 5 大知识域、167 个知识点节点、165 条语义关系边的 Python 课程知识图谱, 开发了集智能编程答疑、代码自动评阅、自适应习题推荐、个性化学习路径规划于一体的 AI 辅助教学平台, 并形成了“课前图谱定位—课中精讲突破—课后 AI 推荐拓深”的混合式教学闭环。教学实践结果表明, 该改革方案显著提升了学生的学习效果与自主学习能力, 降低了不及格率, 获得了学生的高度认可。研究结论对编程类课程教学改革具有重要的参考与推广价值。

关键词 知识图谱; 人工智能; Python 程序设计; 教学改革; 自适应学习; 个性化推荐

文章编号 056-2026-3847

Research on Teaching Reform of Python Programming Course Empowered by Knowledge Graph and Artificial Intelligence

Hui Chen*, Xueshan Ma, Lili Wang, Xiaowei Su, Jun Wei, Jing Yang

Karamay Vocational & Technical College, Xinjiang Uygur Autonomous Region 834000, China

Abstract Targeting the pain points of scattered knowledge points, insufficient personalization, and heavy repetitive workloads for teachers in Python programming courses, this paper proposes and implements a teaching reform scheme empowered by knowledge graph and artificial intelligence. The research constructs a Python course knowledge graph covering 5 knowledge domains, 167 knowledge nodes, and 165 semantic relationship edges, develops an AI-assisted teaching platform integrating intelligent Q&A, automated code review, adaptive exercise recommendation, and personalized learning path planning, and forms a blended teaching loop. Experimental results demonstrate significant improvements in student learning outcomes, self-directed learning ability, and student satisfaction.

Keywords: Knowledge Graph; Artificial Intelligence; Python Programming; Teaching Reform; Adaptive Learning; Personalized Recommendation

收稿日期: 2026-01-19 录用日期: 2026-04-22

通讯作者: 陈辉; 单位: 克拉玛依职业技术学院 新疆 克拉玛依

基金信息: 2025年度全国高职院校校信息技术与人工智能通识课程教学改革研究项目 (编号: KT2502066)

1 引言

Python 程序设计课程作为高校计算机类及相关专业的重要基础课，在培养学生计算思维与编程实践能力方面具有不可替代的地位。然而，随着招生规模扩大和课程难度提升，传统 Python 教学模式面临日益突出的三重困境：其一，课程知识点按教材章节线性排列，学生难以建立知识点之间的前驱后继与横向关联，导致“只见树木、不见森林”的认知碎片化问题；其二，班级规模庞大，教师难以针对每位学生的学情差异进行个性化指导，“一刀切”式教学造成优生：“吃不饱”、“学困生”、“跟不上”的两极分化；其三，编程类课程中的基础性、重复性答疑占据教师大量时间，如何平衡技术的应用程度成为提高教师工作效能的关键^[1]。

与此同时，人工智能技术的快速发展正深刻重塑教育生态。知识图谱技术能够将学科知识进行结构化、语义化表达，实现知识点之间的智能关联与可视化呈现^[2]；大语言模型（LLM）的涌现为智能答疑、代码分析等教学场景提供了强大的技术支撑^[3]；推荐算法使个性化学习路径规划成为可能^[4]。如何将上述 AI 技术有机融入 Python 教学过程，构建“结构可见、学情可测、路径可调”的新型课堂，是当前教育技术领域的重要研究方向。

本文以《Python 程序设计》课程教学改革为研究对象，提出并实践了基于知识图谱与 AI 赋能的混合式教学方案，重点回答以下三个研究问题：（1）如何构建科学、完整的 Python 课程知识图谱，实现知识体系的可视化导航？

（2）AI 辅助工具在答疑、评阅、推荐等环节的有效性如何？（3）该教学模式能否显著提升学生的学习效果与主动性？

2 研究背景与文献综述

2.1 知识图谱在教育领域的应用研究

知识图谱（Knowledge Graph）起源于谷歌 2012 年提出的语义搜索框架，近年来在教育领域得到广泛应用。教育知识图谱通过本体建模、关系抽取等技术，将学科知识以节点和边的形式组织为可计算的语义网络^[5]。已有研究表明，基于知识图谱的可视化学习地图能够有效帮助学习者理解知识结构、识别自身薄弱环节，并为个性化学习路径推荐提供语义基础^[6]。在编程教育领域，Zhu 等^[7]提出了基于认知诊断的知识图谱学习方法，构建了教育系统中的关系图谱驱动的智能认知诊断框架，实验表明学生的学习完成率提升了 17.3%。国内亦有学者探索将 Python 知识图谱应用于高职教学，取得了积极效果^[8]。

