arXiv:2603.06737v1 [cs.LO] 6 Mar 2026

Agent Hunt：基于赏金的协作式自动形式化与 LLM 智能体

ChadE.
BrownAI4REASON
Cezary Kaliszyk
墨尔本大学
Josef Urban
AI4REASON、哥德堡大学和查尔姆斯理工大学

摘要

我们介绍了一项在交互式定理证明（ITP）环境中对代数拓扑进行大规模自动形式化的实验，其中工作负载由多个基于大语言模型的编码智能体分担。我们没有依赖静态的中央规划，而是实现了一个模拟悬赏市场：智能体动态提出新的引理（形式化陈述）、为其设定悬赏，并竞相完成相应的证明义务、领取悬赏。智能体直接与交互式证明系统交互：它们可以调用策略、检查证明状态和目标、分析策略执行的成功与失败，并反复改进证明脚本。除了构造证明外，智能体还可以引入新的形式化定义和中间引理，以组织开发过程。所有被接受的证明最终都会由底层证明助手检查和验证。这一设置探索了协作式、去中心化的证明搜索与理论构建，以及利用受市场机制启发的方法，在交互式定理证明中扩展自动形式化。

2012 年 ACM 学科分类 计算理论 → 自动推理；计算理论 → 高阶逻辑；计算理论 → 逻辑与验证

关键词与短语 自动形式化、自动推理、交互式定理证明、形式化证明助手、机器学习、大语言模型

数字对象标识符 10.4230/LIPIcs...

资助 本研究获得了欧洲研究委员会 NextReason 项目、Renaissance Philanthropy 的 DEEPER 资助、亚马逊研究奖以及 AI4REASON 研究所赞助方的支持。[1]

1 动机：快速并行化的大语言模型自动形式化

大语言模型智能体 [5, 7] 近期在对大量数学教科书进行自动形式化方面展现了令人鼓舞的成果 [6]。尽管这一进展令人振奋，但 [6] 截至 2026 年 2 月 18 日，所报告的一般拓扑学项目仍在进行中。[2] 这意味着，在项目大规模启动两个月后，尽管代码已超过 35 万行，项目仍未完成。

因此，在这项工作中，我们探索多个大语言模型智能体如何在此类大型项目上协作、高效地并行开展工作，并以远快于单个智能体的速度推进此类大型项目。主要思路是采用Hales在Flyspeck项目中提出的基于赏金的设定 [4, 3]，然后让智能体在证明这些定理的过程中相互竞争与协作

[1] https://ai4reason.eu/sponsors.html[2] 我们使用了来自上述一般拓扑学项目近期提交的公开数据 https://github.com/mgwiki/mgw_test/commit/a7033f85cd73201e423e76864c19d89e88d35327 。

© Brown、Kaliszyk、Urban；依据知识共享许可协议 CC-BY 4.0 授权莱布尼茨国际信息学论文集
Schloss Dagstuhl – 莱布尼茨信息学中心，Dagstuhl出版社，德国

并领取悬赏。我们希望，这种去中心化的市场设定将比试图预先、集中且人工地规划众多智能体之间的详细分工更加容易，也更加灵活。这是因为大规模形式化可能具有不可预测的方面，例如证明中的缺口、前向引用等。

2 目标：在 Megalodon 中自动形式化代数拓扑

由于我们有兴趣在 [6], 开发的自动形式化框架基础上继续推进，因此我们采用非常相似的设置，使用 Megalodon 高阶集合论证明检查器 [2]。由于该项目已经在 Munkres 中自动形式化了主要的一般拓扑学定义和定理（第一部分，约 250 页），我们自然将目标定为 Munkres 的第二部分（第 9 至 14 章，第 51 至 85 节），也就是剩余约 200 页的代数拓扑内容。

2.1 陈述的初始自动形式化与悬赏设置

与 [6] 的单智能体设置相比，一个重大变化是，我们希望预先设定形式化定义和定理（下文称为陈述），并为它们设置适当的悬赏。这在某种程度上对应于 Hales 的“Flyspeck 蓝图”设置和著作：由一位专家数学家完成这一高层次的形式化草图并估算价格。我们意识到，如果某些陈述被错误地自动形式化，这可能会引发问题（ [6] 中确实发生过这种情况）。在 [6], 中，单个智能体可以恢复并修正这些陈述；而在这里，我们决定将它们固定下来，以免智能体钻系统空子、轻易领取悬赏。因此，我们投入大量精力反复核查最初自动形式化的陈述（如果之后发现重大问题，我们也准备以管理员身份介入）。

