Anthropic.

研究 • RESEARCH

研究洞察

当 AI 构建自身

Claude 正在帮助 Anthropic 构建下一代 Claude——这究竟意味着什么?

Anthropic 研究团队 • 原文：anthropic.com

在 Anthropic，我们正在见证一件前所未有的事：我们用来构建 Claude 的工具，越来越多地由 Claude 自己编写。AI 正在参与构建下一代 AI——这一现实既令人振奋，也引发了深刻的思考。

这不是科幻小说中的情节，而是我们日常工程工作的真实写照。从代码审查到测试用例生成，从基础设施脚本到模型评估框架，Claude 的"数字指纹"已经遍布我们的开发流程。

代码产出的跃升

过去一年，Anthropic 工程师借助 Claude 完成的代码量出现了显著增长。这不仅仅是"自动补全"层面的辅助——Claude 能够理解复杂的系统架构，提出设计方案，并独立完成相当规模的编程任务。

数据显示，在某些关键开发领域，AI 辅助生成的代码占比已经超过了人工直接编写的代码。更重要的是，这些代码的质量经过严格评审，达到了生产级别的标准。如今，Anthropic 工程师平均每个季度的代码产出量是此前的 8 倍。

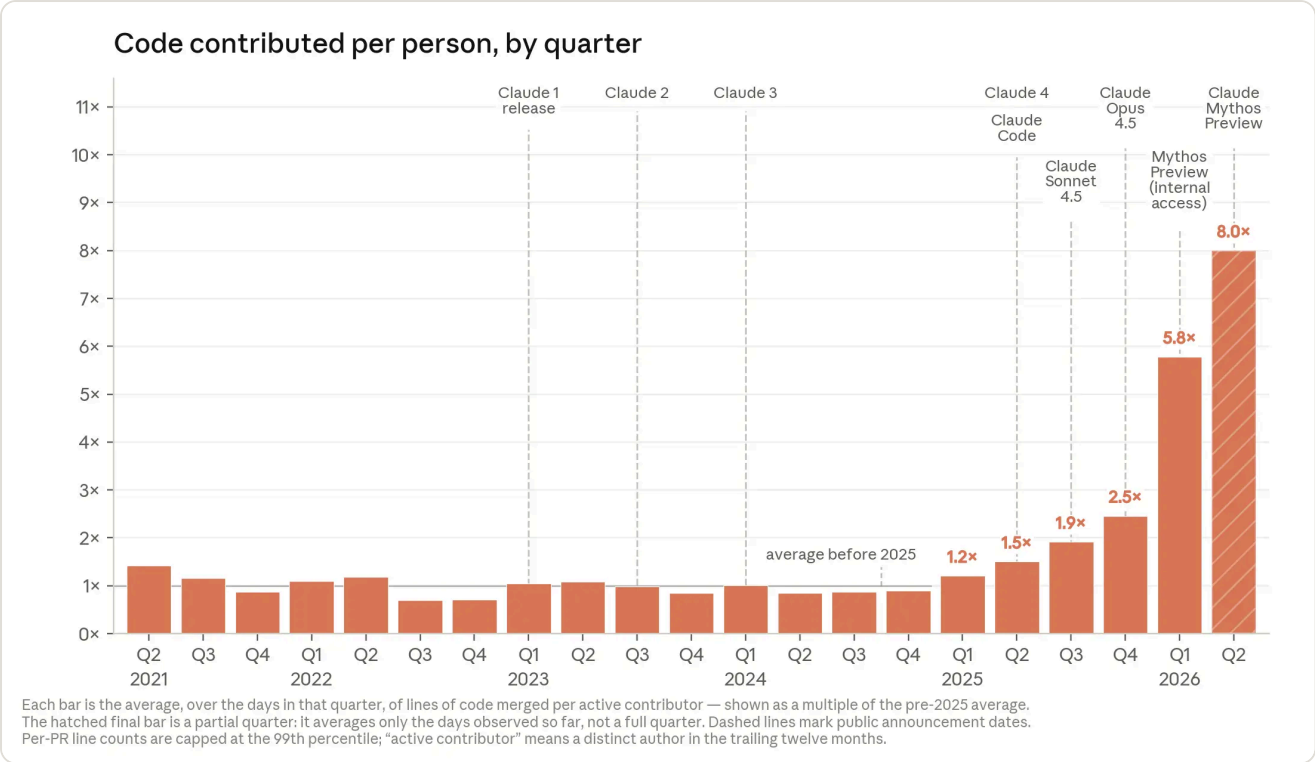

图 1：Anthropic 工程团队中，AI 辅助代码产出量的增长趋势。纵轴为相对代码量，横轴为时间。数据显示，在关键开发周期内，AI 生成代码的占比持续攀升。

这一趋势背后，是 Claude 在理解工程意图方面的持续进步。当工程师描述一个功能需求时，Claude 不仅能生成符合要求的代码，还能主动识别潜在的边界情况、提出更健壮的实现方案，甚至指出与现有代码库的兼容性问题。