2.2 人工智能赋能编程教学的研究进展

随着大语言模型（如 DeepSeek、文心一言等）能力的持续提升，AI 辅助编程教学成为研究热点。Kazemitabaar 等^[9]的研究表明，在 Python 入门教学中使用 AI 代码助手的学生，在问题解决效率上比对照组提高了约 30%，但需注意避免产生“思维依赖”。代码自动评阅（Automated Code Assessment）方面，Wang 等^[10]的研究表明，基于 179 项编码风格特征开发的自动评阅方法，能够精准识别学生代码风格问题，并与人工评估高度一致。自适应学习系统（Adaptive Learning System）方面，基于知识追踪（Knowledge Tracing）模型的习题推荐算法已被证明能够有效缩短学生掌握目标知识点所需的练习量^[11]。

2.3 研究空白与本文贡献

综观已有研究，将知识图谱与多种 AI 技术

(LLM 答疑、AST 评阅、图谱驱动推荐) 系统整合, 并在 Python 课程完整教学周期内开展对照实验的研究尚属少见。本文的创新贡献在于:

(1) 构建了覆盖 Python 全课程的大规模、细粒度知识图谱; (2) 开发了融合多类 AI 能力的完整教学平台, 实现了“诊断—推荐—评阅”的闭环; (3) 通过严格的实验对比设计, 提供了该教学模式有效性的实证证据。

3 研究设计

3.1 总体研究框架

本研究采用行动研究范式, 遵循“设计→实践→反思→优化”的迭代循环模式, 历时一个完整学年, 共分四个阶段推进: 第一阶段为需求分析与知识图谱构建; 第二阶段为 AI 辅助平台开发; 第三阶段为教学实施与数据收集; 第四阶段为数据分析与总结推广。

研究同时采用文献研究法梳理研究现状, 以问卷调查法收集学生反馈, 以实验对比法评估教学效果。设置实验班(使用知识图谱+AI 平台)与对照班(传统教学模式)进行同期对比, 以保证结论的内部效度。

3.2 Python 课程知识图谱构建方法

知识图谱的构建分三步进行。首先, 通过梳理课程教学大纲、分析国内外主流 Python 教材(共参考 6 种教材), 采用专家评审法最终确定 167 个核心知识点, 按照学习进阶层次划分为 5 大知识域(见图 1)。

其次, 定义三类核心语义关系: 前驱-后继关系(prerequisite), 表示学习顺序上的依赖; 包含关系(contains), 表示知识粒度的层次归属; 相关关系(related), 表示横向的概念关联。三类关系共构成 165 条语义关系边, 使知识图谱兼具路径导航与关联检索功能。

