DeepSeek-R1：通过强化学习激励 LLM 的推理能力

DeepSeek-AI

research@deepseek.com

摘要

我们介绍了第一代推理模型DeepSeek-R1-Zero和DeepSeek-R1。DeepSeek-R1-Zero是一个通过大规模强化学习（RL）训练出来的模型，没有经过超级微调（SFT）这一初步步骤，但却展示了非凡的推理能力。通过强化学习，DeepSeek-R1-Zero 自然而然地出现了许多强大而有趣的推理行为。然而，它也遇到了可读性差和语言混杂等挑战。为了解决这些问题并进一步提高推理性能，我们引入了DeepSeek-R1，它在RL之前结合了多阶段训练和冷启动数据。DeepSeek- R1在推理任务上取得了与OpenAI-o1-1217相当的性能。为了支持研究社区，我们开源了DeepSeek-R1-Zero、DeepSeek-R1以及基于Qwen和Llama从DeepSeek-R1中提炼出的六个密集模型（1.5B、7B、8B、14B、32B、70B）。

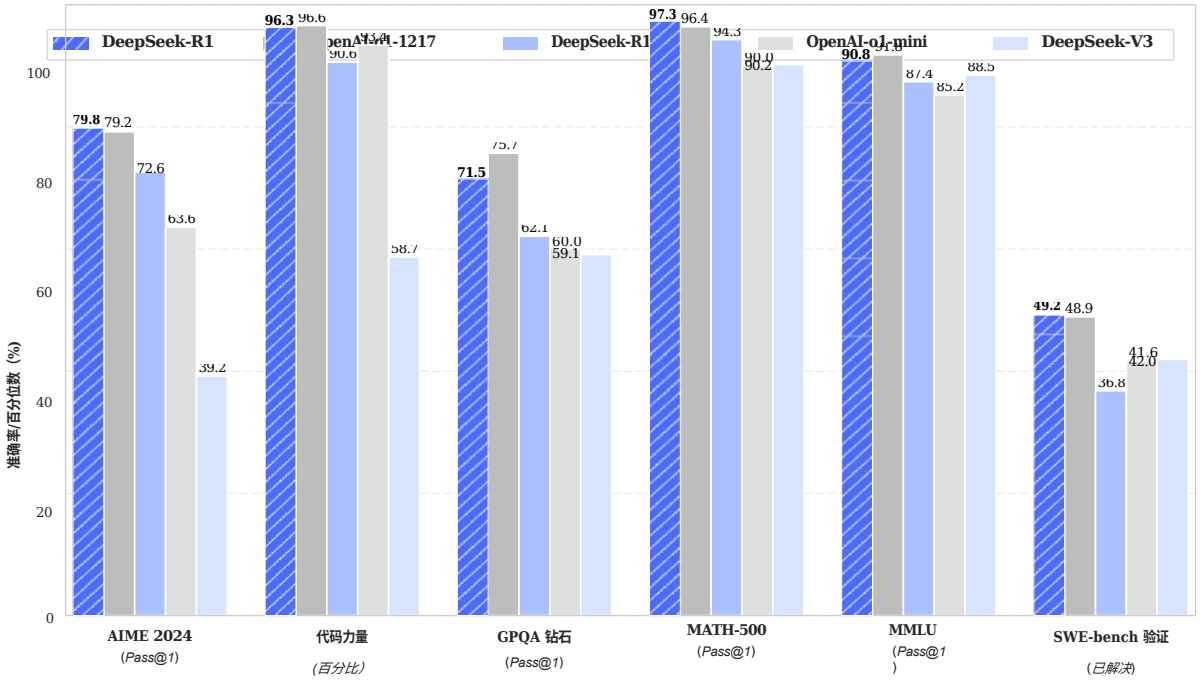

图 1| DeepSeek-R1 的基准性能。

目录

1. 导言

近年来，大型语言模型（LLM）经历了快速的迭代和演变（Anthropic，2024；Google，2024；2024OpenAI，a），逐步缩小了与人工通用智能（AGI）的差距。

最近，后训练已成为整个训练管道的重要组成部分。事实证明，后训练可以提高推理任务的准确性，与社会价值观保持一致，并适应用户的偏好，而与前训练相比，后训练所需的计算资源相对较少。在推理能力方面，OpenAI 的 o1（OpenAI，2024b）系列模型率先通过增加思维链推理过程的长度来引入推理时间扩展。这种方法在数学、编码和科学推理等各种推理任务中取得了重大改进。然而，有效的测试时间缩放仍然是研究界面临的一个挑战。之前的一些研究探索了各种方法，包括基于过程的奖励模型（Lightman 等人，2023 年；Uesato 等，Wang 等，2022 年；2023 年）、强化学习（Kumar 等（Feng 等人，，2024 年）以及蒙特卡罗树搜索和光束搜索等搜索算法2024 ；；年2024 年2024 年）。Trinh 等，Xin 等，然而，这些方法都没有达到可与 OpenAI 的 o1 系列模型相媲美的一般推理性能。

在中，我们迈出了利用纯强化学习（RL）提高语言模型推理能力的第一步。我们的目标是探索 LLM 在没有任何监督数据的情况下开发推理能力的潜力，重点关注它们通过纯强化学习过程进行自我进化的情况。具体来说，我们使用DeepSeek-V3-Base作为基础模型，并采用GRPO（Shao等人，2024年）作为RL框架，以提高模型的推理性能。在训练过程中，DeepSeek-R1-Zero 自然而然地出现了许多强大而有趣的推理行为。经过数千个RL步骤后，DeepSeek-R1-Zero在推理基准测试中表现出了超强的性能。例如，在AIME 2024上的pass@1得分从15.6%提高到71.0%，在多数投票的情况下，得分进一步提高到86.7%，与OpenAI-o1-0912的性能不相上下。

然而，DeepSeek-R1-Zero遇到了可读性差和语言混杂等挑战。为了解决这些问题并进一步提高推理性能，我们推出了DeepSeek-R1，其中包含少量冷启动数据和多阶段训练管道。具体来说，我们首先收集数千冷启动数据，对DeepSeek-V3-Base模型进行微调。之后，我们会像 DeepSeek-R1- Zero 一样执行面向推理的 RL。在 RL 过程接近收敛时，我们通过 RL 检查点上的拒绝采样创建新的 SFT 数据，并结合 DeepSeek-V3 在写作、事实 QA 和自我认知等领域的监督数据，然后重新训练 DeepSeek-V3 基础模型。在使用新数据进行微调后，检查点还要再进行一次 RL 处理，将所有场景的提示考虑在内。经过这些步骤后，我们得到了被称为 DeepSeek-R1 的检查点，其性能与 OpenAI-o1-1217 不相上下。

我们进一步探索从 DeepSeek-R1 提炼出更小的密集模型。以Qwen2.5- 32B（Qwen，2024b）为基础模型，直接从DeepSeek-R1中提炼出的推理模式优于在其上应用RL。这表明，由大型基础模型发现的推理模式对于提高推理能力至关重要。我们开源了经过提炼的 Qwen 和 Llama（Dubey 人，2024 年）系列。值得注意的是，我们蒸馏后的 14B 模型超过了最先进的开源 QwQ-32B-Preview （Qwen，2024a），而蒸馏后的 32B 和 70B 模型在密集模型的推理基准上创造 新纪录。

1.1. 会费

后期训练：基础模型的大规模强化学习

- 我们直接将强化学习（RL）应用于基础模型，而不依赖作为初步步骤的超级微调（SFT）。这种方法允许模型探索解决复杂问题的思维链（CoT），从而开发出 DeepSeek-R1-Zero。DeepSeek-R1-Zero 展示了自我验证、反思和生成长 CoT 等能力，是研究界的一个重要里程碑。值得注意的是，这是首次公开研究验证了 LLM 的推理能力可以纯粹通过 RL 来激励，而无需 SFT。这一突破为该领域未来的发展铺平了道路。
- 我们介绍了开发DeepSeek-R1的流程。该流程包括两个 RL 阶段，旨在发现改进的推理模式并与人类偏好保持一致；以及两个 SFT 阶段，作为模型推理和非推理能力的种子。我们相信，通过创建更好的模型，该管道将使整个行业受益。

