DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models

Zhihong Shao[1,2*†], Peiyi Wang[1,3*†], Qihao Zhu[1,3*†], Runxin Xu[1], Junxiao Song[1]
Xiao Bi[1], Haowei Zhang[1], Mingchuan Zhang[1], Y.K. Li[1], Y. Wu[1], Daya Guo[1*]

[1]DeepSeek-AI, [2]Tsinghua University, [3]Peking University

{zhihongshao,wangpeiyi,zhuqh,guoday}@deepseek.com
https://github.com/deepseek-ai/DeepSeek-Math

Abstract

由于其复杂且结构化的特性，数学推理对语言模型构成了重大挑战。本文介绍了 DeepSeek-Math 7B，该模型在 DeepSeek-Coder-Base-v1.5 7B 的基础上，使用源自 Common Crawl 的 1200 亿个数学相关 token，结合自然语言与代码数据进行了持续预训练。在不依赖外部工具包和投票技术的情况下，DeepSeekMath 7B 在竞赛级 MATH 基准测试中取得了 51.7% 的优异成绩，性能已接近 Gemini-Ultra 和 GPT-4。基于 DeepSeekMath 7B 的 64 个样本进行自一致性推理，在 MATH 上的准确率达到 60.9%。DeepSeekMath 的数学推理能力主要归功于两个关键因素：首先，我们通过精心设计的筛选流程，充分挖掘了公开网络数据的巨大潜力；其次，我们引入了群体相对策略优化（Group Relative Policy Optimization, GRPO），这是一种近端策略优化（Proximal Policy Optimization, PPO）的变体，能够在提升数学推理能力的同时，优化 PPO 的内存占用。

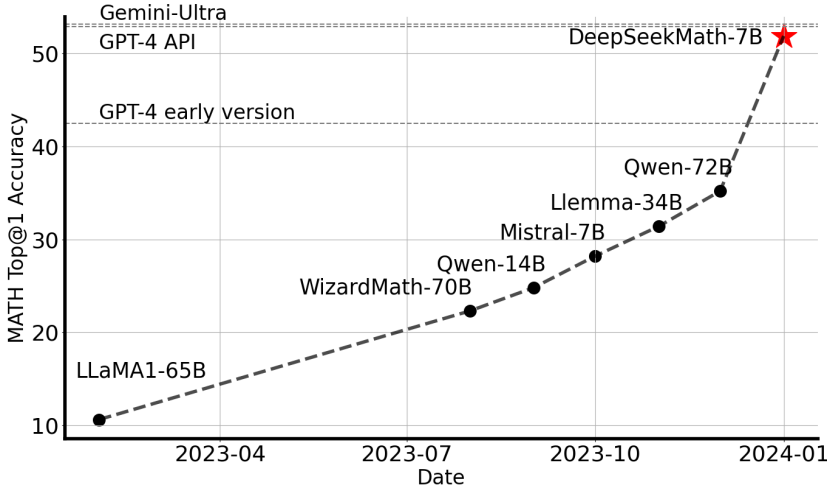

图 1 | 在不使用外部工具包和投票技术的情况下，开源模型在竞赛级 MATH 基准测试 (Hendrycks et al., 2021) 上的 Top1 准确率。

* Core contributors.
† Work done during internship at DeepSeek-AI.

1. 引言

大型语言模型（LLM）彻底改变了人工智能领域处理数学推理的方式，推动了定量推理基准 (Hendrycks et al., 2021) 和几何推理基准 (Trinh et al., 2024) 的显著进步。此外，这些模型已被证明在辅助人类解决复杂数学问题方面发挥着重要作用 (Tao, 2023)。然而，GPT-4 (OpenAI, 2023) 和 Gemini-Ultra (Anil et al., 2023) 等前沿模型并未公开，目前可获取的开源模型在性能上仍大幅落后。

在本研究中，我们介绍了 DeepSeekMath，这是一款领域专用语言模型，其数学能力显著优于现有开源模型，并在学术基准测试中接近 GPT-4 的性能水平。为实现这一目标，我们构建了 DeepSeekMath 语料库，这是一个包含 1200 亿个数学 token 的大规模高质量预训练语料库。该数据集使用基于 fastText 的分类器从 Common Crawl (CC) 中提取 (Joulin et al., 2016)。在初始迭代中，分类器使用来自 OpenWebMath (Paster et al., 2023) 的实例作为正例进行训练，同时纳入多样化的其他网页作为负例。随后，我们利用该分类器从 CC 中挖掘更多的正例，并通过人工标注进行进一步筛选。接着，使用这一增强后的数据集更新分类器以提升其性能。评估结果表明该大规模语料库质量较高，我们的基座模型 DeepSeekMath-Base 7B 在 GSM8K (Cobbe et al., 2021) 上达到 64.2%，在竞赛级 MATH 数据集 (Hendrycks et al., 2021) 上达到 36.2%，优于 Minerva 540B (Lewkowycz et al., 2022a)。此外，DeepSeekMath 语料库为多语言语料，因此我们观察到在中文数学基准测试 (Wei et al., 2023; Zhong et al., 2023) 上的性能也有所提升。我们认为，我们在数学数据处理方面的经验可为研究社区提供一个起点，未来仍有巨大的改进空间。

DeepSeekMath-Base 以 DeepSeek-Coder-Base-v1.5 7B (Guo et al., 2024) 进行初始化，因为我们发现，与通用大语言模型相比，从经过代码训练的模型出发是更好的选择。此外，我们观察到数学训练也提升了模型在 MMLU (Hendrycks et al., 2020) 和 BBH 基准测试 (Suzgun et al., 2022) 上的能力，这表明它不仅增强了模型的数学能力，还放大了通用推理能力。

预训练完成后，我们使用思维链（chain-of-thought）(Wei et al., 2022)、程序思维（program-of-thought）(Chen et al., 2022; Gao et al., 2023) 以及工具集成推理（tool-integrated reasoning）(Gou et al., 2023) 数据对 DeepSeekMath-Base 进行数学指令微调。得到的模型 DeepSeekMath-Instruct 7B 超越了所有 7B 参数规模的同类模型，并与 70B 开源指令微调模型表现相当。

此外，我们引入了组相对策略优化（Group Relative Policy Optimization, GRPO），这是一种近端策略优化（Proximal Policy Optimization, PPO）(Schulman et al., 2017) 的变体强化学习（RL）算法。GRPO 摒弃了 Critic 模型，转而通过组得分来估计基线，从而显著降低了训练资源消耗。仅使用一部分英文指令微调数据，GRPO 就在强化学习阶段使强大的 DeepSeekMath-Instruct 模型获得了显著提升，涵盖域内任务（GSM8K: 82.9% → 88.2%，MATH: 46.8% → 51.7%）和域外数学任务（例如 CMATH: 84.6% → 88.8%）。我们还提供了一个统一的范式来理解不同的方法，例如拒绝采样微调（Rejection Sampling Fine-Tuning, RFT）(Yuan et al., 2023a)、直接偏好优化（Direct Preference Optimization, DPO）(Rafailov et al., 2023)、PPO 和 GRPO。基于这一统一范式，我们发现所有这些方法均可被概念化为直接的或简化的强化学习技术。我

们还进行了大量实验，例如在线与离线训练、结果监督与过程监督、单轮与迭代强化学习等，以深入探究该范式的关键要素。最后，我们解释了为何我们的强化学习能够提升指令微调模型的性能，并基于该统一范式进一步总结了实现更高效强化学习的潜在方向。

1.1. 贡献

我们的贡献包括可扩展的数学预训练，以及对强化学习的探索与分析。

大规模数学预训练

- 我们的研究提供了令人信服的证据，表明公开可用的 Common Crawl 数据包含对数学任务有价值的信息。通过实施精心设计的筛选流程，我们成功构建了 DeepSeekMath 语料库，这是一个包含 120B token 的高质量数据集，源自经过数学内容过滤的网页，其规模几乎是 Minerva (Lewkowycz et al., 2022a) 所用数学网页的 7 倍，以及近期发布的 OpenWebMath (Paster et al., 2023) 的 9 倍。
- 我们的预训练基座模型 DeepSeekMath-Base 7B 取得了与 Minerva 540B (Lewkowycz et al., 2022a) 相当的性能，表明参数量并非数学推理能力的唯一关键因素。在高质量数据上预训练的较小规模模型同样能够取得强劲的性能。
- 我们分享了数学训练实验的发现。在数学训练之前进行代码训练，能够提升模型在使用和不使用工具的情况下解决数学问题的能力。这为长期存在的问题提供了一个部分答案：*代码训练能否提升推理能力？*我们认为确实如此，至少在数学推理方面是这样。
- 尽管在 arXiv 论文上进行训练很常见，尤其是在许多数学相关论文中，但它在本文采用的所有数学基准测试上均未带来显著改进。

强化学习的探索与分析

- 我们引入了组相对策略优化（Group Relative Policy Optimization, GRPO），这是一种高效且有效的强化学习算法。与近端策略优化（Proximal Policy Optimization, PPO）相比，GRPO 摒弃了 Critic 模型，转而通过组得分估计基线，从而显著降低了训练资源消耗。
- 我们证明了仅使用指令微调数据，GRPO 就能显著提升我们的指令微调模型 DeepSeekMath-Instruct 的性能。此外，我们观察到在强化学习过程中，域外性能也得到了提升。
- 我们提供了一个统一的范式来理解不同的方法，例如 RFT、DPO、PPO 和 GRPO。我们还进行了大量实验，例如在线与离线训练、结果监督与过程监督、单轮与迭代强化学习等，以深入探究该范式的关键要素。
- 基于我们的统一范式，我们探讨了强化学习有效性的背后原因，并总结了实现大语言模型更高效强化学习的几个潜在方向。

1.2. 评估与指标总结

- **中英文数学推理**：我们在中英文基准测试上对模型进行了全面评估，涵盖从小学到大学水平的数学问题。英文基准测试包括 GSM8K (Cobbe et al., 2021)、MATH (Hendrycks et al., 2021)、SAT (Azerbayev et al., 2023)、OCW Courses (Lewkowycz et al., 2022a) 和 MMLU-

STEM (Hendrycks et al., 2020)。中文基准测试包括 MGSM-zh (Shi et al., 2023)、CMATH (Wei et al., 2023)、Gaokao-MathCloze (Zhong et al., 2023) 以及 Gaokao-MathQA (Zhong et al., 2023)。我们评估了模型在不使用工具的情况下生成自包含文本解答的能力，以及使用 Python 解决问题的能力。

在英文基准测试上，DeepSeekMath-Base 与闭源模型 Minerva 540B (Lewkowycz et al., 2022a) 具有竞争力，并且超越了所有开源基础模型（例如 Mistral 7B (Jiang et al., 2023) 和 Llemma-34B (Azerbayev et al., 2023)），无论这些模型是否经过数学预训练，通常都领先幅度显著。值得注意的是，DeepSeekMath-Base 在中文基准测试上表现更优，这可能是因为我们没有遵循以往工作 (Azerbayev et al., 2023; Lewkowycz et al., 2022a) 仅收集英文数学预训练数据，而是同时包含了高质量的非英文数据。经过数学指令微调与强化学习后，得到的 DeepSeekMath-Instruct 和 DeepSeekMath-RL 展现出强劲的性能，首次在开源社区中在竞赛级 MATH 数据集上取得了超过 50% 的准确率。

- **形式化数学**：我们使用 (Jiang et al., 2022) 中的非形式化到形式化定理证明任务，在 miniF2F (Zheng et al., 2021) 上对 DeepSeekMath-Base 进行评估，并选用 Isabelle (Wenzel et al., 2008) 作为证明助手。DeepSeekMath-Base 展现出强大的少样本自动形式化性能。
- **自然语言理解、推理与代码**：为了全面刻画模型的通用理解、推理和编程能力，我们在大规模多任务语言理解（MMLU）基准测试 (Hendrycks et al., 2020) 上对 DeepSeekMath-Base 进行评估，该测试包含涵盖多个学科的 57 个多项选择题任务；在 BIG-Bench Hard (BBH) (Suzgun et al., 2022) 上评估，该测试包含 23 个主要需要多步推理才能解决的挑战性任务；以及在 HumanEval (Chen et al., 2021) 和 MBPP (Austin et al., 2021) 上评估，这两个基准被广泛用于评估代码语言模型。数学预训练对语言理解和推理性能均有裨益。

2. 数学预训练

2.1. 数据收集与去污染

在本节中，我们将概述从 Common Crawl 构建 DeepSeekMath 语料库的过程。如图 2 所示，我们提出了一种迭代流水线，展示了如何从 Common Crawl 中系统性地收集大规模数学语料库，该过程从一个种子语料库开始（例如，一个规模较小但高质量的数学相关数据集集合）。值得注意的是，该方法同样适用于其他领域，例如代码编程。

