本文由 AINLP 公众号整理翻译，更多 LLM 资源请扫码关注!

MiMo: Unlocking the Reasoning Potential of Language Model – From Pretraining to Posttraining

Xiaomi LLM-Core Team

Abstract

We present MiMo-7B, a large language model born for reasoning tasks, with optimization across both pre-training and post-training stages. During pre-training, we enhance the data preprocessing pipeline and employ a three-stage data mixing strategy to strengthen the base model's reasoning potential. MiMo-7B-Base is pre-trained on 25 trillion tokens, with additional Multi-Token Prediction objective for enhanced performance and accelerated inference speed. During post-training, we curate a dataset of 130K verifiable mathematics and programming problems for reinforcement learning, integrating a test-difficulty–driven code-reward scheme to alleviate sparse-reward issues and employing strategic data resampling to stabilize training. Extensive evaluations show that MiMo-7B-Base possesses exceptional reasoning potential, outperforming even much larger 32B models. The final RL-tuned model, MiMo-7B-RL, achieves superior performance on mathematics, code and general reasoning tasks, surpassing the performance of OpenAI o1-mini. The model checkpoints are available at https://github.com/xiaomimimo/MiMo.

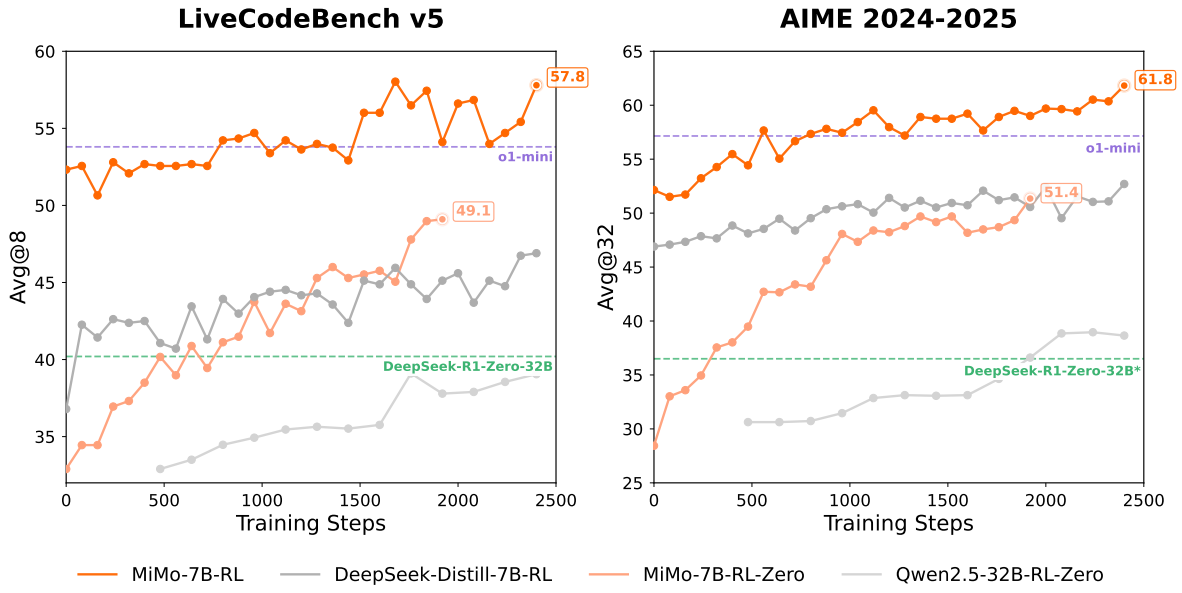

Figure 1 Performance of MiMo-7B in code and math reasoning benchmark.

MiMo：释放语言模型的推理潜能——从预训练到后训练

小米 LLM 核心团队

摘要

我们推出了 MiMo-7B，一款为推理任务而生的大型语言模型，在预训练和后训练两个阶段都进行了优化。在预训练阶段，我们增强了数据预处理流程，并采用三阶段数据混合策略，以增强基础模型的推理潜力。MiMo-7B-Base 在 25 万亿个标记上进行了预训练，加入了多标记预测（Multi-Token Prediction）目标，以提升性能和加快推理速度。在后训练阶段，我们策划了一个包含 130K 个可验证的数学和编程问题的数据集，用于强化学习，结合了基于测试难度的代码奖励方案，以缓解稀疏奖励问题，并采用策略性数据重采样以稳定训练。大量评估显示，MiMo-7B-Base 具有卓越的推理潜力，甚至优于体积更大的 32B 模型。最终经过 RL 调优的模型 MiMo-7B-RL，在数学、代码和通用推理任务中表现出色，超越了 OpenAI o1-mini 的性能。模型检查点可在 `https://github.com/xiaomimimo/MiMo` 获取。

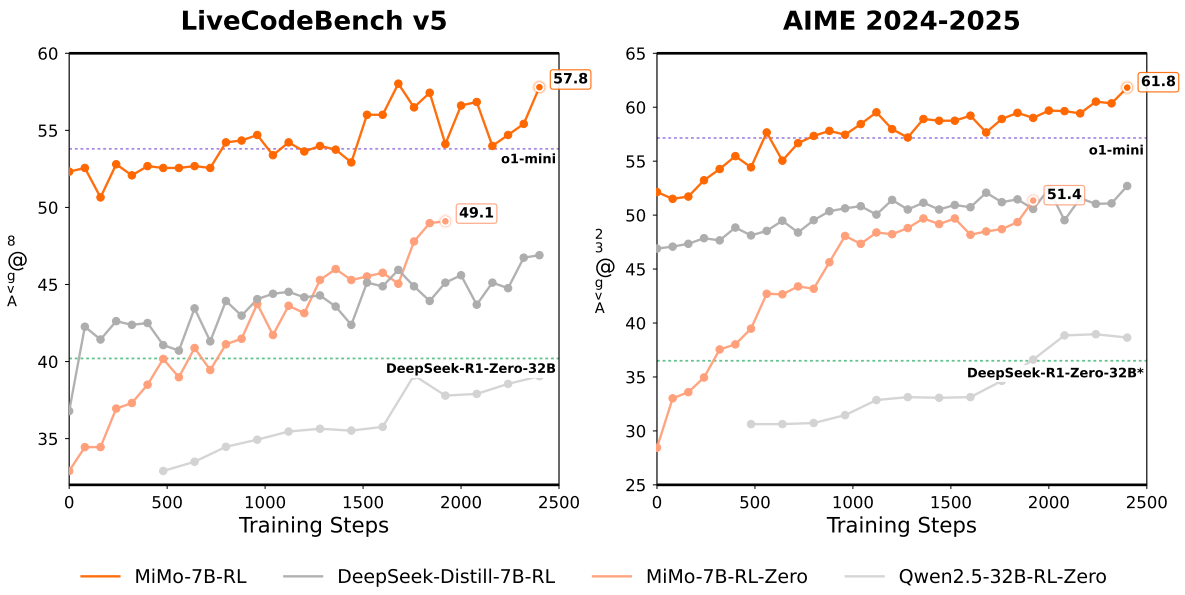

图 1 MiMo-7B 在代码和数学推理基准测试中的表现。

Contents

Contents 目录

1 Introduction

Large language models (LLMs) with advanced reasoning capabilities, such as OpenAI o-series (OpenAI, 2024), DeepSeek R1 (Guo et al., 2025), and Claude 3.7 (Anthropic, 2025), have achieved remarkable performance in complex tasks like mathematical reasoning and code generation. Through large-scale reinforcement learning (RL), these models develop sophisticated reasoning patterns, including step-by-step analysis, self-reflection and backtracking, enabling more robust and accurate problem solving capabilities across diverse domains. This emerging paradigm represents a significant advancement in artificial intelligence's approach for tackling intricate challenges.

Currently, most successful RL works, including open-source research, rely on relatively large base models, e.g., 32B models, particularly for enhancing code reasoning capabilities. Moreover, it was widely considered that achieving uniform and simultaneous improvements in both mathematical and code capabilities within a small model is challenging. Nonetheless, we believe that the effectiveness of the RL trained reasoning model relies on the inherent reasoning potential of the base model. To fully unlock the reasoning potential of language models, efforts must focus not only on post-training but also on pre-training strategies tailored to reasoning.

In this work, we present MiMo-7B, a series of models trained from scratch and born for reasoning tasks. Our RL experiments from MiMo-7B-Base show that our model possesses extraordinary reasoning potential, even outperforming much larger 32B models. Additionally, we perform RL training on a cold-started SFT model, resulting in MiMo-7B-RL, which demonstrates superior performance on both mathematics and code reasoning tasks, surpassing the performance of OpenAI o1-mini. Here are our detailed contributions:

Pre-Training: Base Model Born for Reasoning

- We optimize data preprocessing pipeline, enhancing text extraction toolkits and applying multi-dimensional data filtering to increase reasoning pattern density in pre-training data. We also employ multiple strategies to generate massive diverse synthetic reasoning data.
- We adopt a three-stage data mixture strategy for pre-training. Overall, MiMo-7B-Base is pre-trained on approximately 25 trillion tokens.
- We incorporate Multiple-Token Prediction as an additional training objective, which enhances model performance and accelerates inference.

Post-Training Recipe: Pioneering Reasoning Model

- We curate 130K mathematics and code problems as RL training data, which can be verified by rule-based verifiers. Each problem undergoes careful cleaning and difficulty assessment to ensure quality. We employ only rule-based accuracy rewards to avoid potential reward hacking.
- To mitigate the sparse reward issue for challenging code problems, we introduce a test difficulty driven code reward. By assigning fine-grained scores for test cases with varying difficulty levels, the policy can be more effectively optimized via dense reward signal.
- We implement a data re-sampling strategy to enhance rollout sampling efficiency and stabilize policy updates, particularly in the later phases of RL training.

1 引言

具有先进推理能力的大型语言模型（LLMs），如OpenAI的o系列（OpenAI，2024）、DeepSeek R1（郭等，2025）和Claude 3.7（Anthropic，2025），在数学推理和代码生成等复杂任务中取得了显著的性能。通过大规模强化学习（RL），这些模型发展出复杂的推理模式，包括逐步分析、自我反思和回溯，从而在不同领域实现更强大、更准确的问题解决能力。这一新兴范式代表了人工智能在应对复杂挑战方面的重大进步。

目前，大多数成功的强化学习（RL）工作，包括开源研究，依赖于相对较大的基础模型，例如32B模型，特别是在增强代码推理能力方面。此外，人们普遍认为在一个小模型中实现数学能力和代码能力的均衡且同时的提升具有挑战性。然而，我们相信，RL训练的推理模型的有效性依赖于基础模型固有的推理潜力。为了充分释放语言模型的推理潜能，努力不仅应集中在后训练阶段，还应关注针对推理的预训练策略。

在本工作中，我们提出了 MiMo-7B，一系列从零开始训练、专为推理任务而生的模型。我们在 MiMo-7B-Base 上的强化学习（RL）实验表明，我们的模型具有非凡的推理潜力，甚至优于更大规模的 32B 模型。此外，我们在一个冷启动的 SFT 模型上进行 RL 训练，得到 MiMo-7B-RL，该模型在数学和代码推理任务中表现出色，超越了 OpenAI o1-mini 的性能。以下是我们的详细贡献:

预训练：为推理而生的基础模型

- 我们优化数据预处理流程，增强文本提取工具包，并应用多维数据过滤，以增加预训练数据中的推理模式密度。我们还采用多种策略生成大量多样的合成推理数据。
- 我们采用三阶段的数据混合策略进行预训练。总体而言，MiMo-7B-Base在大约25万亿个标记上进行了预训练。
- 我们将多词预测作为额外的训练目标，这增强了模型的性能并加快了推理速度。

训练后方案：开创性推理模式 l

- 我们整理了13万道数学和代码题作为强化学习训练数据，可以通过基于规则的验证器进行验证。每个题目都经过仔细的清理和难度评估，以确保质量。我们仅采用基于规则的准确性奖励，以避免潜在的奖励操控。
- 为了解决具有挑战性的代码问题中的稀疏奖励问题，我们引入了基于测试难度的代码奖励。通过为不同难度级别的测试用例分配细粒度的分数，策略可以通过密集奖励信号更有效地进行优化。
- 我们实施一种数据重采样策略，以提高滚动采样效率并稳定策略更新，特别是在强化学习训练的后期阶段。

RL Infrastructures

- We develop a Seamless Rollout Engine to accelerate RL training and validation. Our design integrates continuous rollout, asynchronous reward computation, and early termination to minimize GPU idle time, achieving 2.29× faster training and 1.96× faster validation.
- We support MTP in vLLM and enhance the robustness of the inference engine in RL system.

Summary of Evaluation Results

- **MiMo-7B-Base** outperforms SoTA open-source models of approximately 7B parameters, excelling in general knowledge and coding tasks. On BBH, it achieves a score of 75.2, showcasing superior reasoning capabilities. Its strong performance on SuperGPQA further highlights its ability to handle complex graduate-level questions.
- **MiMo-7B-RL-Zero** surpasses the RL training performance of the 32B base model on both mathematics and code tasks. This underscore its efficiency and potential in RL training, positioning MiMo-7B as a compelling candidate for future advancements in RL.
- **MiMo-7B-RL** achieves excellent reasoning performance. It scores 55.4 on AIME 2025, exceeding o1-mini by 4.7 points. In algorithm code generation tasks, MiMo-7B-RL demonstrates extremely impressively results, significantly outperforming OpenAI o1-mini on both LiveCodeBench v5 and the latest v6, demonstrating robust and stable capabilities. MiMo-7B-RL also maintains competitive general performance.

Open-Source We open-source MiMo-7B series, including checkpoints of the base model, SFT model, RL model trained from base model, and RL model trained from the SFT model. We believe this report along with the models will provides valuable insights to develop powerful reasoning LLM that benefit the larger community.

2 Pre-Training

In this section, we first detail our strategies to enhance reasoning capabilities during MiMo-7B pre-training process, encompassing pre-training data construction, model architecture design, and hyper-parameter settings. Then we demonstrate the reasoning potential of MiMo-7B-Base model.

2.1 Pre-Training Data

The pre-training corpus for MiMo-7B integrates diverse sources, including web pages, academic papers, books, programming code, and synthetic data. We believe that incorporating more data with high-quality reasoning patterns during pre-training stage can substantially enhance the reasoning potential of the resulting language model. To achieve this goal, we first optimize our natural text preprocessing pipeline to improve quality and most importantly, reasoning data density. Second, we leverage advanced reasoning models to generate extensive synthetic reasoning data. Finally, we implement a three-stage data mixture strategy to maximize our model's reasoning potential across various tasks and domains.