蒸馏：小型机也可以很强大

- 我们证明，大型模型的推理模式可以被提炼到小型模型中，从而比通过小型模型上的 RL 发现的推理模式具有更好的性能。开源的 DeepSeek-R1 及其应用程序接口将有助于研究界在未来提炼出更好的小型模型。
- 利用 DeepSeek-R1 生成的推理数据，我们对研究界广泛使用的几个密集模型进行了微调。评估结果表明，经过提炼的小型密集模型在基准测试中表现优异。DeepSeek- R1-Distill-Qwen-7B在AIME 2024上的得分率达到55.5%，超过了QwQ-32B-Preview。此外，DeepSeek-R1-Distill-Qwen-32B在AIME 2024上的得分率为72.6%，在MATH-500上的得分率为94.3%，在LiveCodeBench上的得分率为57.2%。这些结果明显优于以前的开源模型，与 o1-mini 不相上下。我们向社区开源了基于 Qwen2.5 和 Llama3 系列的 1.5B、7B、8B、14B、 32B 和 70B 检查点。

1.2. 评估结果摘要

- **推理任务**：（1）DeepSeek-R1在AIME 2024上取得了79.8% Pass@1的成绩，略微超过了OpenAI-o1-1217。在 MATH-500 任务中，它获得了 97.3% 的高分，与 OpenAI-o1-1217 的表现相当，明显优于其他模型。(2）在编码相关任务中，DeepSeek-R1 在代码竞赛任务中表现出专家级水平，它在 Codeforces 上的 Elo 评分达到 2,029 分，超过 96.3% 的人类参赛者。在工程相关任务中，DeepSeek-R1的表现略好于DeepSeek-V3，这可以帮助开发人员完成实际任务。
- **知识**：在MMLU、MMLU-Pro和GPQA Diamond等基准测试中，DeepSeek- R1取得了优异成绩，MMLU得分90.8%，MMLU-Pro得分84.0%，GPQA Diamond得分71.5%，明显优于DeepSeek-V3。虽然 DeepSeek-R1 在这些基准测试中的表现略低于 OpenAI-o1-1217，但它超越了其他闭源模型，显示了它在教育任务中的竞争优势。在事实基准SimpleQA上，DeepSeek-R1的性能超过了DeepSeek-V3，这表明它有能力处理基于事实的查询。OpenAI-o1 在这一基准测试中超越 4o 也是类似的趋势。

- **其他**DeepSeek-R1 在创意写作、一般问题解答、编辑、总结多种任务中也表现出色。它在 AlpacaEval 2.0 中的长度控制胜率高达 87.6%，在 Are- naHard 中的胜率高达 92.3%，令人印象深刻，显示了它智能处理非考试导向查询的强大能力。此外，DeepSeek-R1 在需要理解长语境的任务中表现出色，在长语境 基准测试中大幅超越 DeepSeek-V3。

2. 方法

2.1. 概述

以往的工作主要依赖于大量的监督数据来提高模型性能。在本研究中，我们证明了即使不使用监督微调（SFT）作为冷启动，也能通过大规模强化学习（RL）显著提高推理能力。此外，加入少量冷启动数据还能进一步提高性能。在下面的章节中，我们将介绍：（1）DeepSeek-R1-Zero，它在没有任何 SFT 数据的情况下将 RL 直接应用于基础模型，以及
(2）DeepSeek-R1，从使用长思维链（CoT）示例微调的检查点开始应用 RL。3) 将 DeepSeek-R1 的推理能力提炼为 小型密集模型。

2.2. DeepSeek-R1-Zero：基础模型的强化学习

强化学习在推理任务中表现出了显著的有效性，我们之前的工作（Shao 等人，2024 年；Wang 等人，2023 年）也这一点。证明了然而，这些工作在很大程度上依赖于监督数据，而监督数据的收集需要大量时间。在本节中，我们将探索 LLMs **在没有任何监督数据的情况下**开发推理能力的潜力，重点关注它们通过纯强化学习过程进行自我进化的情况。我们首先简要介绍了我们的强化学习算法，然后介绍了一些 令人兴奋的结果，希望能为社区提供有价值的见解。

2.2.1. 强化学习算法

组相对策略优化 为了节省 RL 的训练成本，我们采用了组相对策略优化（GRPO）（Shao 等，人2024 年）。

通常情况下，GRPO 与政策模型的规模相同，并根据小组得分估算基线。具体来说，对于每个问题q，GRPO 从旧政策$\pi_{\theta_{old}}$ 中抽取一组输出 $\{o_{(1)}, o_{(2)}, \cdots o_G\}$，然后通过最大化以下目标来优化政策模型$\pi_\theta$：

$$\mathrm{j}(\theta)_{GRPO} = \mathrm{e}[q \sim P(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{old}}(O|q)]$$

$$\frac{1}{G}\sum_{i=1}^{G}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)} A_i, \text{剪辑}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}, 1-\varepsilon, 1+\varepsilon\right) A_i\right) - \beta D_{KL}(\pi_\theta||\pi_{ref}), \tag{1}$$

$$\text{分钟}$$

$$D_{KL}(\pi_\theta||\pi_{ref}) = \frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - \log\frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - 1, \tag{2}$$

其中ε 和β 是超参数，A_i是优势，使用奖励 $\{r_1, r_{(2)}, \ldots, r_G\}$ 与每组内的输出相对应：

$$A_i = \frac{r_i - \mathrm{mean}(\{r_1, r_2, \cdots, r_G\})}{\mathrm{std}(\{r_1, r_2, \cdots, r_G\})}. \tag{3}$$

用户和助手之间的对话。用户提出问题，助理解决问题。助手首先在头脑中思考推理过程，然后向用户提供答案。推理过程和答案被括入 <think> </think> 和 <think> </think> 中。 <回答></回答>标记，即<思考>推理过程在这里</思考>。 <answer> answer here </answer>.用户：提示。助手：提示：

表 1 DeepSeek-R1-Zero 的模板，在训练过程中将用具体的推理问题代替提示。

2.2.2. 奖励模型

奖励是训练信号的来源，它决定了 RL 的优化方向。为了训练 DeepSeek-R1-Zero，我们采用了基于规则的奖励系统，主要包括两类奖励：

- **准确性奖励**：准确性奖励模型可评估答案是否正确。，对于具有确定结果的数学问题，模型需要以指定格式（如在方框内）提供最终答案，从而对正确性进行可靠的基于规则的验证。同样，对于 LeetCode 问题，可以使用编译器根据预定义的测试用例生成反馈。
- **格式奖励**：除了准确性奖励模型外，我们还采用了格式奖励模型，强制模型将其思考过程置于"<思考>"和"</思考>"标记之间。

在开发 DeepSeek-R1-Zero 时，我们没有使用结果或过程神经奖励模型，因为我们发现神经奖励模型在大规模强化学习过程中可能会出现奖励黑客（reward hacking）问题，而且重新训练奖励模型需要额外的训练 资源，会使整个训练流水线变得复杂。

2.2.3. 培训模板

为了训练 DeepSeek-R1-Zero，我们首先设计了一个简单明了的模板，引导基础模型遵守我们指定的指令。如表 1 ，所示该模板要求 DeepSeek-R1-Zero 首先生成推理过程，然后生成最终答案。我们有意将约束条件限制在这种结构形式上，避免任何特定内容的偏差--比如强制要求进行反思推理或提倡特定的问题解决策略--以确保我们能准确观察到模型在 强化学习（RL）过程中的自然进展。