Claude Code：自主工程的新前沿

Claude Code 是 Anthropic 推出的一款面向开发者的 AI 编程工具，它代表了我们在"自主工程"方向上的重要探索。与传统的代码补全工具不同，Claude Code 能够在更长的时间跨度内维持对复杂任务的理解，并以接近人类工程师的方式规划和执行多步骤的编程工作。

在内部测试中，我们将 Claude Code 应用于真实的工程任务——包括修复 bug、重构代码、编写测试，以及开发新功能模块。结果令人印象深刻：在许多标准化的工程基准测试上，Claude Code 的成功率已经达到了相当高的水平。

"AI 正在参与构建下一代 AI——这一现实既令人振奋，也引发了深刻的思考。"

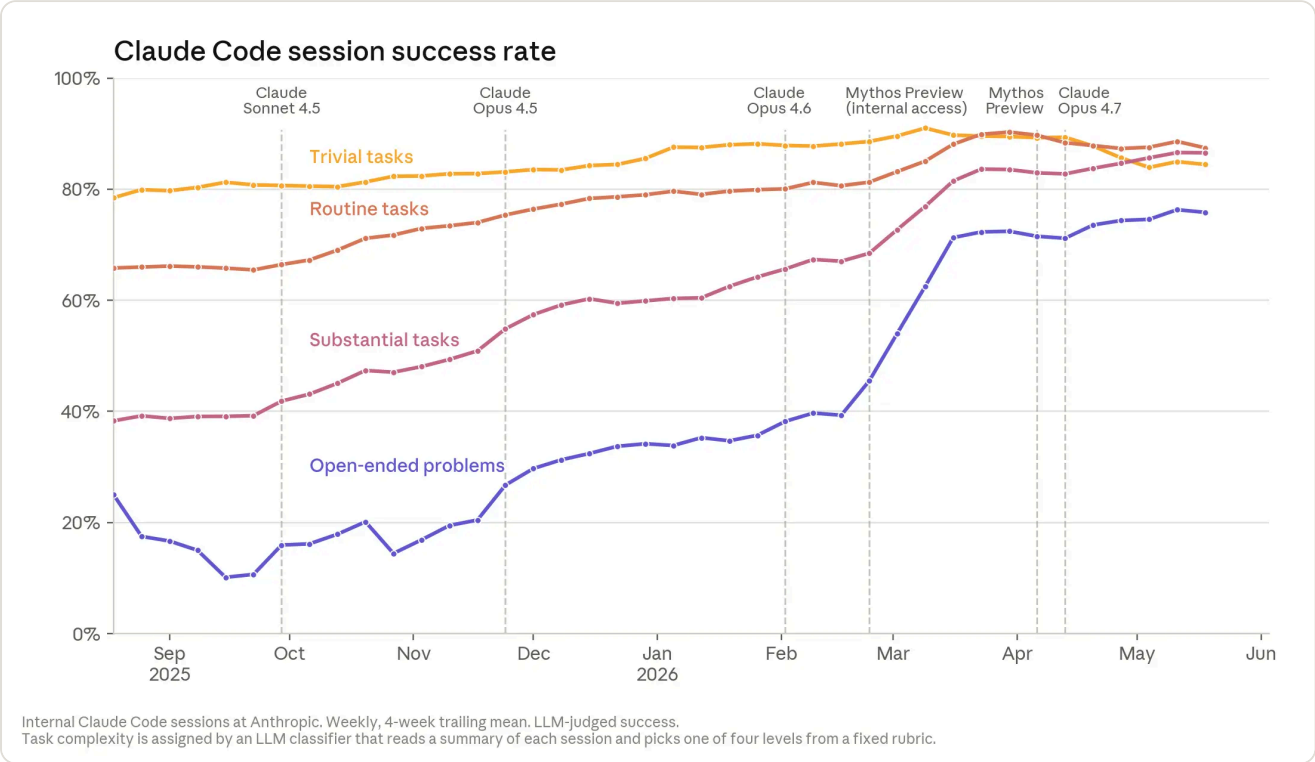

图 2：Claude Code 在不同类型工程任务上的成功率。数据涵盖 bug 修复、功能开发、代码重构、测试编写等多个维度，与基线模型和人类工程师的表现进行对比。