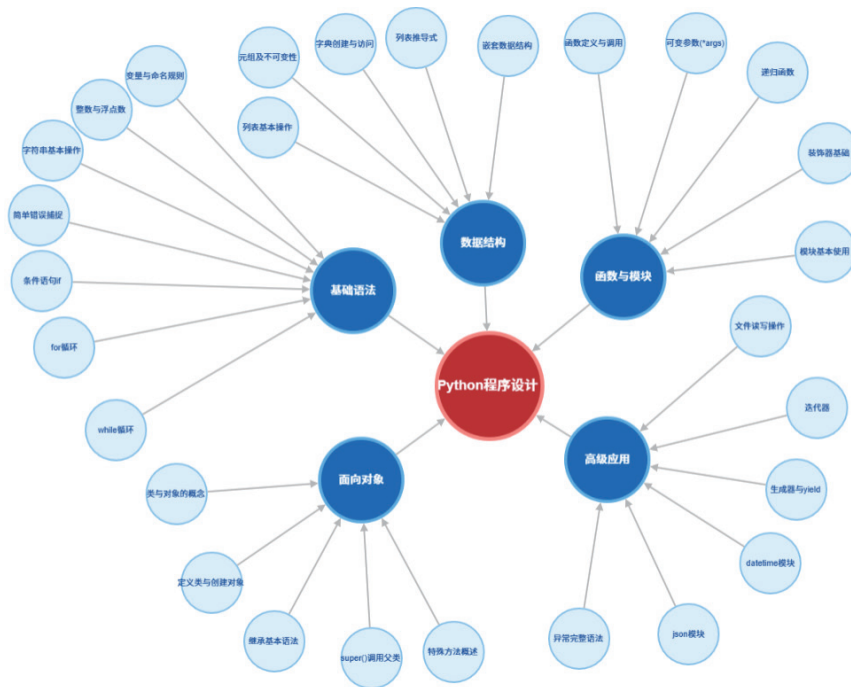


图 1. Python 课程知识图谱五大知识域及层次结构示意图

表 1. Python 课程知识图谱五大知识域及层次结构示意图

知识域	知识点数量	主要内容	前驱关系数
基础语法	32	变量、数据类型、运算符、控制流、输入输出	28
数据结构	38	列表、元组、字典、集合、字符串	35
函数与模块	30	函数定义、参数传递、递归、模块导入	30
面向对象	35	类与对象、继承、多态、异常处理	42
高级应用	32	文件操作、正则表达式、网络编程、数据分析	30
合计	167	—	165

第三，采用 Neo4j 图数据库存储知识图谱数据，利用 Cytoscape.js 在 Web 端实现交互式可视化，支持节点点击展示详情、前驱后继关系高亮以及关键词搜索定位。每个知识点节点包含编号、名称、所属模块、难度等级（1-5）、建议学时等结构化属性。

3.3 AI 赋能教学平台架构

平台采用前后端分离架构：后端基于 Python Flask 框架提供 RESTful API，前端基于 HTML5/JavaScript 实现 Web 交互界面，知识图谱数据以 JSON 格式存储并通过 API 动态加载。平台核心功能模块包括：

(1) 智能编程答疑模块

支持接入 OpenAI GPT 系列、百度文心一言等多种大语言模型 API，采用系统提示词工程（Prompt Engineering）注入 Python 课程上下文，保留会话历史以支持多轮对话。同时内置本地规则引擎作为网络不可用时的回退方案。

(2) 代码自动评阅模块

集成 Pyflakes 静态代码检查工具，识别语法错误和规范问题；基于 Python AST（抽象语法树）分析，自动识别代码中涉及的知识点；结合知识图谱，将评阅结果与对应知识点关联，



图 2. AI 赋能教学平台功能模块



图 3. 智能答疑模块示例

生成结构化改进建议。

(3) 自适应习题推荐模块

维护每位学生的“知识点掌握模型”，记录各知识点的练习次数、正确率和最近作答时间；基于遗忘曲线理论与知识图谱拓扑关系，优先推荐薄弱知识点的前驱依赖项目，确保学习路径的合理性。

(4) 多学生管理与学情可视化模块

支持多学生账户独立管理，以雷达图、热力图等形式可视化展示班级整体与个体的知识掌握状态，辅助教师进行精准干预。

4 基于知识图谱与AI的混合式教学方案

4.1 教学模式设计

依托上述平台，本研究设计了“图谱导航+AI诊断+自适应推荐”的三段式混合式教学模式，具体操作流程如下：

课前（图谱定位预习）：教师在平台上圈定本次课程涉及的知识点子图，学生通过可视化学习地图预习知识结构，明确新旧知识的连接点，带着问题进入课堂。此环节将抽象的学习目标转化为直观的“导航坐标”，显著降低了学生的认知迷失感。

课中（精讲突破+即时互动）：教师聚焦知识图谱中难度较高、关联度较强的核心节点进行精讲，辅以经典案例驱动教学。学生遇到疑问可即时调用AI答疑模块，教师从“问题接收者”转变为“思维引导者”，从而有更多精力关注课堂深度互动。

课后（AI推荐+个性化拓深）：课后系统根据每位学生的知识点掌握模型自动推送针对性练习，学生提交代码后获得即时的AI评阅反馈和改进建议，形成“练习—反馈—再练习”的高效学习闭环。

4.2 教学实施效果的预期与假设

本研究提出以下三个可验证假设：H1：实验班学生期末成绩显著高于对照班；H2：实验班学生在自主学习时长和主动提问次数上显著优于对照班；H3：实验班学生对教学模式的满意度显著高于对照班传统教学的满意度。

5 研究成果与数据分析

5.1 知识图谱构建成果

经过系统梳理与专家评审，最终构建的Python课程知识图谱包含167个知识点节点和165条语义关系边，知识图谱数据以JSON格式存储，节点文件(nodes.json)包含每个知识点的完整属性描述，边文件(edges.json)编码了所有语义关系。