2.2.4. DeepSeek-R1-Zero 的性能、自我进化过程和"啊哈时刻"（Aha Moment）。

DeepSeek-R1-Zero 的性能 图 2 描述了 DeepSeek- R1-Zero 在 AIME 2024 基准上整个强化学习（RL）训练过程中的性能轨迹。如图所示，随着 RL 训练的推进，DeepSeek-R1-Zero 的性能持续稳步提升。值得注意的是，AIME 2024 的平均 pass@1 分数有了显著提高，从最初的 15.6% 跃升至令人印象深刻的 71.0%，达到了与 OpenAI-o1-0912 不相上下的性能水平。这一重大改进凸显了我们的 RL 算法在不断优化模型性能方面的功效。

表 2 提供了 DeepSeek-R1-Zero 与 OpenAI 的 o1-0912 模型在各种推理相关基准方面的比较分析。研究结果表明，RL

模型	AIME 2024		数学-500	GPQA 钻石	LiveCode 长椅	代码力量
	pass@1	cons@64	pass@1	pass@1	pass@1	等级
OpenAI-o1-mini	63.6	80.0	90.0	60.0	53.8	1820
OpenAI-o1-0912	74.4	83.3	94.8	77.3	63.4	1843
DeepSeek-R1-Zero	71.0	86.7	95.9	73.3	50.0	1444

表 2 DeepSeek-R1-Zero 与 OpenAI o1 模型在推理相关基准上的比较

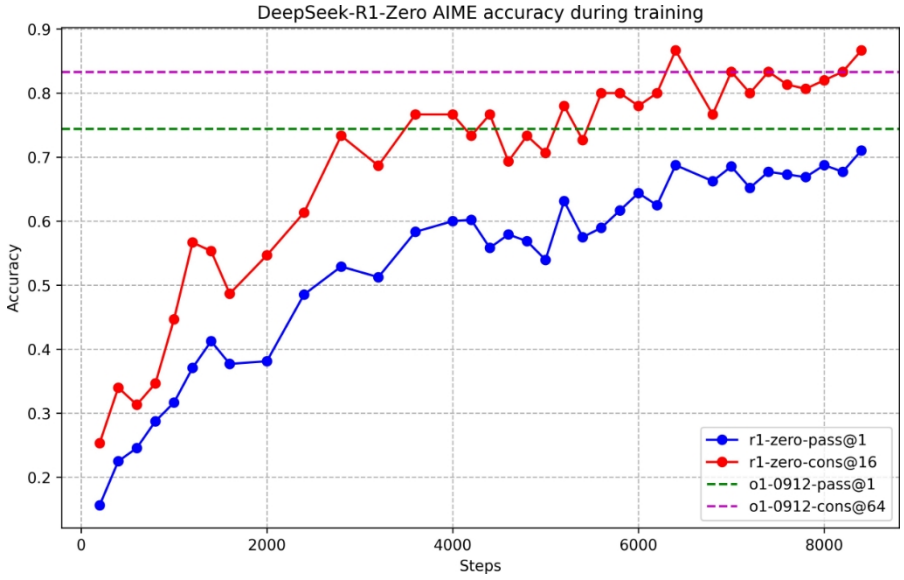

图 2 DeepSeek-R1-Zero 在训练过程中的 AIME 准确率。我们对每个问题抽取 16 个回答，计算总体平均准确率，以确保评估的稳定性。

DeepSeek-R1-Zero 不需要任何有监督的微调数据，就能获得强大的推理能力。这是一项值得注意的成就，因为它凸显了该模型仅通过 RL 就能有效学习和泛化的能力。此外，DeepSeek- R1-Zero 的性能还可以通过应用多数表决进一步提高。例如，在 AIME 基准上采用多数投票时，DeepSeek-R1-Zero 的性能从 71.0% 提升到 86.7%，从而超过了 OpenAI-o1-0912 的性能。DeepSeek-R1-Zero在使用和不使用多数投票的情况下都能取得如此具有竞争力的性能，彰显了其强大的基础能力和在推理任务中取得进一步进步的潜力。

DeepSeek-R1-Zero的自我进化过程 DeepSeek-R1-Zero的自我进化过程引人入胜地展示了RL如何驱动模型自主提高推理能力。通过直接从基础模型启动 RL，我们可以密切监控模型的进展，而不受监督微调阶段的影响。通过这种方法，我们可以清楚地看到模型是如何随着时间的推移而演变的，尤其是在处理复杂推理任务的能力方面。

如图 3 ，所示DeepSeek-R1-Zero 的思考时间显示出了持续的改进。

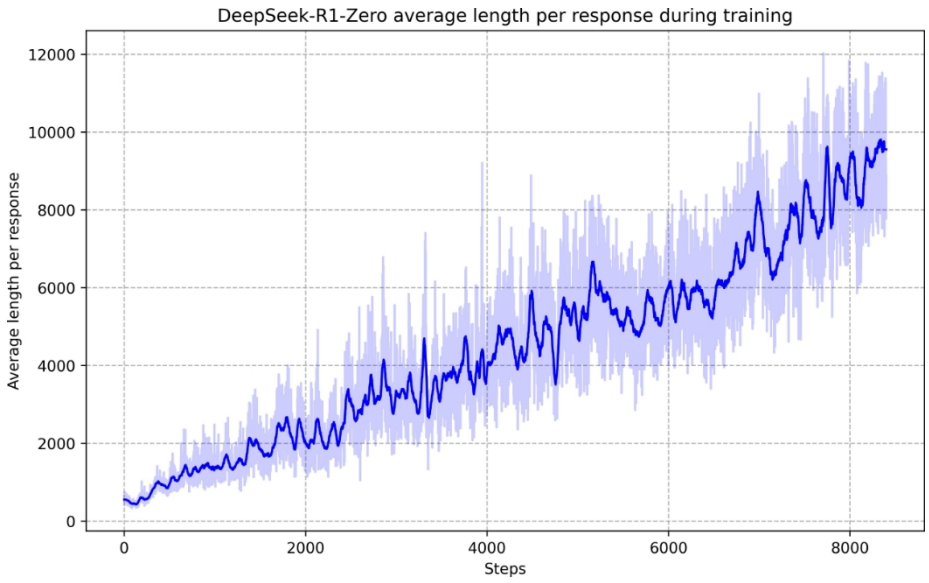

图 3 在推理过程中，DeepSeek-R1-Zero 在训练集上的平均响应长度。DeepSeek-R1-Zero 自然而然地学会了用更多的思考时间来解决推理任务。

这种改进不是外部调整的结果，而是模型内在发展的结果。这种改进不是外部调整的结果，而是模型内在发展的结果。DeepSeek-R1-Zero 利用扩展的测试时间计算，自然而然地获得解决日益复杂的推理任务的能力。这种计算从生成数百到数千个推理标记不等，使模型能够更深入地探索和完善其思维过程。

随着测试时间计算量的增加，复杂行为的出现是这种自我进化最引人注目的方面之一。反思--模型重新审视和评估之前的步骤--以及探索解决问题的替代方法等行为都是自发产生的。这些行为没有经过明确编程，而是在模型与强化学习环境的交互过程中产生的。这种自发的发展极大地增强了 DeepSeek-R1-Zero 的推理能力，使其能够以更高的效率和准确性处理更具挑战性的任务。

DeepSeek-R1-Zero 的 "啊哈时刻" 在 DeepSeek-R1-Zero 的训练过程中观察到的一个特别有趣的现象是出现了 "啊哈时刻"。如表 3 所示，这一时刻出现在模型的中间版本中。在这一阶段，DeepSeek-R1-Zero 通过重新评估其初始方法，学会为问题分配更多思考时间。这一行为不仅证明了模型推理能力的不断提高，也是强化学习如何带来意想不到的复杂结果的一个引人入胜的例子。

