面向开源协同能力培养的研究生前沿课程改革与实践*

张宇霞** 谭鑫 刘辉

北京理工大学计算机学院，北京 100081 北京航空航天大学计算机学院，北京 100191 北京理工大学计算机学院，北京 100081

摘 要 针对当前研究生计算机课程在开源人才培养中与产业需求相脱节的问题，本文结合开源软件的演进趋势与课程教学实际，提出一套面向开源人才协同能力培养的计算机前沿课程改革方案。该方案以成果导向教育（OBE）理念为指导，将理论讲授、案例分析、项目实践和开放交流四个环节有机融合，形成循序渐进的教学模式。在教学组织上，采用“教师引导、学生探究、同伴互动”相结合的混合式策略；在实施流程上，依次推进课堂讲授与前沿导入、小组汇报与轻量化开源实践、反思反馈与持续改进三个环节。文章系统介绍了该改革方案在《计算机科学与技术前沿》课程中的教学实践与成效。课程反馈显示，超过 96%的学生认为课程有助于加深对开源知识的理解，超过 90%的学生认为课程显著提升了工程实践能力和团队协作能力。研究表明，将开源教育有机嵌入计算机前沿课程，能够有效实现知识学习、能力训练与价值塑造的协同推进，为高校开源人才培养提供具有推广价值的课程改革路径。

关键词 开源软件，课程改革，开源人才培养，项目式教学

Reform and Practice Of A Graduate Frontier Course For Cultivating Open Source Collaboration Competencies *

Yuxia Zhang Xin Tan Hui Liu

School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China yuxiazh@bit.edu.cn

School of Computer Science and Technology, Beihang University, Beijing 100191, China xintan@buaa.edu.cn

School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China liuhui08@bit.edu.cn

Abstract—To address the disconnection between current graduate-level computer science courses and industry demands in open-source talent cultivation, this paper proposes a reform scheme for advanced computer science curricula aimed at fostering collaborative competencies in open-source development, grounded in both the evolutionary trends of open-source software and actual teaching practices. Guided by the Outcome-Based Education (OBE) concept, the scheme organically integrates four components—theoretical instruction, case analysis, project practice, and open exchange—into a progressive teaching model. In terms of course organization, a blended approach combining “teacher guidance, student inquiry, and peer interaction” is adopted. The implementation proceeds through three sequential phases: lecture-based frontier introduction, group presentations coupled with lightweight open-source practice, and reflective feedback for continuous improvement. This paper systematically presents the teaching practice and outcomes of implementing this reform in the course *Frontiers of Computer Science and Technology*. Course feedback indicates that over 96% of students found the course helpful for understanding and learning open-source knowledge, and more than 90% reported significant enhancement in their practical engineering skills and teamwork abilities. The findings suggest that embedding open-source education organically into advanced computer science courses effectively synergizes knowledge acquisition, skill development, and value shaping, thereby offering a replicable pathway for open-source talent cultivation in higher education institutions.

Keywords—Open-source Software, Curriculum Reform, Open-source Competency Cultivation, Project-based Learning

1 引 言

开源软件已成为数字基础设施与技术创新体系的重要组成部分。从操作系统、数据库到人工智能框架，开源项目不仅重塑了软件产业的研发组织方式，也深刻改变了技术人才的能力结构要求[1]。在此背景下，产业界对人才的需求已由单一编程能力转向版本协同、

*基金资助：本文得到国家自然科学基金 62572048 和北京理工大学研究生教育培养综合改革一般项目资助

*通讯作者：张宇霞 yuxiazh@bit.edu.cn.

社区沟通、许可证合规、持续集成与跨团队协作等复合能力。然而，当前高校计算机课程体系普遍存在知识讲授与真实工程实践相分离、课程内容与开源生态脱节、评价机制难以有效表征协作贡献等问题，致使学生虽然接触了开源概念与相关工具，却难以形成对开源运行机制与社区参与方式的系统性理解。

作为连接技术前沿与专业培养的重要课程，计算机前沿课程具有内容更新快、覆盖面广、适宜引入新技术范式等特点，是开展开源教育的重要载体。然而，若仍沿用以教师讲授与文献介绍为主的传统教学方式，课程容易停留于前沿知识展示层面，难以有效承担开源人才培育的任务。基于此，有必要围绕课程目标、教学内容、实践载体与评价方式实施系统性改革，推动课程由知识介绍型向能力生成型转变。

