

大模型赋能 Selenium 自动化测试课程教学改革探析

李颖 王景波 赵依扬 李开远

哈尔滨信息工程学院 黑龙江 哈尔滨 150431

【摘要】：大模型技术在 Selenium 自动化测试课程教学改革中的应用，有助于降低编程门槛、培育高阶思维以及实现全天辅导，有望重塑 Selenium 自动化测试课程教学体系，促进学生实操能力提高。基于此，文章围绕 Selenium 自动化测试课程教学内容、教学模式、教学实践以及教学评价四个维度展开研究，提出利用大模型技术重构课程内容、构建人机协同课堂、搭建智能化实践项目、立过程性与发展性相结合的评价体系等对策，希望为该课程改革提供参考。

【关键词】：大模型；Selenium；自动化测试课程；教学改革

DOI:10.12417/2982-3811.26.02.021

为了保证软件产品的质量，在软件开发的各个阶段都需要引入软件测试工作，以期尽早地尽可能多地发现系统中存在的缺陷，最大限度地确保软件产品的质量。Selenium 作为一种主流的开源 Web 自动化测试框架，凭借跨浏览器、多语言支持等优势，已成为高校软件测试课程的核心内容。但传统的 Selenium 教学存在着诸多的不足，如脚本编写要求学生综合掌握 HTML、XPath、CSS 选择器及编程语言，导致学生容易陷入耗时费力的调试困局；传统教学偏重 API 讲解，难以培养学生面对真实项目复杂场景时的问题解决能力。随着大型语言模型技术取得突破性进展，其为解决上述教学难题提供了全新的思路^[1]。大模型能够根据自然语言描述自动生成 Selenium 测试脚本，从而将学生从繁琐的语法记忆与反复调试中解放出来，推动学习重心转向测试策略设计、逻辑验证等高阶思维培养。因此，探析大模型赋能 Selenium 自动化测试课程的教学改革，能够为高校软件测试类课程教学改革提供借鉴。

1 大模型赋能 Selenium 自动化测试课程教学改革的价值

1.1 有助于降低编程门槛

Selenium 自动化测试课程依托 Python、Java 等编程语言开展实操教学，对编程基础薄弱的学生而言，代码编写、语法纠错、元素定位脚本调试等环节难度较大，容易因基础代码问题阻碍自动化测试学习进度，打击学生实训积极性。大模型具备智能代码生成、语法纠错、代码简化以及注释解析功能，可根据测试场景快速生成规范的 Selenium 基础脚本，并针对学生编写的错误脚本精准定位语法漏洞、逻辑缺陷并给出优化方案。大模型还可以辅助学生规避基础编程错误，无需耗费大量精力钻研基础代码编写，能够让学生聚焦测试思路、用例设计、场

景适配等核心知识点，从而降低课程编程准入门槛，帮助零基础、弱基础学生快速入门自动化测试，缩小学生之间的实操能力差距^[2]。

1.2 有助于培育高阶思维

传统 Selenium 自动化测试教学多以固定脚本复刻、标准化案例演练为主，学生长期处于机械复刻、被动执行的学习状态，仅能掌握基础的脚本编写操作，难以形成问题分析、测试优化、场景创新的高阶能力，不利于批判性思维与创新思维的培育。大模型融入课程教学后，可以改变传统的机械式实训模式，学生依托大模型就完成基础脚本编写与纠错，将学习重心从重复编码操作转移到测试方案设计、异常场景分析、脚本优化迭代等深度学习环节^[3]。不仅如此，学生还能够结合复杂测试场景自主探究动态元素适配、测试脚本复用、测试效率优化等进阶问题，能够锻炼自身的逻辑推理、问题研判、创新优化等高阶思维，打破固化的应试学习思维。

1.3 有助于实现全天辅导

传统课堂教学受课时、场地以及师资等资源限制，辅导时间集中在课堂授课时段，课后答疑渠道有限，难以满足学生碎片化、即时性的学习需求。Selenium 自动化测试实操具有极强的实践性，学生课后自主实训时，常遇到元素定位失败、脚本运行报错、测试流程逻辑混乱等各类问题，若问题无法及时解决，会中断实训节奏、积累知识漏洞，降低自主练习积极性。大模型具备全天候在线服务能力，可突破传统教学时空限制，为学生提供全天智能化辅导服务^[4]。学生在课后实训、课后复盘、考前练习等任意时段遇到问题，均可随时向大模型咨询，获取实时答疑、故障排查、知识点讲解和代码调试指导，从而弥补教师课后辅导覆盖不足的问题。