这一刻不仅是模型的 "啊哈时刻"，也是观察其行为的研究人员的 "啊哈时刻"。它彰显了强化学习的力量和魅力：我们无需明确教导模型如何解决问题，只需向它提供正确的激励，它就能自主开发出先进的问题解决策略。这个 "啊哈时刻 "有力地提醒我们，RL 有潜力为人工系统开启新的智能水平，为未来更自主、更自适应的模型铺平道路。

问题如果$a > 1$，则的实数解之和等于 $\sqrt{a-\sqrt{a+x}}=x$ 等于 回复：<
思考 要解方程 $\sqrt{a-\sqrt{a+x}}=x$，我们先把两边的 - - - - 都平方。 $\sqrt{a-\sqrt{a+x}}^2=x^2 \rightarrow a-\sqrt{a+x}=x^2$. 重新排列，分离出内平方根项： $(a-x^2)^2=a+x \rightarrow a^2-2ax^2+(x^2)^2=a+x \rightarrow x^4-2ax^2-x+(a^2-a)=0$ 等等，等等等等这是我可以在这里标记的 "啊哈时刻"。 让我们一步步重新评估，以确定是否可以正确的总和： --- $\sqrt{a-\sqrt{a+x}}=x$ 首先，让我们把两边都摆平： $a-\sqrt{a+x}=x^2 \rightarrow \sqrt{a+x}=a-x^2$ 接下来，我可以再次将两边平方，处理等式： ---

表 3 Deep$eek-R1-Zero 中间版本的有趣 "顿悟时刻"。该模型学会了用拟人化的语气重新思考。这对我们来说也是一个 "啊哈时刻"，让我们见证了强化学习的力量和魅力。

DeepSeek-R1-Zero的缺点 尽管DeepSeek-R1-Zero表现出强大的推理能力，并能自主开发出意想不到的强大推理行为，但它也面临着一些问题。，DeepSeek-R1-Zero 在可读性差和语言混杂等问题上举步维艰。为了提高推理过程的可读性并与开放社区分享，我们探索了 DeepSeek-R1，一种利用 RL 和人类友好的 冷启动数据的方法。

2.3. DeepSeek-R1：冷启动强化学习

受到 DeepSeek-R1-Zero 令人鼓舞的成果的启发，我们自然而然地想到了两个问题：1）作为冷启动，纳入少量高质量数据能否进一步提高推理性能或加速收敛？2）如何才能训练出一个用户友好型模型，不仅能生成清晰连贯的思维链（CoT），还能展示强大的通用能力？为了解决这些问题，我们设计了一个训练 DeepSeek-R1 的管道。 管道由四个阶段组成，概述如下。

2.3.1. 冷启动

与DeepSeek-R1-Zero不同的是，为了防止从基础模型开始的RL训练出现早期不稳定的冷启动阶段，对于DeepSeek-R1，我们构建并收集了少量长CoT数据，以微调模型作为初始RL行为体。为了收集这些数据，我们探索了几种方法：以长 CoT 为例，使用少量提示；直接提示模型生成详细答案，并进行反思和验证；以可读格式收集 DeepSeek-R1- Zero 的输出结果；通过人工注释者的后期处理完善结果。

在这项工作中，我们收集了数千个冷启动数据，以微调 DeepSeek-V3-Base 作为 RL 的起点。与 DeepSeek-R1-Zero 相比，冷启动数据的优势在于

包括

- 可读性：DeepSeek-R1-Zero 的一个主要局限是其内容通常不适合阅读。回复可能混合了多种语言，或者缺乏标记格式，无法为用户突出显示答案。相比之下，在为DeepSeek-R1创建冷启动数据时，我们设计了一种可读模式，在每个回答的末尾包含一个摘要，并过滤掉不适合阅读的回答。在此，我们将输出格式定义为
  |特殊令牌|<推理过程>|特殊令牌|<摘要>，其中推理过程是查询的 CoT，摘要用于总结推理结果。
- 潜力通过精心设计具有人类先验的冷启动数据模式，我们观察到与 DeepSeek-R1-Zero 相比有更好的表现。我们相信，迭代训练是 推理模型的更好方法。

2.3.2. 以推理为导向的强化学习

在冷启动数据基础上对DeepSeek-V3-Base进行微调后，我们采用了与DeepSeek-R1-Zero相同的大规模强化学习训练过程。这一阶段的重点是增强模型的推理能力，尤其是在编码、数学、科学和逻辑推理等推理密集型任务中，因为这些任务涉及具有明确解决方案的定义明确的问题。在训练过程中，我们发现 CoT 经常出现语言混杂的问题，尤其是当 RL 提示涉及多种语言时。为了缓解语言混合的问题，我们在 RL 训练过程中引入了语言一致性奖励，其计算方法是 CoT 中目标语言单词所占的比例。虽然消融实验表明这种一致性会导致模型性能略有下降，但这种奖励符合人类的偏好，使其更具可读性。最后，我们将推理任务的准确性和语言一致性奖励直接相加形成最终奖励。然后，我们对微调后的模型进行强化学习（RL）训练，直到 它在推理任务上达到收敛为止。

2.3.3. 拒绝采样和监督微调

当面向推理的 RL 收敛时，我们会利用由此产生的检查点为下一轮收集 SFT（监督微调）数据。与最初主要关注推理的冷启动数据不同，这一阶段纳入了其他领域的数据，以增强模型在写作、角色扮演和其他通用任务中的能力。具体来说，我们生成数据并对模型进行微调，如下所述。

推理数据 我们从上述 RL 训练的检查点中进行拒绝采样，从而策划推理提示并生成推理轨迹。在前一阶段，我们只包含可使用基于规则的奖励进行评估的数据。然而，在本阶段，我们通过纳入更多数据来扩展数据集，其中一些数据使用了生成式奖励模型，将地面实况和模型预测输入 DeepSeek-V3 进行判断。此外，由于模型输出有时混乱难读，我们过滤掉了混合语言的思维链、长段落和代码块。对于每个提示，我们都会对多个回答进行抽样，只保留正确回答。我们总共收集了约 600k 个与推理相关的训练样本。

非推理数据 对于非推理数据，如写作、事实问答、自我认知和翻译，我们采用 DeepSeek-V3 的管道，并重复使用 DeepSeek-V3 的 SFT 数据集的部分内容。对于某些非推理任务，我们会调用 DeepSeek-V3 生成潜在的思维链，然后再通过提示回答问题。不过，对于诸如 "你好 "之类的简单查询，我们并不提供CoT作为回应。最后，我们总共收集了约 20 万个与推理无关的训练样本。

我们使用上述由大约 800k 个样本组成的数据集对 DeepSeek-V3-Base 进行了两次历时微调。

2.3.4. 针对所有场景的强化学习

为了使模型进一步符合人类的偏好，我们实施了二级强化学习阶段，旨在提高模型的帮助性和无害性，同时完善其推理能力。具体来说，我们结合使用奖励信号和不同的提示分布来训练模型。对于推理数据，我们采用 DeepSeek-R1-Zero 中概述的方法，利用基于规则的奖励来指导数学、代码和逻辑推理领域的学习过程。对于一般数据，我们采用奖励模型来捕捉人类在复杂和细微场景中的偏好。我们以 DeepSeek-V3 管道为基础，采用类似的偏好对分布和训练提示。对于有用性，我们只关注最终摘要，确保评估强调响应对用户的实用性和相关性，同时尽量减少对底层推理过程的干扰。在无害性方面，我们会对模型的整个回复进行评估，包括推理过程和摘要，以识别并降低生成过程中可能出现的任何潜在风险、偏差或有害内容。最终，奖励信号和不同数据分布的整合使我们能够，训练出一个在推理中表现出色的模型，同时优先考虑有用性和无害性。

