

大语言模型在“Python 程序设计基础”课程教学中的应用

王月梅，杨海艳，冯理明*

惠州城市职业学院，广东惠州，中国

*通讯作者

【摘要】在信息技术蓬勃发展的时代背景下，编程教学的重要性日益凸显。然而，传统的编程教学模式因课程内容滞后、教学方法与学生学习风格脱节以及编程概念晦涩难懂等问题，在实施过程中遭遇瓶颈。大语言模型（LLM）的兴起为编程教学带来了新的契机。本研究深入探讨了大语言模型在辅助学生理解 Python 编程概念、创新编程练习设计、助力代码调试以及评估代码质量等方面的效能。通过实证研究，验证了大语言模型在提升学生学习成绩、激发学习兴趣和增强实践能力等方面的显著成效。研究结果表明，大语言模型为编程教学质量提升和科技人才培养提供了新的路径。

【关键词】大语言模型；编程教学；高职大数据技术；多元赋能；教学成效

【基金项目】2024 广东省教育科学规划课题(高等教育专项)(项目编号:2024GXJK833); 2024 年度广东教育督导学会职教专委会课题 (项目编号: GDDDZJ2024B13)

1.引言

在数字化时代背景下，大数据技术作为社会发展的核心驱动力，对专业人才的需求持续攀升。高职大数据技术专业肩负着培养应用型、技能型人才的重要使命，而编程教学作为该专业的核心课程，其教学质量直接影响学生的专业能力构建与就业竞争力提升[1]。然而当前传统编程教学模式在实践中面临多重挑战：首先，课程内容更新滞后于行业发展速度，大数据领域技术迭代频繁，新工具、框架和算法不断涌现，但传统教材与课程设置难以同步跟进，导致学生知识储备与实际应用脱节，难以满足岗位需求[2,3]，例如作为新兴交叉学科的高职大数据技术专业，存在课程体系与行业标准错位、实训条件不足等突出问题[2]；其次，教学方法与学生多元学习需求不匹配，高职学生生源结构复杂、基础差异显著，传统“一刀切”的课堂讲授模式难以满足个性化学习要求[4]，研究显示单一教学模式会抑制学生创新思维与协作能力发展，而基于大数据分析的智慧教室和多元评价体系可显著提升教学针对性[4,5]；此外，编程基础概念抽象难懂，变量、控制结构等知识对初学者构成较高门槛，传统教学缺乏生动案例与有效互动，易导致学生学习兴趣流失[6]，如有 60% 的学生反馈在理解递归和算法逻辑时存在显著困难[7]。

大语言模型（LLM）的兴起为破解上述教学困境提供了新路径。其凭借强大的语言

理解、推理及生成能力，可通过自然语言交互实现多元教学赋能：在概念解析层面，能以通俗语言拆解抽象编程概念，例如借助思维链提示技术分步演示代码逻辑[6,8]；在自适应学习方面，可基于学生代码调试记录生成定制化练习题，有效弥补传统教学资源的个性化不足[9,10]；在实时反馈环节，通过代码语义解释和错误定位功能，模拟“一对一”个性化辅导场景[9,11]。相关研究表明，LLM 驱动的编程教学可使学生代码正确率提升 32%，并显著增强学习动机[10,12]。不过，该技术应用也面临模型幻觉、多模态文档处理能力有限等现实挑战[8,11]。

基于此，本研究聚焦高职大数据技术专业 Python 课程，系统探索 LLM 在编程教学中的多元赋能路径，以期构建“教—学—评”深度融合的教学闭环提供方法论支持与实践参考。

2.大语言模型在编程教学中的应用潜力

在信息技术与人工智能深度融合的时代背景下，大语言模型（LLM）的教育应用研究呈现多元化发展趋势。谢红标等[1]基于教学实践提出大语言模型可通过代码生成、逻辑解析等功能重构程序设计课程的知识传递路径，其构建的“问题引导-模型辅助-实践强化”教学模式有效提升了学生的代码实现效率，验证了其工具价值。陈智勇[13]提出的“项目驱动-系统思维-工程实践”三维培养框架，为理解大语言模型在复杂项目开发