首先，我们选择 OpenWebMath (Paster et al., 2023)（一个高质量的数学网页文本集合）作为初始种子语料库。利用该语料库，我们训练了一个 fastText 模型 (Joulin et al., 2016)，以召回更多类似 OpenWebMath 的数学网页。具体而言，我们从种子语料库中随机选取 50 万个数据点作为正训练样本，并从 Common Crawl 中选取另外 50 万个网页作为负样本。我们使用一个开源库[1]进行训练，将向量维度配置为 256，学习率设为 0.1，词 n-gram 的最大长度设为 3，词的最小出现次数设为 3，训练轮数设为 3。为了减小原始 Common Crawl 的规模，我们采用了基于 URL 的去重和近似去重技术，最终得到 40B 个 HTML 网页。随后，我们使用 fastText 模型从

[1] `https://fasttext.cc`

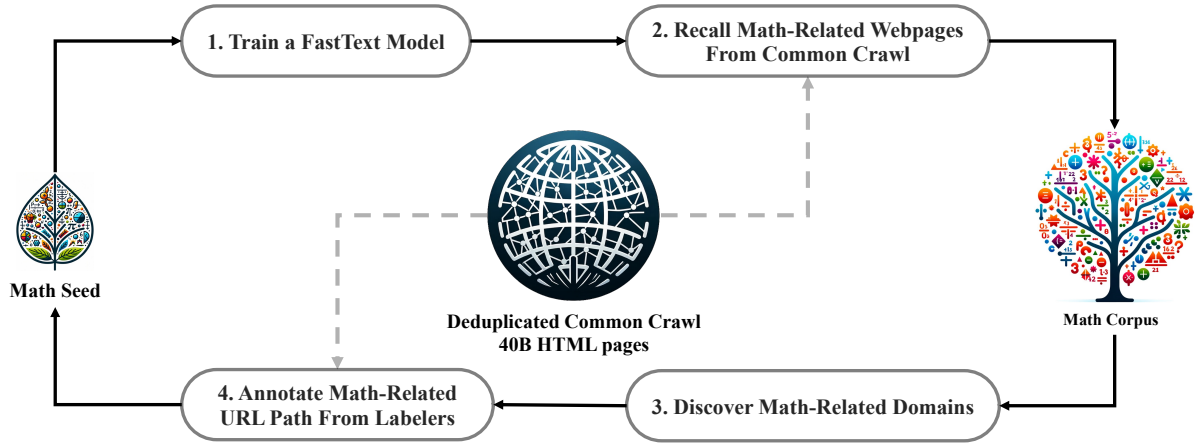

图 2 | 从 Common Crawl 收集数学网页的迭代流水线。

去重后的 Common Crawl 中召回数学网页。为了过滤掉低质量的数学内容，我们根据 fastText 模型预测的分数对收集的网页进行排序，仅保留排名靠前的网页。保留的数据量通过在前 40B、80B、120B 和 160B 个 token 上进行预训练实验来评估。在第一次迭代中，我们选择保留前 40B 个 token。

在第一次数据收集迭代之后，仍有大量数学网页未被收集，这主要是因为 fastText 模型是在一组缺乏足够多样性的正样本上训练的。因此，我们识别了额外的数学网页来源以丰富种子语料库，从而优化 fastText 模型。具体而言，我们首先将整个 Common Crawl 划分为互不相交的域名；一个域名被定义为共享相同基础 URL 的网页集合。对于每个域名，我们计算在第一次迭代中被收集的网页所占的百分比。被收集网页比例超过 10% 的域名被归类为与数学相关（例如 `mathoverflow.net`）。随后，我们手动标注这些已识别域名内与数学内容相关的 URL（例如 `mathoverflow.net/questions`）。链接到这些 URL 但尚未被收集的网页将被添加到种子语料库中。该方法使我们能够收集更多正样本，从而训练出一个改进的 fastText 模型，使其在后续迭代中能够召回更多数学数据。经过四轮数据收集，我们最终获得了 3550 万（35.5M）个数学网页，总计 1200 亿（120B）个词元（tokens）。在第四轮迭代中，我们注意到近 98% 的数据已在第三轮被收集，因此决定停止数据收集。

为避免基准测试数据污染，我们遵循 Guo et al. (2024) 的方法，过滤掉包含 GSM8K (Cobbe et al., 2021) 和 MATH (Hendrycks et al., 2021) 等英文数学基准测试以及 CMATH (Wei et al., 2023) 和 AGIEval (Zhong et al., 2023) 等中文基准测试中题目或答案的网页。过滤标准如下：任何包含与评估基准中任意子串完全匹配的 10-gram 字符串的文本片段都将从我们的数学训练语料库中移除。对于长度不足 10-gram 但至少为 3-gram 的基准文本，我们采用精确匹配来过滤受污染的网页。

2.2. 验证 DeepSeekMath 语料库的质量

我们运行预训练实验，以探究 DeepSeekMath 语料库与近期发布的数学训练语料库相比的表现：

- **MathPile** (Wang et al., 2023c)：一个多源语料库（89 亿词元），聚合自教科书、Wikipedia、ProofWiki、CommonCrawl、StackExchange 和 arXiv，其中大部分（超过 85%）来源于 arXiv；
- **OpenWebMath** (Paster et al., 2023)：经过数学内容过滤的 CommonCrawl 数据，总计 136 亿词元；
- **Proof-Pile-2** (Azerbayev et al., 2023)：一个数学语料库，由 OpenWebMath、Algebraic-Stack（103 亿词元的数学代码）和 arXiv 论文（280 亿词元）组成。在 Proof-Pile-2 上进行实验时，我们遵循 Azerbayev et al. (2023) 的方法，采用 arXiv:Web:Code 比例为 2:4:1。

2.2.1. 训练设置

我们将数学训练应用于一个具有 13 亿参数的通用预训练语言模型，该模型与 DeepSeek LLMs (DeepSeek-AI, 2024) 共享相同的架构，记为 DeepSeek-LLM 1.3B。我们在每个数学语料库上分别训练模型，训练数据量为 1500 亿词元。所有实验均使用高效且轻量级的 HAI-LLM (High-flyer, 2023) 训练框架进行。遵循 DeepSeek LLMs 的训练实践，我们使用 AdamW 优化器 (Loshchilov and Hutter, 2017)，参数设置为 $\beta_1 = 0.9$，$\beta_2 = 0.95$，以及 $\text{weight\_decay} = 0.1$，并采用多步学习率调度策略：学习率在 2,000 步预热后达到峰值，在训练过程的 80% 处下降至峰值的 31.6%，并在训练过程的 90% 处进一步下降至峰值的 10.0%。我们将学习率最大值设置为 5.3e-4，并使用 4M 词元的批次大小和 4K 的上下文长度。

数学语料库	规模	英文基准测试					中文基准测试		
		GSM8K	MATH	OCW	SAT	MMLU STEM	CMATH	Gaokao MathCloze	Gaokao MathQA
无数学训练	N/A	2.9%	3.0%	2.9%	15.6%	19.5%	12.3%	0.8%	17.9%
MathPile	8.9B	2.7%	3.3%	2.2%	12.5%	15.7%	1.2%	0.0%	2.8%
OpenWebMath	13.6B	11.5%	8.9%	3.7%	31.3%	29.6%	16.8%	0.0%	14.2%
Proof-Pile-2	51.9B	14.3%	11.2%	3.7%	43.8%	29.2%	19.9%	5.1%	11.7%
DeepSeekMath 语料库	**120.2B**	**23.8%**	**13.6%**	**4.8%**	**56.3%**	**33.1%**	**41.5%**	**5.9%**	**23.6%**

表 1 | 在不同数学语料库上训练的 DeepSeek-LLM 1.3B 的性能表现，评估采用少样本思维链提示。语料库大小使用词表大小为 100K 的分词器计算得出。

2.2.2. 评估结果

DeepSeekMath 语料库质量高，涵盖多语言数学内容，且规模最大。

- **高质量**：我们使用少样本思维链提示 Wei et al. (2022) 在 8 个数学基准测试上评估了下游性能。如表 1 所示，在 DeepSeekMath 语料库上训练的模型具有明显的性能优势。图 3 显示，在 500 亿词元（即 Proof-Pile-2 的 1 个完整 epoch）时，在 DeepSeekMath 语料库上训练的模型表现优于 Proof-Pile-2，表明 DeepSeekMath 语料库的平均质量更高。
- **多语言**：DeepSeekMath 语料库包含多种语言的数据，其中英语和中文是占比最高的两种语言。如表 1 所示，在 DeepSeekMath 语料库上进行训练能够提升中英文的数学推理性

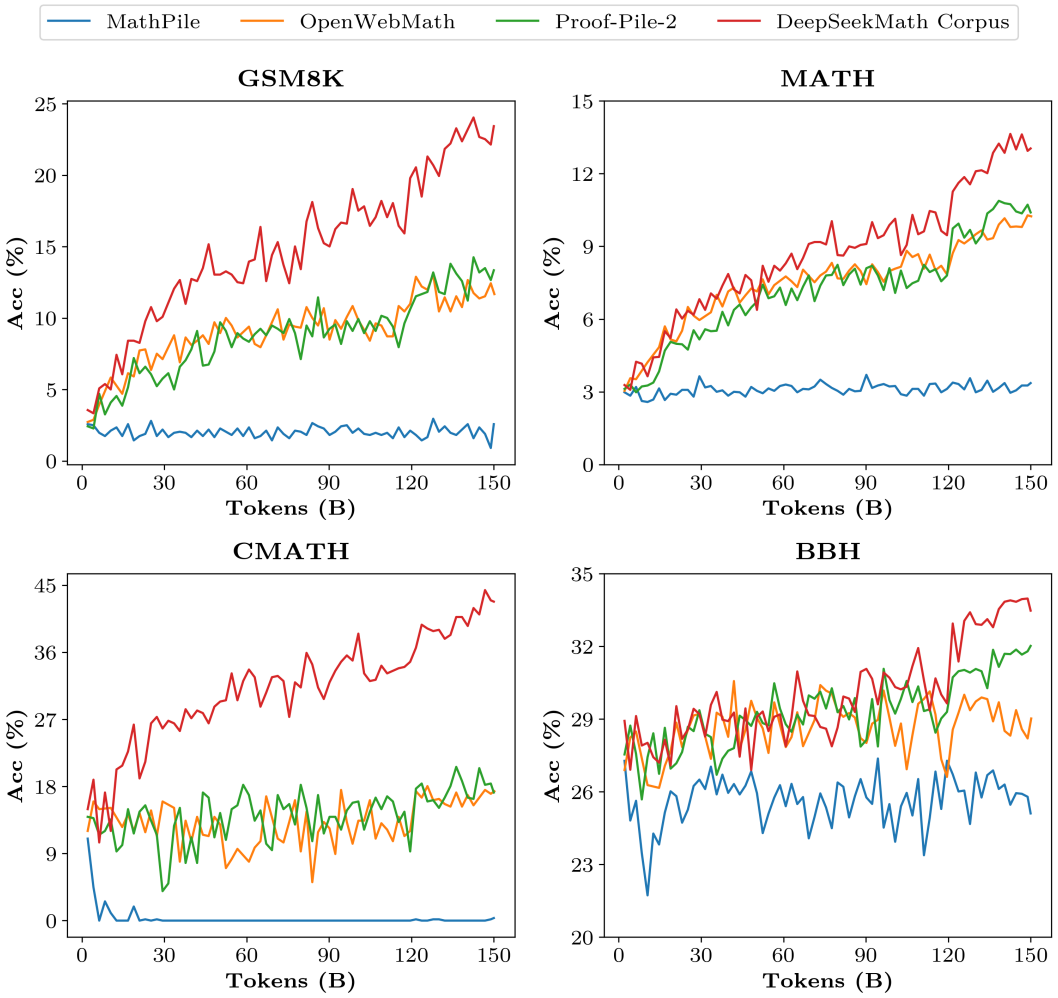

图 3 | 在不同数学语料库上训练的 DeepSeek-LLM 1.3B 的基准测试曲线。

能。相比之下，现有的数学语料库主要以英语为中心，提升效果有限，甚至可能阻碍中文数学推理的性能。

- **大规模**：DeepSeekMath 语料库的规模是现有数学语料库的数倍。如图 3 所示，DeepSeek-LLM 1.3B 在 DeepSeekMath 语料库上训练时，展现出更陡峭的学习曲线以及更持久的性能提升。相比之下，基线语料库规模小得多，且在训练过程中已被重复多个轮次，导致模型性能迅速达到平台期。