Better Reasoning Data Extraction Web pages naturally contain content with high density reasoning patterns, such as coding tutorial and mathematics blogs. However, we discover that commonly used extractors (Barbaresi, 2021) often fail to preserve mathematics equations and

RL 基础设施

- 我们开发了一个无缝部署引擎，以加快强化学习的训练和验证。我们的设计集成了连续部署、异步奖励计算和提前终止，以最大限度地减少GPU空闲时间，实现了2.29×倍的训练速度提升和1.96×倍的验证速度提升。
- 我们在 v 中支持 MTP大型语言模型（LLM）并增强推理引擎的鲁棒性　　在强化学习系统中的ine。

评估结果总结

- MiMo-7B-Base 在大约 7B 参数的开源模型中表现优于最新的技术水平（SoTA），在通用知识和编码任务中表现出色。在 BBH 上，它取得了 75.2 的分数，展示了其卓越的推理能力。它在 SuperGPQA 上的出色表现进一步凸显了其处理复杂研究生级别问题的能力。
- MiMo-7B-RL-Zero 在数学和代码任务上都超越了 32B 基础模型的强化学习训练性能。这凸显了其在强化学习训练中的效率和潜力，使 MiMo-7B 成为未来强化学习发展的有力候选。
- MiMo-7B-RL 实现了出色的推理性能。在 AIME 2025 上得分为 55.4，超过 o1-mini 4.7 分。在算法代码生成任务中，MiMo-7B-RL 展示了极其令人印象深刻的结果，在 LiveCodeBench v5 和最新的 v6 上都显著优于 OpenAI o1-mini，展现出强大且稳定的能力。MiMo-7B-RL 也保持了具有竞争力的整体性能。

开源——我们开源了MiMo-7B系列，包括基础模型、SFT模型、从基础模型训练的RL模型以及从SFT模型训练的RL模型。我们相信，这份报告以及这些模型将为开发强大的推理大型语言模型（LLM）提供宝贵的见解，惠及更广泛的社区。

2 预训练

在本节中，我们首先详细介绍在MiMo-7B预训练过程中提升推理能力的策略，包括预训练数据的构建、模型架构设计和超参数设置。然后，我们展示了MiMo-7B-Base模型的推理潜力。

2.1 预训练数据

MiMo-7B的预训练语料库整合了多样的来源，包括网页、学术论文、书籍、编程代码和合成数据。我们相信，在预训练阶段引入更多具有高质量推理模式的数据，能够显著提升所生成语言模型的推理潜力。为实现这一目标，我们首先优化自然文本预处理流程，以提高质量，最重要的是，增强推理数据的密度。其次，我们利用先进的推理模型生成大量合成推理数据。最后，我们实施三阶段的数据混合策略，以最大化模型在各种任务和领域中的推理潜能。

更好的推理数据提取网页自然包含高密度推理模式的内容，例如编码教程和数学博客。然而，我们发现常用的提取器（Barbaresi，2021）经常无法保留数学方程式和

code snippets embedded in the webpage. To address this limitation, we develop a novel HTML-extraction tool specially optimized for mathematics content (Liu et al., 2024b; Paster et al., 2024; Zhou et al., 2025), code blocks, and forum websites. For papers and books, we enhance PDF parsing toolkits to better handle STEM and code content. With these optimized extraction tools, we successfully preserved massive reasoning patterns for subsequent processing stages.

Fast Global Deduplication Data deduplication plays an important role in improving training efficiency and reducing overfitting. We adopt both URL deduplication and MinHash deduplication (Broder, 1997) across all webpage dumps. Through extreme engineering optimization, we can complete this global deduplication process within a single day. Since deduplication algorithms treat high-quality and low-quality text equally without content awareness, we subsequently adjust the final data distribution according to multi-dimension quality scores.

Multi-Dimensional Data Filtering High-quality pre-training data with rich reasoning patterns is crucial for developing models with strong reasoning capabilities. We find that commonly used heuristic rule-based filters (Penedo et al., 2023, 2024) incorrectly filter high-quality web pages containing substantial mathematical and code content. To address this limitation, we instead fine-tune small LLMs to serve as data quality taggers, performing domain classification and multi-dimensional quality assessment.

Synthetic Reasoning Data Another crucial source for reasoning patterns is synthetic data generated by advanced reasoning models. We employ multiple strategies to generate diverse synthetic reasoning responses. First, we select STEM content tagged with high reasoning depth and prompt models to develop insightful analyses and perform in-depth thinking based on the source materials. Second, we gather mathematics and code problems and prompt reasoning models to solve them. Additionally, we incorporate general domain queries, particularly creative writing tasks. Notably, our preliminary experiments reveal that, unlike non-reasoning data, synthetic reasoning data can be trained for extremely high number of epochs without overfitting risk.

Three-Stage Data Mixture To optimize the pre-training data distribution, we adopt a three-stage data strategy in the final model training:

- **Stage 1**: We incorporate all data sources except synthetic responses for reasoning task queries. We downsample overrepresented content, such as ads, news, job postings, and materials with insufficient knowledge density and reasoning depth. We also upsample high-value data from professional domains with superior quality.
- **Stage 2**: Building on the curated distribution in Stage 1, we significantly increase mathematics and code related data to ~70% of the mixture. This approach is expected to enhance specialized skills without compromising general language abilities (Zhu et al., 2024). The first two stages are trained with an 8,192-token context length.
- **Stage 3**: To boost the capabilities for solving complex tasks, we further incorporate ~10% synthetic responses for mathematics, code, and creative writing queries. Simultaneously, we extend the context length from 8,192 to 32,768 in the final stage.

Through this process, we build a large high-quality pre-training dataset comprising approximately **25 trillion** tokens.

嵌入网页中的代码片段。为了解决这一限制，我们开发了一种新颖的HTML提取工具，专门针对数学内容（Liu et al., 2024b; Paster et al., 2024; Zhou et al., 2025）、代码块和论坛网站进行优化。对于论文和书籍，我们增强了PDF解析工具包，以更好地处理STEM和代码内容。借助这些优化的提取工具，我们成功地保留了大量的推理模式，以供后续处理阶段使用。

快速全局去重 数据去重在提高训练效率和减少过拟合方面起着重要作用。我们在所有网页数据中同时采用URL去重和MinHash去重（Broder, 1997）。通过极致的工程优化，我们可以在一天之内完成这一全局去重过程。由于去重算法对高质量和低质量文本一视同仁，且没有内容感知能力，因此我们随后根据多维度的质量评分调整最终的数据分布。

多维数据过滤 高质量的预训练数据，具有丰富推理模式，对于开发具有强大推理能力的模型至关重要。我们发现，常用的启发式规则过滤器（Penedo 等，2023，2024）错误地过滤了包含大量数学和代码内容的高质量网页。为了解决这一限制，我们改为微调小型大模型（LLMs），使其作为数据质量标注器，进行领域分类和多维度质量评估。

合成推理数据 另一个推理模式的关键来源是由先进推理模型生成的合成数据。我们采用多种策略来生成多样的合成推理响应。首先，我们选择带有高推理深度标签的STEM内容，并提示模型进行深刻的分析，基于源材料进行深入思考。其次，我们收集数学和编码问题，并提示推理模型进行解答。此外，我们还融入一般领域的查询，特别是创造性写作任务。值得注意的是，我们的初步实验显示，与非推理数据不同，合成推理数据可以在极高的训练轮数下进行训练而不出现过拟合风险。

三阶段数据混合 为了优化预训练数据分布，我们在最终模型训练中采用三阶段数据策略：

- 阶段1：我们整合除合成响应以外的所有数据源，用于推理任务的查询。我们对过度代表的内容进行抽样，例如广告、新闻、职位发布以及知识密度和推理深度不足的材料。我们还对来自专业领域的高价值高质量数据进行上采样。

- 第二阶段：在第一阶段精心筛选的分布基础上，我们将数学和代码相关数据显著增加到混合中的~70%。预计这种方法将增强专业技能，同时不影响通用语言能力（Zhu 等，2024）。前两个阶段的训练采用8,192令牌的上下文长度。

- 阶段3：为了增强解决复杂任务的能力，我们在最后阶段进一步加入了~ 10%的合成响应，用于数学、代码和创意写作的查询。同时，我们将上下文长度从8,192扩展到32,768。

通过这个过程，我们构建了一个包含大约25万亿个标记的高质量预训练数据集。

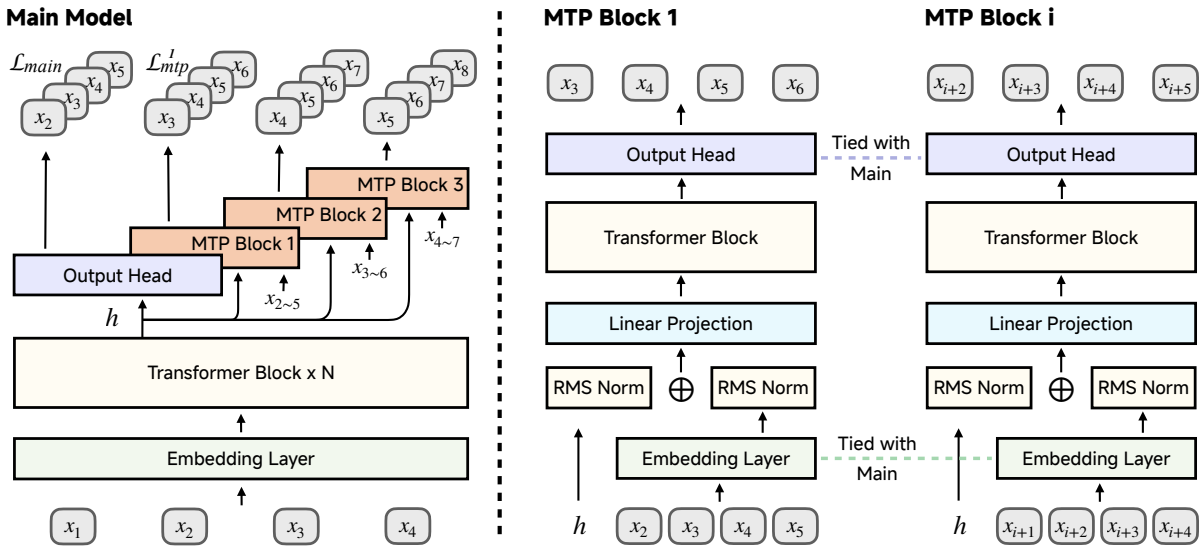

Figure 2 Implementation of Multi-Token Prediction with MiMo-7B. During pre-training we use a single MTP layer, while the inference stage can use multiple MTP layers for additional speedup.

2.2 Model Architecture

MiMo-7B follows the general decoder-only Transformer architecture (Radford et al., 2018; Vaswani et al., 2017), and consists of Grouped-Query Attention (GQA, Ainslie et al. 2023), pre-RMSNorm (Zhang and Sennrich, 2019), SwiGLU activation (Dauphin et al., 2017) and Rotary Positional Embedding (RoPE, Su et al. 2024), similar to Llama (Grattafiori et al., 2024; Touvron et al., 2023) and Qwen (Yang et al., 2024).

Reasoning models often face an inference speed bottleneck due to their lengthy auto-regressive generation process, despite the high correlation and predictability observed among consecutive tokens in their reasoning paths.

MTP Modules Inspired by DeepSeek-V3 (Liu et al., 2024a), we incorporate Multi-Token Prediction (MTP) (Gloeckle et al., 2024) as an additional training objective. This approach enables the model to strategically pre-plan and generate representations that facilitate more accurate and potentially faster prediction of future tokens. As shown in Figure 2, we implement distinct MTP setups for pre-training and inference. During pre-training, we utilize only a single MTP layer, as our preliminary studies show that multiple MTP layers yield no further improvement. In contrast, we find that multiple parallel MTP layers significantly accelerate inference through speculative decoding. To implement this, after pre-training, we replicate the pre-trained single MTP layer into two identical copies. Then, with the main model and first MTP layer frozen, we fine-tune two new MTP layers for inference speedup.

MTP Inference Speedup During inference, these MTP layers can be utilized for speculative decoding (Leviathan et al., 2023; Xia et al., 2023) to reduce generation latency. We evaluated the performance of the MTP layers on the AIME24 benchmark. The first MTP layer achieves a remarkably high acceptance rate about 90%, while even the third MTP layer maintains an acceptance rate above 75%. This high acceptance rate enables MiMo-7B to deliver enhanced decoding speed, particularly in reasoning scenarios requiring extremely long outputs.

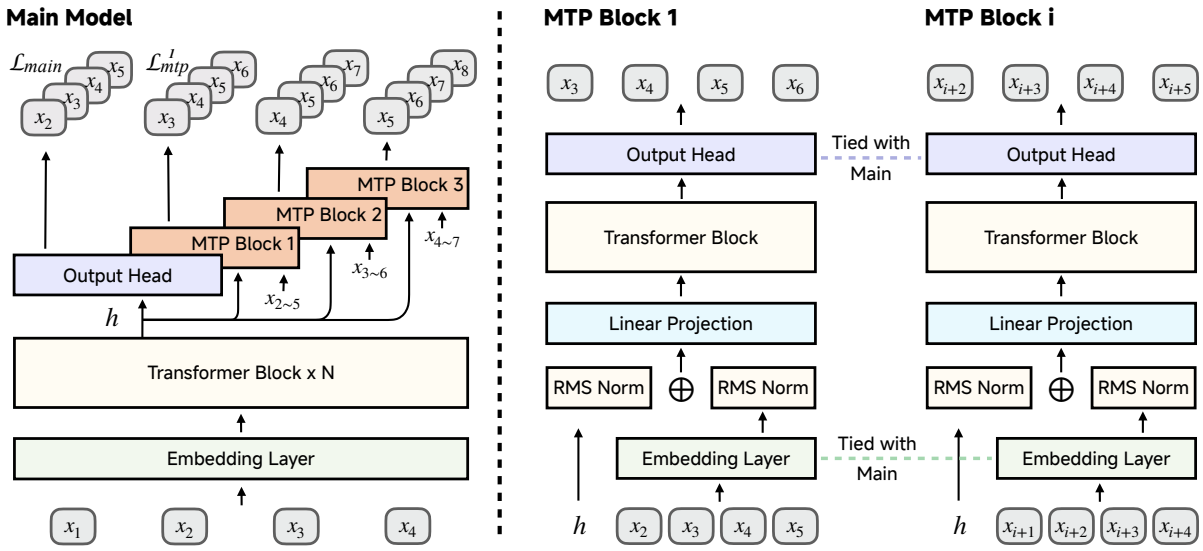

图 2 使用 MiMo-7B 实现多令牌预测。在预训练阶段，我们使用单个 MTP 层，而推理阶段可以使用多个 MTP 层以实现额外的加速。

2.2 模型架构

MiMo-7B 采用通用的仅解码器 Transformer 架构（Radford 等，2018；Vaswani 等，2017），由分组查询注意力（GQA，Ainslie 等，2023）、预归一化（pre-RMSNorm，Zhang 和 Sennrich，2019）、SwiGLU 激活（Dauphin 等，2017）和旋转位置嵌入（RoPE，Su 等，2024）组成，类似于 Llama（Grattafiori 等，2024；Touvron 等，2023）和 Qwen（Yang 等，2024）。

推理模型常常由于其冗长的自回归生成过程而面临推理速度瓶颈，尽管在其推理路径中连续标记之间观察到高度相关性和可预测性。

受深度搜索-V3 (Liu et al., 2024a) 启发的 MTP 模块，我们引入多令牌预测 (MTP) (Gloeckle et al., 2024) 作为额外的训练目标。这种方法使模型能够有策略地预先规划并生成有助于更准确且可能更快预测未来令牌的表示。如图 2 所示，我们为预训练和推理实现了不同的 MTP 设置。在预训练期间，我们仅使用单个 MTP 层，因为我们的初步研究表明多个 MTP 层并不能带来额外的提升。相反，我们发现多个并行的 MTP 层通过投机性解码显著加快了推理速度。为了实现这一点，在预训练后，我们将预训练的单个 MTP 层复制成两个相同的副本。然后，在主模型和第一个 MTP 层被冻结的情况下，我们微调两个新的 MTP 层以加快推理速度。

在推理过程中，这些 MTP 层可以用于推测解码（Leviathan 等，2023；Xia 等，2023），以减少生成延迟。我们在 AIME24 基准测试中评估了 MTP 层的性能。第一个 MTP 层的接受率达到了惊人的 90% 左右，而即使是第三个 MTP 层也保持在 75% 以上的接受率。这一高接受率使得 MiMo-7B 能够提供更快的解码速度，特别是在需要极长输出的推理场景中。

2.3 Hyper-Parameters

Model Hyper-Parameters We set the number of Transformer layers to 36 and the hidden dimension to 4,096. The intermediate hidden dimension of FFN is set to 11,008. The number of attention heads is 32 and there are 8 key-value groups.

