知识图谱与大模型双驱动的《软件工程》课程人机协作教学模式研究

王晨华　孙　姗 *　陈玉姝　祝倩倩　王丽楠
（新疆科技学院）

DOI：10.65285/JYLLYJ.V2I6.025

摘　要：应用型本科高校软件工程课程存在一个结构性矛盾：知识体系具有过程性，教学组织却按线性方式推进。大语言模型进入课堂后，“效率提升”与“能力替代”之间的张力进一步显现。本文提出一种人机协作教学模式，其核心是“知识图谱认知约束 + 大模型角色情境”。在这一模式中，知识图谱不再只是知识表示工具，而是工程决策的认知锚点；大模型也不只是问答工具，而是工程情境模拟器。围绕《软件工程》课程，本文设计了三类角色情境：“模糊客户”“挑剔评审”“需求变更者”，并构建“图谱导航—情境演练—决策记录”教学路径。初步实践显示，该模式能够利用 AI 情境的丰富性，同时让学生保持工程判断主体的地位。

关键词：软件工程；人机协作；知识图谱；大语言模型；工程判断

Research on the Human-Machine Collaborative Teaching Model of the Software Engineering Course Driven by Knowledge Graph and Large Language Model

Chenhua Wang, Shan Sun*, Yushu Chen, Qianqian Zhu, Linan Wang
Xinjiang University of Science and Technology

Abstract: Software engineering courses in application-oriented universities have a structural contradiction: the knowledge system is procedural, whereas teaching is arranged in a linear way. After large language models are introduced into classrooms, the tension between “efficiency improvement” and “competence substitution” becomes more prominent. This paper puts forward a human-machine collaborative teaching model with the core of “cognitive constraints of knowledge graph + role scenarios of large model”. In this model, the knowledge graph is not only a knowledge representation tool but also a cognitive anchor for engineering decision-making. The large model is not merely a Q&A tool, but an engineering scenario simulator. Targeting the Software Engineering course, this paper designs three role scenarios: "vague client", "critical reviewer" and "requirement changer", and builds a teaching path of "graph navigation — scenario drill — decision recording". Preliminary practice indicates that this model can take advantage of rich AI scenarios while enabling students to remain the main subjects of engineering judgment.

Keywords: software engineering; human-machine collaboration; knowledge graph; large language model; engineering judgment

前言：

《软件工程》课程要培养学生在真实工程情境中做出有依据判断的能力。当前教学面临三重困难。

第一，工程知识具有过程性，教学组织却偏线性。软件工程知识围绕生命周期展开，需求、设计、测试相互依赖。教材与课堂往往按阶段推进，学生容易形成“各阶段相互独立”的认知偏差。

第二，大模型辅助教学可能从“效率提升”滑向“能力替代”。AIGC 工具链可以提升需求分析效率，代码生成准确率也较高，但效率提升不等于能力培养[1]。学生能提交可运行代码，却无法追踪路径、识别隐藏假设，也解释不了依赖关系。

第三，现有知识图谱实践多把图谱定位为“知识可视化与学习路径推荐”工具，尚未充分发挥它在约束 AI 输出、支撑工程决策训练方面的认知功能。

针对上述问题，本文尝试一种新的教学组织方式：把“认知约束”和“角色情境”作为两个抓手。这样做，是想回答一个现实问题：AI 已经进入课堂，学生的工程判断力会不会被削弱？如果不会，靠什么撑住？

一、应用型本科高校软件工程教学现状

（一）课程体系与行业需求的“过程性脱节”

应用型本科高校的软件工程专业，课程体系与行业需求之间常有落差，教学内容更新也偏慢[2]。这种落差不只是技术新旧的问题，更在于知识怎么组织。传统课堂多采用“方法讲授 + 课程设计”的教学方式，教材按章节顺序推进，背后默认了一个前提：需求、设计、测试各管各的。真实

项目不是这样。设计阶段可能发现需求前后矛盾，只能回头改；测试阶段暴露的缺陷，根子可能在设计，不在编码。学生如果没建立起知识依赖和反馈关系的整体感，到了职场就很难跟上真实项目的迭代节奏。

（二）AI 工具使用中的“隐性依赖”现象

大语言模型进入学习场景后，有个现象值得留意：能力幻觉。学生用 AI 生成一份完整项目文档，或者跑通一段代码，就容易觉得自己“已经会了”。可一旦让他解释设计依据，或者把需求一路追到测试用例，再或者临时改一个设计约束，理解就卡在表面。这里的问题不在于学生故意偷懒，而在于他们手里没有一套审视 AI 输出的工具。知识图谱可以补上这个缺口。它像一张可查证的底图。AI 给出建议时，学生能顺着图谱看这个节点跟上下游是什么关系，再判断建议跟需求约束是否一致、测试上是否可验证。