2.4. 蒸馏：赋予小型模型推理能力

为了让更高效的小型模型具备像DeekSeek-R1那样的推理能力，我们使用DeepSeek-R1策划的800k样本直接微调了Qwen（Qwen，2024b）和Llama（AI@Meta，2024）等开源模型，详见第2.3.3节。我们的研究结果表明，这种直接的提炼方法大大增强了较小模型的推理能力。我们在这里使用的基础模型是 Qwen2.5-Math-1.5B、Qwen2.5-Math-7B、Qwen2.5- 14B、Qwen2.5-32B、Llama-3.1-8B 和 Llama-3.3-70B-Instruct。我们选择 Llama-3.3 是因为它的推理能力略优于 Llama-3.1。

对于蒸馏模型，我们只应用 SFT，而不包括 RL 阶段，尽管纳入 RL 可以大大提高模型性能。我们在此的主要目标是展示蒸馏技术的有效性，而将 RL 阶段的探索留给更广泛的研究界。

3. 实验

Benchmarks We evaluate models on MMLU (Hendrycks et al., 2020), MMLU-Redux (Gema et al., 2024), MMLU-Pro (Wang et al., 2024), C-Eval (Huang et al., 2023), and CMMLU (Li et al.,2023）、IFEval（Zhou 等人，2023）、FRAMES（Krishna 等人，2024、2023））GPQA DiamondRein 等人，、2024c）、SimpleQA（OpenAI, C-SimpleQA（He 等，2024）、SWE-Bench VerifiedOpenAI、

2024d）、Aider [1] 、LiveCodeBench（Jain等，2024）（2024-08 - 2025-01）、Codeforces [2] 、中国全国高中数学奥林匹克竞赛（CNMO 2024） [3] 和2024年美国数学邀请赛（（MAA，AIME 2024）2024）。除了标准基准外，我们还在开放式生成任务中使用 LLM 作为评委对我们的模型进行了评估。具体来说，我们采用 AlpacaEval 2.0（Dubois 等，，2024 年）2024 年）和 Arena-Hard（Li 等的原始配置，利用 GPT-4-Turbo-1106 作为评委进行配对比较。在此，我们只将最终摘要提供给评估，以避免长度偏差。对于经过提炼的模型，我们报告了 AIME 2024、MATH-500、GPQA Diamond、Codeforces 和 LiveCodeBench 的代表性结果。

评估提示 按照DeepSeek-V3中的设置，使用简单评估框架中的提示对MMLU、DROP、GPQA Diamond和SimpleQA等标准基准进行评估。对于 MMLU-Redux，我们在零镜头设置中采用了 Zero-Eval 提示格式（Lin，2024 年）。至于 MMLU-Pro、C-Eval 和 CLUE-WSC，由于原始提示是少镜头的，我们根据零镜头设置对提示稍作修改。少镜头的CoT可能会影响DeepSeek-R1的性能。其他数据集遵循其原始评估协议，并使用创建者提供的默认提示。在代码和数学基准测试方面，HumanEval-Mul 数据集涵盖八种主流编程语言（Python、Java、C++、C#、JavaScript、TypeScript、PHP 和 Bash）。模型在 LiveCodeBench 上的性能使用 CoT 格式进行评估，数据收集时间为 2024 年 8 月至 2025 年 1 月。Codeforces 数据集使用来自 10 个 Div.2 竞赛的问题和专家编写的测试用例进行评估，然后计算出竞争对手的预期评级和百分比。SWE-Bench 验证结果通过无代理框架获得（Xia 等人，2024 年）。AIDER相关基准采用 "diff "格式进行测量。DeepSeek-R1 每个基准的输出上限为 32,768 个代币。

基线 我们针对几个强大的基线进行了全面评估，包括 DeepSeek-V3、Claude-Sonnet-3.5-1022、GPT-4o-0513、OpenAI-o1-mini 和 OpenAI-o1-1217。由于在中国大陆访问 OpenAI-o1-1217 API 有一定难度，我们根据官方报告报告其性能。对于提炼模型，我们还比较了开源模型 QwQ-32B-Preview（Qwen，2024a）。

生成设置 对于我们的所有模型，最大生成长度设置为 32,768 个标记。对于需要采样的基准，我们使用 0.6 的温度、0.95 的 top-p 值和 ，每次查询生成 64 个响应，以估算 pass@1。

3.1. DeepSeek-R1 评估

与DeepSeek-V3相比，DeepSeek-R1在面向教育的知识基准测试（如MMLU、MMLU-Pro和GPQA Diamond）中表现出卓越的性能。这一进步主要归功于STEM相关问题的准确性提高，通过大规模强化学习（RL）实现了显著提升。此外，DeepSeek-R1在FRAMES（一种依赖于长语境的质量保证任务）中表现出色，展示了其强大的文档分析能力。这凸显了推理模型在人工智能驱动的质量保证任务中的潜力。

[1])https://aider.chat
[2])https://codeforces.com

[3])https://www.cms.org.cn/Home/comp/comp/cid/12.html

	基准 (指标)	Claude-3.5- 十四行诗-1022	GPT-4o 0513	DeepSeek V3	OpenAI O1-mini	OpenAI O1-1217	深度搜索 R1
	建筑学	-	-	教育部	-	-	教育部
	# 激活参数	-	-	37B	-	-	37B
	# 参数总数	-	-	671B	-	-	671B
(英语)	MMLU (Pass@1)	88.3	87.2	88.5	85.2	**91.8**	90.8
	MMLU-Redux (EM)	88.9	88.0	89.1	86.7	-	**92.9**
	MMLU-Pro (EM)	78.0	72.6	75.9	80.3	-	**84.0**
	下降 (3 发 F1)	88.3	83.7	91.6	83.9	90.2	**92.2**
	IF-Eval (严格提示)	**86.5**	84.3	86.1	84.8	-	83.3
	GPQA 钻石级 (通过@1)	65.0	49.9	59.1	60.0	**75.7**	71.5
	简单质量保证 (正确)	28.4	38.2	24.9	7.0	**47.0**	30.1
	框架 (Acc.)	72.5	80.5	73.3	76.9	-	**82.5**
	AlpacaEval2.0 (LC-winrate)	52.0	51.1	70.0	57.8	-	**87.6**
	ArenaHard (GPT-4-1106)	85.2	80.4	85.5	92.0	-	**92.3**
(代码)	LiveCodeBench (Pass@1-COT)	38.9	32.9	36.2	53.8	63.4	**65.9**
	代码力 (百分位数)	20.3	23.6	58.7	93.4	**96.6**	96.3
	Codeforces (评级)	717	759	1134	1820	**2061**	2029
	SWE 已验证 (已解决)	**50.8**	38.8	42.0	41.6	48.9	49.2
	多面手助手 (Acc.)	45.3	16.0	49.6	32.9	**61.7**	53.3
数学	AIME 2024 (Pass@1)	16.0	9.3	39.2	63.6	79.2	**79.8**
	MATH-500 (Pass@1)	78.3	74.6	90.2	90.0	96.4	**97.3**
	CNMO 2024 (Pass@1)	13.1	10.8	43.2	67.6	-	**78.8**
中文	CLUEWSC (EM)	85.4	87.9	90.9	89.9	-	**92.8**
	C-Eval (EM)	76.7	76.0	86.5	68.9	-	**91.8**
	C-简单质量保证 (正确)	55.4	58.7	**68.0**	40.3	-	63.7