2.3. DeepSeekMath-Base 7B 的训练与评估

在本节中，我们介绍 DeepSeekMath-Base 7B，这是一个具有强大推理能力的基础模型，尤其在数学领域表现突出。我们的模型以 DeepSeek-Coder-Base-v1.5 7B (Guo et al., 2024) 为初始化起点，并训练了 500B 个 token。数据分布如下：56% 来自 DeepSeekMath 语料库，4% 来自 AlgebraicStack，10% 来自 arXiv，20% 为 Github 代码，其余 10% 为 Common Crawl 中的中英文自然语言数据。我们主要采用第 2.2.1 节中指定的训练设置，仅将学习率最大值设为 4.2e-4，并使用 10M tokens 的批次大小。

我们对 DeepSeekMath-Base 7B 的数学能力进行了全面评估，重点关注其在不依赖外部工具的情况下生成自包含数学解答的能力、使用工具解决数学问题的能力，以及进行形式化定理证明的能力。除数学领域外，我们还提供了该基础模型更全面的性能概况，包括其在自然语言理解、推理和编程技能方面的表现。

基于逐步推理的数学问题解决 我们使用少样本思维链提示（few-shot chain-of-thought prompting）(Wei et al., 2022)，在八个中英文基准上评估了 DeepSeekMath-Base 解决数学问题的性能。这些基准涵盖了定量推理（如 GSM8K (Cobbe et al., 2021)、MATH (Hendrycks et al., 2021) 和 CMATH (Wei et al., 2023)）与选择题（如 MMLU-STEM (Hendrycks et al., 2020) 和 Gaokao-MathQA (Zhong et al., 2023)），覆盖了从小学到大学难度的多个数学领域。

如表 2 所示，在开源基础模型中（包括广泛使用的通用模型 Mistral 7B (Jiang et al., 2023) 以及近期发布、在 Proof-Pile-2 (Azerbayev et al., 2023) 上经过数学训练的 Llemma 34B (Azerbayev et al., 2023)），DeepSeekMath-Base 7B 在所有八个基准上的性能均位居前列。值得注意的是，在竞赛级 MATH 数据集上，DeepSeekMath-Base 的绝对性能超过现有开源基础模型 10% 以上，并优于基于 PaLM (Lewkowycz et al., 2022b) 构建、规模为其 77 倍且进一步在数学文本上训练的闭源基础模型 Minerva 540B (Lewkowycz et al., 2022a)。

模型	规模	英文基准					中文基准		
		GSM8K	MATH	OCW	SAT	MMLU STEM	CMATH	Gaokao MathCloze	Gaokao MathQA
闭源基础模型									
Minerva	7B	16.2%	14.1%	7.7%	-	35.6%	-	-	-
Minerva	62B	52.4%	27.6%	12.0%	-	53.9%	-	-	-
Minerva	540B	58.8%	33.6%	17.6%	-	63.9%	-	-	-
开源基础模型									
Mistral	7B	40.3%	14.3%	9.2%	71.9%	51.1%	44.9%	5.1%	23.4%
Llemma	7B	37.4%	18.1%	6.3%	59.4%	43.1%	43.4%	11.9%	23.6%
Llemma	34B	54.0%	25.3%	10.3%	71.9%	52.9%	56.1%	11.9%	26.2%
DeepSeekMath-Base	7B	**64.2%**	**36.2%**	**15.4%**	**84.4%**	**56.5%**	**71.7%**	**20.3%**	**35.3%**

表 2 | DeepSeekMath-Base 7B 与强基础模型在中英文数学基准上的对比。模型均使用思维链提示进行评估。Minerva 的结果引用自 Lewkowycz et al. (2022a)。

基于工具使用的数学问题解决 我们使用少样本程序思维提示（few-shot program-of-thought prompting）(Chen et al., 2022; Gao et al., 2023)，在 GSM8K 和 MATH 上评估了程序辅助的数学推理能力。模型被提示通过编写 Python 程序来解决每个问题，其中可利用 *math* 和 *sympy* 等库进行复杂计算。程序的执行结果将被作为最终答案进行评估。如表 3 所示，DeepSeekMath-Base 7B 的性能优于先前的最先进模型 Llemma 34B。

Model	Size	使用工具解题		非形式化到形式化证明	
		GSM8K+Python	MATH+Python	miniF2F-valid	miniF2F-test
Mistral	7B	48.5%	18.2%	18.9%	18.0%
CodeLlama	7B	27.1%	17.2%	16.3%	17.6%
CodeLlama	34B	52.7%	23.5%	18.5%	18.0%
Llemma	7B	41.0%	18.6%	20.6%	22.1%
Llemma	34B	64.6%	26.3%	21.0%	21.3%
DeepSeekMath-Base	7B	**66.9%**	**31.4%**	**25.8%**	**24.6%**

表 3 | 基座模型在使用工具解决数学问题以及使用 Isabelle 进行非形式化到形式化定理证明能力上的少样本评估。

形式化数学 形式化证明自动化有助于确保数学证明的准确性与可靠性并提升效率，近年来受到越来越多的关注。我们在 (Jiang et al., 2022) 提出的非形式化到形式化证明任务上评估了 DeepSeekMath-Base 7B，该任务旨在根据非形式化陈述、该陈述的形式化对应版本以及非形式化证明来生成形式化证明。我们在面向形式化奥林匹克级别数学的基准测试 miniF2F (Zheng et al., 2021) 上进行评估，并通过少样本提示为每个问题生成 Isabelle 形式化证明。遵循 Jiang et al. (2022) 的方法，我们利用模型生成证明大纲，并执行现成的自动化证明器 Sledgehammer (Paulson, 2010) 以补充缺失的细节。如表 3 所示，DeepSeekMath-Base 7B 在证明自动形式化方面表现出强劲的性能。

Model	Size	MMLU	BBH	HumanEval (Pass@1)	MBPP (Pass@1)
Mistral	7B	**62.4%**	55.7%	28.0%	41.4%
DeepSeek-Coder-Base-v1.5†	7B	42.9%	42.9%	40.2%	52.6%
DeepSeek-Coder-Base-v1.5	7B	49.1%	55.2%	**43.2%**	**60.4%**
DeepSeekMath-Base	7B	54.9%	**59.5%**	40.9%	52.6%

表 4 | 在自然语言理解、推理和代码基准测试上的评估结果。DeepSeek-Coder-Base-v1.5† 是学习率衰减前的检查点，用于训练 DeepSeekMath-Base。在 MMLU 和 BBH 上，我们使用少样本思维链提示。在 HumanEval 和 MBPP 上，我们分别在零样本和少样本设置下评估模型性能。

自然语言理解、推理与代码 我们在 MMLU (Hendrycks et al., 2020) 上评估模型的自然语言理解能力，在 BBH (Suzgun et al., 2022) 上评估推理能力，并在 HumanEval (Chen et al., 2021) 和 MBPP (Austin et al., 2021) 上评估代码生成能力。如表 4 所示，与其前身 DeepSeek-Coder-Base-v1.5 (Guo et al., 2024) 相比，DeepSeekMath-Base 7B 在 MMLU 和 BBH 上的性能显著提升，这表明数学训练对语言理解和推理具有积极的促进作用。此外，通过在持续训练中引入代码词元，DeepSeekMath-Base 7B 有效保持了 DeepSeek-Coder-Base-v1.5 在这两个代码基准测试上的性能。总体而言，在这三个推理和代码基准测试上，DeepSeekMath-Base 7B 显著优于通用模型 Mistral 7B (Jiang et al., 2023)。

3. 监督微调

3.1. SFT 数据构建

我们构建了一个数学指令微调数据集，涵盖来自不同数学领域且难度各异的英文和中文问题：每个问题均配有思维链（CoT）(Wei et al., 2022)、程序思维（PoT）(Chen et al., 2022; Gao et al., 2023) 以及工具集成推理格式 (Gou et al., 2023) 的解答。训练样本总数为 776K。

- **英文数学数据集**：我们为 GSM8K 和 MATH 问题标注了工具集成解答，并采用了 MathInstruct (Yue et al., 2023) 的子集以及 Lila-OOD (Mishra et al., 2022) 的训练集（其中问题通过 CoT 或 PoT 求解）。我们的英文语料涵盖了代数、概率、数论、微积分和几何等多样化的数学领域。
- **中文数学数据集**：我们收集了涵盖线性方程等 76 个子主题的中文 K-12 数学问题，并为解答标注了 CoT 和工具集成推理两种格式。

3.2. DeepSeekMath-Instruct 7B 的训练与评估

在本节中，我们介绍了 DeepSeekMath-Instruct 7B，该模型基于 DeepSeekMath-Base 进行了数学指令微调。训练样本被随机拼接，直到达到 4K tokens 的最大上下文长度。我们以 256 的批次大小和 5e-5 的恒定学习率对模型进行了 500 步的训练。

我们在 4 个中英文定量推理基准上，评估了模型在使用和不使用工具情况下的数学性能。我们将我们的模型与当时的领先模型进行了基准对比：

- **闭源模型**包括：(1) GPT 系列，其中 GPT-4 (OpenAI, 2023) 和 GPT-4 Code Interpreter [2] 能力最强，(2) Gemini Ultra 和 Pro (Anil et al., 2023)，(3) Inflection-2 (Inflection AI, 2023)，(4) Grok-1 [3]，以及中国公司近期发布的模型，包括 (5) Baichuan-3 [4]，(6) GLM 系列中的最新模型 GLM-4 [5] (Du et al., 2022)。这些模型均为通用模型，其中大多数已经过一系列的对齐流程。
- **开源模型**包括：通用模型如 (1) DeepSeek-LLM-Chat 67B (DeepSeek-AI, 2024)、(2) Qwen 72B (Bai et al., 2023)、(3) SeaLLM-v2 7B (Nguyen et al., 2023) 和 (4) ChatGLM3 6B (ChatGLM3 Team, 2023)，以及在数学方面有所增强的模型，包括 (5) InternLM2-Math 20B [6]，该模型基于 InternLM2 构建，并经过数学训练和指令微调，(6) Math-Shepherd-Mistral 7B，该模型使用过程监督奖励模型对 Mistral 7B (Jiang et al., 2023) 应用了 PPO 训练 (Schulman et al., 2017)，(7) WizardMath 系列 (Luo et al., 2023)，该系列使用 evolve-instruct（即使用 AI 演化指令的指令微调变体）和 PPO 训练提升了 Mistral 7B 和 Llama-2 70B (Touvron et al., 2023) 的数学推理能力，训练问题主要来源于 GSM8K 和 MATH，(8)

[2] https://openai.com/blog/chatgpt-plugins#code-interpreter

[3] https://x.ai/model-card

[4] https://www.baichuan-ai.com

[5] https://open.bigmodel.cn/dev/api#glm-4

[6] https://github.com/InternLM/InternLM-Math

MetaMath 70B (Yu et al., 2023)，该模型是在 GSM8K 和 MATH 的增强版本上微调的 Llama-2 70B，(9) ToRA 34B Gou et al. (2023)，该模型是微调后的 CodeLlama 34B，用于进行工具集成的数学推理，(10) MAmmoTH 70B (Yue et al., 2023)，该模型是在 MathInstruct 上指令微调的 Llama-2 70B。

如表 5 所示，在禁止使用工具的评估设置下，DeepSeekMath-Instruct 7B 展现了强大的逐步推理能力。值得注意的是，在竞赛级 MATH 数据集上，我们的模型以至少 9% 的绝对优势超越了所有开源模型和大多数专有模型（例如 Inflection-2 和 Gemini Pro）。即使对于规模大得多的模型（例如 Qwen 72B）或通过数学专项强化学习专门增强的模型（例如 WizardMath-v1.1 7B），这一结论依然成立。尽管 DeepSeekMath-Instruct 在 MATH 数据集上的表现与中国专有模型 GLM-4 和 Baichuan-3 相当，但仍不及 GPT-4 和 Gemini Ultra。

在允许模型结合自然语言推理与基于程序的工具使用来解决问题的评估设置下，DeepSeekMath-Instruct 7B 在 MATH 数据集上的准确率接近 60%，超越了所有现有的开源模型。在其他基准测试上，我们的模型与先前规模最大 10 倍的领先模型 DeepSeek-LLM-Chat 67B 具有相当的竞争力。

4. 强化学习

4.1. 组相对策略优化

强化学习（RL）已被证明在监督微调（SFT）阶段之后能进一步提升大语言模型（LLM）的数学推理能力 (Luo et al., 2023; Wang et al., 2023b)。在本节中，我们介绍我们提出的一种高效且有效的强化学习算法：组相对策略优化（GRPO）。

4.1.1. 从 PPO 到 GRPO

近端策略优化（PPO）(Schulman et al., 2017) 是一种广泛适用于大语言模型强化学习微调阶段的 Actor-Critic 强化学习算法 (Ouyang et al., 2022)。具体而言，它通过最大化以下代理目标来优化大语言模型：

$$\mathcal{J}_{PPO}(\theta) = \mathbb{E}[q \sim P(Q), o \sim \pi_{\theta_{old}}(O|q)] \frac{1}{|o|} \sum_{t=1}^{|o|} \min \left[\frac{\pi_\theta(o_t|q,o_{<t})}{\pi_{\theta_{old}}(o_t|q,o_{<t})} A_t, \text{clip}\left(\frac{\pi_\theta(o_t|q,o_{<t})}{\pi_{\theta_{old}}(o_t|q,o_{<t})}, 1-\varepsilon, 1+\varepsilon \right) A_t \right], \quad (1)$$