基于上述认识，本文结合北京理工大学“计算机前沿”课程建设实践，在梳理开源人才培养需求与高校开源教育痛点的基础上，提出面向开源人才培养的课程改革总体框架，并结合 85 份学生问卷反馈对课程实施效果进行实证分析，以期为同类课程建设提供可借鉴的经验。

2 改革动因与问题分析

2.1 开源人才培养的现实需求

开源已成为技术创新的重要组织方式[1]。一方面，Linux、Apache、TensorFlow、Kubernetes 等代表性项目推动了关键技术的快速演化；另一方面，企业对人才的需求逐渐从“熟悉某项技术”转向“能够在开放协作环境中持续贡献”[2]。这意味着高校在人才培养中不仅要关注知识体系，还要同步培养学生的问题发现能力、公共表达能力、工程协同能力与规则意识。

从国家战略层面看，开源生态建设已经与基础软件自主可控、数字经济发展和高水平科技自立自强紧密相关，“十四五”规划将开源纳入国家战略，工信部推动建立开源应用示范标杆。产业界已经形成了多元参与格局，华为、百度等企业通过开源认证课程培养专业人才，PingCAP 等开源商业公司构建培训认证体系。高校既是开源理念传播的重要阵地，也是高水平开发者与研究者的源头供给方。因此，将开源教育嵌入专业课程，不仅是课程内容更新的问题，更是人才培养体系适应产业变革的必然要求。

2.2 传统前沿课程面临的主要问题

结合课程前期教学观察与文献研究，传统计算机前沿课程在开源人才培养方面的不足主要体现在内容体系与实践链条层面。

就内容体系而言，教学往往偏重计算机算法，特色 AI 技术相关的专题介绍，缺乏围绕开源生态的理论与实践教学。学生对开源协议、社区治理、协作模式和商业化路径等核心开源理论体系缺乏了解。进一步看，实践环节相对薄弱，难以形成“从认知到参与”的教学闭环。个别课程虽然引入了开源项目案例，但学生并未真正进入代码托管平台、Issue 讨论、Pull Request 审阅等真实情境，实践体验仍停留在外围观察层面。

与此同时，学习共同体虽已初具雏形，但组织机制仍显粗疏。小组汇报、课堂提问和共享交流能够显著提升参与度，但一旦缺少结构化设计，便容易出现提问形式化、主题分散、评价失真等问题，从而影响教学质量与学生体验。相反，小组课题汇报是天然的开源实践训练场。若以开源协作的方式组织小组任务的开展，将大大提高学生对开源协作的感性认知以及技能掌握，同时也有助于提升课程考核的公平公正性。

3 面向开源人才培养的课程改革设计

3.1 改革目标

围绕开源人才培养要求，课程将教学目标落实为相互衔接的四个维度。其中，认知目标旨在帮助学生理解开源软件的发展逻辑、社区文化、许可证规则与产业生态，建立对开源的整体性认识；在此基础上，工具目标进一步强调 Git、代码托管平台、Issue 协作、持续集成等基本方法的掌握，使学生具备进入真实开源场景的技术基础。随着学习层次的推进，课程将协作能力纳入核心培养范围，通过小组汇报、跨组提问和项目实践提升学生的信息整合、表达沟通、分工协作与同行互评能力；最终，课程希望引导学生将开源学习与自身研究方向相结合，在分析前沿项目和参与社区实践的过程中逐步成长为合格的开源人才。

3.2 教学内容模块化设计

课程采用“专题讲授+项目驱动”的模块化架构，设置三大核心模块：

（1）开源理论基础模块：通过《开源软件及其开发概述》《开源开发的思想和原则》等专题，系统讲解开源协议（如 GPL、Apache）[8]、社区治理模式（如 Linux 基金会运作机制）及开源文化内涵，结合 GitHub 开源项目代码库分析，帮助学生构建开源生态认知框架。

（2）技术工具实践模块：依托真实开源项目（如 Apache Kafka、TensorFlow）开展案例教学，演示 Git 版本控制、持续集成(CI/CD)工具链及代码审查流程，通过作业提交和开源项目实践等方式，使学生掌握 Pull Request 等开源协作开发范式。