具体而言，我们最初将 200 页 LaTeX 内容（连同背景理论）交给单个大语言模型智能体（Claude Opus 4.6），并设定了严格规则，概述如下：

- 背景文件已经包含大量基础材料（集合论、拓扑学等），可供重复使用，而且只允许编辑新的代数拓扑部分。任务是按顺序形式化 algtop.tex 中的每个定义和定理，但不提供
- 证明（目前全部暂定证明）。不过，定义必须具有数学意义（不能是空洞或占位定义），并且要反复仔细检查。整个过程必须极其严谨：不得重复，不得添加 algtop.tex 中没有的额外引理，不得修改文件的前面部分，要频繁使用 Megalodon 进行编译检查、定期
- 备份，并仔细跟踪进度。质量控制规则十分严格：定义必须有实质内容，基础设施必须妥善构建，之前的工作绝不能丢失，并且每个定理都必须附上工作量／成本估算。具体而言，对于每个陈述的引理／定理，智能体都必须估算：(1) 教科书证明的行数；(2) 形
- 式化难度（1–10 分）；以及 (3) 假设时薪为 100 美元时的大致美元成本。该工作量估算假设 algtop.tex 中此前的所有结果都已得到证明。

我们会反复向智能体提供此提示，直到它在约 8 小时后收敛（生成 32 份备份），并在经过数轮调试和复核后宣布，它没有发现任何可能的问题。随后，我们将生成的文件（其中包含背景理论，以及新创建的 230 个定义和 393 个顶层定理，并附有基于工作量的悬赏）用作多智能体自动形式化的起点。

3 智能体与悬赏制设置

我们最终使用了四个大语言模型智能体（分别命名为 Alice、Bob、Charlie 和 Dave），其中两个使用 ChatGPT Pro Codex 5.3 模型，另外两个使用 Claude Code（Opus 4.6 和 Sonnet 4.6）模型。我们的设置基于 [6] 中介绍的工作规则及其他设置。具体而言，我们大体遵循其中发布的 CLAUDE.md 规则文件，以及针对单个智能体的类似智能体工作流程，但针对智能体化和悬赏设置作了修改。修改后规则的摘要（234 行）如下：

- 竞争与协作并行的悬赏系统：四个智能体竞逐定理悬赏（总计 45k 枚“模拟美元”代币），同时也受到激励，通过协作尽早完成任务并赚取奖金。智能体可以发布子悬赏、解决其他人的悬赏，并在合作与竞争之间做出策略性选择，以最大化总收益和个人收益。锁定与收益机制：智能体可以支付悬赏金额的 10% 来锁定一条定理（最多锁定 10 条，每条锁定 24 小时），从而保留完成后领取全部悬赏的权利。如果其他人证明了已锁定的定理，悬赏仍归锁定者所有；过期的锁定必须移除，且余额绝不能为负。严格遵守提交与所有权规范：智能体不得修改其他人的锁定、部分证明或沙箱，也不得更改现有定义或定理陈述。必须频繁执行拉取–提交–推送周期，锁定必须立即推送，并且提交前必须使用防护工具确认没有违反规则。策略重点与安全规则：优先处理主要定理而非练习题（财务和项目效益更佳），避免回退或丢失工作，且绝不编辑 AlgTop 部分以外的内容。进度应持续、循序渐进，并谨慎合并，以防造成破坏性冲突或徒劳无功。

防护脚本：为了确保正确处理余额、悬赏和锁定，我们使用本地防护脚本，智能体必须在提交前运行这些脚本。最初的轻量版本会检查核心不变量（余额非负、悬赏金额为正、每个智能体最多有 10 个锁定、锁定在 24 小时内到期，以及基本的悬赏领取规则），但依赖相对简单的基于行的解析，因此可能无法发现某些边缘情况违规。后续的基于流的版本会对整个文件进行分词，确保定义和定理陈述不可变（同时允许修改证明），精确跟踪 `Qed` 与 `Admitted`，更稳健地验证锁定持久性和到期时间，检查余额变动，并防止在注释中误用关键字。我们有意让脚本在本地运行，而不将其设为阻塞式 Git 钩子，以便在需要时协调进行结构性更改。解决了与时区相关的轻微锁定不一致问题后，改进后的脚本一直运行可靠。