Training Hyper-Parameters For optimization, we use AdamW (Loshchilov and Hutter, 2019) with $\beta_1 = 0.9$, $\beta_2 = 0.95$, and weight decay of 0.1. We apply gradient clipping with a maximum norm of 1.0.

During the first two pre-training stages, the maximum sequence length is 8,192 tokens with the RoPE base of 10,000. Stage 3 expands these parameters to 32,768 tokens and 640,000, respectively.

Our learning rate schedule begins in Stage 1 with a linear warmup from 0 to 1.07×10^{-4} over the first 84B tokens, followed by a constant phase at 1.07×10^{-4} for 10.2T tokens, and concludes with a cosine decay to 3×10^{-5} over 7.5T tokens. This rate of 3×10^{-5} is maintained throughout Stage 2 (4T tokens) and for the first 1.5T tokens of Stage 3. Subsequently, the learning rate decays via a cosine schedule to 1×10^{-5} over the final 500B tokens.

We implement a linear batch size warmup to 2,560 over the first 168B tokens and maintain this value throughout the remainder of Stage 1 and Stage 2. In Stage 3, the batch size is fixed at 640.

The MTP loss weight is set to 0.3 for the first 10.3T tokens, then reduced to 0.1 for the remainder of pre-training.

2.4 Pre-Training Evaluation

2.4.1 Evaluation Setup

We evaluate MiMo-7B-Base on a series of benchmarks, encompassing natural language understanding and reasoning, scientific question answering, reading comprehension, mathematics reasoning, coding, Chinese understanding, and long-context comprehension capabilities:

Language understanding and reasoning: BBH (Suzgun et al., 2023), MMLU Hendrycks et al. (2021a), MMLU-Redux (Gema et al., 2024), MMLU-Pro (Wang et al., 2024), ARC (Clark et al., 2018), HellaSwag (Zellers et al., 2019), PIQA (Bisk et al., 2020).

Closed-book question answering: TriviaQA (Joshi et al., 2017), NaturalQuestions (Kwiatkowski et al., 2019).

Scientific question answering: GPQA (Rein et al., 2024), SuperGPQA (Du et al., 2025).

Reading comprehension: DROP (Dua et al., 2019), RACE (Lai et al., 2017).

Mathematics reasoning: AIME (MAA, 2024), GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021b).

Coding: LiveCodeBench (Jain et al., 2024), HumanEval (Chen et al., 2021), HumanEval+ (Liu et al., 2023), MBPP (Austin et al., 2021), MBPP+ (Liu et al., 2023), CRUXEval (Gu et al., 2024).

Miscellaneous: WinoGrande (Sakaguchi et al., 2020), AGIEval (Zhong et al., 2024a).

Chinese understanding: C-Eval (Huang et al., 2023), CMMLU (Li et al., 2023).

Long-Context Comprehension: RULER (Hsieh et al., 2024)

2.3 超参数

模型超参数 我们将Transformer层数设为36层，隐藏维度为4,096。FFN的中间隐藏维度设为11,008。注意力头的数量为32个，具有8个键值组。

训练超参数 为了优化，我们使用 AdamW（Loshchilov 和 Hutter，2019），其参数为 $\beta_1 = 0.9$，$\beta_2 = 0.95$，权重衰减为 0.1。我们采用梯度裁剪，最大范数为 1.0。

在前两个预训练阶段，最大序列长度为8,192个标记，RoPE基础为10,000。第3阶段将这些参数扩展到32,768个标记和640,000。

我们的学习率调度在第1阶段开始，首先在前84B个标记中进行线性预热，从0到1.07×10^{-4}，然后在接下来的10.2T个标记中保持恒定在1.07×10^{-4}，最后通过余弦衰减到3×10^{-5}，持续7.5T个标记。在整个第2阶段（4T个标记）以及第3阶段的前1.5T个标记中，学习率保持在3×10^{-5}。随后，学习率通过余弦调度在最后的500B个标记中衰减到1×10^{-5}。

我们在前168B个标记中实现线性批量大小预热至2,560，并在第1阶段和第2阶段的其余部分中保持该值。在第3阶段，批量大小固定为640。

MTP损失权重在前10.3T个tokens中设置为0.3，然后在剩余的预训练中降低到0.1。

2.4 预训练评估

2.4.1 评估设置

我们在一系列基准测试中评估了MiMo-7B-Base，包括自然语言理解与推理、科学问答、阅读理解、数学推理、编码、中文理解以及长上下文理解能力：

语言理解与推理：BBH（Suzgun 等，2023）、MMLU Hendrycks 等（2021a）、MMLU-Redux（Gema 等，2024）、MMLU-Pro（Wang 等，2024）、ARC（Clark 等，2018）、HellaSwag（Zellers 等，2019）、PIQA（Bisk 等，2020）。

闭卷问答：TriviaQA（Joshi 等，2017），NaturalQuestions（Kwiatkowski 等，2019）。

科学问题回答：GPQA（Rein 等，2024），SuperGPQA（Du 等，2025）。

阅读理解：DROP（Dua 等，2019），RACE（Lai 等，2017）。

数学推理：AIME（MAA，2024）、GSM8K（Cobbe 等，2021）、MATH（Hendrycks 等，2021b）。

编码：LiveCodeBench（Jain 等，2024），HumanEval（Chen 等，2021），HumanEval+ (Liu 等，2023)，MBPP（Austin 等，2021），MBPP+ (Liu 等，2023)，CRUXEval（Gu 等，2024）。

杂项：WinoGrande（Sakaguchi 等人，2020），AGIEval（Zhong 等人，2024a）。

中文理解：C-Eval（Huang 等，2023）、CMMLU（Li 等，2023）。

长上下文理解：RULER（Hsieh 等，2024）

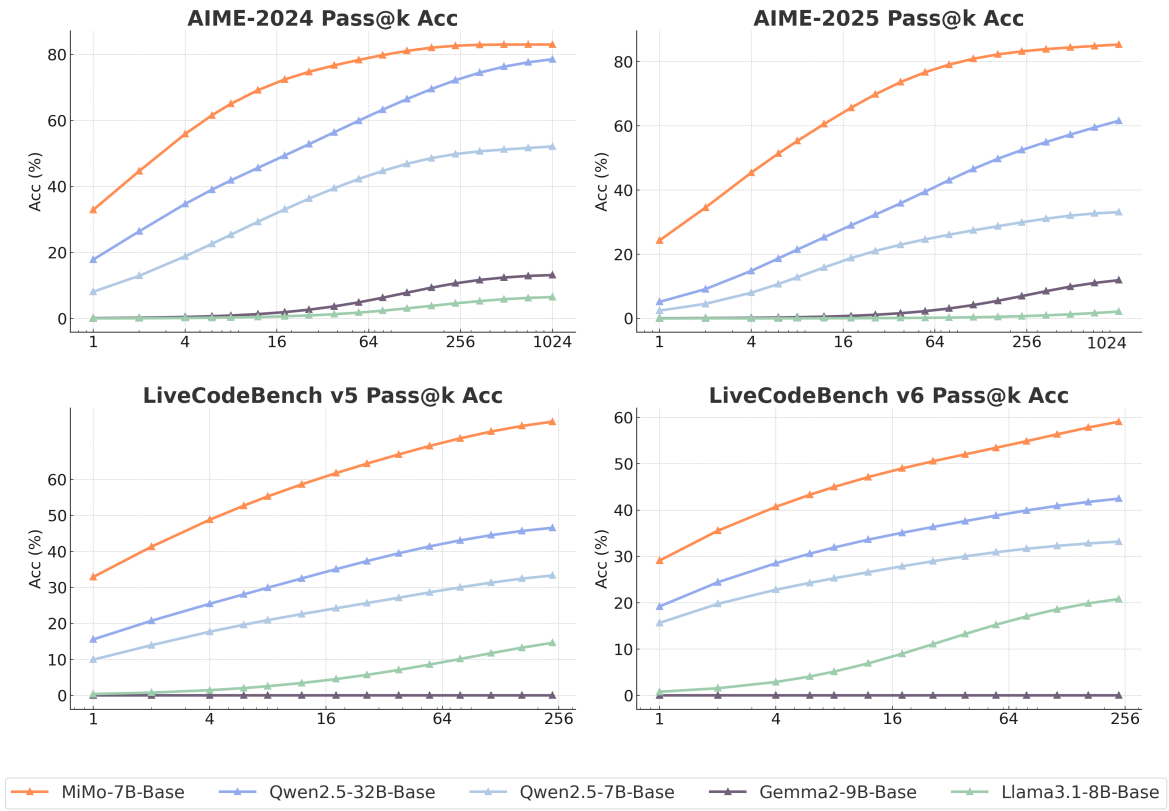

Figure 3 Pass@k curves of different base models across multiple reasoning benchmarks.

We compare MiMo-7B-Base with other open-source base models of comparable size, including Llama-3.1-8B (Grattafiori et al., 2024), Gemma-2-9B (Team, 2024), and Qwen2.5-7B (Yang et al., 2024). The evaluation of all models shares the same evaluation settings.

2.4.2 Upper Bounds of Reasoning Capability

Traditional evaluation methods often underestimate a model's true reasoning potential by relying on single-pass success rates or average performance across multiple samplings. Following Yue et al. (2025), we adopt the pass@k metric, which considers a problem solved if any of k sampled solution is correct, to better assess the reasoning capacity boundary of different models.

As illustrated in Figure 3, MiMo-7B-Base achieves significantly higher pass@k scores than all compared models, including the 32B baseline, across all benchmarks and evaluated k values. Notably, the performance gap between MiMo-7B-Base and other baselines widens steadily as k increases, particularly on LiveCodeBench. These results demonstrates the superior reasoning potential of MiMo-7B-Base, which establishes a strong base policy for RL training.

2.4.3 Evaluation Results

General Reasoning MiMo-7B-Base achieves superior performance in general knowledge and reasoning, outperforming open-source models of comparable size. On BBH, a benchmark evaluating language reasoning abilities, MiMo-7B-Base scores 75.2, surpassing Qwen2.5-7B by about 5 points. Furthermore, SuperGPQA results show our model's robust performance in solving graduate-level problems. On DROP, a reading comprehension benchmark, MiMo-7B-Base outperforms compared

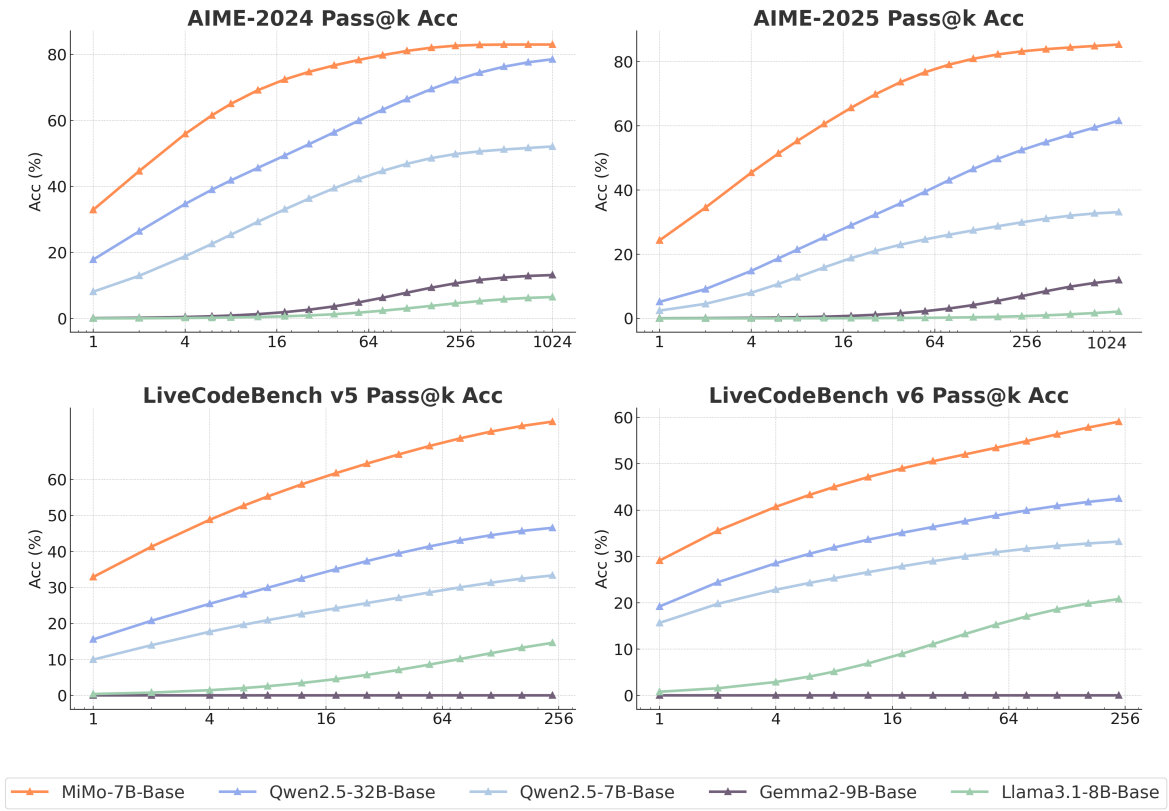

图 3 不同基础模型在多个推理基准测试中的 Pass@k 曲线。

我们将MiMo-7B-Base与其他规模相当的开源基础模型进行了比较，包括Llama-3.1-8B（Grattafiori等，2024）、Gemma-2-9B（Team，2024）和Qwen2.5-7B（Yang等，2024）。所有模型的评估都采用相同的评估设置。