（三）工程判断能力培养的缺失维度

工程判断的本质是在不确定性中做出有依据的决策，需要三个认知条件：信息的不完整性、方案的多可能性和决策的可解释性。传统课堂难以提供这三个条件。教材中的案例往往是需求明确、约束清晰、最佳实践明确。即使有课程设计项目，学生也倾向于选择自己熟悉的方案，而非探索不确定性。AI 的引入若使用不当，反而会加剧问题。学生可以直接要求 AI“给出最优方案”，绕过决策过程直接获得答案。

二、双驱动人机协作教学模式设计

已有实践表明，将知识图谱与大模型结合可以部分解决上述问题。例如，有教师构建 “知识图谱 + 大模型 + 智能开发工具”的人机协作新范式，覆盖 520 名学生，86.2% 认为知识图谱有助于理解方法关联 [3]。但该模式主要聚焦“方法选择”，而非“决策判断”。本节在此基础上，提出“知识图谱认知约束 + 大模型角色情境”的双驱动模式设计。

（一）知识图谱的软件工程教学建模

软件工程的核心关系是“过程流转与决策依赖”。因此，知识图谱的构建需引入过程节点作为核心实体。每个过程节点标注三类属性：输入依赖、输出产出、反馈关系。这些过程节点及其依赖关系构成学生审视 AI 建议的“认知锚点”。

在此基础上，本模式进一步强调过程节点之间的决策影响关系。选择某种架构风格将约束后续的可测试性设计；接受某项需求变更将改变测试计划的覆盖范围。这些影响关系构成学生审视 AI 建议的推理链条。当 AI 建议采用微服务架构时，学生可沿图谱追溯：微服务架构对测试策略提出了什么要求？当前团队能力是否匹配？部署复杂度是否在可控范围内？

（二）大模型辅助的软件工程教学实践

传统“问答模式”中，学生将 AI 输出视为“参考方案”，而非“待检验的假设”。角色化使用旨在改变这一关系。学生不再是“提问者—接受者”，而是“追问者—判断者”。

“模糊其词的客户”（需求分析阶段）。大模型扮演对需求有大致想法但表达不清晰、对技术约束缺乏认知的客户。学生需主动追问来澄清需求边界、识别隐含约束。

“挑剔的技术评审”（设计阶段）。大模型扮演有经验的评审者，对学生的设计方案从可维护性、性能、团队能力匹配度等角度提出质疑。学生必须为设计选择提供依据，而非简单采纳 AI 生成的“最佳实践”。

“中途变更的需求方”（项目迭代阶段）。大模型扮演在开发中途提出需求变更的利益相关者。学生需判断变更是否合理、影响范围有多大、如何调整测试策略。这类训练直击软件工程的核心难点：在变化中维持工程秩序 [4]。

（三）人机协作教学模式的补充设计

在角色情境之外，还需设计支撑人机协作的基础机制。已有研究讨论过“师—生—AI”三元协同架构的探索 [5][7]，本文在此基础上做了三层设计。

第一层，把任务分工摆到明面上。每个教学节点都标清楚：哪些事交给 AI，哪些事必须学生自己来。AI 可以生成方案草案、给备选项、检索资料、模拟对话回应。学生则要负责追问和澄清、评价和选择方案、说出决策依据、批判性地看 AI 输出。这张分工表直接画在知识图谱里，学生一眼能看出：什么时候可以靠 AI，什么时候必须自己判断。

第二层，设“手动闸门”。CARLWIN 框架里有个做法，指令手动复制、输出手动粘贴 [6]。我们把它搬到关键决策点上。学生不能直接让 AI 生成最终产出，必须走完“提需求 → AI 生成→人工审核→修订完善”这一圈。这么做，是为了保住学生的认知参与，不让“一键生成”把学习过程绕过去。

第三层，评价不只盯最终产出，也看协作过程。具体看三个数：学生追问 AI 的频次和深度、识别出 AI 建议不合理的比例、在决策日志里引用知识图谱做验证的次数。

三、双驱动模式的教学路径构建

（一）图谱导航与认知准备

课程启动阶段，教师向学生展示完整的软件工程知识图谱，重点阐释三类关系：阶段依赖关系、产出物流转关系、反馈迭代关系。每个教学单元开始前，通过图谱定位明确“本节点在生命周期中的位置”“上游依赖是什么”“下游影响是什么”。

进入人机协作情境之前，要求学生完成一项“认知准备任务”：根据知识图谱中当前节点的输入输出关系，列出“在与 AI 对话时需要澄清的关键信息清单”。这一准备任务将图谱的“约束功能”转化为学生主动构建提问框架的认知活动。

（二）情境演练与决策记录

每个核心知识节点配备一组角色情境任务，典型流程包括四个步骤：图谱定位与假设生成；角色对话，系统记录对话轮次与追问深度；决策记录，学生撰写“决策日志”，包含判断、替代方案、选择依据、对 AI 建议的采纳或拒绝理由；图谱反馈，根据决策日志质量和对话表现更新知识图谱中相应节点的掌握度标记。

这一流程的关键设计意图是将学生的认知努力从“生成方案”转移到“做出判断”。AI 可以生成方案，但 AI 判断的依据、替代方案的评价、采纳与拒绝的决策必须由学生完成。决策日志的形式迫使这一认知过程外显化，使之可评价、可反思。