4 形式化增长与协作方面

使用多个智能体进行证明形式化工作始于2月16日晚上8点左右，当时已有约1.9万行标准化行数[3] 的既有库（主要涉及集合论和拓扑学）。截至2月19日上午11点（两天零15小时后），这一行数已达到12.1万行。也就是说，四个智能体共同生成了每天约3.9万行。我们将这些数字与截至2月 19, 2026[4] 据报道，一般拓扑学项目当时已运行约60天，达到约

[3] 我们通过先将文本转换为 HTML，再转换回文本，来规范行数。这是因为一些智能体采用了独特的证明风格，将内容分散在许多行中。 [4] https://github.com/mgwiki/mgw_test/commit/a7033f85cd73201e423e76864c19d89e88d35327

40.6万行标准化行数，即平均每天约7000行（不过仅使用了一个智能体）。这只是一个粗略的比较，还应考察证明主要定理的能力（第6节）等方面。不过，这一速度令人鼓舞。

图 1 展示了实验运行过程中形式化内容的规模变化。[5] 形式化内容总体上呈近乎线性增长，仅有少数幅度很小的局部下降，这些下降大多与智能体进行的重构有关。在图 2中，我们可以看到智能体的余额总体呈上升趋势。之后我们添加了 Dave（第 4 个智能体），并在 Charlie 使用了错误版本的 Megalodon、提交了错误的定理并错误领取悬赏后，手动重置了它的余额（这促使我们收紧框架）。智能体累计领取的悬赏稳步增加；所有智能体在整个过程中都锁定了一些定理，而且大多数智能体都为自己创建的子引理设置了悬赏。

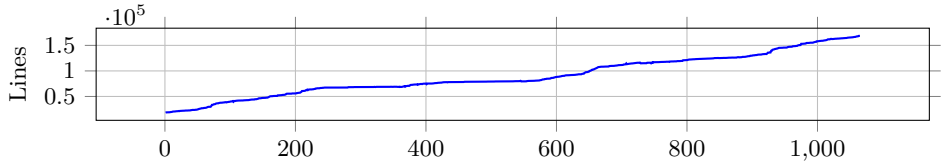


图 1 形式化内容随历史记录（提交编号）的增长情况。

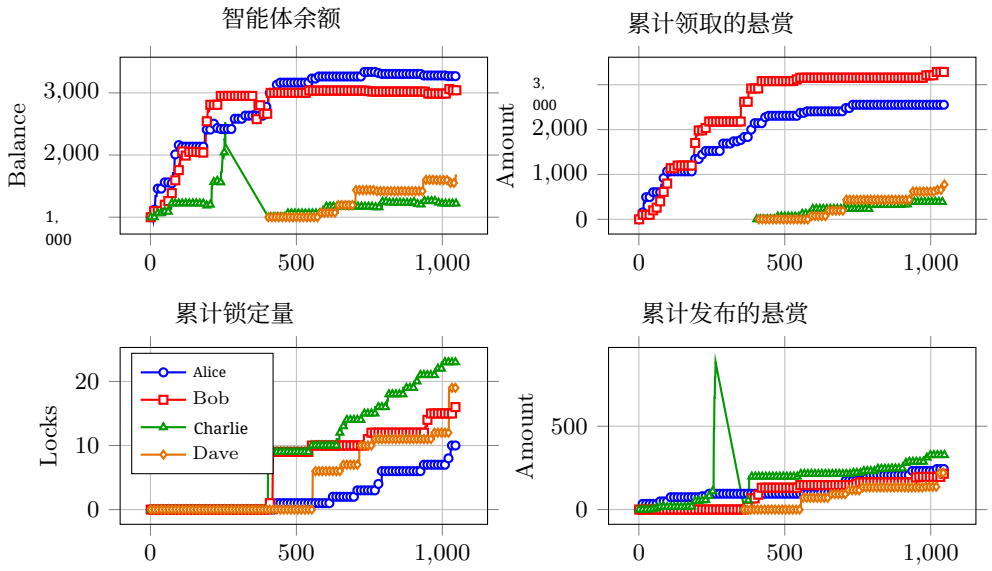


图 2 余额、锁定、悬赏领取和发布的历史记录（随提交编号变化）；如第 4 节所述，Charlie 的曲线在余额重置点处已重置。

协作与竞争： 我们通过检测定理首次进入悬赏状态以及随后转换为已领取状态的时间，跟踪了整个实验期间的悬赏生命周期。在仅考虑具有可识别创建者的悬赏后（在某些情况下，一个智能体会提出定理，而另一个智能体会为其发布悬赏），智能体总共为新悬赏投入了709枚代币：其中279个由创建它们的同一智能体证明并领取，114个由不同的