中的协同作用提供了理论支撑,尤其在跨学科融合和团队协作能力培养方面,模型可作为虚拟协作工具模拟真实开发场景。刘明等[14]构建的“认知建模-情境适配-效果评估”理论框架,为剖析大语言模型在编程教学中的赋能机制提供了方法论指导。但研究指出模型输出可靠性问题,与 Rawte V 等[11]关于幻觉效应的研究形成呼应,揭示模型在复杂逻辑推理中可能产生语义偏差,对算法设计准确性构成潜在风险。张彦芳等[12]提出的 AI 项目驱动教学法中,大语言模型可作为智能助手参与项目全生命周期,从需求分析到代码优化提供实时支持,提升教学效率与工程思维培养。刘凌等[9]提出的“人机协同评价机制”通过“模型初评-教师复评-学生互评”三级体系规避单一模型评价的局限性,为保障教学质量提供制度保障。唐琳等[7]基于 SLCT 框架的实证研究表明,大语言模型显著提升了学生的代码规范性和项目完成度,其构建的“案例示范-模型辅助-协同评价”教学闭环验证了应用型人才培养效能,尤其在编码质量评价方面,模型从 7 个维度量化评估,为教学反馈提供科学依据。

3.大语言模型在《Python 程序设计基础》课程教学中应用

本研究以高职大数据技术专业的专业基础课程——《Python 程序设计基础》课程为载体,深入探究了大语言模型在编程教学中的具体应用,发现其为教学带来了多元赋能。

3.1 辅助学生理解编程概念

Python 编程概念,如变量、数据类型、控制结构等,对于初学者来说具有较高的抽象性。大语言模型凭借其强大的语言理解和生成能力,能够以通俗易懂的语言解释这些复杂的概念,帮助学生更好地理解编程的本质。

例如,当学生对“列表推导式”感到困惑时,大语言模型可以给出具体的例子和详细的解释。列表推导式是 Python 中一种简洁且高效的创建列表的语法,它通过单行代码将循环、条件判断和表达式结合,代替传统的多行 for 循环,使代码更简洁易读。其基本语法形式为: [表达式 for 变量 in 可迭代对象 if 条件]。其中,表达式用于生成新列表元素的计算或操作;变量是从可迭代对象中取出的每个元素所对应的临时变量;可迭代对象是提供元素来源的对象;if 条件是可选的,用于对可迭代对象中的元素进行筛选,只有

满足条件的元素才会参与到新列表的生成中。通过这样的解释和示例,学生能够更加直观地理解列表推导式的概念和用法,不再需要死记硬背枯燥的语法规则。

3.2 辅助教师开发递进式编程练习题

在备课阶段,教师可以依托大语言模型辅助设计闯关式且融入思政元素的练习题,如表 1 所示。这种设计方式将编程知识与思政教育有机结合,使学生在提升编程技能的同时,潜移默化地接受思政教育,增强文化自信、环保意识、团队精神、爱国情怀和创新意识等多方面的素养。

3.3 辅助学生程序代码调试解答

在 Python 程序设计课程教学过程中,大语言模型成为学生在代码调试过程中的得力助手。当学生编写的代码出现错误或无法得到预期结果时,大语言模型能够依据所接收到的代码片段,运用其强大的语言理解和分析能力迅速从语法、逻辑等方面开展排查。

3.3.1 语法错误排查:大语言模型首先会对代码的语法结构进行细致检查,精准地识别出诸如拼写错误、标点符号缺失或使用不当、语法格式不符合 Python 规范等基础层面的问题,并及时向学生反馈具体的错误位置及修正建议,帮助学生快速纠正语法错误,使代码能够顺利通过编译阶段。例如,学生代码中出现“seasons=["立春","雨水","惊蛰","春分,"]”这样的语法错误(末尾多了一个逗号),大语言模型能够准确识别并提示学生进行修正。