其中 π_θ 和 $\pi_{\theta_{old}}$ 分别为当前策略模型和旧策略模型，q, o 分别表示从问题数据集和旧策略 $\pi_{\theta_{old}}$ 中采样的问题与输出。ε 是 PPO 中为稳定训练而引入的与裁剪相关的超参数。A_t 为优势函数，其基于奖励 $\{r_{\geq t}\}$ 和已学习的价值函数 V_ψ，通过应用广义优势估计（GAE）(Schulman et al., 2015) 计算得出。因此，在 PPO 中，价值函数需要与策略模型一同训练；为缓解奖励模型的过度优化问题，标准做法是在每个 token 的奖励中加入来自参考模型的逐 token KL 惩罚项 (Ouyang et al., 2022)，即：

$$r_t = r_\varphi(q, o_{\leq t}) - \beta \log \frac{\pi_\theta(o_t|q,o_{<t})}{\pi_{ref}(o_t|q,o_{<t})}, \quad (2)$$

其中 r_φ 为奖励模型，π_{ref} 为参考模型（通常为初始 SFT 模型），β 为 KL 惩罚系数。

模型	参数量	英文基准		中文基准	
		GSM8K	MATH	MGSM-zh	CMATH
思维链推理					
闭源模型					
Gemini Ultra	-	94.4%	53.2%	-	-
GPT-4	-	92.0%	52.9%	-	86.0%
Inflection-2	-	81.4%	34.8%	-	-
GPT-3.5	-	80.8%	34.1%	-	73.8%
Gemini Pro	-	86.5%	32.6%	-	-
Grok-1	-	62.9%	23.9%	-	-
Baichuan-3	-	88.2%	49.2%	-	-
GLM-4	-	87.6%	47.9%	-	-
开源模型					
InternLM2-Math	20B	82.6%	37.7%	-	-
Qwen	72B	78.9%	35.2%	-	-
Math-Shepherd-Mistral	7B	84.1%	33.0%	-	-
WizardMath-v1.1	7B	83.2%	33.0%	-	-
DeepSeek-LLM-Chat	67B	84.1%	32.6%	74.0%	80.3%
MetaMath	70B	82.3%	26.6%	66.4%	70.9%
SeaLLM-v2	7B	78.2%	27.5%	64.8%	-
ChatGLM3	6B	72.3%	25.7%	-	-
WizardMath-v1.0	70B	81.6%	22.7%	64.8%	65.4%
DeepSeekMath-Instruct	7B	82.9%	46.8%	73.2%	84.6%
DeepSeekMath-RL	7B	**88.2%**	**51.7%**	**79.6%**	**88.8%**
工具集成推理					
闭源模型					
GPT-4 Code Interpreter	-	97.0%	69.7%	-	-
开源模型					
InternLM2-Math	20B	80.7%	54.3%	-	-
DeepSeek-LLM-Chat	67B	86.7%	51.1%	76.4%	85.4%
ToRA	34B	80.7%	50.8%	41.2%	53.4%
MAmmoTH	70B	76.9%	41.8%	-	-
DeepSeekMath-Instruct	7B	83.7%	57.4%	72.0%	84.3%
DeepSeekMath-RL	7B	**86.7%**	**58.8%**	**78.4%**	**87.6%**

表 5 | 开放源与闭源模型在英文和中文基准测试上结合思维链与工具集成推理的性能表现。灰色 分数表示基于 32 个候选答案的多数投票结果；其余为 Top1 分数。DeepSeekMath-RL 7B 超越了所有 7B 至 70B 规模的开源模型，以及大多数闭源模型。尽管 DeepSeekMath-RL 7B 仅在 GSM8K 和 MATH 的思维链格式指令微调数据上进行了进一步训练，但它在所有基准测试上均优于 DeepSeekMath-Instruct 7B。

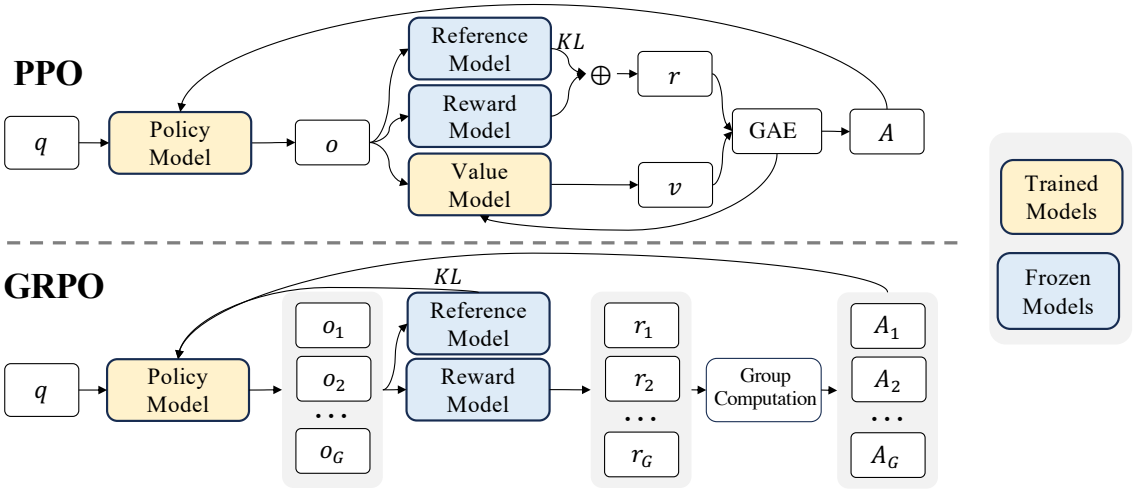

图 4 | PPO 与我们提出的 GRPO 的示意图。GRPO 摒弃了价值模型，转而通过组得分估计基线，从而显著降低了训练资源消耗。

由于 PPO 中使用的价值函数通常是一个与策略模型规模相当的独立模型，这会带来巨大的内存和计算负担。此外，在强化学习训练过程中，价值函数在优势计算中作为基线以实现方差缩减。然而在大语言模型的语境下，奖励模型通常仅对最后一个 token 分配奖励分数，这可能会使训练在每个 token 上都准确的价值函数变得复杂。为解决这一问题，如图 4 所示，我们提出了组相对策略优化（GRPO）。该算法免去了 PPO 中额外的价值函数近似需求，转而将针对同一问题生成的多个采样输出的平均奖励作为基线。具体而言，对于每个问题 q，GRPO 从旧策略 $\pi_{\theta_{old}}$ 中采样一组输出 $\{o_1, o_2, \cdots, o_G\}$，然后通过最大化以下目标函数来优化策略模型：

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{old}}(O|q)]$$
$$\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left\{\min\left[\frac{\pi_\theta(o_{i,t}|q,o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q,o_{i,<t})}\hat{A}_{i,t}, \text{clip}\left(\frac{\pi_\theta(o_{i,t}|q,o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q,o_{i,<t})}, 1-\varepsilon, 1+\varepsilon\right)\hat{A}_{i,t}\right] - \beta\mathbb{D}_{KL}\left[\pi_\theta||\pi_{ref}\right]\right\}, \tag{3}$$

其中 ε 和 β 为超参数，$\hat{A}_{i,t}$ 是仅基于组内输出相对奖励计算的优势值，具体细节将在后续小节中详述。GRPO 采用的组相对优势计算方法与奖励模型固有的比较特性高度契合，因为奖励模型通常是在针对同一问题的不同输出进行比较的数据集上进行训练的。此外需注意，GRPO 并未在奖励中直接添加 KL 惩罚项，而是通过将训练策略与参考策略之间的 KL 散度直接加入损失函数来实现正则化，从而避免了 $\hat{A}_{i,t}$ 计算的复杂化。与 (2) 中使用的 KL 惩罚项不同，我们采用以下无偏估计量来估计 KL 散度 (Schulman, 2020)：

$$\mathbb{D}_{KL}\left[\pi_\theta||\pi_{ref}\right] = \frac{\pi_{ref}(o_{i,t}|q,o_{i,<t})}{\pi_\theta(o_{i,t}|q,o_{i,<t})} - \log\frac{\pi_{ref}(o_{i,t}|q,o_{i,<t})}{\pi_\theta(o_{i,t}|q,o_{i,<t})} - 1, \tag{4}$$

该估计量保证恒为正。

4.1.2. 基于 *GRPO* 的结果监督强化学习

形式上，对于每个问题 q，从旧策略模型 $\pi_{\theta_{old}}$ 中采样一组输出 $\{o_1, o_2, \cdots, o_G\}$。随后使用奖励模型对输出进行评分，相应地得到 G 个奖励 $\mathbf{r} = \{r_1, r_2, \cdots, r_G\}$。接着，通过减去组平均值并除以组标准差对这些奖励进行归一化。结果监督在每个输出 o_i 的末尾提供归一化奖励，并将输出中

Algorithm 1 迭代组相对策略优化

输入初始策略模型 $\pi_{\theta_{\text{init}}}$；奖励模型 r_φ；任务提示 $\mathcal{D}$；超参数 ε, β, μ

策略模型 $\pi_\theta \leftarrow \pi_{\theta_{\text{init}}}$
for 迭代次数 = 1, ..., I **do**
  参考模型 $\pi_{ref} \leftarrow \pi_\theta$
  for 步骤 = 1, ..., M **do**
    从 $\mathcal{D}$ 中采样一个批次 $\mathcal{D}_b$
    更新旧策略模型 $\pi_{\theta_{old}} \leftarrow \pi_\theta$
    对每个问题 $q \in \mathcal{D}_b$ 采样 G 个输出 $\{o_i\}_{i=1}^{G} \sim \pi_{\theta_{old}}(\cdot \mid q)$
    运行 r_φ 计算每个采样输出 o_i 的奖励 $\{r_i\}_{i=1}^{G}$
    通过组相对优势估计计算 o_i 第 t 个 token 的 $\hat{A}_{i,t}$
    for GRPO 迭代 = 1, ..., μ **do**
      通过最大化 GRPO 目标函数（公式 21）更新策略模型 π_θ
  使用重放机制持续训练更新 r_φ

输出 π_θ

所有 token 的优势 $\hat{A}_{i,t}$ 设为该归一化奖励，即 $\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$，随后通过最大化公式 (3) 定义的目标函数来优化策略。

4.1.3. 基于 *GRPO* 的过程监督强化学习

结果监督仅在每个输出的末尾提供奖励，这在复杂的数学任务中可能不足以高效地监督策略。遵循 Wang et al. (2023b) 的工作，我们也探索了过程监督，它在每个推理步骤的末尾提供奖励。形式上，给定问题 q 和 G 个采样输出 $\{o_1, o_2, \cdots, o_G\}$，使用过程奖励模型对输出的每个步骤进行评分，得到相应的奖励：$\mathbf{R} = \{\{r_1^{index(1)}, \cdots, r_1^{index(K_1)}\}, \cdots, \{r_G^{index(1)}, \cdots, r_G^{index(K_G)}\}\}$，其中 $index(j)$ 是第 j 步的结束 token 索引，K_i 是第 i 个输出中的总步数。我们同样使用平均值和标准差对这些奖励进行归一化，即 $\widetilde{r}_i^{index(j)} = \frac{r_i^{index(j)} - \text{mean}(\mathbf{R})}{\text{std}(\mathbf{R})}$。随后，过程监督将每个 token 的优势计算为后续步骤归一化奖励的总和，即 $\hat{A}_{i,t} = \sum_{index(j) \geq t} \widetilde{r}_i^{index(j)}$，然后通过最大化公式 (3) 定义的目标函数来优化策略。

4.1.4. 基于 *GRPO* 的迭代强化学习

随着强化学习训练过程的推进，旧的奖励模型可能不足以监督当前的策略模型。因此，我们也探索了基于 GRPO 的迭代强化学习。如算法 1 所示，在迭代 GRPO 中，我们基于策略模型的采样结果为奖励模型生成新的训练集，并使用包含 10% 历史数据的重放机制持续训练旧的奖励模型。随后，我们将参考模型设置为当前策略模型，并使用新的奖励模型持续训练策略模型。

4.2. DeepSeekMath-RL 的训练与评估

我们基于 DeepSeekMath-Instruct 7B 进行强化学习训练。强化学习的训练数据来自 SFT 数据中与 GSM8K 和 MATH 相关的思维链格式问题，共计约 14.4 万道题目。我们排除了其他 SFT 问题，以研究强化学习对在整个 RL 阶段缺乏数据的基准测试的影响。我们遵循 (Wang et al., 2023b) 构建奖励模型的训练集。我们基于 DeepSeekMath-Base 7B 以 2e-5 的学习率训练初始奖励模型。对于 GRPO，我们将策略模型的学习率设置为 1e-6。KL 系数为 0.04。对于每个问