2 大模型赋能 Selenium 自动化测试课程教学改革路径

2.1 依托大模型重构课程教学内容

传统 Selenium 自动化测试课程内容多遵循从环境配置、API 讲解到简单案例模仿的线性逻辑，知识点相对孤立，各板块之间缺乏有机联动，学生即便掌握了单个 API 的用法，面对真实项目中复杂的业务场景与动态页面时仍会感到无从下手。大模型技术的引入为课程内容重构提供了新的可能，其核心在于打破以语法操作为中心的内容编排方式，转而以测试思维培养为主线，将元素定位策略设计、测试框架搭建、断言逻辑构建等高阶认知活动前置并贯穿始终。在这一思路下，课程内容可从“工具使用手册”式结构升级为“问题驱动-模型辅助-反思深化”的递进式模块，每个模块围绕典型测试难题展开，先引导学生分析问题本质，再借助大模型生成初步方案，最后通过对比评估与优化迭代，加深学生对测试原理的理解，从而实现从“教工具”到“育思维”的转变。

以“动态元素定位策略”教学为例。传统教学中，教师通常会罗列 ID、Name、XPath、CSS 选择器等定位方式的语法规则，再提供几个静态页面的练习让学生熟悉用法，但学生一旦遇到动态生成的元素 ID、异步加载的内容或频繁变动的页面结构，定位脚本便容易失效，挫败感随之而来。针对该问题，教师可先呈现一个实际场景，如某电商平台商品详情页的部分元素 ID 带有随机后缀，现有脚本因无法稳定定位元素而频繁报错。教师引导学生分析问题特征，识别这是典型的动态属性导致定位失败。随后，学生向大模型描述这一场景，模型将推荐基于部分属性匹配的 XPath 策略、基于文本内容的定位方案或相对位置定位等多种可选方案，并附上各方案的适用场景说明与示例代码。学生根据模型建议进行实验验证，观察不同方案在多次运行中的稳定性差异，进而总结出动态元素的定位方法论。这一过程中，大模型充当了即时的方案提供者与代码生成助手，而学生的核心任务则从记忆语法规则转变为评估方案优劣、理解策略适用边界，高阶认知能力在“人机协同”的真实实践中得以有效发展。

2.2 借助大模型创新课堂教学模式

大模型技术的介入，为 Selenium 课堂教学模式创新开辟了新的空间，即推动传统课堂中的“教师演示-学生模仿”模式向“人机协作”转变。传统 Selenium 课堂教学中，教师通常先讲解 API 语法，再演示操作步骤，学生跟随模仿完成固定练习，整个过程以知识复现为主，学生独立思考与深度探究的机会相对有限。引入大模型后，课堂可重构为教师引导下的探究式学习场域，教师聚焦核心概念与典型难题进行精要讲解，随后设置开放性的测试设计任务，学生在任务完成过程中随时向大模型提问以获取代码建议、定位策略或错误诊断，教师则根据学生的实时反馈与模型输出进行针对性点拨。这一模式将大模型

定位为即时学习支架而非标准答案提供者，学生需要在模型给出的多种方案中进行比较评估，判断其适用性并加以调整，从而在对话与反思中完成知识内化。

以“Selenium 脚本健壮性设计”教学为例，教师可以先设置真实任务场景，如一段针对某新闻网站信息采集的 Selenium 脚本在运行中频繁因弹窗广告出现而中断、因网络延迟导致元素等待超时。教师要求学生提高该脚本的健壮性，但暂不给出具体方案。学生接到任务后，先自主分析脚本中断的可能原因，然后向大模型描述具体场景和报错信息。大模型会给出多种建议，如添加显式等待与弹窗检测逻辑、引入重试机制、设计状态检查点等，并生成相应的代码片段。学生需要对模型输出进行验证，判断哪些建议适用于当前场景，哪些可能存在性能隐患或过度设计。此时教师巡视课堂，观察学生与模型的交互过程，收集典型案例进行集中剖析，引导学生思考：“为什么同样的异常在不同业务场景中处理策略不同？”“显式等待的时间参数如何合理设定？”“重试次数如何权衡效率与可靠性？”。学生完成任务后，教师组织学生展示各自的改进方案并进行对比讨论，学生不仅可以看到多样的解决路径，也可以在辩论中加深对测试设计权衡思维的理解。整个过程以真实问题驱动，大模型承担了基础方案生成与错误解释的职能，学生则在比较、验证与质疑中完成了高阶思维训练，教师真正成为课堂学习的设计者与思维深化的促进者。