（三）双驱动模式的初步实践

从课堂看，角色情境确实把追问行为带起来了。以前学生向 AI 提问，问一句就完。换成角色扮演后，提问频率高了，追问也更深。面对“模糊客户”，学生只能一轮一轮进行追问来获取信息。这种认知负荷，比直接向 AI 提问更接近真实的需求沟通。

决策日志则把“隐性依赖”翻了出来。有些学生在角色对话里很积极，可日志里只写一句“AI 建议用微服务，我觉得可以”。没有独立评价，也没有推理。日志的好处就在这里：它把这种依赖摆到台面上，教师能看见，也就能针对性干预。

知识图谱的约束功能，还得专门教。部分学生能看懂图谱，但审视 AI 建议时并不主动拿图谱去追溯、去验证。这说明“会用图谱”和“在决策中会用图谱”是两回事。后者需要专门设计教学环节，反复练，才能变成习惯。

四、过程性评价的改造

传统软件工程课程评价主要是团队项目成果和期末考试，很难追踪每个学生在各个知识节点上的决策质量。本文在人机协作框架下，将过程性评价重新设计如下：

图谱学习轨迹与准备质量，占 20%。记录学生在各知识节点的认知准备任务完成情况，以及能不能从图谱关系里认出关键决策点。

角色对话质量，占 30%。记录学生在追问中的表现：学生能不能提出覆盖图谱关键依赖的问题？学生能不能识别对话中透露的隐含约束？学生能不能对模糊信息进行了有效澄清？

决策记录质量，占 50%。该部分是评价的核心维度，评分主要看三个要素：判断有没有依据、有没有考虑替代方案、对 AI 建议是采纳还是拒绝，逻辑说清楚没有。

已有研究指出，生成式 AI 很容易把“任务完成”和“学习发生”搅在一起[8]。因此，评价要有意去抓“生成交互之后”的能力证据：口头解释、独立测试、设计理由说明、对 AI 输出的批判性评述等。决策记录就是把这些证据系统收上来的采集方式。

结语：

大语言模型进软件工程课堂，已是不可逆的趋势。现在要问的不是“用不用 AI”，而是“怎么用 AI，才不至于把学生的工程判断力越教越弱”。本文思路是让 AI 去制造工程情境里的不确定性，让学生在不确定性里练判断。知识图谱管两件事，一是把知识结构画清楚，二是给学生一个审视 AI 输出的参照点。角色化大模型则提供传统课堂缺的那块：情境压力。

本模式的实施对教师提出较高要求：需要构建与维护知识图谱，设计有效的角色情境任务，同时对学习数据持续关注和解读。后续工作将主要集中在降低图谱构建成本、开发标准化角色情境模板、优化提示词策略以增强角色扮演稳定性，同时在更大的教学中检验框架的有效性。

软件工程教育到底在教什么？也许可以这么说：让学生在一个学期的“安全环境”里，过一遍浓缩的工程人生——面对模糊需求，做出有依据的设计选择，在变化里调整计划。知识图谱与大模型的双驱动设计，最后都指向同一个目标：让这段浓缩经历更真实、更可反思、也更以判断为核心。

参考文献：

[1]Yang X, Guo P. Practical Research on the Application of the CDIO Model Optimized by AIGC Toolchain in the Integrated Training of Software Technology[C]//ACM, 2026.

[2] 数字化转型背景下的应用型高校软件工程专业人才培养模式研究与实践 [C]//2025 年高等教育发展论坛论文集 .

[3] 从知识图谱到人机协同 : 软件测试课程三阶递进教学案例 [R]. 2026 全国高校软件工程教育大会 ,2026.

[4]From Specifications to Implementation in the Gen-AI Era: Lessons from a Project-Based Software Engineering Course[C]//ACM, 2026.

[5]Research on the Application of an AI-Tutor-Driven Human-Computer Collaborative Learning Model in Vocational College Website Front-End Teaching[C]//ACM, 2026.

[6]The CARLWIN Framework: A Learner-Centered Pedagogical Methodology for Human-AI Collaboration in Software Engineering Capstone Projects[C]//ACM, 2026.

[7] 基于雨课堂 +DeepSeek 的解析几何“师 - 生 -AI”三元协同教学模式构建 [J]. 长江文库 ,2025.

[8]Research on the Effectiveness of Human-Machine Collaborative Teaching Based on Data Analysis in the Era of Digital Intelligence[C]//IEEE, 2025.

作者简介：王晨华（1994 年 -）、女、汉族、河北唐山、硕士研究生、新疆科技学院、讲师、研究方向：边缘计算。

通讯作者：孙姗（1994 年 -）、女、汉族、山东临沂、硕士研究生、新疆科技学院、讲师、研究方向：高等教育。

基金项目：2025 年新疆科技学院“基于知识图谱与 AI 大模型的《操作系统》课程教学模式探究”（项目编号：JGPT-2025-13）。