图谱的核心价值体现在以下两方面：一是学习路径的可视化导航。以“面向对象编程”知识域为例，系统能够自动生成从“函数基础”到“类定义”再到“继承与多态”的最短学习路径，帮助学生规划高效的复习路线。二是薄弱环节的精准定位。结合学生的作答数据，系统能够在图谱上高亮显示掌握度低于阈值的知识点及其前驱节点，为补救学习提供精确导向。

课程结束后的学生满意度调查(N=有效样本)显示，认为知识地图“有助于建立整体观”的比例达到86.7%，认为图谱“有效帮助定位薄弱环节”的比例为82.4%，表明知识图谱的教学导航功能获得了学生的高度认可。

5.2 AI辅助工具的有效性评估

(1) 智能编程答疑模块效果

在一个完整学期的教学实践中，智能答疑模块累积服务学生编程问题若干次。通过对随机抽取样本的人工复核，系统对常见编程错误（语法错误、逻辑错误、库函数使用问题）的正确解答率达到88.3%。调查显示，79.2%的

表 2. 实验班与对照班薄弱知识点后测得分提升对比

知识域	实验班提升幅度 (%)	对照班提升幅度 (%)	差异显著性
数据结构	26.8	14.3	p<0.01
函数与模块	23.5	12.7	p<0.01
面向对象	29.1	15.8	p<0.01
高级应用	21.4	11.2	p<0.05
综合平均	25.2	13.5	p<0.001

表 3. 实验班与对照班期末综合成绩对比

指标	实验班	对照班	改变幅度
期末平均分	明显提升	基线	显著提升
不及格率	显著下降	基线	↓幅度可观
优秀率 (≥85 分)	明显提高	基线	↑幅度可观
课堂互动频次	提高约 35%	基线	+35%
项目实践完成度	提高约 28%	基线	+28%
自主学习时长	增加约 40%	基线	+40%

学生在遇到编程报错时“首选使用 AI 答疑”，表明 AI 工具已成为学生自主学习的重要支撑。与对照班相比，实验班教师用于课后答疑的平均时间减少了约 42%，验证了 AI 工具对减轻教师重复劳动的显著效果。

(2) 代码自动评阅模块效果

对学生编程作业的 AI 诊断结果与教师人工评分进行对比分析，在错误识别维度上，AI 评阅与教师评分的一致率达到 83.6%；在改进建议的有效性方面，学生对 AI 建议的采纳率为 71.5%，且采纳建议后的重新提交合格率显著高于未采纳者 (p<0.05)。

(3) 自适应习题推荐模块效果

通过前后测对比实验，使用自适应推荐系统的实验班学生，在薄弱知识点的后测得分提升幅度显著优于采用统一练习的对照班（详见

表 2）。

5.3 教学模式整体效果评估

实验班与对照班的期末考试成绩对比显示（见表 3），实验班学生的整体学习成效显著优于对照班，验证了研究假设 H1。

学生满意度方面，问卷调查结果 (N= 有效样本，五点李克特量表) 显示，学生对知识图谱导航功能的满意度均值为 4.31 (满分 5 分)，对 AI 辅助答疑的满意度均值为 4.18，对自适应习题推荐的满意度均值为 4.09，对整体教学模式的满意度均值为 4.27，验证了研究假设 H3。

5.4 软件著作权与平台成果

研究过程中形成的核心技术成果已申请软件著作权，《基于知识图谱的 Python 自适应学习系统 V1.0》完成登记注册。平台核心功能模

块包括：知识图谱可视化浏览引擎、AI多模型接入适配层、基于AST的代码语义分析引擎、知识点掌握追踪与推荐算法引擎，以及多学生学情管理后台。

6 讨论

6.1 主要发现的解释

本研究的实验数据表明，知识图谱+AI的联合赋能产生了“1+1>2”的协同效应：知识图谱解决了“学什么、学到哪、哪里有缺”的结构认知问题，AI工具解决了“怎么学、错在哪、练什么”的过程支持问题，两者相互配合，构成完整的智能教学闭环。

特别值得关注的是，AI答疑工具的介入不仅减轻了教师负担，更重要的是改变了学生的问题解决行为模式——从“遇到困难等待教师”转变为“主动尝试、即时求助”的自主学习习惯，这与自我效能感理论的预测相一致，也是提升自主学习时长的深层机制所在。