2.3 运用大模型强化项目实践训练

项目实践是 Selenium 自动化测试课程中连接知识学习与工程应用的关键环节，但传统项目训练往往陷入两种困境：或项目场景过于简化，难以复现真实测试环境的复杂性；或学生面对综合性任务时在环境搭建、脚本调试等基础环节耗费大量时间，高阶测试设计能力反而得不到充分锻炼。大模型的引入则可以将那些重复性高、认知层次低的操作转化为即时可得的辅助输出，从而将学生有限的精力聚焦于测试策略制定、用例优先级排序、缺陷分析与报告等更具专业含金量的任务，使项目实践训练的深度得以提升。

以“电商系统全链路自动化测试”项目为例。项目启动后，学生首先需要面对的是测试环境搭建与项目结构设计问题。此时学生与大模型对话，描述项目需求与技术栈，大模型可快速输出基于 TestNG 或 pytest 的项目目录结构、Maven 或 pip 依赖配置文件以及各模块的测试基类代码，学生在此基础上理解框架设计意图并加以调整。接着，学生需要将“包含优惠券叠加使用场景的订单结算流程”这样复杂的业务描述输入大模型，模型会生成涵盖正常流程、优惠券冲突、库存不足等多种情况的用例草案。学生的核心任务并非照搬这些用例，而是结合等价类划分、边界值分析等测试方法对其合理性进行评判，增补遗漏场景并删除冗余用例。最后，学生需要将完成的测试方案与报告摘要提交给大模型进行评审，模型从用例覆盖率、异常

处理完备性、断言有效性等维度提出改进建议，学生据此反思修订并撰写项目总结。整个过程中，大模型可以消解环境配置、样板代码编写等低效劳动，学生在与模型的持续互动中完成了场景分析、策略选择、方案验证等高阶认知活动，项目实践真正从简单的代码编写升华为测试工程思维的全面训练。

2.4 通过大模型优化教学评价体系

传统 Selenium 自动化测试课程的评价方式多以脚本能否成功运行、是否实现指定功能作为主要判据，这种结果导向的评价难以反映学生在测试策略设计、问题定位、方案优化等方面的思维品质。大模型技术的融入为构建过程性与发展性相结合的评价体系提供了技术可能。在评价数据采集层面，教师可要求学生提交与大模型的完整交互记录，包括每次提问的内容、模型返回的建议、学生采纳或修改的痕迹以及最终方案的迭代过程，这些数据构成了反映学生问题分析路径和决策逻辑的思维外化证据，使教师能够穿透代码表象，洞察学生真实的认知水平与进步轨迹。在评价维度设计层面，评价指标可从单一的功能实现度拓展为多维度的测试能力画像，包括场景分析能力即能否准确描述测试需求，策略设计能力即能否在模型给

出的多种方案中做出合理选择，批判性评估能力即能否指出模型生成代码的潜在缺陷，反思归纳能力即能否从具体案例中提炼出可迁移的测试方法。教师可通过分析交互记录对上述维度进行评分，从而形成对学生测试工程思维的立体评价。在评价反馈机制层面，大模型可辅助教师完成部分评价信息的整理与初步分析，例如，自动提取学生与模型交互中的高频错误类型、定位策略选择的偏好分布等，为教师提供班级整体学情的可视化洞察。

3 结语

综上所述，大模型赋能 Selenium 自动化测试课程教学改革，是推动软件测试教育从工具操作训练向工程思维培养转型的重要方法。本研究提出的对策，有望解决传统教学中学生陷入低效调试、高阶能力发展不足等突出问题，有助于构建以学生思维发展为中心的人机协同教学形态，为高校应对智能时代测试人才培养需求提供参考。未来研究需要进一步关注大模型在解决具体问题方面的应用，从而使其赋能作用更具针对性，最终促进课堂提质增效。

参考文献：

- [1] 卢慧雅,郝志卿.POM 设计模式在 Web UI 自动化测试教学中的能力培养研究[J/OL].软件导刊,1-6[2026-06-02].
- [2] 郑丽丽.基于 CC2020 胜任力模型的软件测试课程教学改革研究[J].电脑知识与技术,2025,21(20):56-59+103.
- [3] 彭玲.AI 时代高校软件测试专业实践教学测试平台选择与应用策略[J].现代商贸工业,2025,(8):60-62.
- [4] 徐宏宁,韦昌法,彭荧荧,等.新工科与工程教育认证下“软件测试技术”课程教学改革实践[J].计算机时代,2023,(12):205-208.