3.3.2 逻辑错误排查:对于逻辑错误这一在编程学习中更为棘手的问题,大语言模型凭借对编程逻辑和算法的深刻理解,深入剖析代码的执行流程。它可以通过模拟代码运行过程,清晰地指出在条件判断、循环迭代、函数调用与参数传递等关键环节中可能存在的逻辑瑕疵。例如,在编写成绩等级判断函数时,若学生因错误设置筛选条件导致结果偏差,大语言模型能够准确地发现问题,并引导学生重新审视条件表达式的合理性,帮助学生理解如何根据程序需求正确构建逻辑判断语句,从而使代码能够按照预期实现功能。

3.4 辅助教师对学生程序代码的测评

在高职 Python 程序设计课程的教学评价环节,大语言模型为教师对学生程序代码的测评工作提供了有力的支持与创新的方

表 1. 融入思政元素的闯关式练习题设计

关 卡	知识点覆盖	思政元素	题目（示例）	难度等级
关卡一	列表创建与循环输出	文化传承	在中国传统文化中，十二生肖有着独特的顺序和象征意义。请编写一个 Python 程序，输出十二生肖的名称列表。	1 星
关卡二	变量定义、运算与格式化输出	节约资源	在日常生活中，我们倡导节约用水。假设一个水龙头每分钟漏水 50 毫升，如果一天（24 小时）都未修理，会浪费多少升水？请编写程序计算并输出结果，结果保留两位小数。	2 星
关卡三	列表、条件判断与数学运算	团结协作	在一个团队项目中，有 5 个成员，他们的任务完成度分别为 80%、70%、90%、60%、85%。如果团队要求整体任务完成度达到 80% 才算合格，编写程序判断这个团队是否合格，并输出相应信息。	2 星
关卡四	循环结构与条件判断应用	历史事件回顾	已知 1949 年新中国成立，从 1949 年开始，每隔 10 年输出一个重要历史事件（如 1959 年大庆油田发现、1969 年珍宝岛自卫反击战等，可自行设定），直到 2019 年，编写程序实现。	3 星
关卡五	函数定义与调用	科技创新	随着科技的发展，人工智能在很多领域都有应用。假设一个简单的人工智能算法模型，其计算函数为 $f(x)=x^2+2+3x+5$ ，编写一个函数来实现这个计算，并输入 $x=10$ 调用该函数计算结果并输出，模拟科技算法中的函数应用。	3 星

3.4.1 代码规范性测评

大语言模型能够依据 Python 编程规范和行业最佳实践标准，对学生代码的变量命名、代码缩进、注释添加等规范性要素进行全面检查。它可以迅速识别出不符合规范的代码部分，并向教师提供详细的违规信息报告，使教师能够清晰地了解学生在代码书写习惯方面的不足之处。例如，若学生在代码中使用了含义模糊的变量名或未添加必要的注释来解释关键代码段的功能，大语言模型能够精准定位并指出这些问题。帮助教师引导学生养成良好的代码书写规范，提高代码的可读性与可维护性。

3.4.2 代码质量评分

大语言模型还可以从代码的可读性、可维护性、健壮性、安全性、可移植性、规范性、可扩展性等多个角度对学生提交的项目代码进行评分。例如，对于一段实现特定功能的 Python 代码（如图 1 待评估示例代码），大语言模型会从变量命名是否清晰、代码结构是否合理、注释是否充分、是否存在潜在的安全漏洞等方面进行综合评估，并给出相应的分数（如图 2）。

