

LLMs 赋能 Python 认知冲突的教学设计与实践探索*

吴怀广 王晓 金松河 何亚琼 王榕

郑州轻工业大学计算机科学与技术学院, 郑州 450001

摘 要 当前编程教育中, 学生对抽象编程概念的理解常流于表面, 难以实现高阶认知与深度迁移。针对这一问题, 本文构建了基于大语言模型 (LLMs) 的认知冲突教学框架, 提出“动态触发—精准调节—深度转化”三阶段模型, 系统融合冲突类型设计、认知负荷调控与概念转化工具, 旨在提升学生对 Python 核心概念的理解质量。在教学实施中, 借助 LLM 的自然语言生成与语义分析能力, 动态构建个性化的认知冲突情境, 引发学习者原有知识结构中的不一致; 通过智能对话与多模态反馈机制, 精准识别学习过程中的认知负荷状态并进行调节; 最终结合可视化、类比推理等策略, 引导学生完成编程概念的深度转化与内化。实证研究表明, 与传统教学方法相比, 该框架在提升学生对概念的理解深度与迁移能力方面效果显著, 班级平均成绩提升 18.4%, 跨学科问题解决正确率提升 16.4%, 同时编程代码能力和学习积极性显著提高。研究成果为高校教师在智能支持环境下开展认知导向教学、实现深度学习目标提供了新路径和方法支持, 具有较强的推广价值与实践意义。

关键字 认知冲突, 大语言模型, Python 教学, 概念转变, 深度学习

Teaching Design and Practical Exploration of Cognitive Conflict-Based Python Instruction Empowered by Large Language Models (LLMs)

WU Huaiguang, WANG Xiao, JIN Songhe, HE Yaqiong, WANG Rong

School of Computer Science and Technology
Zhengzhou University of Light Industry
Zhengzhou 450000, China

hgwu@zzuli.edu.cn wangxiao@zzuli.edu.cn 9889170@qq.com
heyaqiong@zzuli.edu.cn wangrong@zzuli.edu.cn

Abstract—In programming education, students often struggle to grasp abstract concepts beyond a superficial level, hindering higher-order thinking and knowledge transfer. To address this, we propose a cognitive conflict teaching framework powered by large language models (LLMs), featuring a three-phase model: "Dynamic Activation - Precise Regulation - Deep Transformation." This framework integrates conflict type design, cognitive load regulation, and conceptual transformation tools to enhance understanding of core Python concepts. Using LLMs' natural language generation and semantic analysis capabilities, the framework dynamically creates personalized conflict scenarios that expose inconsistencies in learners' prior knowledge. Through intelligent dialogue and multimodal feedback, it monitors and regulates cognitive load. Finally, strategies like visualization and analogical reasoning support deep conceptual transformation and internalization. Empirical results show that compared with traditional methods, the framework significantly improves students' conceptual depth and transfer ability—average scores increased by 18.4%, and interdisciplinary problem-solving accuracy by 16.4%. Programming competence and learning engagement also improved notably. This study offers a practical and scalable approach for implementing cognitively-oriented instruction in intelligent environments, supporting deeper learning in programming education.

Keywords—Cognitive Conflict, Large Language Models (LLMs), Python Instruction, Conceptual Change, Deep Learning in Education

1 引言

当前编程教育面临“高投入低转化”的全球性挑

战。学习者普遍陷入“语法记忆陷阱”, 表现为能准确复述循环结构语法规则, 却无法在解决实际问题时自主选择迭代策略, 能背诵面向对象三大特性, 但在设计类结构时仍机械套用模板^[1]。这种浅层认知的根源在于教学过程中过度强调代码规范而忽视计算思维的本质性建构。传统认知冲突策略在此情境下面临双

*基金资助: 本文得到郑州轻工业大学研究生全英文课程(群)建设项目(13109000016)、河南省研究生教育改革与质量提升工程项目(YJS2025AL37)、全国高等院校计算机基础教育研究会项目(2024-AFCEC-170)的资助。

重掣肘:(1)人工设计的冲突案例具有静态固化特征,难以实时适配不同学习者的认知基线。2)普适性教学设计无法精准定位个体认知盲区。这种矛盾在编程教育领域尤为突出,因为程序调试过程天然具备认知冲突属性,但人工干预往往滞后于动态演变的思维过程。