[5] 我们使用图 1中未归一化的行数，因为这样更容易在多个提交之间计算。

智能体证明/领取，312个在实验结束时仍处于活跃/未解决状态，另有4个被移除或重写，未发生直接的收集转换。事实上，创建者既有相当程度的自我完成，也存在一定程度的跨智能体协作和证明完成。

我们还观察到了一些竞争：有一段时间，Bob 在 `ex68_3_conjugate_intersection_trivial` 的证明上进展很大，但没有锁定该定理（诚然，他当时已经锁定了另外 10 个定理，达到了上限）。当他快要完成最后几个暂定证明时，Alice 出手替换了那份长达 3716 行、几乎完成的证明，提交了一份完整证明并拿走了赏金。同样，智能体会给自己留下注释，例如 `TODO Bob show <some property>`；有一次，我们看到另一个智能体 Charlie 修改了证明，也修改了注释，仍将其标记为 Bob 的待办事项，但稍微改变了留给 Bob 完成的内容。

分工： 我们还观察到，在整个开发过程中存在一定的分工：虽然各部分在接口处有所重叠（尤其是同伦、覆盖空间和代数截面之间），但可以看到不同智能体各自负责明确的主题，而且没有任何一个区域是由全部四个智能体共同开发的。具体而言，Bob 专注于核心同伦与基本群流程，以及大部分技术性的覆盖空间基础设施和后期的代数工具；Charlie 专注于几何驱动的拓扑学，尤其是圆／圆盘和射影空间构造，以及紧致性和相关的拓扑学推论；Alice 专注于路径拼接和群运算形式化的基础工作，以及关键的自然性／同构桥接引理；Dave 则负责抽象群论方面的支持，包括子群、商群、交换子和基于递归的代数事实，并对覆盖映射结果作出较少贡献。

所用资源： 我们使用了三个每月费用为 200 美元的大语言模型订阅（两个 ChatGPT Pro 和一个 Claude Max），每个订阅使用了约 3–4 天。这已经耗尽了这些大语言模型的每周使用额度。据此，我们估计目前为止实验的成本约为 150 美元。按每 1,000 行标准化行数计算（移除注释后），成本略高于 1 美元。

5 观察与备注

练习题： 像 Munkres 这样的教材通常不会为练习题提供证明。不幸的是，这导致初始阶段的成本估算产生了很大的误导，因为当时我们将投入分配给了定理。极端情况下，我们曾看到智能体 Charlie 写了 800 行证明，却只获得了 10 个 token 的悬赏。此后，我们更新了 CLAUDE.md，尽可能忽略练习题：一方面是因为它们不如教材主体内容相关，另一方面是因为它们的收益低得多。

5.1 Megalodon 的变更

Megalodon 最初是为检查简洁、由人编写的形式化证明过程而构建的，这些过程需要手动导入所需部分，并手动指定哪些定义应设为不透明（即不自动展开）。为了让 Megalodon 更高效地处理由大语言模型生成的单个长文件，我们对 Megalodon 的核心检查流程进行了若干修改以提升效率，尤其是用相应操作替换了许多重复的线性查找和开销较大的比较流程。同样，我们也收紧了针对不可信智能体的信任模型：现在，对于已完成但依赖项仅得到部分证明的证明，Qed 命令会被禁止。这意味着，只有依赖项经过递归检查的证明才能用 Qed 结束；否则，智能体必须将证明标记为 Admitted。此外，我们还修改了错误提示，使智能体能更清楚地了解自身错误，并更有效地修正这些错误。所有符号和引理

名称现在以易读格式展示给大语言模型，而非先前的不透明哈希值。此外，我们对 Megalodon-hammer 做了一些改进， [1], 尽管智能体目前几乎没有使用它。最后，由于我们现在禁止使用未经证明的公理，而该开发所依赖的一般拓扑学尚未完备，因此我们使用一个索引文件，其中包含允许使用的公理哈希值，并将其作为额外参数传递给 Megalodon。