（3）商业创新拓展模块：《开源商业模式》[7] 专题以顶尖开源软件项目，如 Linux 内核等为案例，

解析企业基于开源的商业模型，组织学生开展开源项目商业价值分析沙盘推演，培养技术经济复合思维。

3.3 教学组织与评价机制

在教学组织上，课程采用“教师引导+学生探究+同伴互动”的混合模式。教师负责知识框架建构与关键问题引导，学生围绕选题开展资料检索、项目分析与课堂展示，其他同学则通过提问、点评与讨论参与共同建构。该设计有助于将单向输入式课堂转化为多主体参与的学习共同体。

4 教学实施路径

4.1 课堂讲授与前沿导入

课程安排 6 学时系统介绍开源发展历史、典型社区、关键工具和前沿趋势。与传统“概念堆砌式”讲授不同，课程强调通过真实项目和产业案例组织教学内容，使学生认识到开源不仅是一种软件开发方式，更是一种知识生产机制、协作机制和创新机制，并能够在较短时间内对开源的整体脉络与未来方向形成全面认知。课程案例库以“技术前沿性—社区活跃度—教学适配性”为筛选标准，聚焦计算机科学与技术领域的核心开源项目，覆盖基础工具、分布式系统、人工智能框架等方向，形成了理论与实践深度融合的教学资源体系。教师在导入过程中有意识地将技术演化、产业变革和人才能力要求结合起来，增强课程的现实针对性。

4.2 小组汇报与轻量化开源实践

小组汇报是课程的核心组织形式之一。学生围绕选定的前沿开源主题开展文献研读与项目调研，并在课堂上进行系统汇报。与单纯的论文复述式展示相比，课程更强调从问题背景、技术路线、社区生态、产业应用与发展趋势等多个维度展开综合分析，以促进学生形成结构化、学术化的表达方式。

共享交流环节进一步将汇报课堂转化为协作学习场域。其他同学需结合报告内容提出问题、补充观点，或联系自身研究方向展开讨论。该环节不仅强化了全体学生的参与意识，也提升了知识在群体中的传播与再加工效率。教学实践表明，这种报告驱动的同伴学习方式较单一讲授更有利于前沿知识的理解、内化与迁移。

考虑到前沿课程学时有限且学生知识背景存在差异，课程并未机械追求高强度代码贡献，而是采用轻量化、阶梯式的实践路径：首先通过文档研读、环境熟悉、社区观察与流程理解帮助学生建立参与信心，随后逐步引导其开展问题分析、文档改进、资源整理或简单贡献尝试。该设计在降低参与门槛的同时，也使开源实践更符合课程教学规律与学生成长节奏。

课后思考与提问则是课堂学习的重要延伸。教师在每次课程或小组汇报结束后鼓励学生积极提问，提问方向既包括对开源协议、开发模式的理解，也涉及前沿开源项目的最新动态，鼓励学生主动思考。学生通过线上平台整理问题与心得，小组汇报成员在课后进行答疑和拓展，逐步形成互动式的学习氛围。

4.3 反思反馈与持续改进

课程将学生反馈视为教学改革闭环中的关键环节。除课堂即时反馈外，教学团队还在课程结束后通过问卷系统收集学生对课程内容、教学形式、小组作业与工具使用等方面的评价，并对开放性建议进行归纳分析。由此，课程优化不再主要依赖教师个体经验判断，而是建立在证据支持的持续改进机制之上。

在评价方式上，课程坚持过程性评价与结果性评价并重。结果性评价主要考察小组报告质量、课程论文与最终展示效果，过程性评价则关注资料准备、课堂互动、项目实践痕迹及团队协作表现。

5 教学成效与数据分析

5.1 问卷样本与分析方法

课程期末共获得 85 份有效问卷，其中本科 1 人，硕士 48 人，博士 36 人。问卷包含 4 个 Likert 量表题和 1 个开放性问题，分别考察课程对学生开源知识理解、开源参与实践与工具使用、以及开放共享交流方式的帮助情况。本文将“A.帮助很大”“B.帮助较大”“C.帮助一般”“D.帮助很小”“E.没有帮助”分别赋值为 5、4、3、2、1，以便进行均值统计和横向比较。同时，对开放题反馈进行主题归类，用于提炼课程改革的深层改进需求。