当前,大语言模型(LLMs)^[2]在Python教学中的应用主要集中在代码生成、案例优化和创新能力培养等方面。研究表明,这类工具能自动生成代码框架、提供多模态演示(如算法可视化),并支持逆向学习模式,即从源码反推问题设计,促进学生深度思考^[3,4]。然而,认知冲突未被有效利用,大语言模型通常直接提供答案,减少了学生通过试错和调试代码来构建知识的机会,弱化了编程学习中的认知冲突(如错误排查、逻辑修正),而这恰恰是计算思维培养的关键环节,同时,生成内容的滞后性和潜在错误可能误导学生,且过度依赖模型会抑制自主探究能力^[5]。

大语言模型(LLMs)凭借其代码语义理解与动态生成能力,为认知冲突的智能化实施开辟新路径^[6]。本研究从教育目标重构、教学过程再造、师生角色转型三个维度,构建LLMs赋能的Python教学框架,重点阐明其在教学实践中的操作路径与实施要点,推动编程教育从知识传递向思维进化的范式转型。

2 理论基础与设计原则

认知冲突策略作为促进深度学习的关键教学手段,其有效性建立在概念转变理论的四维条件基础之上。第一维度强调认知失衡的产生机制,要求通过教学设计暴露学习者原有认知图式与客观现象的矛盾冲突。

第二维度关注冲突强度的调控边界,需将认知冲突的复杂度控制在超出当前认知水平但可通过支架支持解决的区间范围。第三维度聚焦新概念的建构标准,要求替代性概念系统不仅需要逻辑自洽性,更需在解释力、预测力等方面展现显著优越性。第四维度强调知识迁移的通道建设,需设计概念应用的多样化情境。

LLMs的技术特性为认知冲突策略的智能化实施提供创新解决方案。首先,LLMs的自然语言交互能力支持冲突情境的动态建构。通过对话状态跟踪技术,系统可实时捕捉学习者的认知盲点,如当学生反复误解化学平衡移动原理时,自动生成生活化类比情境(如电梯载客量的动态平衡),这种具象化表达显著降低抽象概念的认知门槛。其次,模型内嵌的代码分析模块能够实现认知偏差的精准定位。在编程教育场景中,系统不仅识别语法错误,更能通过抽象语法树分析追溯错误背后的逻辑误区,如将循环控制变量误解为全局变量等深层次认知偏差。更为重要的是,生成式AI的即时反馈特性为冲突调节提供动态支持,系统可根据实时交互数据调整认知负荷:当检测到学生出现连续错误时,自动插入阶段性小结与可视化支架;当应答准确率提升时,则逐步增加开放性问题的比重。这种“冲突-支持”的动态平衡机制,完美复现了专家教师“搭脚手架”的教学智慧。

3 框架设计:三层次系统模型

框架采用层次化设计理念,构建包含冲突触发层、动态调节层和概念转化层的三维结构,框架结构图如图1所示。

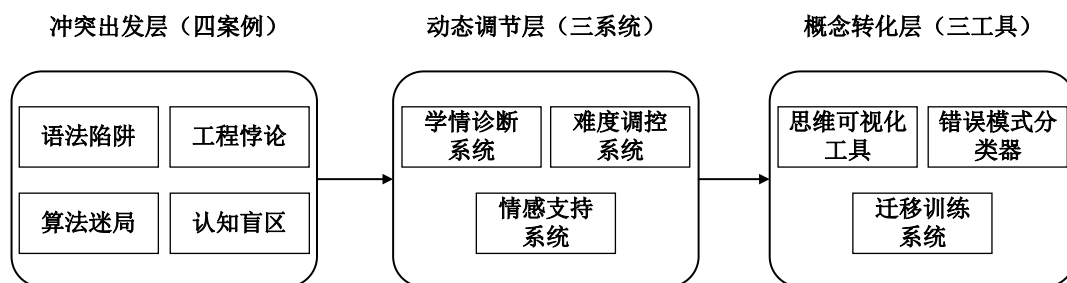


图1 三层次系统模型框架图

冲突触发层聚焦四类核心问题设计:语法陷阱类案例通过表面合规但隐含语义错误的代码,揭示形式语法与实际运行的认知鸿沟;工程悖论类案例利用符合编码规范但违背设计原则的程序,引发软件工程思维的深度反思;算法迷局类案例通过多算法路径的对比分析,培养计算决策的系统思维;认知盲区类案例则通过特殊场景模拟,将程序员的隐性经验转化为显性知识。