2.4.2 推理能力的上限

传统的评估方法通常通过依赖单次成功率或多次采样的平均表现，低估了模型的真实推理潜力。根据Yue等人（2025年）的研究，我们采用pass@k指标，即如果k个采样解中的任何一个是正确的，则认为问题已解决，以更好地评估不同模型的推理能力边界。

如图3所示，MiMo-7B-Base在所有基准测试和评估的k值中，均显著高于所有比较模型，包括32B基线。在所有基准和k值中，MiMo-7B-Base的pass@k得分明显优于其他基线，尤其是在LiveCodeBench上。 这些结果展示了MiMo-7B-Base的优越推理潜力，为RL训练奠定了坚实的基础策略。

2.4.3 评估结果

通用推理 MiMo-7B-Base 在通用知识和推理方面表现优越，优于同等规模的开源模型。在评估语言推理能力的基准测试 BBH 上，MiMo-7B-Base 得分为 75.2，超过 Qwen2.5-7B 大约 5 分。此外，SuperGPQA 的结果显示我们的模型在解决研究生水平问题方面表现出色。在阅读理解基准测试 DROP 上，MiMo-7B-Base 的表现优于其他模型。

Benchmark	# Shots	Llama-3.1 8B Base	Gemma-2 9B Base	Qwen2.5 7B Base	MiMo-7B Base
General					
BBH (EM)	3-shot	64.2	69.4	70.4	75.2
GPQA-Diamond (EM)	5-shot	33.3	24.2	35.4	25.8
SuperGPQA (EM)	5-shot	19.9*	22.6*	24.6*	25.1
DROP (F1)	3-shot	59.5	67.9*	61.5*	69.2
MMLU (EM)	5-shot	65.3	71.2	74.2	71.2
MMLU-Redux (EM)	5-shot	58.4*	67.9	71.1	65.3
MMLU-Pro (EM)	5-shot	37.1	44.7	45.0	41.9
ARC-Easy (EM)	25-shot	84.3	88.3	86.4	85.2
ARC-Challenge (EM)	25-shot	57.7	68.2	63.8	62.3
HellaSwag (EM)	10-shot	82.0	81.9	80.4	80.0
PIQA (EM)	0-shot	80.3	81.9	78.5	79.4
WinoGrande (EM)	5-shot	60.5	73.9*	75.9	78.0
RACE-High (EM)	5-shot	44.3	48.3	46.8	44.1
TriviaQA (EM)	5-shot	70.6	76.5	60.0	60.8
NaturalQuestions (EM)	5-shot	27.7	29.2	24.1	24.5
AGIEval (EM)	0-shot	38.2*	21.6*	44.4	48.3
Mathematics					
AIME 2024 (Pass@1)	0-shot	0.3*	0.0*	10.1*	32.9
AIME 2025 (Pass@1)	0-shot	0.0*	0.0*	4.3*	24.3
GSM8K (EM)	8-shot	48.5*	70.2*	80.2*	75.2
MATH (EM)	4-shot	16.9*	36.4*	44.3*	37.4
Code					
LiveCodeBench v5 (Pass@1)	0-shot	0.4*	0.0*	5.0*	32.9
HumanEval (Pass@1)	1-shot	37.8*	41.5*	56.7*	51.8
HumanEval+ (Pass@1)	1-shot	31.7*	31.1*	50.0*	44.5
MBPP (Pass@1)	3-shot	58.4	63.9	76.7	69.2
MBPP+ (Pass@1)	3-shot	49.9	52.9	64.2	56.6
CRUXEval-I (EM)	2-shot	41.5	49.8	52.4	47.6
CRUXEval-O (EM)	2-shot	36.8	42.4	48.5	56.3
Chinese					
C-Eval (EM)	5-shot	52.2	57.0	81.8	68.7
CMMLU (EM)	5-shot	52.1	58.4	82.7	70.9

Table 1 Comparison among MiMo-7B-Base and other open-source base models of comparable size. Results marked with * are obtained using our internal evaluation framework.

Benchmark	# Shots	Llama-3.1 8B Base	Gemma-2 9B Base	Qwen2.5 7B Base	MiMo-7B Base
General					
BBH (EM)	3-shot	64.2	69.4	70.4	75.2
GPQA-Diamond (EM)	5-shot	33.3	24.2	35.4	25.8
SuperGPQA (EM)	5-shot	19.9*	22.6*	24.6*	25.1
DROP (F1)	3-shot	59.5	67.9*	61.5*	69.2
MMLU (EM)	5-shot	65.3	71.2	74.2	71.2
MMLU-Redux (EM)	5-shot	58.4*	67.9	71.1	65.3
MMLU-Pro (EM)	5-shot	37.1	44.7	45.0	41.9
ARC-Easy (EM)	25-shot	84.3	88.3	86.4	85.2
ARC-Challenge (EM)	25-shot	57.7	68.2	63.8	62.3
HellaSwag (EM)	10-shot	82.0	81.9	80.4	80.0
PIQA (EM)	0-shot	80.3	81.9	78.5	79.4
WinoGrande (EM)	5-shot	60.5	73.9*	75.9	78.0
RACE-High (EM)	5-shot	44.3	48.3	46.8	44.1
TriviaQA (EM)	5-shot	70.6	76.5	60.0	60.8
NaturalQuestions (EM)	5-shot	27.7	29.2	24.1	24.5
AGIEval (EM)	0-shot	38.2*	21.6*	44.4	48.3
Mathematics					
AIME 2024 (Pass@1)	0-shot	0.3*	0.0*	10.1*	32.9
AIME 2025 (Pass@1)	0-shot	0.0*	0.0*	4.3*	24.3
GSM8K (EM)	8-shot	48.5*	70.2*	80.2*	75.2
MATH (EM)	4-shot	16.9*	36.4*	44.3*	37.4
Code					
LiveCodeBench v5 (Pass@1)	0-shot	0.4*	0.0*	5.0*	32.9
HumanEval (Pass@1)	1-shot	37.8*	41.5*	56.7*	51.8
HumanEval+ (Pass@1)	1-shot	31.7*	31.1*	50.0*	44.5
MBPP (Pass@1)	3-shot	58.4	63.9	76.7	69.2
MBPP+ (Pass@1)	3-shot	49.9	52.9	64.2	56.6
CRUXEval-I (EM)	2-shot	41.5	49.8	52.4	47.6
CRUXEval-O (EM)	2-shot	36.8	42.4	48.5	56.3
Chinese					
C-Eval (EM)	5-shot	52.2	57.0	81.8	68.7
CMMLU (EM)	5-shot	52.1	58.4	82.7	70.9

表1 比较了MiMo-7B-Base与其他同等规模的开源基础模型。带*的结果是使用我们的内部评估框架获得的。

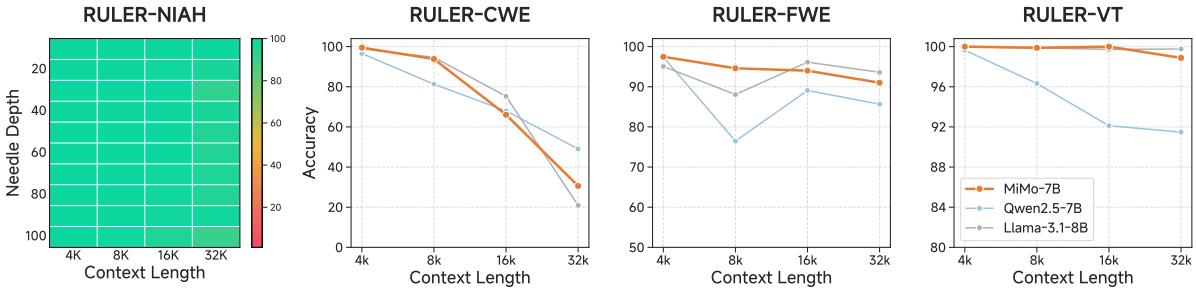

Figure 4 Results of long-context comprehension on RULER. Our MiMo-7B-Base achieves near-perfect NIAH retrieval performance within the supported 32K context length, and delivers remarkable performance on Common Words Extraction (CWE), Frequent Words Extraction (FWE), and Variable Tracking (VT) that emphasizes long-context reasoning beyond retrieval.

models, showing advanced language understanding capability.

Code and Mathematics Reasoning MiMo-7B-Base demonstrates strong proficiency in coding and mathematics tasks. On LiveCodeBench v5, it scores 32.9, far surpassing Llama-3.1-8B and Qwen-2.5-7B. Similarly, on AIME 2024, our model achieves 32.9, significantly outperforming other comparably sized base models. These results highlight MiMo-7B-Base's extraordinary problem-solving abilities and its huge potential for complex reasoning tasks.

Long-Context Comprehension The ability to understand and reason over long contexts is essential for modern thinking models (Liu et al., 2025), as it enables them to produce long and complex reasoning chains.

For the needle-in-a-haystack (NIAH) tasks (Single, Multi-keys, Multi-values, and Multi-queries NIAH) that focus on long-context retrieval, we aggregate their accuracy across varying depths and context lengths, as depicted in the leftmost panel of Figure 4. We observe that MiMo-7B achieves near-perfect retrieval performance across all positions within the 32K context window.

Beyond pure retrieval, MiMo-7B excels in tasks requiring long-context reasoning, including Common Words Extraction (CWE), Frequent Words Extraction (FWE), and Variable Tracking (VT). It delivers remarkable performance and surpasses Qwen2.5-7B in most scenarios. These results validate the efficacy of our strategy to incorporate diverse data with high-quality reasoning patterns during pre-training.

3 Post-Training

After the pre-training stage, post-training are implemented on MiMo-7B-Base. Specifically, we develop MiMo-7B-RL-Zero through direct RL from MiMo-7B-Base, and MiMo-7B-RL trained from an SFT version of MiMo-7B.

3.1 Supervised Fine-Tuning

SFT Data The SFT data consists of a combination of open-source and proprietary distilled data. To ensure optimal quality and diversity, we implement a three stage preprocessing pipeline. First, we eliminate all training queries that have 16-gram overlap with evaluation benchmarks to prevent data leakage. Then, we exclude samples with mixing language or incomplete response. Finally, we

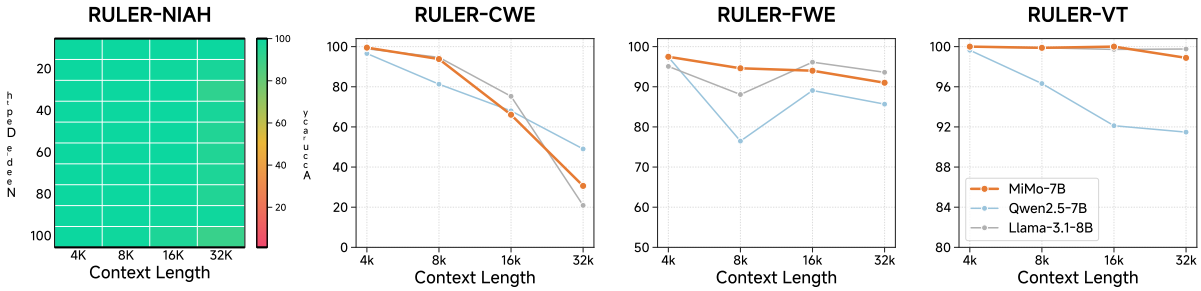

图4 RULER上长上下文理解的结果。我们的MiMo-7B-Base在支持的32K上下文长度内实现了几乎完美的NIAH检索性能，并在常用词提取（CWE）、频繁词提取（FWE）和变量追踪（VT）方面表现出色，强调超越检索的长上下文推理能力。

模型，展示了先进的语言理解能力。

代码与数学推理 MiMo-7B-Base 在编码和数学任务中表现出强大的能力。在 LiveCodeBench v5 上，它的得分为 32.9，远超 Llama-3.1-8B 和 Qwen-2.5-7B。同样，在 AIME 2024 上，我们的模型也取得了 32.9 的成绩，显著优于其他同等规模的基础模型。这些结果突显了 MiMo-7B-Base 在解决问题方面的非凡能力以及其在复杂推理任务中的巨大潜力。

长上下文理解 长上下文理解能力对于现代思维模型（Liu et al., 2025）至关重要，因为它使它们能够产生长而复杂的推理链。

对于针在大海捞针（NIAH）任务（单一、多键、多值和多查询NIAH），这些任务专注于长上下文检索，我们将它们在不同深度和上下文长度上的准确率进行汇总，如图4最左侧的面板所示。我们观察到，MiMo-7B在32K上下文窗口内的所有位置都实现了接近完美的检索性能。

超越纯检索，MiMo-7B 在需要长上下文推理的任务中表现出色，包括常用词提取（CWE）、频繁词提取（FWE）和变量追踪（VT）。它展现出卓越的性能，在大多数场景中超越 Qwen2.5-7B。这些结果验证了我们在预训练过程中结合多样化数据和高质量推理模式策略的有效性。

3 训练后

在预训练阶段之后，在MiMo-7B-Base上实施后训练。具体而言，我们通过直接RL从MiMo-7B-Base开发了MiMo-7B-RL-Zero，而MiMo-7B-RL则是从MiMo-7B的SFT版本训练而来。

3.1 监督微调

SFT 数据 SFT 数据由开源和专有的提炼数据组合而成。为了确保最佳的质量和多样性，我们实施了一个三阶段的预处理流程。首先，我们消除所有与评估基准具有 16-gram 重叠的训练查询，以防止数据泄漏。然后，我们排除包含混合语言或响应不完整的样本。最后，我们

capped the number of responses per query at eight, striking a balance between preserving diversity and preventing redundancy. Following this preprocessing, our final SFT dataset comprises about 500K samples.

SFT Hyper-parameters We fine-tune the MiMo-7B-Base model with a constant learning rate of 3×10^{-5} and batch size of 128. Samples are packed to the maximum length of 32,768 tokens during training.

3.2 RL Data Curation

We utilize two categories of verifiable problems, mathematics and code, to formulate our RL training data. Our preliminary studies demonstrate that high-quality problem sets plays a critical role in stabilizing the RL training process and further enhancing the LLM's reasoning capabilities.

Mathematical Data Our mathematical problem set is drawn from diverse sources, including open-source datasets and proprietary collected competition-level collections. To mitigate the risk of reward hacking, we employ an LLM to filter proof-based and multiple-choice problems. Unlike recent approaches that modify problems to ensure integer answers, we preserve original problems to minimize reward hacking. Additionally, we perform global n-gram deduplication and carefully decontaminate of our problem set with evaluation benchmarks.