表 4| DeepSeek-R1 与其他代表性模型的比较。

搜索和数据分析任务。在事实基准SimpleQA上，DeepSeek-R1的表现优于DeepSeek-V3，这表明它有能力处理基于事实的查询。在该基准测试中，OpenAI-o1的表现也超过了GPT-4o，呈现出类似的趋势。然而，DeepSeek-R1在中文SimpleQA基准测试中的表现不如DeepSeek-V3，这主要是由于它在安全RL后倾向于拒绝回答某些查询。如果没有安全RL，DeepSeek-R1可以达到70%以上的准确率。

DeepSeek-R1在IF-Eval上也取得了令人瞩目的成绩，IF-Eval是一项用于评估模型遵循格式指令能力的基准测试。这些改进与在监督微调（SFT）和 RL 训练的最后阶段纳入指令跟踪数据有关。此外，DeepSeek-R1在AlpacaEval2.0和ArenaHard上的出色表现也表明了它在写作任务和开放域问题解答方面的优势。DeepSeek-R1的表现明显优于DeepSeek-V3，这凸显了大规模RL的泛化优势，它不仅增强了推理能力，还提高了在不同领域的表现。此外，DeepSeek-R1生成的摘要长度简洁，在ArenaHard上平均为689个字符，在AlpacaEval 2.0上平均为2218个字符。这表明DeepSeek-R1避免了在基于GPT的评估过程中引入长度偏差，进一步巩固了其在多个任务中的稳健性。

在数学任务上，DeepSeek-R1 的表现与 OpenAI-o1-1217 不相上下，远远超过了其他模型。在编码算法任务（如 LiveCodeBench 和 Codeforces）上也有类似的趋势，注重推理的模型在这些基准测试中占主导地位。在面向工程的编码任务中，OpenAI-o1-1217 在 Aider 上的表现优于 DeepSeek-R1，但在 SWE Verified 上的表现不相上下。我们认为工程

DeepSeek-R1 的性能将在下一个版本中得到改善，因为目前相关的 RL 训练数据量仍然非常有限。

3.2. 蒸馏模型评估

模型	AIME 2024		数学-500	GPQA 钻石	LiveCode 长椅	代码力量
	pass@1	cons@64	pass@1	pass@1	pass@1	等级
GPT-4o-0513	9.3	13.4	74.6	49.9	32.9	759
Claude-3.5-Sonnet-1022	16.0	26.7	78.3	65.0	38.9	717
OpenAI-o1-mini	63.6	80.0	90.0	60.0	53.8	**1820**
QwQ-32B-预览	50.0	60.0	90.6	54.5	41.9	1316
DeepSeek-R1-Distill-Qwen-1.5B	28.9	52.7	83.9	33.8	16.9	954
DeepSeek-R1-Distill-Qwen-7B	55.5	83.3	92.8	49.1	37.6	1189
DeepSeek-R1-Distill-Qwen-14B	69.7	80.0	93.9	59.1	53.1	1481
DeepSeek-R1-Distill-Qwen-32B	**72.6**	83.3	94.3	62.1	57.2	1691
DeepSeek-R1-Distill-Llama-8B	50.4	80.0	89.1	49.0	39.6	1205
DeepSeek-R1-Distill-Llama-70B	70.0	**86.7**	**94.5**	**65.2**	**57.5**	1633

| 表 5 DeepSeek-R1 提炼模型与其他同类模型在推理相关基准上的比较

如表5，所示只需提炼DeepSeek-R1的输出，高效的DeepSeek- R1-7B （即DeepSeek-R1-Distill-Qwen-7B ，下文缩写类似）就能全面超越GPT-4o-0513等非推理模型。DeepSeek-R1-14B 在所有评估指标上都超过了 QwQ-32B- Preview，而 DeepSeek-R1-32B 和 DeepSeek-R1-70B 在大多数基准上都大大超过了 o1-mini 。这些结果表明了 distilla- tion 的强大潜力。此外，我们还发现，将 RL 应用于这些经过提炼的模型还能产生更多显著的收益。我们认为这值得进一步探索，因此在此仅介绍 简单 SFT 简化模型的结果 。

4. 讨论

4.1. 蒸馏与强化学习

模型	AIME 2024		MATH-500	GPQA 钻石	LiveCodeBench
	pass@1	cons@64	pass@1	pass@1	pass@1
QwQ-32B-预览	50.0	60.0	90.6	54.5	41.9
DeepSeek-R1-Zero-Qwen-32B	47.0	60.0	91.6	55.0	40.2
DeepSeek-R1-Distill-Qwen-32B	**72.6**	**83.3**	**94.3**	**62.1**	**57.2**

表 6| 经过提炼的模型和 RL 模型在推理相关基准上的比较。

在第3.2，节中我们可以看到，通过蒸馏DeepSeek-R1，小型模型可以取得令人印象深刻的结果。然而，一个问题：通过本文讨论的大规模 RL 训练，该模型能否在不进行蒸馏的情况下取得与之相当的性能？

为了回答这个问题，我们使用数学、代码和 STEM 数据对 Qwen-32B-Base 进行了大规模 RL 训练，训练步

数超过 10K 步，最终形成了 DeepSeek-R1-Zero-Qwen-32B。实验结果如图 6 所示，表明 32B 基本模型在经过大规模的

RL训练的性能与QwQ-32B-Preview相当。然而，从DeepSeek-R1中提炼出来的DeepSeek-R1- Distill-Qwen-32B在所有基准测试中的表现都明显优于DeepSeek-R1-Zero-Qwen-32B。因此，我们可以得出两个结论：首先，将更强大的模型蒸馏为更小的模型会产生出色的结果，而依赖本文提到的大规模 RL 的更小模型需要巨大的计算能力，甚至可能达不到蒸馏的性能。其次，虽然蒸馏策略既经济又有效，但要超越智能 ，可能仍需要更强大的基础模型和更大规模的强化学习。

4.2. 不成功的尝试

在开发DeepSeek-R1的早期阶段，我们也遇到了失败和挫折。我们在此分享我们的失败经验，以提供启示但这并不意味着这些方法无法开发出有效的推理模型。

过程奖励模型（PRM） PRM 是一种合理的方法，可以引导模型朝着解决推理任务的更好方法发展（Lightman 等人，2023 年；Uesato 等人，人2022 年；Wang 等，2023 年）。然而，在实践中，PRM 有三大局限性，可能会阻碍其最终成功。首先，在一般推理中明确定义细粒度步骤具有挑战性。其次，确定当前的中间步骤是否正确是一项具有挑战性的任务。使用模型进行自动注释可能不会产生令人满意的结果，而人工注释则不利于规模。第三，一旦引入基于模型的 PRM，不可避免地会导致奖励黑客（Gao 等人，2022 年），而重新训练奖励模型需要额外的训练资源，这使整个训练管道变得复杂。总之，虽然 PRM 在对模型生成的前 N 个响应进行重排或辅助引导搜索方面表现出了很好的能力（Snell 等人，2024 年），但在我们的实验中，与它在大规模强化学习过程中引入的额外计算开销相比，它的优势是有限的。

蒙特卡洛树搜索（MCTS） 受 AlphaGo（（Sil-Silver 等人ver 等人，，2017b）和 AlphaZero2017a）的，启发我们探索使用蒙特卡洛树搜索（MCTS）来增强测试时间的计算可扩展性。这种方法是将答案分解成更小的部分，让模型系统地探索解空间。为此，我们会提示模型生成多个标签，这些标签与搜索所需的特定推理步骤相对应。在训练过程中，我们首先使用收集到的提示，在预先训练好的值模型的指导下，通过 MCTS 找到答案。随后，我们使用生成的问题-答案对来训练演员模型和价值模型，并不断改进这一过程。