动态调节层建立智能适配机制,通过学情诊断、难度调控、情感支持三个子系统实现冲突强度的精准把控。学情诊断系统解析学习者历史数据,构建包含概念掌握度、错误模式、思维特征的知识画像;难度调控系统根据实时诊断结果,动态选择基础型、进阶型、挑战型三类案例;情感支持系统则通过语义分析识别学习焦虑,适时调整干预策略,如插入认知脚手架或切换冲突类型。

概念转化层开发系列认知强化工具，促进冲突经验的深度内化。思维可视化工具采用知识图谱技术，将调试过程转化为可追溯的认知路径图；错误模式分类器建立包含语法误用、逻辑缺陷、设计失当等维度的标签体系，支持精准归因分析；迁移训练系统设计跨学科应用场景，例如将循环结构教学与基因序列分析任务结合，促进编程思维的泛化迁移。

4 实践路径

教学框架的落地实施遵循五阶段螺旋式推进路径（见图2）。

预判阶段，教师利用 LLMs 分析学习者作业代码与提问记录，识别如可变对象误解、作用域混淆等典型认知偏差，建立个性化教学基线。

触发阶段，系统推送定制化冲突案例，例如展示函数修改全局变量引发的意外结果，通过预期与实际的矛盾引发认知失衡。

探究阶段，采用双师协同模式——LLMs 负责代码调试的技术引导，教师侧重思维过程的元认知提问，共同帮助学生发现矛盾本质。

抽象阶段，引导学生提炼编程原则，如通过迭代器陷阱案例总结“遍历不可变副本”的操作规范，将具体经验上升为设计范式。

迁移阶段，设计跨学科应用任务，例如用 Python 实现物理实验数据处理，检验概念理解的深度与灵活性。

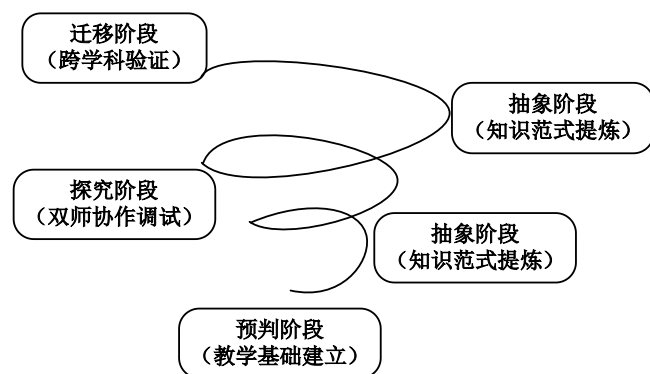


图2 五阶段螺旋式推进路径

5 实践探索

本节将通过典型教学案例验证教学框架的操作逻辑。例如，在 Python 语言的“列表别名现象”教学中，LLMs 生成两套表面相同但内存指向不同的列表操作代码，学生在调试中发现修改副本影响原变量的异常现象。教师引导学生观察 id() 函数的输出结果，并结合可视化工具展示内存地址变化，最终使学生能够达成对“可变对象传引用”的深层认知。此过程遵循“现象冲突→技术验证→原理阐释→模式提炼”的认知发

展路径，体现框架促进概念转变的内在机理。如下为可变对象传递的认知重构示例代码：

```

# 教师提供的初始代码
def modify_element(data):
    data[0] = 100
    original_list = [1, 2, 3]
    modify_element(original_list)
    print("修改后的列表:", original_list) # 输出
[100,2,3], 与学生预期可能冲突
# LLM 生成的冲突代码变体
def modify_reassign(data):
    data = data + [4] # 创建新对象
    original_list = [1, 2, 3]
    modify_reassign(original_list)
    print("重新赋值后的列表:", original_list) #
仍输出[1,2,3]
  