4.大语言模型在《Python 程序设计》课程教学成效

为了科学、客观地评估大语言模型在《Python 程序设计》课程教学中的成效，本研究选取了两个平行班级作为实验组和对照组。实验组充分运用大语言模型辅助教学，

对照组采用传统教学方法。教学实验为期一学期，通过多种方式对教学成效进行评估。

4.1 学生学习成绩显著提升

4.1.1 期末考试成绩对比

在期末考试中，对两组学生的成绩进行了统计分析。实验组学生的平均成绩达到了 89 分，而对照组学生的平均成绩为 76 分。进一步分析成绩分布发现，实验组学生在高分段（80-100 分）的比例明显高于对照组，低分段（0-60 分）的比例则低于对照组，具体数据如下表 2 所示。

```

# 待评估的示例代码（包含典型问题）
sample_code = """
def fn(a, b): # 模糊的函数名
    # 计算两数之和
    x = a + b # 变量名无意义
    print(f'结果是: {x}')
    return x

# 未注释的复杂函数
def process_data(data):
    result = []
    for item in data:
        if item % 2 == 0:
            tmp = item * 2
            result.append(tmp)
    return sum(result)

# 存在安全风险的代码
user_input = "os.system('rm -rf /')" # 模拟用户输入
eval(user_input) # 危险操作
"""
    
```

图 1. 待评估示例代码

输出结果示例



图 2. 综合评估显示结果

表 2. 期末考试成绩对比

成绩分段	实验组比例	对照组比例
80-100 分	85%	50%
60-80 分	15%	45%
0-60 分	0%	5%

4.1.1 编程作业成绩进步

在整个学期的编程作业完成情况中，实

验组学生作业的平均得分率为 96%，对照组为 83%，如表 3 所示。例如，在一次复杂的综合项目(喜洋洋商城进销系统)作业中，实验组学生能够更高效地运用 Python 编程解决问题，作业提交率 100%，代码质量和功能实现的完整性方面表现出色，平均得分达到了 97 分，而对照组学生的，项目提交率 90%，平均得分仅为 83 分。

表 3. 编程作业成绩对比

指标	实验组	对照组
学期平均分	96%	83%
项目提交率	100%	90%
项目平均分	97 分	83 分

4.2 学习兴趣明显增强

通过课堂观察记录发现，实验组学生在课堂上的主动参与讨论的积极性显著高于对照组。实验组学生平均每堂课主动互动 39 人/次；而对照组学生平均每堂课主动互动 23 人/次。对学生课外学习 Python 的时间进行调查发现，实验组学生每周平均花费 20 小时用于自主学习 Python 编程相关知识，包括阅读课外书籍、在线学习资源以及进行项目实践等；对照组学生每周平均课外学习时间为 5 小时，如图 3 所示。