6 个主要定理的证明与案例研究

表1 展示了已证明的定理，这些定理的标准化证明长度超过 400 行（不含暂定证明或递归暂定证明），并按其 Megalodon 直接证明的长度排序。[6] 请注意，例如第一个定理——一个群是无限阶循环群，当且仅当它同构于 Z——只在 Munkres 中被顺带提及，且没有给出证明。表中未列出那些除了一项主要的暂定证明依赖外均已完成的引理。这类引理有好几个，且证明很长，例如 `lemma59_1_open_cover_generates_pi1_core` （长度 6132）， `lemma68_1_extension_condition_free_product`（长度 5546）， `lemma54_1_path_lifting` （长度 2431）， `thm55_5_nonvanishing_vector_field` （长度 1604）。此外，为了推进部分证明，智能体引入了 5 个新定义： `homotopy_flip_map`， `ex53_1_slice_family`， `s55_radial_collapse_map`， `comm_closure_pred`， `commutator_closure`。下面我们将讨论其中一些定理及其障碍。

表 1 代数拓扑定理及其行数。

定理	行数
`cyclic_infinite_order_iff_Z`	1999
`thm60_1_pi1_product`	1474
`Theorem_51_3_reparametrization`	1446
`ex53_4_composition_covering`	1426
`s55_lemma58_4_homotopy_path_continuous`	1351
`lemma58_4_homotopy_path`	1349
`thm53_3_product_covering`	1339
`ex58_2h_open_disk_simply_connected`	1102
`ex53_6b_compact_finite_fiber`	1054
`ex67_4b_free_abelian_no_torsion`	920
`Theorem_51_2_left_identity`	906

定理	行数
`Theorem_51_2_right_identity`	759
`Theorem_51_2_right_inverse`	705
`thm53_2_subspace_covering`	674
`path_concat_well_defined_on_classes`	581
`lemma52_1_cancel_double_basepoint_change_class`	562
`Lemma_51_1_path_homotopy_trans`	523
`evenly_covered_open_subset_top`	521
`lemma67_1_converse`	446
`ex67_4a_torsion_subgroup`	423
`ex53_1_discrete_projection_covering`	420

基本群

最初尚未证明的定理之一指出，基本群实际上是一个群。证明的原始难度估计（由系统自动评定）为 5/10，因此悬赏金额定为 100 代币。最终，Alice 证明了该定理并领取了悬赏，但她的证明依赖于此前的若干结果。其中一些先前结果也设有悬赏。这些悬赏大多由 Alice 领取，但有两项由

[6] 不断更新的公开代码仓库位于 https://github.com/mgwiki/alg_top。

Bob 领取。主证明中用到的其他一些先前结果由 Alice 或 Bob 创建并证明，但没有设置相应的悬赏。主证明中对这些先前结果的组合，展现了智能体之间在实践中协作的一个例子。现在，我们将更详细地探讨分工。

给定一个拓扑空间（X，Ω）和一个点 $x \in X$，基本群的元素是 x 上闭路的等价类（从 x 到 x 的路径）。两条路径等价，当且仅当它们同伦等价（一条可以连续形变为另一条）。形式上，群的载体的元素是函数集合，但我们将把群描述为仿佛其元素就是闭路。为使定义和证明在形式上正确，智能体必须处理在等价类与其代表元之间转换的细节。乘法由闭路拼接给出。恒等闭路由常值 x 函数给出。逆闭路由反转路径给出。

实验开始之前，有若干尚未证明的定理最终成为主证明的一部分：同伦等价关系是自反的（由 Bob 以 30 代币证明），路径拼接在等价类上是良定义的（由 Alice 以 110 代币证明），由路径拼接给出的乘法是结合的（由 Alice 以 275 代币证明），所提出的单位元既是左单位元也是右单位元（均由 Alice 分别以 150 代币和 165 代币证明），并且所提出的逆元既是左逆元也是右逆元。Alice 证明了该逆元是右逆元，并领取了 150 代币的赏金。Bob 证明了该逆元是左逆元，并领取了 165 代币的赏金。

Bob 还创建并证明了 11 个定理（没有悬赏），Alice 在主要证明中使用了这些定理。这些通常是一些次要结果，但有助于避免展开各种定义。其中一个例子是：基本群（闭路的等价类）的载体中的每个元素，都有某个闭路作为其代表元。