5.2 量化结果分析

从总体结果看，5 个量表题的平均分均高于 4.40，表明学生对课程改革的整体评价较高，不同教学环节均产生了较为积极的学习效应。具体结果见表 1。

从表 1 可以看出，与开源课程直接相关的四项指标呈现出较为清晰的递进关系。开源背景与知识学习的平均分为 4.47，“帮助很大”与“帮助较大”的合计比例达到 95.30%，表明课程在帮助学生建立开源基本概念、理解开源生态及相关发展背景方面取得了较好成效。与之相比，开源参与和工具使用的平均分为 4.42，积极评价比例为 89.41%，虽然总体评价依然较高，但“帮助一般”及以下的比例相对更高，说明学生在将知识理解进一步转化为工具运用和实际参与能力时仍面临一定门槛。课程中的交流互动环节包括课

堂提问、小组讨论和开放共享交流等。开放共享交流环节的平均分达到 4.52，高于前两项指标，积极评价比例同样达到 95.30%，说明交流讨论不仅有效承接了知识讲授，也在答疑释惑、拓展视野和增强学习投入方面发挥了重要作用。课程小组对于个人协作能力提升的平均分达到 4.60，尤其是选择“帮助很大”的学生占比最高，比例为 64.71%，仅 4.71%学生选择了“帮助一般”，且没有学生选择“帮助很小”或“没有帮助”，这表明小组作业在促进学生开放式实践协作方面效果显著。

表 1 课程改革效果问卷统计结果

问题	A. 帮助很大（占比/%）	B.帮助较大（占比/%）	C.帮助一般（占比/%）	D.帮助很小（占比/%）	E.没有帮助（占比/%）	平均分
开源背景与知识	**54.12**	41.18	3.53	0.00	1.18	4.47
开源参与和工具	**56.47**	32.94	8.24	1.18	1.18	4.42
开放共享交流	**57.65**	37.65	3.53	1.18	0.00	4.52
协作能力提升	**64.71**	30.59	4.71	0.00	0.00	4.60

表 2 不同学历阶段学生对了解与学习开源背景与知识帮助的反馈

学历	A. 帮助很大（占比/%）	B.帮助较大（占比/%）	C.帮助一般（占比/%）	D.帮助很小（占比/%）	E.没有帮助（占比/%）	平均分
硕士	**47.92**	43.75	6.25	0.00	2.08	4.35
博士	**63.89**	36.11	0.00	0.00	0.00	4.64

表 3 不同学历阶段学生对多人协作能力提升的反馈

学历	A. 帮助很大（占比/%）	B.帮助较大（占比/%）	C.帮助一般（占比/%）	D.帮助很小（占比/%）	E.没有帮助（占比/%）	平均分
硕士	**62.50**	29.17	8.33	0.00	2.08	4.54
博士	**69.44**	30.56	0.00	0.00	0.00	4.69

进一步从不同学历阶段学生的反馈情况来看，如表 2、表 3 所示，硕士生与博士生均对课程改革表现出较高认可度，且博士生在各项指标上的积极反馈比例整体高于硕士生。如表 2，在开源背景与知识学习方面，博士生选择“帮助很大”的比例达到 63.89%，硕士生的则是 47.92%，两类学生的平均分分别达到 4.64 和 4.35，均体现出课程在促进学生理解开源理念、技术背景以及生态体系方面的积极作用。不同学历阶段学生反馈差异在一定程度上反映了学生已有知识基础、科研经历和学习方式的不同，但并不影响课程改革对各类学生能力发展的促进作用。表 3 展示的在小组协作环节对能力提升的方面，博士生同样表现出更高的认可度，选择“帮助很大”的比例达到 69.44%，平均分达到 4.69，硕士生平均分为 4.54，表明以真实项目为载体的小组实践模式能够适应不同学习阶段学生的能力发展需求。

