科研人员提出大语言模型复杂执行路径推理能力分析框架

作者：writer 来源：中国科学院

本文原地址：https://www.iikx.com/news/progress/41630.html

本文仅供学习交流之用，版权归原作者所有，请勿用于商业用途！

近日，中国科学院软件研究所等围绕动态类型程序语言中复杂执行路径难以有效分析的问题，分析大语言模型的路径约束推理能力，并构建覆盖测试生成、路径可行性判定和缺陷检测的评测基准。

执行路径由程序从入口到出口依次经过的语句和分支构成，是理解程序语义、生成高覆盖测试和发现特定路径缺陷的基础。传统符号执行通过收集路径约束，并借助SMT求解器求解。但面对Python等语言的动态类型、复杂数据结构、对象行为及大量外部接口调用时，传统符号往往难以建立准确的约束模型。大语言模型在代码生成和理解方面的进展，为不依赖SMT约束求解的路径推理提供了可能，但能否沿复杂控制流准确推理且转化为真实软件测试收益，仍缺少研究。

为评估大语言模型能否进行正确的执行路径推理，研究团队围绕三个问题展开。一是给定目标执行路径，模型能否生成输入并准确复现该路径；二是给出路径和分支条件，模型能否判断路径可达、不可达或会触发特定类型缺陷；三是在包含第三方库和复杂依赖的真实Python项目中，路径信息能否帮助模型生成覆盖度更高的单元测试。由此，团队选取14款主流大语言模型，构建出包含基准构建、实证评估的大模型复杂执行路径推理能力分析框架。同时，团队提出了基于插桩和控制流图最小覆盖遍历的路径提取方法，用于数据集构建，并通过实际执行、覆盖率统计和人工标注检验模型输出。

实验显示，大语言模型已初步具备复杂路径约束推理能力。在平均包含233条语句、103个分支的测试生成任务中，推理模型o4-mini达到最高的65.6%路径复现率，表明旗舰推理模型已拥有较强的路径推理能力。在路径分类与缺陷检测任务中，Qwen3-235B识别出全部120个含除零缺陷的程序；与之相反，推理模型在这一任务上没有优势，说明过长推理思维链可能推翻模型正确判断。在真实软件仓库中，路径信息可提升大部分模型的总体行覆盖率，但第三方库和接口的理解能力以及测试的可执行性仍是将路径推理能力转化为工程收益的主要瓶颈。

研究表明，大语言模型现阶段更适合作为符号执行的互补式启发工具。模型可利用代码语义和外部接口知识提供候选输入、缺陷线索和测试方向；传统程序分析负责路径提取、约束校验和结果验证。二者结合，有望形成面向真实软件系统的混合式路径推理与智能测试方法。

相关论文被软件工程领域国际期刊TOSEM 2026录用。

论文链接

大语言模型执行路径约束求解与测试流程

研究团队单位：软件研究所

更多 科学进展 请访问 https://www.iikx.com/news/progress/

本文版权归原作者所有，请勿用于商业用途，爱科学iikx.com转发