值得注意的是，Claude Code 的优势不仅体现在"能做什么"上，更体现在"如何做"上。它倾向于先理解问题的全貌，再动手编写代码；遇到不确定的地方，会主动提问而不是盲目猜测；完成任务后，还会自我检查并解释关键的设计决策。这种工作方式，与优秀的人类工程师高度相似。

超越代码：模型决策的参与

AI 参与 AI 开发，并不仅限于编写代码。在 Anthropic，Claude 还被用于更深层的研究工作：帮助分析实验结果、评估模型行为、识别训练数据中的潜在问题，甚至参与关于模型设计方向的讨论。

这引出了一个微妙但重要的问题：当 AI 参与决定自身"继任者"的设计时，我们如何确保这一过程的透明度和可控性？

Anthropic 的答案是：建立严格的人类监督机制，并持续研究 AI 系统的决策过程。我们相信，只有深入理解 AI 如何思考和决策，才能在赋予它更多自主权的同时，确保它的行为符合人类的价值观和利益。

图 3：不同版本模型在复杂推理与决策任务上的能力对比。图表展示了模型在需要多步推理、权衡取舍和不确定性处理等场景下的表现差异，以及随迭代版本的能力演进轨迹。

上图中的数据揭示了一个关键趋势：随着模型能力的提升，AI 在复杂决策场景中的表现正在逼近——在某些维度上甚至超越——人类专家的水平。这既是技术进步的体现，也是我们需要保持高度警惕的信号。

三种未来图景

当我们展望 AI 深度参与自身开发的未来，至少存在三种截然不同的可能性。理解这些可能性，有助于我们做出更明智的选择。

> **情景一：良性加速**
>
> AI 辅助开发形成正向循环：更强的 AI 帮助构建更好的工具，更好的工具加速 AI 研究，研究成果反哺工具改进。在这一情景中，人类工程师从重复性工作中解放出来，专注于创造性问题和价值判断，AI 则承担越来越多的执行层工作。技术进步的速度大幅提升，但人类始终掌握方向盘。

> **情景二：能力鸿沟**
>
> 随着 AI 系统变得越来越复杂，人类工程师逐渐难以完全理解 AI 生成的代码和设计决策。这并非因为 AI 有意隐瞒，而是因为复杂系统本身的不透明性。在这一情景中，人类监督的有效性可能下降，我们需要开发新的工具和方法来维持对 AI 行为的真正理解和控制。

> **情景三：价值对齐的考验**
>
> 当 AI 参与设计下一代 AI 时，它的价值观、偏见和盲点可能会被放大和传递。即使每一步都经过人类审查，累积的偏差也可能在长期内产生显著影响。这一情景要求我们在 AI 参与开发的每个环节，都保持对价值对齐问题的高度敏感。

Anthropic 的立场是：我们不应该因为担忧后两种情景而放弃 AI 辅助开发的巨大潜力，但我们必须以负责任的方式推进——建立完善的监督机制，投入大量资源研究 AI 安全，并对公众保持透明。

透明度的承诺

发布这篇文章，本身就是我们透明度承诺的一部分。我们认为，公众有权了解 AI 系统是如何被开发出来的，包括 AI 在这一过程中扮演的角色。

我们也希望这篇文章能够引发更广泛的讨论：社会应该如何看待 AI 参与 AI 开发这一现象？哪些边界是不应逾越的？哪些监督机制是必要的？这些问题没有简单的答案，但它们值得我们认真对待。

在 Anthropic，我们每天都在与这些问题正面交锋。我们相信，只有直面这些挑战，才能真正构建出对人类有益的 AI。

注释与参考

1. 本文中的数据和图表来自 Anthropic 内部研究，部分数据经过归一化处理以保护商业敏感信息。
2. "代码产出"的统计口径包括：由 AI 直接生成并经人类审查后合并的代码，以及由 AI 提供关键逻辑、人类进行修改完善的代码。纯粹的代码补全（单行或少量行）不计入统计。
3. Claude Code 的成功率数据基于内部工程基准测试集，该测试集涵盖了 Anthropic 实际工程工作中的典型任务类型，并经过独立团队的盲评验证。
4. 关于 AI 参与 AI 开发的伦理和安全问题，Anthropic 已发布多篇研究论文，详见 anthropic.com/research。
5. 本文为 Anthropic 原文的中文编译版本，部分内容经过适当调整以符合中文读者的阅读习惯。如有出入，以英文原文为准。

Anthropic · 本文为非官方中文译文，译者：Ren@FakeMaidenMaker。原文：anthropic.com/institute/recursive-self-improvement

译者：Ren@FakeMaidenMaker