```

教学流程如下：

阶段1：认知冲突触发

（1）代码预测

学生分别预测两个函数的输出结果，多数可能认为：

第一个函数会修改原列表。

第二个函数也会修改原列表。

（2）执行结果与部分学生的预期产生冲突：

第一个函数成功修改原列表。

第二个函数未修改原列表。

阶段2：概念探索

（1）内存可视化

使用在线 Python Tutor 工具演示，代码如下所示，展示对象 ID 的变化过程。

```

def modify(data):
    print("函数内初始 id:", id(data))
    data[0] = 100
    data = data + [4]
    print("函数内最终 id:", id(data))

lst = [1,2,3]
print("原始 id:", id(lst))
modify(lst)
print("最终 id:", id(lst))
  
```

（2）关键提问链

为什么两种修改方式结果不同？

函数参数传递的本质是什么？

如何避免意外修改原始数据？

阶段3：规律抽象

提炼编程原则：

（1）引用传递原则：可变对象通过引用传递

（2）安全修改三要素：

- ①明确是否需要原位修改
- ②必要时创建副本（list.copy()或deepcopy）
- ③在文档字符串中注明副作用

阶段 4：迁移训练

LLMs 生成变式练习，代码如下所示，学生通过预测→验证→解释流程强化理解。

```
# 变体 1：字典修改
def update_dict(d):
    d['key'] = 'new'
    original = {}
    update_dict(original)
    print(original) # ?

# 变体 2：嵌套列表
def modify_nested(lst):
    lst[0][1] = 99
    data = [[1,2], [3,4]]
    modify_nested(data)
    print(data) # ?
```

6 质量保障机制

为确保持续教学效能，构建科学完备的人工智能教育三维质量保障体系，制定教育性、技术性、伦理性的三维评估标准，切实保障智能教育系统的教学效能与伦理安全。

在资源质量控制层面，建立跨学科联合评审机制，研制包含教育适配度、技术性和伦理性三大维度的评估指标体系。其中教育性标准着重考察生成内容是否符合学生认知发展阶段特征，技术性标准侧重检验算法输出的逻辑自治性与知识准确性，伦理性标准则重点防范文化偏见与价值观偏差。具体实施中采用专家联席评审模式，由教育学家负责内容适切性评估，算法工程师开展技术可行性验证，伦理学家进行社会影响审查，通过三方会签制度确保教学资源质量^[7]。

针对教师专业发展需求，构建分层递进的数字化能力培养体系。依托教师专业发展标准框架，设计包含智能教学法应用、学习诊断数据分析、课堂冲突智能调控三大模块的进阶式课程群。采用“理论学习-案例研讨-实践演练”的混合培养模式，通过教学设计方案评审、模拟课堂演练、教育大数据分析报告等多元化评价方式，构建教师智能教育素养发展的可视化成长路径。设置智能教学冲突场景处理模拟机制，通过虚拟仿真系统训练教师应对算法生成内容与现实教学情境的融合挑战。

在系统运行安全维度，实施动态智能监测与分级干预策略。基于教育神经科学关于认知负荷的研究成果，建立多参数融合的实时监测系统，通过语义分析捕捉情绪关键词、交互频率追踪注意力波动、答题模式识别认知负荷等多模态数据，智能评估学生学习状态。设置三级风险预警机制：设置单日认知冲突触发

阈值；当系统检测到连续三次错误响应时自动启动熔断机制，切换至基础知识强化模式，防止认知超载引发的学习挫败；累计达到日安全阈值后，启动强制干预程序并生成学情分析报告。所有干预措施均遵循“降温-诊断-恢复”的医学干预模型，在保障系统安全性的同时预留学习者自主恢复空间。

该质量保障体系通过前端的资源过滤机制、中端的能力提升机制、末端的风险控制机制形成立体防护网络，其特色在于将教育规律与智能技术深度融合，既充分发挥人工智能的教学优势，又有效规避技术滥用风险，为智能教育系统的可持续发展提供系统性解决方案。