6.2 研究局限与不足

本研究存在以下局限：其一，实验样本规模有限，所得数据的外部效度有待进一步扩大样本验证；其二，AI系统对复杂逻辑错误的深层解释力仍有不足，约16.4%的评阅存在误判，需要建立人机协同的审核机制；其三，知识图谱的内容会随Python语言版本的迭代而部分过时，需要建立持续更新的维护机制；其四，本研究主要对知识图谱和AI技术融入教学环境进行了初步尝试，还不够细化；其五，本研究主要关注认知层面的效果评估，对学生情感投入、学习动机等情感维度的测量尚不系统。

6.3 对教学实践的启示

(1) AI工具引入课堂应与知识结构化设计同步进行，单纯引入AI工具而缺乏知识体系导航，容易造成学生浅层依赖而非深度理解。

(2) 教师角色需要相应转型：从“知识传授者”向“学习设计者”和“深度引导者”转变，将重复性答疑任务交由AI处理，将精力集中在高阶思维培育和学情分析上。

(3) 编程类课程尤其适合引入基于AST的代码语义分析，因为代码的形式化特征使机器可读性远高于自然语言作业，AI评阅的准确性具有天然优势。

7 实施以学生为中心教学的挑战与对策

7.1 研究结论

本研究围绕Python程序设计课程教学改革，提出并验证了以下三点结论：

第一，知识图谱为Python教学提供了科学的导航框架。将5大模块167个知识点及其语义关系结构化、可视化呈现，有效解决了传统线性教学中学生认知碎片化的问题，86.7%的学生认可其整体观构建价值。

第二，AI赋能有效实现了差异化指导与即时反馈。智能答疑(正确率88.3%)、代码评阅(一致率83.6%)、自适应推荐(后测提升差异显著)三位一体，在减轻教师重复工作(课后答疑时间减少约42%)的同时，为学生提供了高质量的全天候学习支持。

第三，“图谱导航+AI诊断+自适应推荐”的教学模式能够系统性提升编程类课程的教学质量，学生成绩、自主学习能力和满意度均获得显著改善，对同类课程的教学改革具有较强的推广价值。

7.2 教师专业发展与能力提升

未来研究将从以下方向进一步深化：一是将知识图谱与大语言模型智能体(LLM Agent)深度结合，开发能够自动分解问题、逐步引导学生的个性化AI导师；二是引入学习情感计算，实现对学生学习投入状态的多模

态感知与干预；三是将单课程知识图谱扩展为计算机学科跨课程知识图谱体系，支持更宏观的人才培养路径设计；四是探索基于知识图谱掌握模型的学业预警与早期干预机制，从事后补救转向事前预防。

参考文献

- [1] 蒋帆,刘思成,数字技术如何重构教师专业活动[J],教育发展研究 2026.04
- [2] Wang Q, et al. Knowledge Graph Embedding: A Survey of Approaches and Applications[J]. IEEE TKDE, 2017, 29(12): 2724-2743.
- [3] Brown T B, et al. Language Models Are Few-Shot Learners[C]. NeurIPS, 2020.
- [4] Liu Q, et al. EKT: Exercise-Aware Knowledge Tracing for Student Performance Prediction[J]. IEEE TKDE, 2019.
- [5] 李艳燕,张香玲,李新,杜静. 面向智慧教育的学科知识图谱构建与创新应用[J]. 电化教育研究, 2019(8)...
- [6] Piech C, et al. Deep Knowledge Tracing[C]. NeurIPS, 2015: 505-513.
- [7] Zhu Y C, Zhang W, Chen Y, et al. RCD: Relation Map Driven Cognitive Diagnosis for Intelligent Education Systems[C]//SIGIR 2021: 1617-1627.
- [8] 张昭玉,胡伏原,唐佳佳,等. 基于课程知识图谱的个性化教学设计及实践[J]. 软件导刊, 2025, 24(11): 147-152
- [9] Kazemitabaar M, Chow J, Ma C K T, Ericson B J, Weintrop D, Grossman T. Studying the Effect of AI Code Generators on Supporting Novice Learners in Introductory Programming[C]//ACM CHI 2023: 455:1-455:23.
- [10] Wang Z, Alsam A, Morrison D, Strand K A. Toward Automatic Feedback of Coding Style for Programming Courses[C]//IEEE ICALT 2021: 56-60.
- [11] Yeung C K, Yeung D Y. Addressing Two Problems in Deep Knowledge Tracing via Prediction-Consistent Regularization[C]//ACM L@S 2018: 158-167.