Model-based difficulty assessment is used to further improve the quality of out dataset. Initially, we filter out problems that cannot be solved by advanced reasoning models, identifying those that are either too difficult or contain incorrect answers. For the remaining problems, we rollout an SFT version of MiMo-7B 16 times, eliminating problems with a passrate exceeding 90%. Notably, this process removes approximately 50% of easy problems from the original problem set. After data cleaning, we establish a mathematical training set comprising 100K problems.

Code Data For coding problems, we curate a high-quality training set comprising open-source datasets and our newly collected problem set. We remove problems without test cases. For problems with golden solutions, we exclude those where the golden solution failed to pass all test cases. For problems without golden solution, we discard problems where no test case can be solved in 16 rollouts of advanced reasoning models. Similar to math data, we utilize an SFT version of MiMo-7B to filter out easy problems that are perfectly solved in all 16 rollouts. This rigorous cleaning process yields 30K code problems.

During each RL iteration, we evaluate thousands of problems to compute the rewards, with each problem potentially containing hundreds of test cases. To improve reward computing efficiency and eliminate GPU idle time, we developed an online judge environment that enables parallel execution of extremely high-volume unit tests.

Reward Function We employ only rule-based accuracy rewards in our training process. For mathematics data, we use the rule-based Math-Verify library to evaluate response correctness. For code problems, we implement a test difficulty driven reward as detailed in Section 3.3.1. No additional rewards, such as format reward and length penalty reward, is incorporated.

3.3 RL Training Recipe

We employ a modified version of Group Relative Policy Optimization (GRPO) (Shao et al., 2024) with recently proposed improvement from the research community (Hu et al., 2025; Yu et al.,

将每个查询的响应数量限制在八个，平衡了保持多样性和防止重复之间的关系。在此预处理之后，我们的最终SFT数据集大约包含50万条样本。

SFT 超参数 我们以恒定的学习率 3×10^{-5} 和批量大小为 128 对 MiMo-7B-Base 模型进行微调。在训练过程中，样本被打包到最大长度 32,768 个标记。

3.2 强化学习数据整理

我们利用两类可验证的问题，数学和代码，来构建我们的强化学习训练数据。我们的初步研究表明，高质量的问题集在稳定强化学习训练过程和进一步提升大模型的推理能力方面起着关键作用。

数学数据我们的数学题集来自多样的来源，包括开源数据集和专有的竞赛级收集。为了降低奖励操控的风险，我们使用大型语言模型（LLM）筛选基于证明和多项选择的问题。与近期通过修改题目以确保整数答案的方法不同，我们保留原始题目以最大程度减少奖励操控。此外，我们还进行了全局n-gram去重，并对我们的题集进行了仔细的去污染处理，确保与评估基准的纯净性。

基于模型的难度评估用于进一步提升我们数据集的质量。最初，我们筛选出无法被先进推理模型解决的问题，识别出那些过于困难或包含错误答案的问题。对于剩余的问题，我们对MiMo-7B的SFT版本进行16次 rollout，剔除通过率超过90%的问题。值得注意的是，这一过程从原始题集中移除了大约50%的简单题。在数据清洗后，我们建立了一个包含10万道题目的数学训练集。

代码数据　对于编码问题，我们整理了一个高质量的训练集，包括开源数据集和我们新收集的问题集。我们会删除没有测试用例的问题。对于有黄金解的问题，我们排除那些黄金解未能通过所有测试用例的问题。对于没有黄金解的问题，我们会舍弃那些在16次高级推理模型的滚动中无法解决的测试用例的问题。类似于数学数据，我们使用MiMo-7B的SFT版本来筛选出在所有16次滚动中都能完美解决的简单问题。这一严格的清理过程产生了30K个编码问题。

在每次强化学习迭代中，我们评估数千个问题以计算奖励，每个问题可能包含数百个测试用例。为了提高奖励计算效率并消除GPU空闲时间，我们开发了一个在线判题环境，支持极高容量的单元测试的并行执行。

奖励函数 在我们的训练过程中，我们仅采用基于规则的准确性奖励。对于数学数据，我们使用基于规则的 Math-Verify 库来评估回答的正确性。对于代码问题，我们实现了如第3.3.1节所述的测试难度驱动奖励。不包括其他奖励，例如格式奖励和长度惩罚奖励。

3.3 强化学习训练方案

我们采用了经过修改的群组相对策略优化（GRPO）（Shao 等，2024）版本，并结合了近期研究界提出的改进（Hu 等，2025；Yu 等，）

2025). For each problem q, the algorithm samples a group of responses $\{o_1, o_2, ..., o_G\}$ from the old policy $\pi_{\theta_{old}}$, and update the policy π_θ by maximizing the following objective:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{q\sim D, \{o_i\}_{i=1}^{G}\sim\pi_\theta(\cdot|q)} \left[\frac{1}{\sum_{i=1}^{G}|o_i|} \sum_{i=1}^{G}\sum_{j=1}^{|o_i|} \min\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)} A_{i,j}, \text{clip}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}, 1-\varepsilon_{\text{low}}, 1+\varepsilon_{\text{high}} \right) A_{i,j} \right) \right] \tag{1}$$

where ε_{low} and $\varepsilon_{\text{high}}$ are hyper-parameters. $A_{i,j}$ is the advantage, which is computed by the rewards $\{r_1, r_2, ..., r_G\}$ of responses in the same group:

$$A_{i,j} = \frac{r_i - \text{mean}(\{r_i\}_{i=1}^{G})}{\text{std}(\{r_i\}_{i=1}^{G})} \tag{2}$$

Upon the original GRPO algorithm, we incorporate several enhancements from recent research:

- **Removal of KL Loss** (He et al., 2025; Hu et al., 2025): simply removing the KL loss effectively unleashes the full potential of the policy model without compromising training stability.
- **Dynamic Sampling** (Yu et al., 2025): in RL rollout phase, we over-sample and filter out prompts with passrate equal to 1 and 0, leaving all prompts in the batch with effective gradients while maintaining a consistent batch size. This strategy automatically calibrates problem difficulty throughout policy training.
- **Clip-Higher** (Yu et al., 2025): we increase the upper clip bounds $\varepsilon_{\text{high}}$ in Eq. 1, with a fixed lower clip bounds ε_{low}. It can mitigate the entropy convergence problem and facilitate the policy to explore new solutions.

During training, we identify two key challenges affecting model performance: sparse rewards for code problems and diminishing sampling efficiency of dynamic sampling. Therefore, we propose **test complexity driven reward** function and **easy data re-sampling** approach, respectively.

3.3.1 Test Difficulty Driven Reward

Currently, for algorithm code generation tasks, existing RL works such as Deepseek-R1 Guo et al. (2025) adopt a rule-based reward strategy, where a solution is rewarded only if the generated code passes all the test cases for a given problem. However, for difficult algorithmic problems, the model might never receive any reward, preventing it from learning from these challenging cases and reducing training efficiency for dynamic sampling.

Various Test Difficulty in IOI Scoring Rules To address this limitation, we propose a new reward mechanism, test difficulty driven reward. The design is inspired by the scoring rule of the International Olympiad in Informatics (IOI, IOI 2024). In IOI contests, each complete problem is divided into multiple subtasks, and participants will obtain points for each subtask they successfully complete. Each subtask will have tests with different difficulty. Assigning different scores to subtasks better reflects how humans solve problems. For challenging problems, the model can still earn partial scores by solving some of the subtasks, which allows better utilization of these difficult examples during training.

Assigning Difficulty to Tests Based on Pass Rates We propose a technique for grouping test cases based on their difficulty. We utilize several models to perform multiple rollouts on each

2025)。对于每个问题 q，算法从旧策略 $\pi_{\theta_{old}}$ 中采样一组响应 $\{o_1, o_2, ..., o_G\}$，并通过最大化以下目标来更新策略 π_θ：

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{q\sim D,\{o_i\}_{i=1}^{G}\sim\pi_\theta(\cdot|q)} \left[\frac{1}{\sum_{i=1}^{G}|o_i|} \sum_{i=1}^{G}\sum_{j=1}^{|o_i|} \min\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)} A_{i,j}, \text{clip}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}, 1-\varepsilon_{\text{low}}, 1+\varepsilon_{\text{high}} \right) A_{i,j} \right) \right] \quad (1)$$

其中 ε_{low} 和 $\varepsilon_{\text{high}}$ 是超参数。$A_{i,j}$ 是优势值，通过同一组中响应的奖励 $\{r_1, r_2, ..., r_G\}$ 计算得出：

$$A_{i,j} = \frac{r_i - \text{mean}(\{r_i\}_{i=1}^{G})}{\text{std}(\{r_i\}_{i=1}^{G})} \quad (2)$$

在原始的GRPO算法基础上，我们结合了近期研究中的若干改进：

- 去除KL损失（He等，2025；Hu等，2025）：仅仅去除KL损失就能有效释放策略模型的全部潜力，而不会影响训练的稳定性。
- 动态采样（Yu 等，2025）：在强化学习展开阶段，我们进行过采样并过滤掉通过率等于1和0的提示，只保留具有有效梯度的批次中的所有提示，同时保持批次大小的一致性。这一策略在整个策略训练过程中自动校准问题的难度。
- Clip-Higher（Yu 等人，2025）：我们在式（1）中增加上限剪裁值 $\varepsilon_{\text{high}}$，同时保持下限剪裁值 ε_{low} 不变。这样可以缓解熵收敛问题，并促进策略探索新的解决方案。

在训练过程中，我们识别出影响模型性能的两个关键挑战：代码问题的稀疏奖励和动态采样的采样效率递减。因此，我们分别提出了测试复杂度驱动的奖励函数和简易的数据重采样方法。

3.3.1 测试难度驱动的奖励

目前，对于算法代码生成任务，现有的强化学习方法如 Deepseek-R1 Guo 等人（2025）采用基于规则的奖励策略，即只有当生成的代码通过给定问题的所有测试用例时，才给予奖励。然而，对于困难的算法问题，模型可能永远不会获得任何奖励，阻碍其从这些具有挑战性的案例中学习，降低了动态采样的训练效率。

IOI评分规则中的各种测试难度 为了解决这一限制，我们提出了一种新的奖励机制——测试难度驱动奖励。该设计受到国际信息学奥林匹克（IOI，IOI 2024）评分规则的启发。在IOI比赛中，每个完整的问题被划分为多个子任务，参赛者完成每个子任务后将获得相应的分数。每个子任务都包含不同难度的测试。为子任务分配不同的分数更好地反映了人类解决问题的方式。对于具有挑战性的问题，模型仍然可以通过解决部分子任务获得部分分数，从而在训练过程中更好地利用这些难度较大的示例。

根据通过率为测试分配难度我们提出了一种基于难度对测试用例进行分组的技术。我们利用多种模型对每个测试用例进行多次模拟。

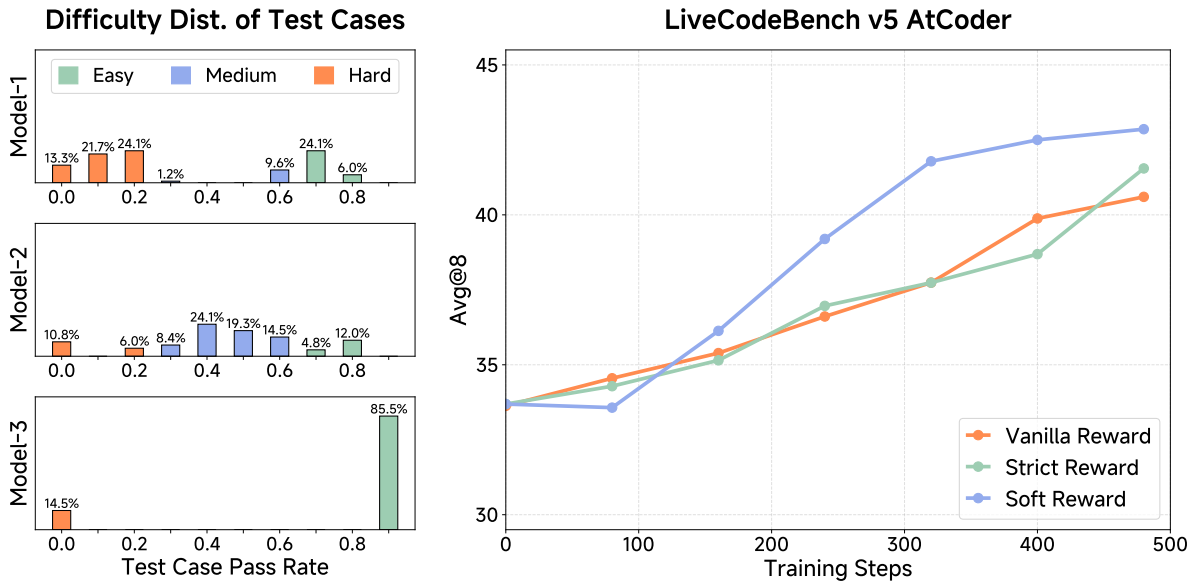

Figure 5 Experiments with test difficulty driven reward.

problem, and calculate the pass rate for each test case across all model-generated solutions. We then cluster the test cases into different difficulty levels according to their pass rates, with lower pass rates indicating higher difficulty. The left part of Figure 5 presents the pass rates and difficulty levels for each test case of certain problem. The results reveal a clear stratification of test difficulty, and demonstrate that more capable models achieve higher pass rates.

Reward Rules After categorizing the tests into different difficulty levels, we design two reward schemes based on these difficulty levels: a strict scheme and a soft scheme. (1) *Strict Reward*. Under the strict reward scheme, a solution receives the reward corresponding to a difficulty level only if it passes all tests in that group as well as in all lower-difficulty groups. (2) *Soft Reward*. In contrast, the soft reward scheme distributes the total score of each group equally among its tests. The final reward is the sum of the scores for all passed tests. The right part of Figure 5 compares the performance achieved by two reward schemes against the baseline without test difficulty driven reward.

3.3.2 Easy Data Filter and Re-Sampling

During RL training, as the policy improves, an increasing number of problems achieve a perfect pass rate of 1. Under dynamic sampling mechanism, these problems are subsequently filtered from the batch for policy update. This filtration leads to drastic sampling efficiency degradation, as more rollouts are required to construct a batch of fixed size. A straightforward approach to address this efficiency issue would be to entirely remove problems with perfect pass rates from the training data. However, our preliminary studies show that this method introduces significant instability in policy updates.

To improve sampling efficiency without risking policy collapse, we developed an easy data re-sampling strategy. During the training process, we maintain an easy data pool, where problems with perfect pass rates are stored. When performing rollouts, there is a probability α (10% in our experiments) to sample data from this easy data pool. This strategy effectively stabilizes the policy update while improving sampling efficiency, especially in the later phases of RL training.