题，我们采样 64 个输出。最大长度设置为 1024，训练批次大小为 1024。策略模型在每次探索阶段后仅进行一次更新。我们遵循 DeepSeekMath-Instruct 7B 的评估方式，在基准测试上对 DeepSeekMath-RL 7B 进行评估。对于 DeepSeekMath-RL 7B，采用思维链推理的 GSM8K 和 MATH 可视为域内任务，其余所有基准测试均可视为域外任务。

表 5 展示了开源与闭源模型在英文和中文基准测试上结合思维链与工具集成推理的性能。我们发现：1) 利用思维链推理，DeepSeekMath-RL 7B 在 GSM8K 和 MATH 上的准确率分别达到 88.2% 和 51.7%。该性能超越了 7B 至 70B 参数规模的所有开源模型，以及大多数闭源模型。2) 关键在于，DeepSeekMath-RL 7B 仅基于 DeepSeekMath-Instruct 7B，使用 GSM8K 和 MATH 的思维链格式指令微调数据进行训练。尽管训练数据范围有限，它在所有评估指标上均优于 DeepSeekMath-Instruct 7B，彰显了强化学习的有效性。

5. 讨论

在本节中，我们将分享我们在预训练和强化学习实验中的发现。

5.1. 预训练中的经验教训

我们首先分享预训练方面的经验。除非另有说明，我们将遵循第 2.2.1 节中概述的训练设置。值得注意的是，在本节中提到 DeepSeekMath 语料库时，我们使用的是数据收集流程第二次迭代中获得的 89B token 数据集。

5.1.1. 代码训练有助于数学推理

一个流行但尚未得到验证的假设认为，代码训练能够提升推理能力。我们尝试对此给出部分解答，特别是在数学领域：代码训练能够提升模型在使用和不使用工具时的数学推理能力。

为了研究代码训练如何影响数学推理，我们实验了以下两阶段训练和单阶段训练设置：

两阶段训练

- **400B Token 代码训练 → 150B Token 数学训练**：我们使用 400B 代码 token 训练 DeepSeek-LLM 1.3B，随后使用 150B 数学 token 进行训练；
- **400B Token 通用训练 → 150B Token 数学训练**：作为对照实验，我们在训练的第一阶段使用通用 token（从 DeepSeek-AI 构建的大规模通用语料库中采样）替代代码 token，旨在探究代码 token 相较于通用 token 在提升数学推理方面的优势。

单阶段训练

- **150B Token 数学训练**：我们使用 150B 数学 token 训练 DeepSeek-LLM 1.3B；
- **400B 代码 Token 与 150B 数学 Token 混合训练**：在代码训练后进行数学训练会降低编码性能。我们探究当代码 token 与数学 token 混合进行单阶段训练时，是否仍能提升数学推理能力，并缓解灾难性遗忘问题。

训练设置	训练 Token 数			不使用工具			使用工具	
	通用	代码	数学	GSM8K	MATH	CMATH	GSM8K+Python	MATH+Python
无持续训练	–	–	–	2.9%	3.0%	12.3%	2.7%	2.3%
两阶段训练								
阶段 1：通用训练	400B	–	–	2.9%	3.2%	14.8%	3.3%	2.3%
阶段 2：数学训练	–	–	150B	19.1%	14.4%	37.2%	14.3%	6.7%
阶段 1：代码训练	–	400B	–	5.9%	3.6%	19.9%	12.4%	10.0%
阶段 2：数学训练	–	–	150B	**21.9%**	**15.3%**	**39.7%**	17.4%	9.4%
单阶段训练								
数学训练	–	–	150B	20.5%	13.1%	37.6%	11.4%	6.5%
代码与数学混合训练	–	400B	150B	17.6%	12.1%	36.3%	**19.7%**	**13.5%**

训练设置	训练 Token 数			MMLU	BBH	HumanEval (Pass@1)	MBPP (Pass@1)
	通用	代码	数学				
无持续训练	–	–	–	24.5%	28.1%	12.2%	13.0%
两阶段训练							
阶段 1：通用训练	400B	–	–	25.9%	27.7%	15.2%	13.6%
阶段 2：数学训练	–	–	150B	33.1%	32.7%	12.8%	13.2%
阶段 1：代码训练	–	400B	–	25.0%	31.5%	25.0%	**40.0%**
阶段 2：数学训练	–	–	150B	**36.2%**	35.3%	12.2%	17.0%
单阶段训练							
数学训练	–	–	150B	32.3%	32.5%	11.6%	13.2%
代码与数学混合训练	–	400B	150B	33.5%	**35.6%**	**29.3%**	39.4%

表 6 | 探究代码和数学训练的不同设置如何影响模型在语言理解、推理和编程方面的性能。我们使用 DeepSeek-LLM 1.3B 进行实验。我们使用少样本思维链提示在 MMLU 和 BBH 上评估模型。在 HumanEval 和 MBPP 上，我们分别进行零样本和少样本评估。

结果 表 **??** 和表 6 展示了不同训练设置下的下游性能。

在两阶段训练和单阶段训练设置下，代码训练均有益于程序辅助的数学推理。如表 **??** 所示，在两阶段训练设置下，仅进行代码训练已能显著提升使用 Python 解决 GSM8K 和 MATH 问题的能力。第二阶段的数学训练带来了进一步的提升。有趣的是，在单阶段训练设置下，混合代码 token 和数学 token 有效缓解了两阶段训练引发的灾难性遗忘问题，同时也协同提升了编程能力（表 6）和程序辅助的数学推理能力（表 **??**）。

代码训练也提升了不使用工具时的数学推理能力。在两阶段训练设置下，初始阶段的代码训练已带来适度的提升。它还提高了后续数学训练的效率，最终取得了最佳性能。然而，将代码 token 和数学 token 结合进行单阶段训练会削弱不使用工具时的数学推理能力。一种推测是，DeepSeek-LLM 1.3B 由于规模有限，缺乏同时充分吸收代码和数学数据的能力。

模型	规模	ArXiv 语料库	英文基准测试					中文基准测试		
			GSM8K	MATH	OCW	SAT	MMLU STEM	CMATH	Gaokao MathCloze	Gaokao MathQA
DeepSeek-LLM	1.3B	无数学训练	2.9%	3.0%	2.9%	15.6%	19.5%	12.3%	0.8%	17.9%
		MathPile	2.7%	3.3%	2.2%	12.5%	15.7%	1.2%	0.0%	2.8%
		ArXiv-RedPajama	3.3%	3.4%	4.0%	9.4%	9.0%	7.4%	0.8%	2.3%
DeepSeek-Coder-Base-v1.5	7B	无数学训练	29.0%	12.5%	6.6%	40.6%	38.1%	45.9%	5.9%	21.1%
		MathPile	23.6%	11.5%	7.0%	46.9%	35.8%	37.9%	4.2%	25.6%
		ArXiv-RedPajama	28.1%	11.1%	7.7%	50.0%	35.2%	42.6%	7.6%	24.8%

表 7 | 数学训练对不同 arXiv 数据集的影响。模型性能通过少样本思维链提示进行评估。

ArXiv 语料库	miniF2F-valid	miniF2F-test
无数学训练	20.1%	21.7%
MathPile	16.8%	16.4%
ArXiv-RedPajama	14.8%	11.9%

表 8 | 数学训练对不同 arXiv 语料库的影响，基础模型为 DeepSeek-Coder-Base-v1.5 7B。我们在 Isabelle 中评估非形式化到形式化的证明能力。

5.1.2. ArXiv 论文似乎对提升数学推理能力无效

ArXiv 论文通常作为数学预训练数据的一部分被纳入 (Azerbayev et al., 2023; Lewkowycz et al., 2022a; Polu and Sutskever, 2020; Wang et al., 2023c)。然而，关于其对数学推理能力影响的详细分析尚未得到充分开展。或许与直觉相悖，根据我们的实验结果，ArXiv 论文似乎对提升数学推理能力并无显著效果。我们使用经过不同处理流程的 ArXiv 语料库，对不同规模的模型进行了实验，包括 DeepSeek-LLM 1.3B 和 DeepSeek-Coder-Base-v1.5 7B (Guo et al., 2024)：

- **MathPile** (Wang et al., 2023c)：一个包含 89 亿 token 的语料库，采用清洗和过滤启发式规则构建，其中超过 85% 为科学类 ArXiv 论文；
- **ArXiv-RedPajama** (Computer, 2023)：完整的 ArXiv LaTeX 文件，已移除导言区、注释、宏和参考文献，总计 280 亿 token。

在我们的实验中，我们分别在每个 ArXiv 语料库上对 DeepSeek-LLM 1.3B 训练了 1500 亿 token，对 DeepSeek-Coder-Base-v1.5 7B 训练了 400 亿 token。结果表明，ArXiv 论文似乎对提升数学推理能力无效。当仅在 ArXiv 语料库上进行训练时，这两种模型在本研究采用的各种不同复杂度的数学基准测试中均未表现出显著改善，甚至出现性能下降。这些基准测试包括定量推理数据集（如 GSM8K 和 MATH，见表 7）、多项选择题挑战（如 MMLU-STEM，见表 7）以及形式化数学任务（如 miniF2F，见表 8）。

然而，这一结论存在局限性，应持审慎态度。我们尚未研究以下方面：

- ArXiv token 对本研究未涵盖的特定数学相关任务的影响，例如定理非形式化（即将形式化陈述或证明转换为其非形式化版本）；
- ArXiv token 与其他类型数据结合时的效果；
- 在更大规模的模型上，ArXiv 论文的优势是否会显现出来。

方法	数据源	奖励函数	梯度系数
SFT	$q, o \sim P_{sft}(Q, O)$	-	1
RFT	$q \sim P_{sft}(Q)$, $o \sim \pi_{sft}(O\|q)$	规则	公式 10
DPO	$q \sim P_{sft}(Q)$, $o^+, o^- \sim \pi_{sft}(O\|q)$	规则	公式 14
在线 RFT	$q \sim P_{sft}(Q)$, $o \sim \pi_\theta(O\|q)$	规则	公式 10
PPO	$q \sim P_{sft}(Q)$, $o \sim \pi_\theta(O\|q)$	模型	公式 18
GRPO	$q \sim P_{sft}(Q)$, $\{o_i\}_{i=1}^{G} \sim \pi_\theta(O\|q)$	模型	公式 21

表 9 | 不同方法的数据源与梯度系数。P_{sft} 表示监督微调数据集的数据分布。$\pi_{\theta_{sft}}$ 和 π_θ 分别表示监督微调模型和在线训练过程中的实时策略模型。

因此，仍需进一步探索，我们将此留待未来研究。

5.2. 强化学习的启示

5.2.1. 迈向统一范式

在本节中，我们提出一个统一范式来分析不同的训练方法（如 SFT、RFT、DPO、PPO、GRPO），并进一步开展实验以探索该统一范式中的关键因素。一般而言，某种训练方法关于参数 θ 的梯度可表示为：

$$\nabla_\theta \mathcal{J}_{\mathcal{A}}(\theta) = \mathbb{E}[\underbrace{(q,o) \sim \mathcal{D}}_{Data\ \ Source}] \left(\frac{1}{|o|} \sum_{t=1}^{|o|} \underbrace{GC_{\mathcal{A}}(q,o,t,\pi_{rf})}_{Gradient\ \ Coefficient} \nabla_\theta \log \pi_\theta(o_t|q,o_{<t}) \right). \tag{5}$$

该范式包含三个关键组件：1) *数据源* $\mathcal{D}$，决定训练数据；2) *奖励函数* π_{rf}，作为训练奖励信号的来源；3) *算法* $\mathcal{A}$：将训练数据和奖励信号处理为梯度系数 GC，该系数决定对数据的惩罚或强化程度。我们基于该统一范式分析了几种代表性方法：

- **监督微调（SFT）**：SFT 在人工筛选的 SFT 数据上对预训练模型进行微调。
- **拒绝采样微调（RFT）**：RFT 基于 SFT 问题从 SFT 模型采样的过滤输出，对 SFT 模型进行进一步微调。RFT 根据答案的正确性对输出进行过滤。
- **直接偏好优化（DPO）**：DPO 使用成对 DPO 损失，在从 SFT 模型采样的增强输出上对其进行微调，从而进一步优化 SFT 模型。
- **在线拒绝采样微调（Online RFT）**：与 RFT 不同，Online RFT 使用 SFT 模型初始化策略模型，并通过使用从实时策略模型采样的增强输出进行微调来优化它。
- **PPO/GRPO**: PPO/GRPO 使用 SFT 模型初始化策略模型，并利用从实时策略模型采样的输出对其进行强化。