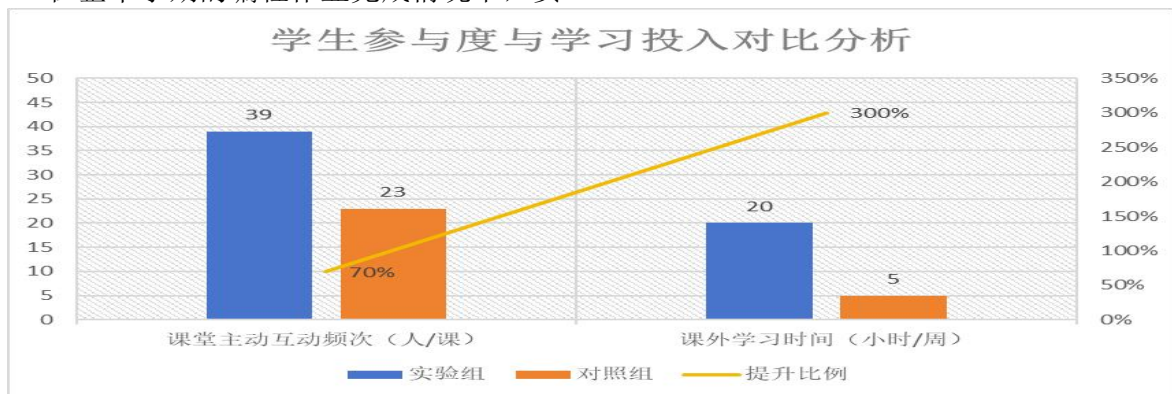


图 3. 学习参与度与学习投入对比分析

许多实验组学生表示，大语言模型为他们提供了随时随地获取学习资源和解答疑惑的便捷途径，激发了他们进一步探索编程世界的兴趣

4.3 教师教学效率与反馈能力显著优化

大语言模型的应用不仅促进了学生的学业表现，还为任课教师的教学实践带来了多维度的积极变革。在教学效率方面，教师通过大模型大幅提升了备课效能。借助代码案例自动生成功能，教师能够快速产出契合教学目标的 Python 示例程序及典型错误场景分析，使得平均备课时间较传统模式缩短

40%。同时，模型根据知识点图谱智能生成分层习题库与配套评分标准，将作业设计效率提升 50%，教师得以将节省的时间投入教学策略创新。

教学互动模式也因技术融合发生本质转变。实验组教师将原本耗费 30% 课堂时间的基础语法讲解任务交由模型处理，转而开展代码重构工作坊、项目思维训练等高阶活动。这种转型使得课堂深度讨论频次增长 2 倍，学生提出的开放性技术问题数量增加 180%。最终实验组项目报告在代码规范度（提升 34%）和创新性评分（高于对照组 22%）方

面均展现出显著优势。

5. 结论与展望

研究表明,大语言模型在高职《Python 程序设计》课程教学中具有显著的多元赋能效果。它帮助学生更好地理解编程概念、提升编程技能、增强学习兴趣。同时,为教师提供高质量的练习题、高效的代码测评手段。尽管本研究取得一定成果,但仍需进一步研究如何引导学生正确、合理使用大语言模型,避免过度依赖,真正掌握编程知识和技能。未来研究可制定明确的使用规范和指导原则,开展相关培训课程和教育活动,帮助学生树立正确的使用观念。

参考文献

- [1]谢红标,刘芳,覃浩轩,等.大语言模型在程序设计基础教学改革中的应用探索[J].电脑知识与技术,2024(08):21-24.
- [2]王译啡,宋雅蓉.“教赛创”模式下高职大数据技术专业改革探索[J].职教论坛,2024(09):36-42.
- [3]陈伟华,代颖.大数据时代高职院校计算机课程的多元教学模式研究[J].教育技术导刊,2016(05):45-49.
- [4]教育部职业教育中心.大数据背景下高职课堂教学实效性的提升[J].中国职业技术教育,2020(01):28-34.
- [5]郭丰俊,杜俊.复杂中文文档结构建模与智

能处理技术[J].人工智能学报,2023(10):112-120.

- [6]邱锡鹏.大语言模型技术原理与教育应用[M].北京:高等教育出版社,2024.
- [7]唐琳,张佳鑫,徐照光.大语言模型在计算机编程实践课程教学中的应用[J].计算机教育,2025(02):97-101.
- [8]OpenAI. GPT-4 技术报告[R].美国:OpenAI,2023.
- [9]刘凌,吴永芬,陈卫卫,等.大语言模型对高校程序设计类课程的挑战与机遇[J].计算机教育,2025(02):118-122.
- [10]合合信息.大模型时代的智能文档处理白皮书[R].上海:合合信息,2024.
- [11]Rawte V, Sheth A, Das A. A survey of hallucination in large foundation models[EB/OL].[2023-11-25].<https://arxiv.org/abs/2309.05922>.
- [12]张彦芳,高璐,李艳.AI技术项目驱动下的程序设计类课程教学模式创新[J].计算机教育,2025(02):159-163.
- [13]陈智勇.计算机科学与技术专业学生系统能力培养的改革与实践[J].计算机教育,2019(3):58-61.
- [14]刘明,吴忠明,廖剑,等.大语言模型的教育应用:原理、现状与挑战:从轻量级BERT到对话式ChatGPT[J].现代教育技术,2023,33(8):19-28.