7 教学成效分析

LLMs赋能Python认知冲突的教学设计在郑州轻工业大学开设的Python 程序设计课程中进行了五个学期的实践验证。每个学期将学生分成两组，每组60人。实践结果表明：采用认知冲突教学模式培养的学生在能力提升、课堂互动、高阶对话以及后续学习中解决跨学科问题的正确率都明显高于传统模式培养的学生。具体统计数据如表1所示。

任课老师针对采用认知冲突教学模式培养的学生进行了问卷调查，同时获得反馈信息，如图6显示，93.5%的学生认为改进的教学环节对自己的学习效果提升，而在代码能力、概念理解和学习主动性方面具有提升的学生分别占据71.3%、56.8%和45.3%，提升效果显著。

表 1 学生平均成绩、评教成绩和课程目标达成度

指标类型	评估指标	传统教学模式	本文提出的教学模式	变化率
能力提升	核心技能考核成绩	72.3分	85.6分	+18.4%
互动频次	课堂主动发言次数	17.3次	28.5次	+64.7%
高阶对话	课堂高阶思维对话	3.2次	4.7次	+46.8%
长效影响	跨学科问题解决正确率	65.9%	76.7%	+16.4%

8 结论与展望

本研究构建的框架为 LLMs 支持下的认知冲突教学提供了系统化实施方案。通过冲突触发、智能调节、深度转化的三阶模型，以及五阶段教学流程与质量保障机制，形成完整的实践路径体系。框架的创新价值体现在三个方面：教学目标上实现从语法记忆到计算思维培养的跃迁，教学过程上建立人机协同的新型教学样态，教学评价上发展基于认知过程分析的形成性评估。

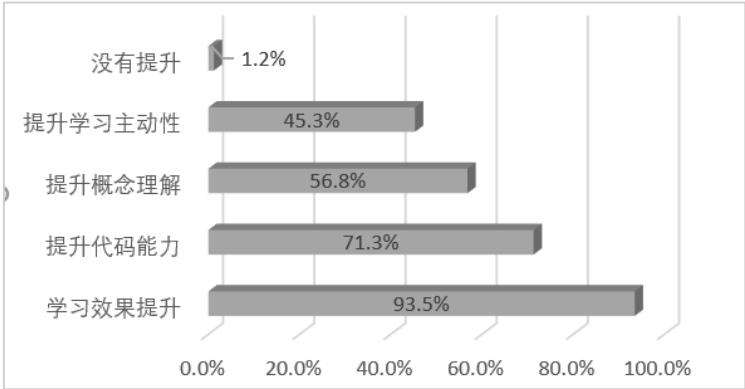


图 6 引入基于 LLMs 的 python 教学的学生评价

后续研究可在三个方向深化探索：其一，开发学科融合型冲突案例，促进编程思维向数学建模、数据科学等领域的迁移；其二，研究不同认知风格学习者的冲突适配策略，提升个性化教学效能；其三，构建教育大模型的伦理审查制度，防范技术应用中的隐私泄露与算法偏见风险。本研究旨在为 LLMs 赋能高等教育改革与创新提供可参考的思路。

参考文献

[1] 刘宇桦, 杨玉东. 运用认知冲突促进学生概念转变的循证研究[J]. 教育研究与评论, 2025(01):76-82.

[2] 张伟. 智慧教育赋能教育强国研究：大语言模型视角[J]. 中国教育信息化, 2024, 30(12):3-12.

[3] 郭曦, 王建勇. 生成式人工智能在 Python 教学中的作用与思考[J], 计算机技术与教育学报, 2024, (12) 02:31-36.

[4] 刘明剑, 张思佳, 李奇蔚, 葛泰亨. 依托“Python 程序设计”课程的学生创新能力培养模式研究与实践[J], 计算机技术与教育学报, 2024, 12(4):40-44.

[5] 李亚坤, 颜荣恩, 杨波, 等. 生成式人工智能背景下高校软件工程课程的教学改革与探索[J], 计算机技术与教育学报, 2024, 12(5):8-12.

[6] 杨小涵, 卢宇. 生成式人工智能助力课堂教学[J]. 湖北教育(教育教学), 2025 (03):8-10.

[7] 徐晓飞, 张策. 生成式人工智能赋能工程教育和学生能力培养、测评与认证体系[J]. 高等工程教育研究, 2025 (03):1-14.