我们在表 9 中总结了这些方法的组成部分。更详细的推导过程请参阅附录 A.1。

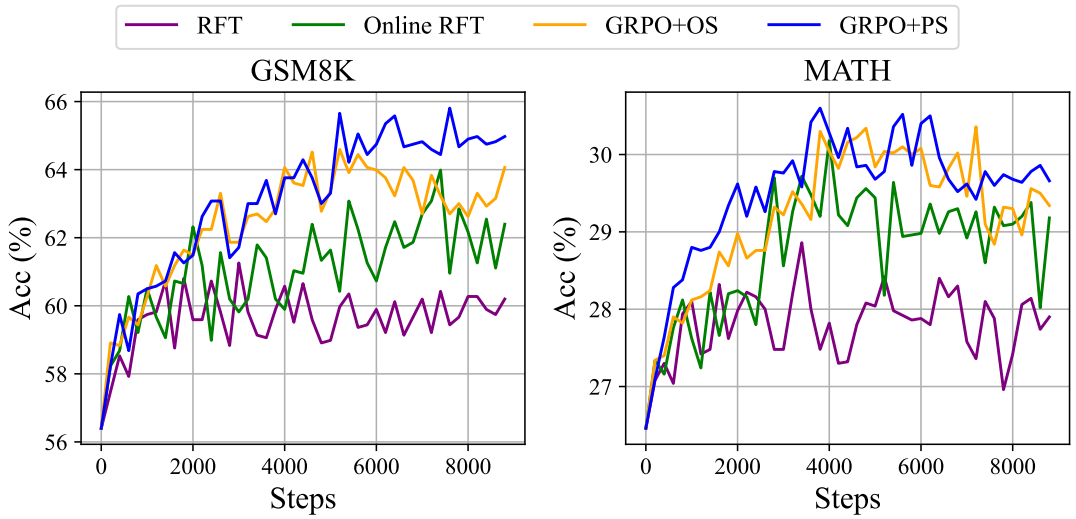

图 5 | 使用不同方法进一步训练的 DeepSeekMath-Instruct 1.3B 模型在两个基准测试上的性能。

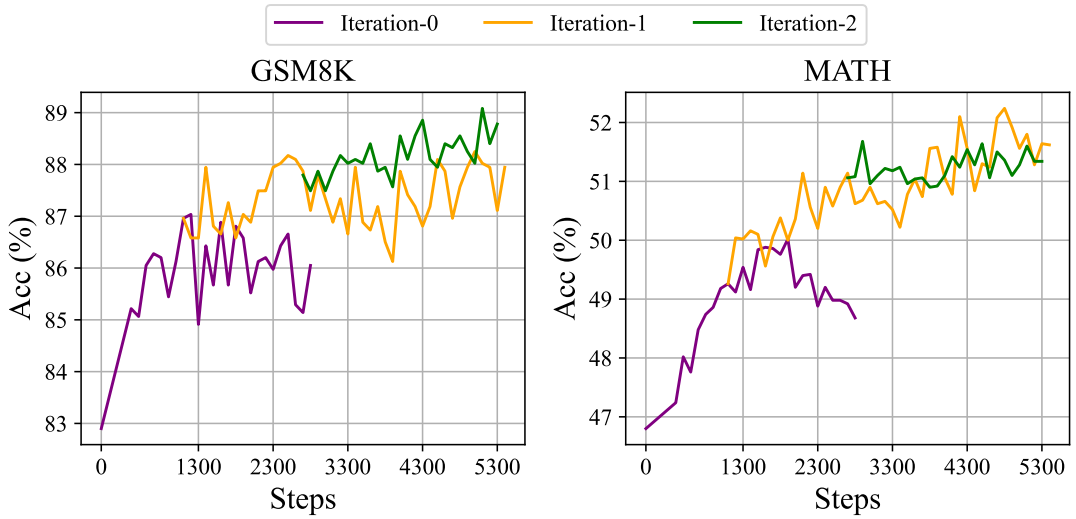

图 6 | DeepSeekMath-Instruct 7B 迭代强化学习在两个基准测试上的性能。

关于数据源的观察 我们将数据源分为两类：在线采样和离线采样。在线采样表示训练数据来自实时训练策略模型的探索结果，而离线采样表示训练数据来自初始 SFT 模型的采样结果。RFT 和 DPO 遵循离线模式，而 Online RFT 和 GRPO 遵循在线模式。

如图 5 所示，我们发现 Online RFT 在两个基准测试上均显著优于 RFT。具体而言，Online RFT 在训练初期与 RFT 表现相当，但在后期获得了绝对优势，证明了在线训练的优越性。这是符合直觉的，因为在初始阶段，Actor 模型与 SFT 模型表现出高度相似性，采样数据仅显示出微小差异。然而，在后期阶段，从 Actor 模型采样的数据将表现出更显著的差异，实时数据采样将带来更大的优势。

关于梯度系数的观察 该算法将奖励信号转化为梯度系数，以更新模型参数。在我们的实验中，我们将奖励函数分为“规则（Rule）”和“模型（Model）”两类。“规则”指根据答案的正确性判

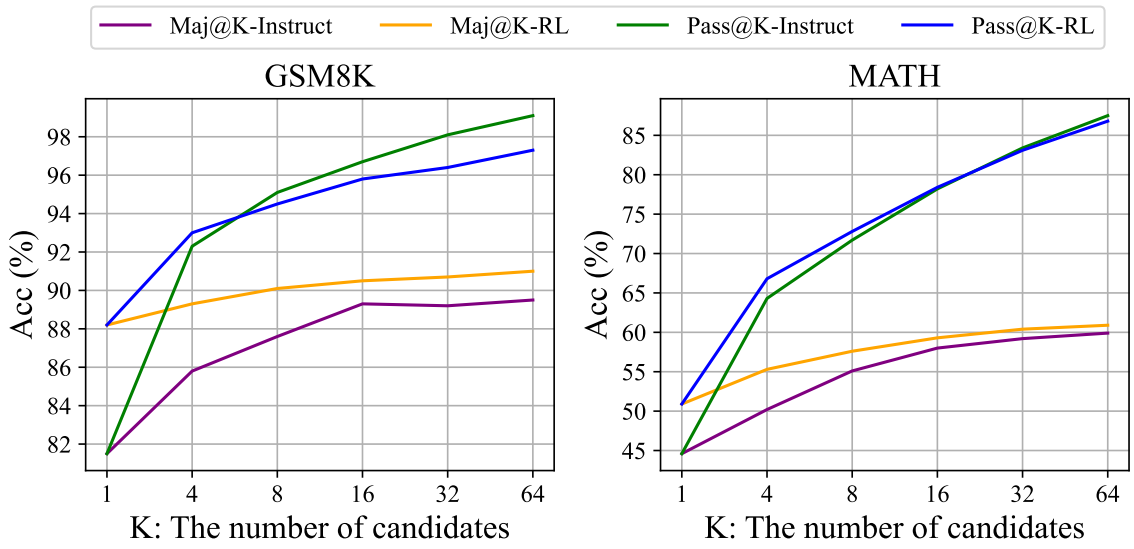

图 7 | SFT 和 RL DeepSeekMath 7B 在 GSM8K 和 MATH 数据集上的 Maj@K 与 Pass@K 指标（温度 0.7）。值得注意的是，RL 提升了 Maj@K 但未提升 Pass@K。

断回复的质量，“模型”指我们训练一个奖励模型来为每个回复打分。奖励模型的训练数据基于规则判断。公式 10 和 21 突出了 GRPO 与 Online RFT 之间的一个关键区别：GRPO 独特地根据奖励模型提供的奖励值调整其梯度系数。这使得能够根据奖励值的不同大小对回复进行差异化的强化与惩罚。相比之下，Online RFT 缺乏这一特性；它不对错误回复进行惩罚，而是以相同的强度统一强化所有正确答案的回复。

如图 5 所示，GRPO 的表现优于 Online RFT，从而凸显了调整正负梯度系数的有效性。此外，与 GRPO+OS 相比，GRPO+PS 表现出更优的性能，表明使用细粒度、步骤感知（step-aware）梯度系数的好处。此外，我们探索了迭代强化学习，在实验中我们进行了两轮迭代。如图 6 所示，我们注意到迭代强化学习显著提升了性能，尤其是在第一次迭代时。

5.2.2. 为何强化学习有效？

在本文中，我们基于指令微调数据的一个子集进行强化学习，并在指令微调模型的基础上实现了显著的性能提升。为了进一步阐明强化学习有效的原因，我们在两个基准测试上评估了 Instruct 模型和 RL 模型的 Pass@K 与 Maj@K 准确率。如图 7 所示，RL 提升了 Maj@K 的性能，但未提升 Pass@K。这些发现表明，RL 通过使输出分布更加稳健来提升模型的整体性能，换言之，**这种提升似乎归因于提高了 TopK 中正确回复的比例，而非基础能力的增强**。类似地，(Wang et al., 2023a) 指出了 SFT 模型在推理任务中存在的**对齐偏差问题（misalignment problem）**，表明 SFT 模型的推理性能可以通过一系列偏好对齐策略得到提升 (Song et al., 2023; Wang et al., 2023a; Yuan et al., 2023b)。

5.2.3. 如何实现更有效的强化学习？

我们证明了 RL 在数学推理任务中表现良好。我们还提供了一个统一的范式来理解不同的代表性训练方法。在该范式中，所有方法均被概念化为直接的或简化的 RL 技术。如公式 5 所总结，存在三个关键组成部分：数据源、算法和奖励函数。我们针对这三个组成部分提出了一些潜在的未来研究方向。

数据源 数据源是所有训练方法的原材料。在强化学习（RL）的语境下，我们特指数据源为带有从策略模型中采样输出的未标注问题。在本文中，我们仅使用指令微调阶段的问题，并采用基础的核采样（nucleus sampling）来采样输出。我们认为，这可能是我们的强化学习流程仅能提升 Maj@K 性能的一个潜在原因。未来，我们将结合**高级采样（解码）策略**（例如基于树搜索的方法 (Yao et al., 2023)），在分布外（out-of-distribution）问题提示上探索我们的强化学习流程。此外，决定策略模型探索效率的**高效推理技术** (Kwon et al., 2023; Leviathan et al., 2023; Xia et al., 2023, 2024) 也发挥着极其重要的作用。

算法 算法将数据和奖励信号处理为梯度系数，以更新模型参数。基于公式 5，在某种程度上，目前的所有方法都完全**信任（TRUST）**奖励函数的信号，以增加或降低特定 token 的条件概率。然而，无法保证奖励信号始终可靠，尤其是在极其复杂的任务中。例如，即使是由经过良好训练的标注员仔细标注的 PRM800K 数据集 (Lightman et al., 2023)，仍然包含约 20% 的错误标注[7]。为此，我们将探索对噪声奖励信号具有鲁棒性的强化学习算法。我们相信此类**由弱至强（WEAK-TO-STRONG）** (Burns et al., 2023) 的对齐方法将为学习算法带来根本性的变革。

奖励函数 奖励函数是训练信号的来源。在强化学习中，奖励函数通常是神经奖励模型。我们认为奖励模型存在三个重要的研究方向：1) **如何提升奖励模型的泛化能力**。奖励模型必须能够有效泛化以处理分布外问题和高级解码输出；否则，强化学习可能仅仅稳定了大语言模型（LLM）的分布，而非提升其基础能力；2) **如何反映奖励模型的不确定性**。不确定性有望在弱奖励模型与由弱至强的学习算法之间充当连接桥梁；3) **如何高效构建高质量的过程奖励模型**，以为推理过程提供细粒度的训练信号 (Lightman et al., 2023; Wang et al., 2023b).