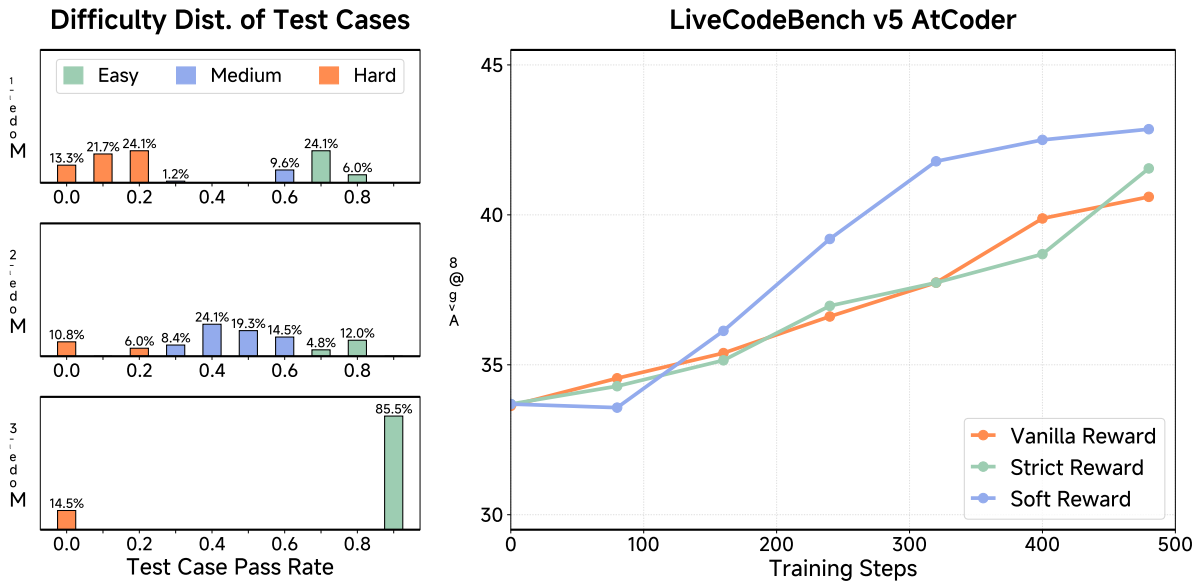

图 5 基于测试难度驱动奖励的实验

问题，并计算每个测试用例在所有模型生成的解答中的通过率。然后，我们根据通过率将测试用例划分为不同的难度等级，较低的通过率表示更高的难度。图5的左侧展示了某个问题的每个测试用例的通过率和难度等级。结果显示出测试难度的明显分层，并证明了更强大的模型能够实现更高的通过率。

奖励规则　将测试按不同难度等级分类后，我们基于这些难度等级设计了两种奖励方案：严格方案和宽松方案。 (1) *Strict Reward.* 在严格奖励方案下，只有当一个解答通过该组所有测试以及所有较低难度组的测试时，才会获得对应难度等级的奖励。 (2) *Soft Reward.* 相比之下，宽松奖励方案将每个组的总分平均分配到其所有测试中。最终奖励是所有通过测试的得分之和。图5的右侧部分比较了两种奖励方案在没有测试难度驱动奖励的基线条件下的表现。

3.3.2 简单数据过滤和重采样

在强化学习训练过程中，随着策略的改进，越来越多的问题达到完美通过率1。在动态采样机制下，这些问题随后会从批次中被筛除以进行策略更新。这种筛选导致采样效率大幅下降，因为需要更多的模拟运行来构建一个固定大小的批次。一种解决这一效率问题的直接方法是完全将完美通过率的问题从训练数据中移除。然而，我们的初步研究表明，这种方法会在策略更新中引入显著的不稳定性。

为了在不冒政策崩溃风险的情况下提高采样效率，我们开发了一种简便的数据重采样策略。在训练过程中，我们维护一个简单数据池，存放通过率完美的问题。在进行滚动时，我们的实验中有α (10%的概率)从这个简单数据池中采样数据。该策略有效地稳定了策略更新，同时提高了采样效率，特别是在强化学习训练的后期阶段。

3.3.3 Hyper-Parameters

In our experiment, we employed a training batch size of 512, with an actor mini-batch size of 32. We executed 16 gradient updates per training iteration at a learning rate of 1e-6. The maximum sequence length was set to 32,768 tokens to facilitate complex reasoning tasks. During the training phase, both temperature and top-p parameters were configured at 1.0 to promote output diversity.

3.4 RL Infrastructures

We develop the Seamless Rollout Engine and enhance vLLM's robustness to enable efficient dynamic-sampling-based RL training. We construct our RL system based on verl (Sheng et al., 2024), an open-source RL training library. The library uses Ray (Moritz et al., 2018) to manage computation and communication, implementing the rollout and training phases in Ray Actors and exchanging training data through Ray Objects. Although verl supports flexible implementations of various RL algorithms, it suffers from GPU idle time during both rollout and reward computation phases. Due to the skewness in response lengths, we observe that most GPUs remain idle while waiting for a few long-sequence rollout workers, resulting in wasted computational resources and a slow training process. Several prior works have identified this issue and proposed system-level solutions (Seed et al., 2025; Team et al., 2025; Zhong et al., 2024b). However, most of these solutions rely on asynchronous training, which modifies the underlying algorithm and introduces staleness in long-sequence responses. Rule-based reward computation is also time-consuming, particularly for code data, leading to idle periods for valuable GPU resources. Our use of dynamic sampling, while improving sample efficiency, exacerbates GPU idle time, and leads to wasted examples during multi-turn rollouts. To simultaneously optimize GPU utilization and reduce sample waste, we develop the Seamless Rollout Engine, opportunistically filling sample batches into rollout while performing asynchronous reward computation. Our system builds on the vLLM inference engine (Kwon et al., 2023), and we collaborate with the open-source community to enhance the robustness of vLLM's "external launch" mode within the verl framework. Additionally, we implement MTP in vLLM to support both MiMo-7B and MiMo-7B-RL.

3.4.1 Seamless Rollout Engine

Seamless Rollout Engine optimizes GPU utilization in rollout workers through efficient task scheduling, minimizing idle time during continuous operation. The engine consists of the following components: (a) continuous rollout, (b) asynchronous reward computation, and (c) early termination. It achieves a 2.29× speedup in training and a 1.96× speedup in validation.

Continuous Rollout The core of Seamless Rollout Engine lies in proactively handling completed rollout tasks and initiating new rollouts. Unlike naive dynamic sampling implementations that delay reward computation until all rollout workers complete, Seamless Rollout Engine eliminates synchronization barriers between generation and reward phases. It actively monitors completed workers, immediately computes their rewards, and triggers new rollouts on demand. After computing rewards, we update the number of valid samples and the current step's pass-rate statistics, then launch new rollout tasks if active tasks are insufficient to meet training demands based on these statistics. As illustrated in Figure 6, the Seamless Rollout Engine initiates a new task upon completing rollout tasks ③④①⑥ to meet demand, whereas after finishing tasks ②⑤⑦, it predicts that ongoing tasks are sufficient and thus schedules no additional ones.

3.3.3 超参数

在我们的实验中，我们采用了训练批次大小为 512，演员迷你批次大小为 32。在每次训练迭代中执行 16 次梯度更新，学习率为 1e-6。最大序列长度设置为 32,768 个标记，以便进行复杂的推理任务。在训练阶段，温度和 top-p 参数均设置为 1.0，以促进输出的多样性。

3.4 强化学习基础设施

我们开发了无缝滚动引擎，并增强了 vLLM 的鲁棒性，以实现高效的基于动态采样的强化学习训练。我们基于 verl (Sheng et al., 2024) 构建了我们的强化学习系统，verl 是一个开源的强化学习训练库。该库使用 Ray (Moritz et al., 2018) 来管理计算和通信，在 Ray Actor 中实现滚动和训练阶段，并通过 Ray Object 交换训练数据。虽然 verl 支持各种强化学习算法的灵活实现，但在滚动和奖励计算阶段会出现 GPU 空闲时间。由于响应长度的偏斜，我们观察到大多数 GPU 在等待少数长序列滚动工作者时处于空闲状态，导致计算资源浪费和训练速度变慢。此前的多项研究已识别出这一问题并提出了系统级解决方案 (Seed et al., 2025; Team et al., 2025; Zhong et al., 2024b)。然而，这些方案大多依赖异步训练，修改了底层算法，并引入了长序列响应的陈旧性。基于规则的奖励计算也非常耗时，尤其是对于代码数据，导致宝贵的 GPU 资源出现空闲。我们采用的动态采样虽然提高了样本效率，但加剧了 GPU 空闲时间，并在多轮滚动中造成样本浪费。为了同时优化 GPU 利用率并减少样本浪费，我们开发了无缝滚动引擎，在进行异步奖励计算的同时，机会性地将样本批次填充到滚动中。我们的系统基于 vLLM 推理引擎 (Kwon et al., 2023)，并与开源社区合作，增强 verl 框架中 vLLM “外部启动” 模式的鲁棒性。此外，我们在 vLLM 中实现了 MTP，以支持 MiMo-7B 和 MiMo-7B-RL。

3.4.1 无缝部署引擎

无缝部署引擎通过高效的任务调度优化 rollout 工作器中的 GPU 利用率，最大限度地减少连续运行中的空闲时间。该引擎由以下组件组成：(a) 连续 rollout，(b) 异步奖励计算，以及 (c) 提前终止。它在训练中实现了 2.29× 的加速，在验证中实现了 1.96× 的加速。

连续滚动 Seamless Rollout 引擎的核心在于主动处理已完成的滚动任务并启动新的滚动。不同于天真的动态采样实现，后者在所有滚动工作者完成之前延迟奖励计算，Seamless Rollout 引擎消除了生成阶段和奖励阶段之间的同步障碍。它主动监控已完成的工作者，立即计算它们的奖励，并根据需要触发新的滚动。当计算出奖励后，我们会更新有效样本的数量和当前步骤的通过率统计，然后根据这些统计数据，如果活跃任务不足以满足训练需求，就启动新的滚动任务。如图6所示，Seamless Rollout 引擎在完成滚动任务 ③④①⑥ 后启动新任务以满足需求，而在完成任务 ②⑤⑦ 后，预测进行中的任务已足够，因此不安排额外的任务。

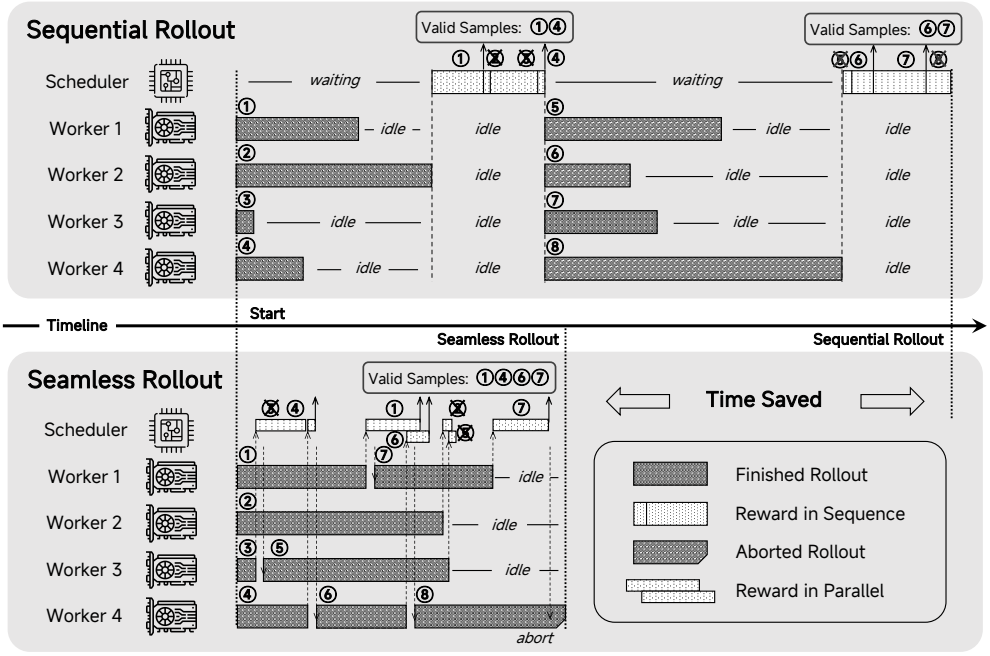

Figure 6 An overview of the Seamless Rollout Engine for MiMo-7B-RL.

Asynchronous Reward Computation While reward computation for math data is rapid, judging code-related data incurs significant overhead, leading to prolonged GPU idle time. Additionally, the sequential nature of naive reward computation fails to utilize the multiprocessing capabilities of modern processing units. To resolve these issues, we employ Ray to launch asynchronous reward computation, which facilitates concurrent management of rollout and reward tasks. Upon task completion, the system dynamically forwards rollout outputs for reward evaluation or aggregates results to update the sample state, as shown in Figure 6. Dedicated servers are allocated for code-specific reward computation to prevent bottlenecks in the rollout pipeline.

Early Termination When the number of valid samples exceeds the required training batch size, careful management of ongoing tasks becomes essential. Abrupt termination of ongoing tasks tends to suppress the generation of long-sequence responses, which could destabilize RL training dynamics. A straightforward solution involves waiting for all active tasks to complete before randomly sampling required batch from the outputs. However, this approach may extend waiting times if a long-sequence rollout initiates near the end of the dynamic sampling phase. To mitigate this delay while preserving data distribution integrity, we implement a first-in-first-out selection strategy. We terminate ongoing tasks only if the valid sample count meets the batch requirement and all tasks initiated prior to these selected samples have completed. In Figure 6, the last rollout is aborted since earlier samples already reach the required batch size.