整体来看，课程改革已较好实现了从开源认知建构到交流深化的教学目标，但在由认知走向实践参与的环节上仍需进一步加强过程指导与实践支撑。

5.3 开放性反馈分析

开放性问题共收集到 51 条有效建议。归类分析后可以发现，学生的意见主要集中在三个方面：

其一，课程应进一步强化开源实践环节，不少学生希望增加代码练习、真实项目参与和基于代码托管平台的协作训练，以缩短从开源知识学习到实际贡献之间的距离。约 30%的学生建议进一步增加实验课程内容、引入更多前沿开源技术报告，并优化课堂提问

形式，希望通过更加丰富的实践教学提升对开源技术和工程能力的理解。

其二，课程需要更充分地呈现开源协作过程，包括项目参与方式、协作规范和贡献路径等内容，使学生不仅“理解开源”，还能够逐步进入“参与开源”的真实场景。

其三，部分学生期待课程在选题组织上进一步向开源前沿和交叉应用方向延伸，例如结合多模态大模型、交叉学科应用或更加细分的技术主题，以增强课程内容与开源生态演化趋势之间的关联性。其中，约20%的学生建议进一步增加跨学科内容，统一课程考核方式，并引入更多真实开源项目参与机会，使课程更加贴近当前开源产业的发展需求，进一步提升学生的综合能力和创新意识。

总体来看，开放性反馈并未削弱课程改革的基本方向，而是进一步表明学生对开源实践深度、协作机制体验和前沿主题拓展具有较高期待。这说明课程已在开源认知建构层面取得较好基础，后续优化的重点应放在提升学生真实参与开源项目的能力与体验上。

6 结束语

面向开源人才培养的计算机前沿课程改革，本质上是在课程教学中引入开源生态中的真实规则、真实问题与真实协作方式，使学生在前沿知识学习过程中同步形成工程意识、协作意识与创新意识。本文基于课程建设实践，提出了以成果导向教育为引领的课程改革框架，并通过问卷数据验证了课程在知识理解、交流互动、汇报表达、前沿认知与工具使用等方面的积极成效。研究表明，开源教育与计算机前沿课程之间具有较强的耦合关系。前沿课程为引入开源项目与技术演化案例提供了内容空间，开源教育则为课程注入了真实协作与能力培养维度。二者的有机结合，有助于推动课程由知识传授型向能力生成型、由结果评价型向过程改进型转变。未来课程仍可沿着几个相互支撑的方向继续深化。一方面，应构建更加规范的课堂提问与问题追踪机制，以提升互动质量并增强交流环节的可持续性；另一方面，有必要引入匿名互评、过程留痕与贡献认证机制，以增强小组评价的公平性和可信度。在此基础上，还应进一步拓展真实开源项目的参与深度，形成“课程学习、项目实践与社区融入”的递进式培养路径。通过持续迭代，该课程有望成为高校开源人才培养的重要支点，并为我国开源生态建设与计算机教育改革提供更有力的支撑。

参 考 文 献

[1] 金芝,周明辉,张宇霞.开源软件与开源软件生态:现状与趋势[J].科技导报,2016,34(14):42-48.

[2] 陈凯华, 温馨, 朱其罡. 我国企业参与开源创新的现状, 挑战与政策建议[J]. 中国科学院院刊, 2025, 40(3): 477-482.

[3] 周明辉, 张宇霞, 谭鑫. 软件数字社会学[J]. 中国科学: 信息科学, 2019, 11: 1399-1411.

[4] 中国计算机学会.CNCC 第三届开源生态发展与开源人才培育一体化建设之路[EB/OL].(2024-10-17)[2025-3-14].https://www.ccf.org.cn/Focus/2024-10-17/831348.shtml.

[5] 毛新军. 基于开源和群智的软件工程实践教学方法[J]. 软件导刊, 2020, 19(1): 1-6.

[6] 谭鑫,林泽燕,张宇霞,周明辉.代码文件贡献组成模式的分析[J].软件学报,2018,29(08):2283-2293.

[7] Zhang Y, Zhou M, Mockus A, et al. Companies' participation in oss development–an empirical study of openstack[J]. IEEE Transactions on Software Engineering, 2019, 47(10): 2242-2259.

[8] Xu W, He H, Gao K, et al. Understanding and remediating open-source license incompatibilities in the pypi ecosystem[C]//2023 38th IEEE/ACM International Conference on Automated Software Engineering (ASE). IEEE, 2023: 178-190.