6. 结论、局限性与未来工作

我们提出了 DeepSeekMath，其在竞赛级 MATH 基准测试上的表现优于所有开源模型，并接近闭源模型的性能。DeepSeekMath 以 DeepSeek-Coder-v1.5 7B 为初始化基础，并进行了 5000 亿 token 的持续训练，其中训练数据的重要组成部分是来自 Common Crawl 的 1200 亿数学 token。我们广泛的消融实验表明，网页数据在高质量数学数据方面具有巨大潜力，而 arXiv 的收益可能不及我们预期。我们引入了群组相对策略优化（Group Relative Policy Optimization, GRPO），这是近端策略优化（Proximal Policy Optimization, PPO）的一种变体，能够在降低内存消耗的

[7] https://github.com/openai/prm800k/issues/12#issuecomment-1728491852

同时显著提升数学推理能力。实验结果表明，即使 DeepSeekMath-Instruct 7B 在基准测试中已取得高分，GRPO 依然有效。我们还提供了一个统一的范式来理解一系列方法，并总结了实现更高效强化学习的几个潜在方向。

尽管 DeepSeekMath 在定量推理基准测试中取得了令人瞩目的成绩，但其在几何和定理证明方面的能力相对弱于闭源模型。例如，在我们的初步测试中，该模型无法处理与三角形和椭圆相关的问题，这可能表明预训练和微调阶段存在数据选择偏差。此外，受限于模型规模，DeepSeekMath 在少样本（few-shot）能力上不如 GPT-4。GPT-4 能够通过少样本输入提升性能，而 DeepSeekMath 在零样本和少样本评估中表现相似。未来，我们将进一步优化人工设计的数据选择流程，以构建更多高质量的预训练语料。此外，我们将探索大语言模型更高效强化学习的潜在方向（第 5.2.3 节）。

参考文献

R. Anil, S. Borgeaud, Y. Wu, J. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican, D. Silver, S. Petrov, M. Johnson, I. Antonoglou, J. Schrittwieser, A. Glaese, J. Chen, E. Pitler, T. P. Lillicrap, A. Lazaridou, O. Firat, J. Molloy, M. Isard, P. R. Barham, T. Hennigan, B. Lee, F. Viola, M. Reynolds, Y. Xu, R. Doherty, E. Collins, C. Meyer, E. Rutherford, E. Moreira, K. Ayoub, M. Goel, G. Tucker, E. Piqueras, M. Krikun, I. Barr, N. Savinov, I. Danihelka, B. Roelofs, A. White, A. Andreassen, T. von Glehn, L. Yagati, M. Kazemi, L. Gonzalez, M. Khalman, J. Sygnowski, and et al. Gemini: A family of highly capable multimodal models. CoRR, abs/2312.11805, 2023. doi: 10.48550/ARXIV.2312.11805. URL https://doi.org/10.48550/arXiv.2312.11805.

J. Austin, A. Odena, M. Nye, M. Bosma, H. Michalewski, D. Dohan, E. Jiang, C. Cai, M. Terry, Q. Le, et al. Program synthesis with large language models. arXiv preprint arXiv:2108.07732, 2021.

Z. Azerbayev, H. Schoelkopf, K. Paster, M. D. Santos, S. McAleer, A. Q. Jiang, J. Deng, S. Biderman, and S. Welleck. Llemma: An open language model for mathematics. arXiv preprint arXiv:2310.10631, 2023.

J. Bai, S. Bai, Y. Chu, Z. Cui, K. Dang, X. Deng, Y. Fan, W. Ge, Y. Han, F. Huang, et al. Qwen technical report. arXiv preprint arXiv:2309.16609, 2023.

C. Burns, P. Izmailov, J. H. Kirchner, B. Baker, L. Gao, L. Aschenbrenner, Y. Chen, A. Ecoffet, M. Joglekar, J. Leike, et al. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. arXiv preprint arXiv:2312.09390, 2023.

ChatGLM3 Team. Chatglm3 series: Open bilingual chat llms, 2023. URL https://github.com/THUDM/ChatGLM3.

M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. de Oliveira Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman, A. Ray, R. Puri, G. Krueger, M. Petrov, H. Khlaaf, G. Sastry, P. Mishkin, B. Chan, S. Gray, N. Ryder, M. Pavlov, A. Power, L. Kaiser, M. Bavarian, C. Winter, P. Tillet, F. P. Such, D. Cummings, M. Plappert, F. Chantzis, E. Barnes, A. Herbert-Voss, W. H. Guss, A. Nichol, A. Paino, N. Tezak, J. Tang, I. Babuschkin, S. Balaji, S. Jain, W. Saunders, C. Hesse, A. N. Carr, J. Leike, J. Achiam, V. Misra, E. Morikawa, A. Radford, M. Knight, M. Brundage, M. Murati, K. Mayer, P. Welinder, B. McGrew, D. Amodei, S. McCandlish, I. Sutskever, and W. Zaremba. Evaluating large language models trained on code. CoRR, abs/2107.03374, 2021. URL https://arxiv.org/abs/2107.03374.

W. Chen, X. Ma, X. Wang, and W. W. Cohen. Program of thoughts prompting: Disentangling

computation from reasoning for numerical reasoning tasks. CoRR, abs/2211.12588, 2022. doi: 10.48550/ARXIV.2211.12588. URL https://doi.org/10.48550/arXiv.2211.12588.

K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, et al. Training verifiers to solve math word problems. arXiv preprint arXiv:2110.14168, 2021.

T. Computer. Redpajama: an open dataset for training large language models, Oct. 2023. URL https://github.com/togethercomputer/RedPajama-Data.

DeepSeek-AI. Deepseek LLM: scaling open-source language models with longtermism. CoRR, abs/2401.02954, 2024. doi: 10.48550/ARXIV.2401.02954. URL https://doi.org/10.48550/arXiv.2401.02954.

Z. Du, Y. Qian, X. Liu, M. Ding, J. Qiu, Z. Yang, and J. Tang. Glm: General language model pretraining with autoregressive blank infilling. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 320–335, 2022.

L. Gao, A. Madaan, S. Zhou, U. Alon, P. Liu, Y. Yang, J. Callan, and G. Neubig. PAL: program-aided language models. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, editors, International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA, volume 202 of Proceedings of Machine Learning Research, pages 10764–10799. PMLR, 2023. URL https://proceedings.mlr.press/v202/gao23f.html.

Z. Gou, Z. Shao, Y. Gong, Y. Shen, Y. Yang, M. Huang, N. Duan, and W. Chen. Tora: A tool-integrated reasoning agent for mathematical problem solving. CoRR, abs/2309.17452, 2023. doi: 10.48550/ARXIV.2309.17452. URL https://doi.org/10.48550/arXiv.2309.17452.

D. Guo, Q. Zhu, D. Yang, Z. Xie, K. Dong, W. Zhang, G. Chen, X. Bi, Y. Wu, Y. K. Li, F. Luo, Y. Xiong, and W. Liang. Deepseek-coder: When the large language model meets programming – the rise of code intelligence, 2024.

D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt. Measuring massive multitask language understanding. arXiv preprint arXiv:2009.03300, 2020.

D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang, D. Song, and J. Steinhardt. Measuring mathematical problem solving with the math dataset. arXiv preprint arXiv:2103.03874, 2021.

High-flyer. Hai-llm: 高效且轻量的大模型训练工具, 2023. URL https://www.high-flyer.cn/en/blog/hai-llm.

Inflection AI. Inflection-2, 2023. URL https://inflection.ai/inflection-2.

A. Q. Jiang, S. Welleck, J. P. Zhou, W. Li, J. Liu, M. Jamnik, T. Lacroix, Y. Wu, and G. Lample. Draft, sketch, and prove: Guiding formal theorem provers with informal proofs. arXiv preprint arXiv:2210.12283, 2022.

A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. d. l. Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, et al. Mistral 7b. arXiv preprint arXiv:2310.06825, 2023.

A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou, and T. Mikolov. Fasttext. zip: Compressing text classification models. arXiv preprint arXiv:1612.03651, 2016.

W. Kwon, Z. Li, S. Zhuang, Y. Sheng, L. Zheng, C. H. Yu, J. E. Gonzalez, H. Zhang, and I. Stoica. Efficient memory management for large language model serving with pagedattention. In Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles, 2023.

Y. Leviathan, M. Kalman, and Y. Matias. Fast inference from transformers via speculative decoding. In International Conference on Machine Learning, pages 19274–19286. PMLR, 2023.

A. Lewkowycz, A. Andreassen, D. Dohan, E. Dyer, H. Michalewski, V. Ramasesh, A. Slone, C. Anil, I. Schlag, T. Gutman-Solo, et al. Solving quantitative reasoning problems with language models. Advances in Neural Information Processing Systems, 35:3843–3857, 2022a.

A. Lewkowycz, A. Andreassen, D. Dohan, E. Dyer, H. Michalewski, V. V. Ramasesh, A. Slone, C. Anil, I. Schlag, T. Gutman-Solo, Y. Wu, B. Neyshabur, G. Gur-Ari, and V. Misra. Solving quantitative reasoning problems with language models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022, 2022b. URL http://papers.nips.cc/paper_files/paper/2022/hash/18abbeef8cfe9203fdf9053c9c4fe191-Abstract-Conference.html.

H. Lightman, V. Kosaraju, Y. Burda, H. Edwards, B. Baker, T. Lee, J. Leike, J. Schulman, I. Sutskever, and K. Cobbe. Let's verify step by step. arXiv preprint arXiv:2305.20050, 2023.

I. Loshchilov and F. Hutter. Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101, 2017.

H. Luo, Q. Sun, C. Xu, P. Zhao, J. Lou, C. Tao, X. Geng, Q. Lin, S. Chen, and D. Zhang. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. arXiv preprint arXiv:2308.09583, 2023.

S. Mishra, M. Finlayson, P. Lu, L. Tang, S. Welleck, C. Baral, T. Rajpurohit, O. Tafjord, A. Sabharwal, P. Clark, and A. Kalyan. LILA: A unified benchmark for mathematical reasoning. In Y. Goldberg, Z. Kozareva, and Y. Zhang, editors, Proceedings of the

2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022, pages 5807–5832. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.EMNLP-MAIN.392. URL https://doi.org/10.18653/v1/2022.emnlp-main.392.

X. Nguyen, W. Zhang, X. Li, M. M. Aljunied, Q. Tan, L. Cheng, G. Chen, Y. Deng, S. Yang, C. Liu, H. Zhang, and L. Bing. Seallms - large language models for southeast asia. CoRR, abs/2312.00738, 2023. doi: 10.48550/ARXIV.2312.00738. URL https://doi.org/10.48550/arXiv.2312.00738.

OpenAI. GPT4 technical report. arXiv preprint arXiv:2303.08774, 2023.

L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. Training language models to follow instructions with human feedback. Advances in Neural Information Processing Systems, 35:27730–27744, 2022.

K. Paster, M. D. Santos, Z. Azerbayev, and J. Ba. Openwebmath: An open dataset of high-quality mathematical web text. CoRR, abs/2310.06786, 2023. doi: 10.48550/ARXIV.2310.06786. URL https://doi.org/10.48550/arXiv.2310.06786.

L. C. Paulson. Three years of experience with sledgehammer, a practical link between automatic and interactive theorem provers. In R. A. Schmidt, S. Schulz, and B. Konev, editors, Proceedings of the 2nd Workshop on Practical Aspects of Automated Reasoning, PAAR-2010, Edinburgh, Scotland, UK, July 14, 2010, volume 9 of EPiC Series in Computing, pages 1–10. EasyChair, 2010. doi: 10.29007/TNFD. URL https://doi.org/10.29007/tnfd.

S. Polu and I. Sutskever. Generative language modeling for automated theorem proving. CoRR, abs/2009.03393, 2020. URL https://arxiv.org/abs/2009.03393.

R. Rafailov, A. Sharma, E. Mitchell, S. Ermon, C. D. Manning, and C. Finn. Direct preference optimization: Your language model is secretly a reward model. 2023.

J. Schulman. Approximating kl divergence, 2020. URL http://joschu.net/blog/kl-approx.html.

J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel. High-dimensional continuous control using generalized advantage estimation. arXiv preprint arXiv:1506.02438, 2015.

J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347, 2017.

F. Shi, M. Suzgun, M. Freitag, X. Wang, S. Srivats, S. Vosoughi, H. W. Chung, Y. Tay, S. Ruder, D. Zhou, D. Das, and J. Wei. Language models are multilingual chain-of-thought reasoners. In The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali,

Rwanda, May 1-5, 2023. OpenReview.net, 2023. URL https://openreview.net/pdf?id=fR3wGCk-IXp.

F. Song, B. Yu, M. Li, H. Yu, F. Huang, Y. Li, and H. Wang. Preference ranking optimization for human alignment. arXiv preprint arXiv:2306.17492, 2023.

M. Suzgun, N. Scales, N. Schärli, S. Gehrmann, Y. Tay, H. W. Chung, A. Chowdhery, Q. V. Le, E. H. Chi, D. Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. arXiv preprint arXiv:2210.09261, 2022.

T. Tao. Embracing change and resetting expectations, 2023. URL https://unlocked.microsoft.com/ai-anthology/terence-tao/.

H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. Canton-Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom. Llama 2: Open foundation and fine-tuned chat models. CoRR, abs/2307.09288, 2023. doi: 10.48550/arXiv.2307.09288. URL https://doi.org/10.48550/arXiv.2307.09288.

T. H. Trinh, Y. Wu, Q. V. Le, H. He, and T. Luong. Solving olympiad geometry without human demonstrations. Nature, 625(7995):476–482, 2024.

P. Wang, L. Li, L. Chen, F. Song, B. Lin, Y. Cao, T. Liu, and Z. Sui. Making large language models better reasoners with alignment. arXiv preprint arXiv:2309.02144, 2023a.

P. Wang, L. Li, Z. Shao, R. Xu, D. Dai, Y. Li, D. Chen, Y. Wu, and Z. Sui. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. CoRR, abs/2312.08935, 2023b.