Experimental Analysis We randomly choose a 5-step training trace to evaluate the performance of Seamless Rollout Engine. The experiment is conducted on 256 H20 GPUs, and the results are presented in Table 2. "Overall Speedup" measures end-to-end RL training efficiency; "Rollout Speedup" shows the acceleration of rollout and reward tasks; "Normalized GPU Idle Time" reflects the total idle GPU hours. The above metrics are normalized with respect to the naive dynamic sampling implementation. "GPU Idle Ratio" quantifies the average proportion of GPU inactivity during rollout and reward computation; "Sample Waste Ratio" represents the ratio of excess valid

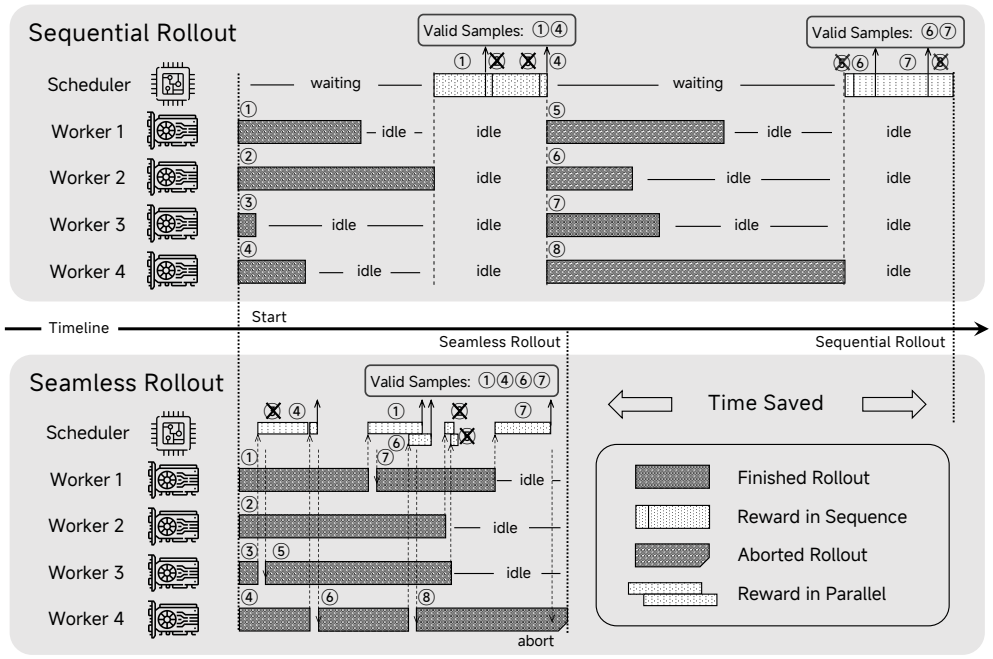

图6 MiMo-7B-RL无缝部署引擎概述。

异步奖励计算　虽然数学数据的奖励计算速度很快，但代码相关数据的判断会带来显著的开销，导致GPU空闲时间延长。此外，天真的奖励计算的顺序性也未能充分利用现代处理单元的多处理能力。为了解决这些问题，我们采用Ray启动异步奖励计算，从而实现回滚和奖励任务的并发管理。在任务完成后，系统会动态地将回滚输出用于奖励评估，或将结果聚合以更新样本状态，如图6所示。为防止回滚流程中的瓶颈，专门为代码相关的奖励计算分配了专用服务器。

提前终止 当有效样本数量超过所需的训练批次大小时，仔细管理进行中的任务变得至关重要。突然终止进行中的任务往往会抑制长序列响应的生成，这可能会使强化学习（RL）训练动态变得不稳定。一种简单的解决方案是等待所有活跃任务完成，然后从输出中随机采样所需的批次。然而，如果在动态采样阶段接近尾声时启动了长序列的滚动，这种方法可能会延长等待时间。为了在保持数据分布完整性的同时减轻这种延迟，我们实现了一种先进先出（FIFO）选择策略。只有当有效样本数满足批次要求，并且在这些已选择样本之前启动的所有任务都已完成时，我们才会终止进行中的任务。在图6中，由于早期样本已达到所需的批次大小，最后一次滚动被中止。

实验分析 我们随机选择一个5步训练轨迹来评估Seamless Rollout Engine的性能。该实验在256块H20 GPU上进行，结果如表2所示。“整体加速比”衡量端到端的强化学习训练效率；“Rollout加速比”显示rollout和奖励任务的加速情况；“归一化GPU空闲时间”反映总的空闲GPU小时数。上述指标均以天真动态采样实现为基准进行归一化。“GPU空闲比”量化了rollout和奖励计算过程中GPU空闲的平均比例；“采样浪费比”表示多余有效采样的比例。

Method	Overall Speedup ↑	Rollout Speedup ↑	Normalized GPU Idle Time ↓	GPU Idle Ratio ↓	Sample Waste Ratio ↓
w/o Dynamic Sampling	2.45×	2.82×	0.36	70.8%	/
Naive Dynamic Sampling	1.00×	1.00×	1.00	69.3%	22.1%
+ Continous Rollout	1.99×	2.20×	0.25	38.8%	13.9%
+ Async. Reward	2.09×	2.34×	0.21	34.0%	16.4%
+ Early Termination	2.29×	2.61×	0.15	27.7%	12.9%

Table 2 The experimental results of Seamless Rollout Engine compared with baseline methods.

samples generated relative to the required batch size. In Seamless Rollout Engine, aborted tasks are considered in GPU idle time.

All three components contribute to faster dynamic sampling and smaller GPU idle time. Though the experiment without dynamic sampling can achieve higher throughput, it incurs significant sample inefficiency due to numerous zero-gradient training samples. These zero-gradient samples not only diminishes the effective training batch size, but also risk destabilizing the training dynamics of the RL algorithm. Given an average sample pass rate of 41% within this 5-step experiment, static sampling achieves a sample efficiency similar to naive dynamic sampling; the latter does not train zero-gradient data but incurs wasted samples. Equipped with all three components, Seamless Rollout Engine achieves a comparable one-step training time compared to static sampling while demonstrating superior sample efficiency. The sample pass rate of 41% leads to a sample waste ratio of 22% in the naive implementation; in practice, this ratio can be larger in different situations. Through continous rollout and dynamic launch scheduling, Seamless Rollout Engine reduces the sample waste ratio to around 15%.

Accelerated Validation During validation, we can directly stream the rollout and reward tasks using Seamless Rollout Engine. Similar to the naive implementation, currently we set the validation batch size equal to the dataset length and launch all rollout tasks simultaneously. Our implementation utilizes asynchronous reward computation, achieving a 1.96× speedup while reducing idle GPU time to 25%, as demonstrated in Table 3. Notably, the experimental results demonstrate Seamless Rollout Engine's potential for static sampling, which also has one-pass rollout and reward computation. If the validation dataset is sufficiently large, further acceleration can be achieved by optimizing the batch size for validation and employing continuous rollout.

Method	Speedup ↑	Normalized GPU Idle Time ↓	GPU Idle Ratio ↓
Naive Validation	1×	1	65.8%
Seamless Rollout Engine	1.96×	0.25	32.9%

Table 3 The validation speedup and GPU idle time of the naive implementation and the Seamless Rollout Engine. The experiment is conducted on 256 H20 GPUs using our full validation dataset.

3.4.2 vLLM-based Inference Engine

Our RL system employs vLLM (Kwon et al., 2023) as the inference engine. To accommodate our model's new features, we have extended the framework with additional functionalities.

MTP Support As described in Section 2.2, our models integrate MTP modules to enhance performance. We have implemented and open-sourced MTP support for our models, enabling

Method	Overall Speedup ↑	Rollout Speedup ↑	Normalized GPU Idle Time ↓	GPU Idle Ratio ↓	Sample Waste Ratio ↓
w/o Dynamic Sampling	2.45×	2.82×	0.36	70.8%	/
Naive Dynamic Sampling	1.00×	1.00×	1.00	69.3%	22.1%
+ Continous Rollout	1.99×	2.20×	0.25	38.8%	13.9%
+ Async. Reward	2.09×	2.34×	0.21	34.0%	16.4%
+ Early Termination	2.29×	2.61×	0.15	27.7%	12.9%

表2 无缝滚动引擎与基线方法的实验结果。

相对于所需批次大小生成的样本。在 Seamless Rollout 引擎中，中止的任务被视为 GPU 空闲时间。

所有三个组件都促成了更快的动态采样和更少的GPU空闲时间。虽然没有动态采样的实验可以实现更高的吞吐量，但由于大量零梯度训练样本，它会带来显著的样本效率低下。这些零梯度样本不仅降低了有效的训练批次大小，还可能导致RL算法的训练动态不稳定。在这次5步实验中，平均样本通过率为41%，静态采样实现了与天真动态采样相似的样本效率；后者不训练零梯度数据，但会造成样本浪费。配备所有三个组件后，Seamless Rollout Engine在一轮训练时间上与静态采样相当，同时展现出更优的样本效率。41%的样本通过率导致天真实现中的样本浪费比为22%；在实际应用中，这一比例在不同情况下可能更大。通过持续的rollout和动态调度，Seamless Rollout Engine将样本浪费比降低到大约15%。

加速验证 在验证过程中，我们可以使用 Seamless Rollout Engine 直接流式处理 rollout 和 reward 任务。与朴素实现类似，目前我们将验证批次大小设置为数据集长度，并同时启动所有 rollout 任务。我们的实现采用异步 reward 计算，在实现了 {v*} 的加速的同时，将空闲 GPU 时间减少到 25%，如表 3 所示。值得注意的是，实验结果展示了 Seamless Rollout Engine 在静态采样中的潜力，该方法也支持一次性 rollout 和 reward 计算。如果验证数据集足够大，还可以通过优化验证的批次大小和采用连续 rollout 来实现进一步加速。

Method	Speedup ↑	Normalized GPU Idle Time ↓	GPU Idle Ratio ↓
Naive Validation	1×	1	65.8%
Seamless Rollout Engine	1.96×	0.25	32.9%

表3 朴素实现和无缝滚动引擎的验证加速比和GPU空闲时间。实验在256个H20 GPU上使用我们的完整验证数据集进行。

3.4.2 基于 vLLM 的推理引擎

我们的强化学习系统采用 vLLM（Kwon 等，2023）作为推理引擎。为了适应我们模型的新特性，我们在框架中添加了额外的功能。

MTP 支持 如第 2.2 节所述，我们的模型集成了 MTP 模块以提升性能。我们已经实现并开源了对模型的 MTP 支持，能够

Benchmark	GPT-4o 0513	Claude-3.5-Sonnet-1022	OpenAI o1-mini	QwQ-32B Preview	R1-Distill-Qwen-14B	R1-Distill-Qwen-7B	MiMo-7B-RL
General							
GPQA Diamond (Pass@1)	49.9	65.0	60.0	54.5	59.1	49.1	54.4
SuperGPQA (Pass@1)	42.4	48.2	45.2	43.6	40.6	28.9	40.5
DROP (3-shot F1)	83.7	88.3	83.9	71.2	85.5	77.0	78.7
MMLU-Pro (EM)	72.6	78.0	80.3	52.0	68.8	53.5	58.6
IF-Eval (Prompt Strict)	84.3	86.5	84.8	40.4	78.3	60.5	61.0
Mathematics							
MATH500 (Pass@1)	74.6	78.3	90.0	90.6	93.9	92.8	95.8
AIME 2024 (Pass@1)	9.3	16.0	63.6	50.0	69.7	55.5	68.2
AIME 2025 (Pass@1)	11.6	7.4	50.7	32.4	48.2	38.8	55.4
Code							
LiveCodeBench v5 (Pass@1)	32.9	38.9	53.8	41.9	53.1	37.6	57.8
LiveCodeBench v6 (Pass@1)	30.9	37.2	46.8	39.1	31.9	23.9	49.3

Table 4 Comparison between MiMo-7B-RL and other representative models.

efficient inference for MTP-equipped architectures.

Better Robustness In verl, vLLM is deployed using the *external launch* mode, which may show instability in some scenarios. We've enhanced engine robustness to address these issues. We clear computed blocks in *prefix caching* during pre-emption to maintain KVCache consistency. We disable asynchronous output processing when increasing the number of scheduler steps to ensure compatibility and optimize performance.

3.5 Post-Training Evaluation

3.5.1 Evaluation Setup

We comprehensively evaluate reasoning models across a diverse range of benchmarks:

Language understanding and reasoning: MMLU-Pro (Wang et al., 2024).

Scientific question answering: GPQA Diamond (Rein et al., 2024) with averaged score of 8 repetitions; SuperGPQA (Du et al., 2025).

Instruction following: IFEval (Zhou et al., 2023) with averaged score of 8 repetitions.

Reading comprehension: DROP (Dua et al., 2019).

Mathematics reasoning: MATH500 (Lightman et al., 2024); AIME 2024 (MAA, 2024) and AIME 2025 (MAA, 2025) with averaged score of 32 repetitions.

Coding: LiveCodeBench v5 (20240801-20250201) (Jain et al., 2024) and LiveCodeBench v6 (20250201-20250501) (Jain et al., 2024) with averaged score of 8 repetitions.

During evaluation, we set the sampling temperature to 0.6 and top-p to 0.95 for all benchmarks. We set the maximum generation length to 32,768 tokens for mathematics reasoning, coding, and scientific question answering benchmarks, and to 8,192 tokens for other benchmarks.

We compare MiMo-7B-RL against several strong baselines, including two non-reasoning models GPT-4o-0513, Claude-Sonnet-3.5-1022, and reasoning models OpenAI-o1-mini, QwQ-32B-Preview, DeepSeek-R1-Distill-Qwen-14B, and DeepSeek-R1-Distill-Qwen-7B.

Benchmark	GPT-4o 0513	Claude-3.5-Sonnet-1022	OpenAI o1-mini	QwQ-32B Preview	R1-Distill-Qwen-14B	R1-Distill-Qwen-7B	MiMo-7B-RL
General							
GPQA Diamond (Pass@1)	49.9	65.0	60.0	54.5	59.1	49.1	54.4
SuperGPQA (Pass@1)	42.4	48.2	45.2	43.6	40.6	28.9	40.5
DROP (3-shot F1)	83.7	88.3	83.9	71.2	85.5	77.0	78.7
MMLU-Pro (EM)	72.6	78.0	80.3	52.0	68.8	53.5	58.6
IF-Eval (Prompt Strict)	84.3	86.5	84.8	40.4	78.3	60.5	61.0
Mathematics							
MATH500 (Pass@1)	74.6	78.3	90.0	90.6	93.9	92.8	95.8
AIME 2024 (Pass@1)	9.3	16.0	63.6	50.0	69.7	55.5	68.2
AIME 2025 (Pass@1)	11.6	7.4	50.7	32.4	48.2	38.8	55.4
Code							
LiveCodeBench v5 (Pass@1)	32.9	38.9	53.8	41.9	53.1	37.6	57.8
LiveCodeBench v6 (Pass@1)	30.9	37.2	46.8	39.1	31.9	23.9	49.3

表4 MiMo-7B-RL 与其他代表性模型的比较。

针对配备MTP的架构的高效推理。

在 verl 中，vLLM 使用 *external launch* 模式部署，可能在某些场景下表现出不稳定。我们增强了引擎的鲁棒性以解决这些问题。在抢占期间，我们会清除 *prefix caching* 中的已计算块，以保持 KVCache 的一致性。当增加调度器步骤的数量时，我们会禁用异步输出处理，以确保兼容性并优化性能。

3.5 训练后评估

3.5.1 评估设置

我们在各种基准测试中全面评估推理模型：

语言理解与推理：MMLU-Pro（Wang 等，2024）。

科学问题回答：GPQA Diamond（Rein 等，2024）在8次重复中的平均得分；SuperGPQA（Du 等，2025）。

指令遵循：使用 IFEval（Zhou 等人，2023）进行8次重复的平均得分。

阅读理解：DROP（Dua 等人，2019）。

数学推理：MATH500（Lightman 等，2024）；AIME 2024（MAA，2024）和 AIME 2025（MAA，2025），平均得分为 32 次重复。

编码：LiveCodeBench v5（20240801-20250201）（Jain 等，2024）和 LiveCodeBench v6（20250201-20250501）（Jain 等，2024），平均得分为 8 次重复。

在评估过程中，我们将所有基准的采样温度设置为0.6，top-p设置为0.95。对于数学推理、编码和科学问答基准，我们将最大生成长度设置为32,768个标记；对于其他基准，则设置为8,192个标记。

我们将MiMo-7B-RL与几个强大的基线模型进行了比较，包括两个非推理模型GPT-4o-0513、Claude-Sonnet-3.5-1022，以及推理模型OpenAI-o1-mini、QwQ-32B-预览、DeepSeek-R1-Distill-Qwen-14B和DeepSeek-R1-Distill-Qwen-7B。

3.5.2 Evaluation Results

Table 4 shows the evaluation results. In mathematics reasoning, MiMo-7B-RL achieves top-tier performance among models of comparable parameter sizes, trailing only slightly behind DeepSeek-R1-Distill-Qwen-14B on AIME 2024. For algorithm code generation tasks, MiMo-7B-RL demonstrates extremely impressively results. On LiveCodeBench v5, it significantly outperforms OpenAI o1-mini, while on the latest LiveCodeBench v6, our model achieves a score of 49.3%, surpassing QwQ-32B-Preview by over 10 points, demonstrating its robust and stable capabilities. Notably, MiMo-7B-RL also maintains strong general performance, exceeding both QwQ-32B-Preview and DeepSeek-R1-Distill-Qwen-7B, though we only include mathematics and code problems for RL.