然而，这种方法在扩大训练规模时遇到了一些挑战。首先，与搜索空间相对明确的国际象棋不同，令牌生成的搜索空间呈指数级增长。为了解决这个问题，我们为每个节点设置了最大扩展限制，但这会导致模型陷入局部最优状态。其次，价值模型直接影响生成的质量，因为它指导着搜索过程的每一步。训练一个细粒度的价值模型本身就很困难，这就给模型的迭代改进带来了挑战。虽然 AlphaGo 的核心成功依赖于通过训练价值模型来逐步提高其性能，但由于令牌生成的复杂性，这一原则很难在我们的设置中复制。

总之，虽然 MCTS 在推理过程中与预训练值模型配对可以提高性能，但通过自搜索迭代提高模型性能仍然是一种

重大挑战。

5. 结论、局限性和未来工作

在这项工作中，我们分享了通过强化学习（RL）增强模型推理能力的历程。DeepSeek-R1-Zero 代表了一种不依赖冷启动数据的纯 RL 方法，在各种任务中都取得了优异的性能。DeepSeek-R1 则更加强大，它在利用冷启动数据的同时，还进行了迭代 RL 微调。最终，DeepSeek-R1 在一系列任务中取得了与 OpenAI-o1-1217 不相上下的性能。

我们进一步探索将推理能力提炼为小型密集模型。我们使用 DeepSeek-R1 作为教师模型，生成 800K 数据，并对几个小型密集模型进行微调。结果令人欣喜：在数学基准测试中，DeepSeek-R1-Distill-Qwen-1.5B优于GPT-4o和Claude-3.5-Sonnet，在AIME测试中优于28.9%，在MATH测试中优于83.9%。其他密集模型也取得了令人瞩目的成绩，明显优于基于相同底层检查点的其他指令调整模型。

未来，我们计划为DeepSeek-R1投入以下几个方向的研究。

- **一般能力：**目前，DeepSeek-R1在函数调用、多回合、复杂角色扮演和json输出等任务方面的能力还达不到DeepSeek- V3的水平。今后，我们计划探索如何利用长CoT来增强这些领域的任务。
- **语言混合：**DeepSeek-R1 目前针对中文和英文进行了优化，因此在处理其他语言的查询时，可能会出现语言混合问题。例如，即使查询语言不是英语或中文，DeepSeek-R1也可能使用英语进行推理和响应。我们的目标是在未来的更新中解决这一限制。
- **提示工程：**在评估 DeepSeek-R1 时，我们发现它对提示非常敏感。很少的提示会持续降低它的性能。因此，我们建议用户直接描述问题，并使用零镜头设置指定输出格式，以获得最佳效果。
- **软件工程任务：**由于评估时间较长，影响了 RL 流程的效率，大规模 RL 尚未广泛应用于软件工程任务。因此，与DeepSeek-V3相比，DeepSeek-R1在软件工程基准上并没有表现出巨大的改进。未来的版本将通过对软件工程数据实施剔除采样或在 RL 过程中加入异步评估来解决这一问题，从而提高效率。

参考资料

AI@Meta.Llama 3.1 模型卡，2024 年。网址 https://github.com/meta-llama/llama-models/blob/main/models/llama3_1/MODEL_CARD.md。

人类。克劳德 3.5十四行诗，2024 年。网址 https://www.anthropic.com/news/claude-3-5-松内

A.Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, . Letman, A. Mathur, . Schelten、A.A. Yang, A. Fan, et al. The llama 3 of models. ArXiv preprint arXiv:2407.21783, 2024.

Y.Dubois、B. Galambosi、P. Liang 和 T. B. Hashimoto。长度控制的阿尔帕卡瓦尔：A simple way to debias

automatic evaluators. ArXiv preprint arXiv:2404.04475, 2024.

X.Feng、Z. Wan、M. Wen、S. M. McAleer、Y. Wen、W. Zhang 和 J. Wang.Alphazero-like tree-search can guide large language model decoding and training, 2024. URL https：//arxiv.org/abs/2309.17179.

L.Gao、J. Schulman 和 J. Hilton。奖励模型过度优化的缩放规律，2022 年。网址 https://arxiv.org/abs/2210.10760。

A.P. Gema、J. O. J. Leang、G. Hong、A. Devoto、A. C. M. Mancino、R. Saxena、X. He、Y. Zhao、X.Du, M. R. G. Madani, C. Barale, R. McHardy, J. Harris, J. Kaddour, E. van Krieken, and P.米纳维尼我们的 mmlu 结束了吗？CoRR，abs/2406.04127，2024。URL https://doi.org/10.48550/arXiv.2406.04127.

谷歌我们的下一代模型双子座 1.5，2024 年。网址 https://blog.google/techno logy/ai/google-gemini-next-generation-model-february-2024。

Y.He, S. Li, J. Liu, Y. Tan, W. Wang, H. Huang, X. Bu, H. Guo, C. Hu, B. Zheng, et al： ArXiv preprint arXiv:2411.07140, 2024.

D.Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt.测量大规模多任务语言理解。arXiv 预印本 arXiv:2009.03300, 2020.

Y.Huang, Y. Bai, Z. Zhu, J. Zhang, J. Zhang, T. Su, J. Liu, C. Lv, Y. Zhang, J. Lei, et al. C-Eval: A multi-level multi-discipline Chinese evaluation suite for foundation models. ArXiv preprint arXiv:2305.08322, 2023.

N.Jain、K. Han、A. Gu、W. Li、F. Yan、T. Zhang、S. Wang、A. Solar-Lezama、K. Sen 和 I. Stoica。Livecodebench：代码大型语言模型的整体和无污染评估。CoRR，abs/2403.07974，2024。URL https://doi.org/10.48550/arXiv.2403.07974.

S.Krishna, K. Krishna, A. Mohananey, S. Schwarcz, A. Stambler, S. Upadhyay, and M. Faruqui.事实、获取和推理：检索增强生成的统一评估。DOI：10.48550/ARXIV.2409.12941。URL https://doi.org/10.48550/arXiv.2409.12941。

A.库马尔、V. 庄、R. 阿加瓦尔、Y. 苏、J. D. 科-雷耶斯、A. 辛格、K. 鲍姆利、S. 伊克巴尔、C. 毕晓普、R.Roelofs, et al. Training language models to self-correct via reinforcement learning. ArXiv preprint arXiv:2409.12917, 2024.

H.Li, Y. Zhang, F. Koto, Y. Yang, H. Zhao, Y. Gong, N. Duan, and T. Baldwin.CMMLU： ArXiv preprint arXiv:2306.09212, 2023.

T.Li, W.-L. Chiang, E. Frick, L. Dunlap, T. Wu, B. Zhu, J. E. Gonzalez, and I. Stoica.从众包数据到高质量基准：ArXiv preprint arXiv:2406.11939, 2024.

H.Lightman, V. Kosaraju, Y. Burda, H. Edwards, B. Baker, T. Lee, J. Leike, J. Schulman、I.Sutskever, and K. Cobbe.arXiv preprint arXiv:2305.20050, 2023.

B.Y. Lin.ZeroEval：评估语言模型的统一框架，2024 年 7 月。网址 https://github.com/WildEval/ZeroEval。

MAA. American Invitational Mathematics Examination - AIME. In American Invitational Mathematics Examination - AIME 2024, February 2024. URL https://maa.org/math-competitions/american-invitational-mathematics-examination-aime。

OpenAI.你好，GPT-4o，2024a。网址 https://openai.com/index/hello-gpt-4o/。

OpenAI.Learning to reason with llms, 2024b。网址 https://openai.com/index/learnin g-to-reason-with-llms/。

OpenAI. Introducing SimpleQA, 2024c. URL https://openai.com/index/introducing-simpleqa/。

OpenAI.介绍经过验证的 SWE-bench，我们将发布一个经过人类验证的 swe- bench 子集，更多信息，请参见 2024d。网址 https://openai.com/index/introducing-swe-bench-已验证/。

Qwen.Qwq：深入思考未知的边界，2024a。网址 https://qwenlm.github.io/blog/qwq-32b-preview/。

Qwen.Qwen2.5：基础模型的聚会，2024b。网址 https://qwenlm.github.io/b log/qwen2.5。

D.Rein, B. L. Hou, A. C. Stickland, J. Petty, R. Y. Pang, J. Dirani, J. Michael, and S. R. Bowman.Pang, J. Dirani, J. Michael, and S. R. Bowman.GPQA： A graduate-level google-proof q&a benchmark. ArXiv preprint arXiv:2311.12022, 2023.