Z. Wang, R. Xia, and P. Liu. Generative AI for math: Part I - mathpile: A billion-token-scale pretraining corpus for math. CoRR, abs/2312.17120, 2023c. doi: 10.48550/ARXIV.2312.17120. URL https://doi.org/10.48550/arXiv.2312.17120.

J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. H. Chi, Q. V. Le, and D. Zhou. Chain-of-thought prompting elicits reasoning in large language models. In NeurIPS, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html.

T. Wei, J. Luan, W. Liu, S. Dong, and B. Wang. Cmath: Can your language model pass chinese elementary school math test?, 2023.

M. Wenzel, L. C. Paulson, and T. Nipkow. The isabelle framework. In O. A. Mohamed, C. A. Muñoz, and S. Tahar, editors, Theorem Proving in Higher Order Logics, 21st International Conference, TPHOLs 2008, Montreal, Canada, August 18-21, 2008. Proceedings, volume 5170 of Lecture Notes in Computer Science, pages 33–38. Springer, 2008. doi: 10.1007/978-3-540-71067-7_7. URL https://doi.org/10.1007/978-3-540-71067-7_7.

H. Xia, T. Ge, P. Wang, S.-Q. Chen, F. Wei, and Z. Sui. Speculative decoding: Exploiting speculative execution for accelerating seq2seq generation. In H. Bouamor, J. Pino, and K. Bali, editors, Findings of the Association for Computational Linguistics: EMNLP 2023, pages 3909–3925, Singapore, Dec. 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.257. URL https://aclanthology.org/2023.findings-emnlp.257.

H. Xia, Z. Yang, Q. Dong, P. Wang, Y. Li, T. Ge, T. Liu, W. Li, and Z. Sui. Unlocking efficiency in large language model inference: A comprehensive survey of speculative decoding. arXiv preprint arXiv:2401.07851, 2024.

S. Yao, D. Yu, J. Zhao, I. Shafran, T. L. Griffiths, Y. Cao, and K. Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. arXiv preprint arXiv:2305.10601, 2023.

L. Yu, W. Jiang, H. Shi, J. Yu, Z. Liu, Y. Zhang, J. T. Kwok, Z. Li, A. Weller, and W. Liu. Metamath: Bootstrap your own mathematical questions for large language models. CoRR, abs/2309.12284, 2023. doi: 10.48550/ARXIV.2309.12284. URL https://doi.org/10.48550/arXiv.2309.12284.

Z. Yuan, H. Yuan, C. Li, G. Dong, C. Tan, and C. Zhou. Scaling relationship on learning mathematical reasoning with large language models. arXiv preprint arXiv:2308.01825, 2023a.

Z. Yuan, H. Yuan, C. Tan, W. Wang, S. Huang, and F. Huang. Rrhf: Rank responses to align language models with human feedback without tears. arXiv preprint arXiv:2304.05302, 2023b.

X. Yue, X. Qu, G. Zhang, Y. Fu, W. Huang, H. Sun, Y. Su, and W. Chen. Mammoth: Building math generalist models through hybrid instruction tuning. CoRR, abs/2309.05653, 2023. doi: 10.48550/ARXIV.2309.05653. URL https://doi.org/10.48550/arXiv.2309.05653.

K. Zheng, J. M. Han, and S. Polu. Minif2f: a cross-system benchmark for formal olympiad-level mathematics. arXiv preprint arXiv:2109.00110, 2021.

W. Zhong, R. Cui, Y. Guo, Y. Liang, S. Lu, Y. Wang, A. Saied, W. Chen, and N. Duan. AGIEval: A human-centric benchmark for evaluating foundation models. CoRR, abs/2304.06364, 2023. doi: 10.48550/arXiv.2304.06364. URL https://doi.org/10.48550/arXiv.2304.06364.

A. 附录

A.1. 强化学习分析

我们提供了各种方法（包括 SFT、RFT、在线 RFT、DPO、PPO 和 GRPO）中数据源和梯度系数（算法与奖励函数）的详细推导。

A.1.1. 监督微调

监督微调的目标是最大化以下目标函数：

$$\mathcal{J}_{SFT}(\theta) = \mathbb{E}[q, o \sim P_{sft}(Q, O)] \left(\frac{1}{|o|} \sum_{t=1}^{|o|} \log \pi_\theta(o_t | q, o_{<t}) \right). \tag{6}$$

$\mathcal{J}_{SFT}(\theta)$ 的梯度为：

$$\nabla_\theta \mathcal{J}_{SFT} = \mathbb{E}[q, o \sim P_{sft}(Q, O)] \left(\frac{1}{|o|} \sum_{t=1}^{|o|} \nabla_\theta \log \pi_\theta(o_t | q, o_{<t}) \right). \tag{7}$$

数据源：用于 SFT 的数据集。奖励函数：可视为人类选择。梯度系数：始终设为 1。

A.1.2. 拒绝采样微调

拒绝采样微调首先针对每个问题从监督微调的大语言模型中采样多个输出，然后在带有正确答案的采样输出上训练大语言模型。形式上，RFT 的目标是最大化以下目标函数：

$$\mathcal{J}_{RFT}(\theta) = \mathbb{E}[q \sim P_{sft}(Q), o \sim \pi_{sft}(O|q)] \left(\frac{1}{|o|} \sum_{t=1}^{|o|} \mathbb{I}(o) \log \pi_\theta(o_t | q, o_{<t}) \right). \tag{8}$$

$\mathcal{J}_{RFT}(\theta)$ 的梯度为：

$$\nabla_\theta \mathcal{J}_{RFT}(\theta) = \mathbb{E}[q \sim P_{sft}(Q), o \sim \pi_{sft}(O|q)] \left(\frac{1}{|o|} \sum_{t=1}^{|o|} \mathbb{I}(o) \nabla_\theta \log \pi_\theta(o_t | q, o_{<t}) \right). \tag{9}$$

数据来源：SFT 数据集中的问题，输出由 SFT 模型采样得到。奖励函数：规则（判断答案是否正确）。梯度系数：

$$GC_{RFT}(q, o, t) = \mathbb{I}(o) = \begin{cases} 1 & o \text{ 的答案正确} \\ 0 & o \text{ 的答案错误} \end{cases} \tag{10}$$

A.1.3. 在线拒绝采样微调

RFT 与在线 RFT 的唯一区别在于，在线 RFT 的输出是从实时策略模型 π_θ 中采样得到的，而非来自 SFT 模型 $\pi_{\theta_{sft}}$。因此，在线 RFT 的梯度为：

$$\nabla_\theta \mathcal{J}_{OnRFT}(\theta) = \mathbb{E}[q \sim P_{sft}(Q), o \sim \pi_\theta(O|q)] \left(\frac{1}{|o|} \sum_{t=1}^{|o|} \mathbb{I}(o) \nabla_\theta \log \pi_\theta(o_t|q, o_{<t}) \right). \tag{11}$$

A.1.4. 直接偏好优化（*DPO*）

DPO 的目标函数为：

$$\mathcal{J}_{DPO}(\theta) = \mathbb{E}[q \sim P_{sft}(Q), o^+, o^- \sim \pi_{sft}(O|q)] \log \sigma \left(\beta \frac{1}{|o^+|} \sum_{t=1}^{|o^+|} \log \frac{\pi_\theta(o_t^+|q, o_{<t}^+)}{\pi_{\text{ref}}(o_t^+|q, o_{<t}^+)} - \beta \frac{1}{|o^-|} \sum_{t=1}^{|o^-|} \log \frac{\pi_\theta(o_{<t}^-|q, o_{<t}^-)}{\pi_{\text{ref}}(o_{<t}^-|q, o_{<t}^-)} \right) \tag{12}$$

$\mathcal{J}_{DPO}(\theta)$ 的梯度为：

$$\begin{aligned} \nabla_\theta \mathcal{J}_{DPO}(\theta) = \mathbb{E}[q \sim P_{sft}(Q), o^+, o^- \sim \pi_{sft}(O|q)] \Bigg(& \frac{1}{|o^+|} \sum_{t=1}^{|o^+|} GC_{DPO}(q, o, t) \nabla_\theta \log \pi_\theta(o_t^+|q, o_{<t}^+) \\ & - \frac{1}{|o^-|} \sum_{t=1}^{|o^-|} GC_{DPO}(q, o, t) \nabla_\theta \log \pi_\theta(o_t^-|q, o_{<t}^-) \Bigg) \end{aligned} \tag{13}$$

数据来源：SFT 数据集中的问题，输出由 SFT 模型采样得到。奖励函数：通用领域的人类偏好（在数学任务中可为“规则”）。梯度系数：

$$GC_{DPO}(q, o, t) = \sigma \left(\beta \log \frac{\pi_\theta(o_t^-|q, o_{<t}^-)}{\pi_{\text{ref}}(o_t^-|q, o_{<t}^-)} - \beta \log \frac{\pi_\theta(o_t^+|q, o_{<t}^+)}{\pi_{\text{ref}}(o_t^+|q, o_{<t}^+)} \right) \tag{14}$$

A.1.5. 近端策略优化（*PPO*）

PPO 的目标函数为：

$$\mathcal{J}_{PPO}(\theta) = \mathbb{E}[q \sim P_{sft}(Q), o \sim \pi_{\theta_{old}}(O|q)] \frac{1}{|o|} \sum_{t=1}^{|o|} \min \left[\frac{\pi_\theta(o_t|q, o_{<t})}{\pi_{\theta_{old}}(o_t|q, o_{<t})} A_t, \text{clip} \left(\frac{\pi_\theta(o_t|q, o_{<t})}{\pi_{\theta_{old}}(o_t|q, o_{<t})}, 1-\varepsilon, 1+\varepsilon \right) A_t \right]. \tag{15}$$

为简化分析，假设模型在每个探索阶段后仅进行一次更新，从而保证 $\pi_{\theta_{old}} = \pi_\theta$。在此情况下，我们可以移除 min 和 clip 操作：

$$\mathcal{J}_{PPO}(\theta) = \mathbb{E}[q \sim P_{sft}(Q), o \sim \pi_{\theta_{old}}(O|q)] \frac{1}{|o|} \sum_{t=1}^{|o|} \frac{\pi_\theta(o_t|q, o_{<t})}{\pi_{\theta_{old}}(o_t|q, o_{<t})} A_t. \tag{16}$$

$\mathcal{J}_{PPO}(\theta)$ 的梯度为：

$$\nabla_\theta \mathcal{J}_{PPO}(\theta) = \mathbb{E}[q \sim P_{sft}(Q), o \sim \pi_{\theta_{old}}(O|q)] \frac{1}{|o|} \sum_{t=1}^{|o|} A_t \nabla_\theta \log \pi_\theta(o_t|q, o_{<t}) \tag{17}$$

数据来源：SFT 数据集中的问题，输出由策略模型采样得到。奖励函数：奖励模型。梯度系数：

$$GC_{PPO}(q, o, t, \pi_{\theta_{rm}}) = A_t, \tag{18}$$

其中 A_t 为优势函数，基于奖励 $\{r_{\geq t}\}$ 和学习的价值函数 V_ψ，通过广义优势估计（GAE）(Schulman et al., 2015) 计算得到。

A.1.6. 组相对策略优化（*GRPO*）

GRPO 的目标函数为（为简化分析，假设 $\pi_{\theta_{old}} = \pi_\theta$）：

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P_{sft}(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{old}}(O|q)]$$
$$\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left[\frac{\pi_\theta(o_{i,t}|q,o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q,o_{i,<t})}\hat{A}_{i,t} - \beta(\frac{\pi_{ref}(o_{i,t}|q,o_{i,<t})}{\pi_\theta(o_{i,t}|q,o_{i,<t})} - \log\frac{\pi_{ref}(o_{i,t}|q,o_{i,<t})}{\pi_\theta(o_{i,t}|q,o_{i,<t})} - 1)\right]. \tag{19}$$

$\mathcal{J}_{GRPO}(\theta)$ 的梯度为：

$$\nabla_\theta\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P_{sft}(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{old}}(O|q)]$$
$$\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left[\hat{A}_{i,t} + \beta\left(\frac{\pi_{ref}(o_{i,t}|o_{i,<t})}{\pi_\theta(o_{i,t}|o_{i,<t})} - 1\right)\right]\nabla_\theta\log\pi_\theta(o_{i,t}|q,o_{i,<t}). \tag{20}$$

数据来源：SFT 数据集中的问题，输出由策略模型采样得到。奖励函数：奖励模型。梯度系数：

$$GC_{GRPO}(q,o,t,\pi_{\theta_{rm}}) = \hat{A}_{i,t} + \beta\left(\frac{\pi_{ref}(o_{i,t}|o_{i,<t})}{\pi_\theta(o_{i,t}|o_{i,<t})} - 1\right), \tag{21}$$

其中 $\hat{A}_{i,t}$ 基于组奖励分数计算得到。