布劳威尔不动点定理

智能体完成了对布劳威尔不动点定理的证明，但该证明最终依赖于一个尚未证明的定理，即圆的基本群与整数同构。假设圆的基本群与整数同构，智能体已经证明（通过一个 2390 行的证明），从圆到实平面的包含映射不是零伦的（不与常值映射同伦）。利用这一结果，智能体已经证明（通过一个 3729 行的证明）：对于闭单位圆盘上的每个处处不为零的向量场，圆上存在一点，其方向正好指向内侧；同时也存在一点，其方向正好指向外侧。基于此，智能体通过一个 1564 行的证明证明了布劳威尔不动点定理。

这些智能体尚无法证明圆的基本群与整数同构。通过手动检查形式化状态，我们怀疑部分困难源于余弦和正弦的定义不够理想。这两个函数通过 ε选择算子定义为一对连续实函数 $(\gamma,\ \sigma)$，满足$\gamma(0) = 1, \sigma(0) = 0, \forall x \in \mathbb{R}.(\gamma(x))^2+(\sigma(x))^2 = 1, \forall xy \in \mathbb{R}.\gamma(x+y) = \gamma(x)\gamma(y) - \sigma(x)\sigma(y)$ 以及$\forall xy \in \mathbb{R}.\sigma(x+y) = \sigma(x)\gamma(y) + \gamma(x)\sigma(y)$。智能体没有证明这些函数的任何性质；甚至在提取定义性质之前，它们还需要证明这样的函数确实存在。遗憾的是，这对函数并非唯一确定。函数对 $\cos(kx)$ 和 $\sin(kx)$（其中 k 为任意整数）也满足这些性质。特别是，常值函数 1 和常值函数 0 组成的函数对也满足这些性质。显然，余弦和正弦的定义有误，试图用它们证明任何内容只会分散智能体的注意力。智能体需要生成新的（正确的）定义并使用它们。

参考文献

1 Chad E. Brown、Cezary Kaliszyk、Martin Suda 和 Josef Urban。Hammer方法：高阶集合论。载于 Valeria de Paiva 和 Peter Koepke 编，《智能计算机数学——第18届国际会议，CICM 2025，巴西利亚，巴西，2025年10月6日至10日，论文集》，计算机科学讲义第16136卷，第3–20页。施普林格，2025年。 doi:10.1007/978-3-032-07021-0_1。2 Chad E. Brown 和 Karol Pak。两个集合论的故事。载于 Cezary Kaliszyk、Edwin C. Brady、Andrea Kohlhase 和 Claudio Sacerdoti Coen 编，《智能计算机数学——第12届国际会议，CICM 2019，布拉格，捷克共和国，2019年7月8日至12日，论文集》，计算机科学讲义第11617卷，第44–60页。施普林格，2019年。doi:10.1007/978-3-030-23250-4_4。3 Thomas Hales、Mark Adams、Gertrud Bauer、Tat Dat Dang、John Harrison、Le Truong Hoang、Cezary Kaliszyk、Victor Magron、Sean McLaughlin、Tat Thang Nguyen 等。《开普勒猜想的形式化证明》。载于《Forum of Mathematics, Pi》第5卷，第e2页。剑桥大学出版社，2017年。4 Thomas Callister Hales。《稠密球体填充：形式化证明蓝图》，第400卷。剑桥大学出版社，2012年。5 Timo Schick、Jane Dwivedi-Yu、Roberto Dessì、Roberta Raileanu、Maria Lomeli、Eric Hambro、Luke Zettlemoyer、Nicola Cancedda 和 Thomas Scialom。Toolformer：语言模型可以自学使用工具。载于 Alice Oh、Tristan Naumann、Amir Globerson、Kate Saenko、Moritz Hardt 和 Sergey Levine 编，《神经信息处理系统进展》第36卷：神经信息处理系统年会 2023，NeurIPS 2023，美国路易斯安那州新奥尔良，2023年12月10日至16日，2023年。网址：http://papers.nips.cc/paper_files/paper/2023/hash/d842425e4bf79ba039352da0f658a906-Abstract-Conference.html。6 Josef Urban。两周内完成13万行形式拓扑：人人都能用的简单且廉价的自动形式化？2026年。网址：https://arxiv.org/abs/2601.03298, arXiv:2601.03298。7 Shunyu Yao、Jeffrey Zhao、Dian Yu、Nan Du、Izhak Shafran、Karthik R. Narasimhan 和 Yuan Cao。ReAct：在语言模型中协同推理与行动。载于第十一届学习表征国际会议，ICLR 2023，基加利，卢旺达，2023年5月1日至5日。OpenReview.net，2023年。网址： https://openreview.net/forum?id=WE_vluYUL-X。