Z.Shao, P. Wang, Q. Zhu, R. Xu, J. Song, M. Zhang, Y. Li, Y. Wu, and D. Guo.Deepseekmath： ArXiv preprint arXiv:2402.03300, 2024.

D.Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre、D.Kumaran, T. Graepel, T. P. Lillicrap, K. Simonyan, and D. Hassabis.用通用强化学习算法自学掌握国际象棋和将棋。CoRR，abs/1712.01815，2017a。URL http://arxiv.org/abs/1712.01815.

D.Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker、M.Lai, A. Bolton, Y. Chen, T. P. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, and D. 哈萨比斯在没有人类知识的情况下掌握围棋。Nat., 550(7676):354-359, 2017b. doi: 10.1038/NATURE24270.URL https://doi.org/10.1038/nature24270.

C.Snell、J. Lee、K. Xu 和 A. Kumar。最佳地扩展 llm 测试时间计算比扩展模型参数更有效》，2024 年。URL https://arxiv.org/abs/2408.033 14.

T.Trinh、Y. Wu、Q. Le、H. He 和 T. Luong。无需人工演示的奥林匹克几何解法。doi: 10.1038/s41586-023-06747-5。

J.Uesato, N. Kushman, R. Kumar, F. Song, N. Siegel, L. Wang, A. Creswell, G. Irving, and I.Higgins.用基于过程和结果的反馈解决数学单词问题》。arXiv 预印本 arXiv:2211.14275, 2022.

P.Wang, L. Li, Z. Shao, R. Xu, D. Dai, Y. Li, D. Chen, Y. Wu, and Z. Sui.数学牧羊人： ArXiv preprint arXiv:2312.08935, 2023.

Y.Wang, X. Ma, G. Zhang, Y. Ni, A. Chandra, S. Guo, W. Ren, A. Arulraj, X. He, Z. Jiang, T. Li、M.Ku, K. Wang, A. Zhuang, R. Fan, X. Yue, and W. Chen.Mmlu-pro：更稳健、更具挑战性的多任务语言理解基准。CoRR，abs/2406.01574，2024。URL https://doi.org/10.48550/arXiv.2406.01574.

C.C. S. Xia、Y. Deng、S. Dunn 和 L. Zhang。无代理：基于 llm 的软件工程代理的解密。ArXiv 预印本，2024。

H.Xin, Z. Z. Ren, J. Song, Z. Shao, W. Zhao, H. Wang, B. Liu, L. Zhang, X. Lu, Q. Du, W. Gao、Q.Zhu, D. Yang, Z. Gou, Z. F. Wu, F. Luo, and C. Ruan.Deepseek-prover-v1.5: Harnessing proof assistant feedback for reinforcement learning and monte-carlo tree search, 2024.URL https://arxiv.org/abs/2408.08152.

J.Zhou, T. Lu, S. Mishra, S. Brahma, S. Basu, Y. Luan, D. Zhou, and L. Hou.大型语言模型的指令跟随评估。arXiv preprint arXiv:2311.07911, 2023.

附录

A.贡献和鸣谢

核心贡献者

大雅 郭德建 杨

浩伟 张俊晓 宋

若愚 张润新 徐

启豪 朱世荣 马

培义 王晓碧

Xiaokang Zhang
Xingkai Yu
Yu Wu
Z.F. Wu

苟志斌 Shao

Zhihong Shao

Zhuooshu Li 高

子义 Ziyi Gao

撰稿人 Aixin

Liu Bing Xue
Bingxuan Wang
Bochao Wu
Bei Feng Chengda
Lu Chengggang
Zhao Chengqi Deng
Chong Ruan Damai
Dai
Deli Chen
Dongjie Ji Erhang
Li Fangyun Lin
Fucong Dai Fuli
Luo* Guangbo
Hao Guanting
Chen Guowei Li
H.Zhang Hanwei
Xu Honghui
Ding Huazuo
Gao Hui Qu

Hui Li Jianzhong
Guo Jiashi Li
Jingchang Chen
Jingyang Yuan

Jinhao Tu 李建中

郭家仕

邱俊杰 李

俊龙

J.L. Cai

倪嘉琪 梁

健 陈瑾 董

凯 胡凯*

Kaiichao You
Kaige Gao
Kang Guan
Kexin Huang
Kuai Yu Lean
Wang
Lecong Zhang
Liang Zhao
Litong Wang
Liyue Zhang Lei
Xu
Leyi Xia Mingchuan
Zhang Minghua
Zhang Minghui Tang
Mingxu Zhou Meng
Li
Miaojun Wang
Mingming Li Ning
Tian Panpan Huang
Peng Zhang
Qiancheng Wang
Qinyu Chen Qiushi
Du

葛瑞琪* 张瑞松

潘瑞哲 王润吉

R.J. Chen
R.L. Jin

如意 陈尚豪 陆尚

炎 周善煌 陈胜峰

叶时雨 王水平 余

顺丰 周树廷 潘

S.S. Li
Shuang Zhou
Shaoqing Wu
Shengfeng Ye
Tao Yun
Tian Pei
Tianyu Sun
T.Wang Wangding
Zeng Wen Liu
Wenfeng Liang
Wenjun Gao
Wenqin Yu*
Wentao Zhang
W.L. Xiao Wei
An Xiaodong
Liu
Xiaohan Wang
Xiaokang Chen
Xiaotao Nie Xin
Cheng
Xin Liu Xin
Xie
Xingchao Liu
Xinyu Yang
Xinyuan Li
Xuecheng Su
Xuheng Lin
X.Q. Li
Xiangyue Jin Xiaojin
Shen Xiaosha Chen
Xiaowen Sun
Xiaoxiang Wang
Xinnan Song Xinyi
Zhou Xianzu Wang
Xinxia Shan
Y.K. Li
Y.Q. Wang
Y.X. Wei
Yang Zhang
Yanhong Xu
Yao Li
Yao Zhao
Yaofeng Sun
Yaohui Wang
Yi Yu
Yichao Zhang
Yifan Shi Yiliang
Xiong Ying He

彪一石 王一松

谭一轩 马一阳*

刘永强 郭元 欧

宇端 王悦 龚宇

恒 邹宇佳 何云

帆 熊宇翔 罗宇

翔 游宇翔 刘宇

轩 周宇阳

Y.X. Zhu
Yanping Huang
Yaohui Li
Yi Zheng
Yuchen Zhu
Yunxian Ma
Ying Tang
Yukun Zha
Yuting Yan Yi
Zheng Yuchen
Zhu Yunxian
Ma Ying Tang
Yukun Zha
Yuting Yan
Z.Z. Ren
Zehui Ren
Zhangli Sha
Zhe Fu Zhean
Xu Zhenda
Xie
Zhengyan Zhang
Zhewen Hao
Zhicheng Ma
Zhigang Yan Zhiyu

Wu 吴志宇

Zihui Gu

Zijia Zhu
Zijun Liu*
Zilin Li Ziwei
Xie Ziyang
Song Zizheng
Pan
Zhen Huang
Zhipeng Xu
Zhongyu Zhang
Zhen Zhang

在每个角色中，作者按姓名字母顺序排列。标有 * 的姓名表示已离开我们团队的人员。