We also presents the evaluation results for different version of MiMo-7B in Table 5. MiMo-7B-RL-Zero is trained from MiMo-7B-Base, while MiMo-7B-RL is trained from MiMo-7B-SFT. As shown, RL from the base model exhibits a stronger growth trend, improving from 32.9% to on AIME 2024 for instance. Nonetheless, RL training from the SFT model achieves a higher performance ceiling, attaining the best results across all evaluated benchmarks.

Benchmark	MiMo-7B-Base	MiMo-7B-RL-Zero	MiMo-7B-SFT	MiMo-7B-RL
Mathematics				
MATH500	37.4	93.6	93.0	95.8
AIME 2024	32.9	56.4	58.7	68.2
AIME 2025	24.3	46.3	44.3	55.4
Code				
LiveCodeBench v5	32.9	49.1	52.3	57.8
LiveCodeBench v6	29.1	42.9	45.5	49.3

Table 5 Evaluation results of MiMo-Series models on mathematics and coding benchmarks

3.6 Discussion

In this section, we share insights and observations from our exploration of MiMo-7B's post-training process, which we hope will benefit the research community.

SFT for Format Alignment In the initial RL training steps from MiMo-7B-Base, we observe that the model primarily learns to adapt the answer extraction function, e.g., "\boxed{}" for mathematics problems. Therefore, we investigate a "light-weight" SFT to help the base model align with the expected answer format. However, as Figure 7 demonstrates, the resulting MiMo-7B-RL-LiteSFT model fails in both reasoning potential and final performance. While MiMo-7B-RL-LiteSFT begins with a higher performance than MiMo-7B-RL-Zero, it falls behind the base model's trajectory after just 500 steps. Furthermore, when compared to MiMo-7B-RL, which undergoes "heavier" SFT, MiMo-7B-RL-LiteSFT exhibits a similar growth trend but significantly

Figure 7 Performance comparison of three MiMo model variants during the RL process.

3.5.2 评估结果

表4显示了评估结果。在数学推理方面，MiMo-7B-RL在参数规模相当的模型中表现出色，仅略逊于DeepSeek-R1-Distill-Qwen-14B在AIME 2024上的表现。在算法代码生成任务中，MiMo-7B-RL展现了极其令人印象深刻的结果。在LiveCodeBench v5上，它显著优于OpenAI o1-mini，而在最新的LiveCodeBench v6上，我们的模型取得了49.3%的得分，超过QwQ-32B-Preview超过10个百分点，展示了其强大且稳定的能力。值得注意的是，MiMo-7B-RL也保持了强大的通用性能，超过了QwQ-32B-Preview和DeepSeek-R1-Distill-Qwen-7B，尽管我们只在RL中包含了数学和代码问题。

我们还在表5中展示了不同版本MiMo-7B的评估结果。MiMo-7B-RL-Zero是从MiMo-7B-Base训练而来，而MiMo-7B-RL则是从MiMo-7B-SFT训练而成。如图所示，从基础模型进行的RL展现出更强的增长趋势，例如在AIME 2024上从32.9%提升。然而，从SFT模型进行的RL训练达到了更高的性能上限，在所有评估基准中都取得了最佳结果。

Benchmark	MiMo-7B-Base	MiMo-7B-RL-Zero	MiMo-7B-SFT	MiMo-7B-RL
Mathematics				
MATH500	37.4	93.6	93.0	95.8
AIME 2024	32.9	56.4	58.7	68.2
AIME 2025	24.3	46.3	44.3	55.4
Code				
LiveCodeBench v5	32.9	49.1	52.3	57.8
LiveCodeBench v6	29.1	42.9	45.5	49.3

表5 MiMo系列模型在数学和编码基准测试中的评估结果

3.6 讨论

在本节中，我们分享了对MiMo-7B训练后过程的探索中的见解和观察，希望能对研究社区有所帮助。

用于格式对齐的SFT 在从MiMo-7B-Base的初始RL训练步骤中，我们观察到模型主要学习调整答案提取函数，例如“\boxed{}”用于数学题目。因此，我们研究了一种“轻量级”SFT，以帮助基础模型与预期的答案格式对齐。然而，正如图7所示，最终的MiMo-7B-RL-LiteSFT模型在推理潜力和最终性能方面都未能达到预期。虽然MiMo-7B-RL-LiteSFT的起始性能高于MiMo-7B-RL-Zero，但在仅仅500步后，它就落后于基础模型的轨迹。此外，与经历“更重”SFT的MiMo-7B-RL相比，MiMo-7B-RL-LiteSFT表现出类似的增长趋势，但明显...

图7 在强化学习过程中三种MiMo模型变体的性能比较。

underperforms due to its inferior starting point, ultimately leading to poorer final results.

Interference Between Different Domains During the later stages of RL training from MiMo-7B-Base, maintaining a performance balance between mathematics and coding tasks proves challenging. Between training steps 2000 and 2500, the model exhibits continuous improvement on code problems, while its performance on mathematical reasoning tasks fluctuates and declines. In contrast, RL training on the cold-started SFT model shows consistent improvements across both domains. Analysis of the model outputs reveals that the base model, with its strong exploration capabilities, tends to hack the reward for mathematics problems. For code problems, however, the test-case-based verifier makes reward exploitation significantly harder. This highlights the critical need for high-quality mathematical problem sets to ensure robust RL training.

Language Mixing Penalty Like DeepSeek-R1-Zero, we also observe language mixing issues during RL training on MiMo-7B-Base. To mitigate this problem, we introduce a language mixing penalty into the reward function. However, we find designing such a penalty function is challenging. While detecting Chinese characters in English responses is straightforward, the reverse is far more difficult, since mathematical equations and code inherently contain English words. As a result, the penalty not only fails to fully resolve language mixing but also introduces the risk of reward hacking, such as always generating English responses regardless of the question language.

4 Conclusion

This work introduces MiMo-7B, a series of LLMs which unlock advanced reasoning capabilities through optimized pre-training and post-training process. Exposed to diverse reasoning patterns during pre-training, MiMo-7B-Base possesses exceptional reasoning potential, outperforming models of significantly larger scale. For post-training, with our robust and efficient RL frameworks, we trained MiMo-7B-RL-Zero and MiMo-7B-RL which demonstrate superior reasoning capabilities across mathematics, code and general tasks. Notably, MiMo-7B-RL achieves 49.3% on LiveCodeBench v6 and 55.4% on AIME 2025, surpassing OpenAI's o1-mini. We hope this work offers insights for developing more powerful reasoning models.

References

J. Ainslie, J. Lee-Thorp, M. de Jong, Y. Zemlyanskiy, F. Lebron, and S. Sanghai. GQA: Training generalized multi-query transformer models from multi-head checkpoints. In H. Bouamor, J. Pino, and K. Bali, editors, Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 4895–4901, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.298. URL https://aclanthology.org/2023.emnlp-main.298.

Anthropic. Claude 3.7 sonnet and claude code, 2025. URL https://www.anthropic.com/claude/sonnet.

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, abs/2108.07732, 2021. URL https://arxiv.org/abs/2108.07732.

A. Barbaresi. Trafilatura: A web scraping library and command-line tool for text discovery and extraction. In H. Ji, J. C. Park, and R. Xia, editors, Proceedings of the 59th Annual

由于起点较低，表现不佳，最终导致较差的最终结果。

在从 MiMo-7B-Base 进行强化学习（RL）训练的后期阶段，不同领域之间的干扰使得在数学和编码任务之间保持性能平衡变得具有挑战性。在训练步骤 2000 到 2500 之间，模型在编码问题上表现出持续的提升，而在数学推理任务上的表现则波动并有所下降。相比之下，在冷启动的 SFT 模型上进行的 RL 训练在两个领域都显示出持续的改进。对模型输出的分析显示，基础模型凭借其强大的探索能力，倾向于“破解”数学问题的奖励。然而，对于编码问题，基于测试用例的验证器使得奖励的利用变得更加困难。这突显了高质量数学题集在确保稳健的 RL 训练中的关键作用。

语言混合惩罚 如 DeepSeek-R1-Zero，我们在 MiMo-7B-Base 上的 RL 训练中也观察到语言混合问题。为了解决这个问题，我们在奖励函数中引入了语言混合惩罚。然而，我们发现设计这样的惩罚函数具有一定的挑战性。虽然检测英文回答中的中文字符相对简单，但反过来则要困难得多，因为数学方程和代码本身就包含英文单词。因此，这种惩罚不仅无法完全解决语言混合问题，还可能引入奖励操控的风险，例如无论问题的语言如何，总是生成英文回答。

4 结论

本工作介绍了MiMo-7B，一系列通过优化预训练和后训练过程释放高级推理能力的LLMs。在预训练期间暴露于多样的推理模式，MiMo-7B-Base具有卓越的推理潜力，性能优于规模明显更大的模型。对于后训练，我们采用了稳健高效的RL框架，训练了MiMo-7B-RL-Zero和MiMo-7B-RL，它们在数学、代码和通用任务中展现出优越的推理能力。值得注意的是，MiMo-7B-RL在LiveCodeBench v6上达到49.3%，在AIME 2025上达到55.4%，超越了OpenAI的o1-mini。我们希望这项工作能为开发更强大的推理模型提供启示。

参考文献

J. Ainslie, J. Lee-Thorp, M. de Jong, Y. Zemlyanskiy, F. Lebron, 和 S. Sanghai. GQA：从多头检查点训练通用多查询变换器模型。收录于 H. Bouamor、J. Pino 和 K. Bali 编著的《2023 年自然语言处理经验方法会议论文集》，第 4895–4901 页，新加坡，2023 年。计算语言学协会。doi: 10.18653/v1/2023.emnlp-main.298。网址 https://aclanthology .org/2023.emnlp-main.298。Anthropic。Claude 3.7 诗篇和 Claude 代码，2025 年。网址 https://www.anthropic.com/claude/sonnet。

J. Austin, A. Odena, M. Nye, M. Bosma, H. Michalewski, D. Dohan, E. Jiang, C. Cai, M. Terry, Q. Le 等. 使用大型语言模型进行程序合成。ArXiv预印本，abs/2108.07732，2021。网址 https://arxiv.org/abs/2108.07732。

A. Barbaresi。Trafilatura：一个用于文本发现和提取的网页抓取库和命令行工具。在H. Ji、J. C. Park和R. Xia编辑的第59届年度会议论文集

Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations, pages 122–131, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-demo.15. URL https://aclanthology.org/2021.acl-demo.15.

Y. Bisk, R. Zellers, R. LeBras, J. Gao, and Y. Choi. PIQA: reasoning about physical commonsense in natural language. In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, pages 7432–7439. AAAI Press, 2020. URL https://aaai.org/ojs/index.php/AAAI/article/view/6239.

A. Z. Broder. On the resemblance and containment of documents. In Proceedings. Compression and Complexity of SEQUENCES 1997 (Cat. No. 97TB100171), pages 21–29. IEEE, 1997.

M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. D. O. Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman, et al. Evaluating large language models trained on code. ArXiv preprint, abs/2107.03374, 2021. URL https://arxiv.org/abs/2107.03374.

P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. ArXiv preprint, abs/1803.05457, 2018. URL https://arxiv.org/abs/1803.05457.

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, abs/2110.14168, 2021. URL https://arxiv.org/abs/2110.14168.

Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier. Language modeling with gated convolutional networks. In D. Precup and Y. W. Teh, editors, Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017, volume 70 of Proceedings of Machine Learning Research, pages 933–941. PMLR, 2017. URL http://proceedings.mlr.press/v70/dauphin17a.html.

X. Du, Y. Yao, K. Ma, B. Wang, T. Zheng, K. Zhu, M. Liu, Y. Liang, X. Jin, Z. Wei, et al. Supergpqa: Scaling llm evaluation across 285 graduate disciplines. ArXiv preprint, abs/2502.14739, 2025. URL https://arxiv.org/abs/2502.14739.

D. Dua, Y. Wang, P. Dasigi, G. Stanovsky, S. Singh, and M. Gardner. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In J. Burstein, C. Doran, and T. Solorio, editors, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 2368–2378, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1246. URL https://aclanthology.org/N19-1246.

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, et al. Are we done with mmlu? ArXiv preprint, abs/2406.04127, 2024. URL https://arxiv.org/abs/2406.04127.

F. Gloeckle, B. Y. Idrissi, B. Rozière, D. Lopez-Paz, and G. Synnaeve. Better & faster large language models via multi-token prediction. In Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024. OpenReview.net, 2024. URL https://openreview.net/forum?id=pEWAcejiU2.

计算语言学协会会议与第十一届国际自然语言处理联合会议：系统演示，页码 122–131，线上，2021年。计算语言学协会。doi: 10.18653/v1/2021.acl-demo.15。网址 https://aclanthology.org/2021.acl-demo.15。Y. Bisk, R. Zellers, R. LeBras, J. Gao 和 Y. Choi。PIQA：关于自然语言中的物理常识推理。在第三十四届人工智能协会会议（AAAI 2020）、第三十二届人工智能创新应用会议（IAAI 2020）、第十届人工智能教育进展研讨会（EAAI 2020），美国纽约，2020年2月7-12日，页码 7432–7439。AAAI出版社，2020年。网址 https://aaai.org/ojs/index.php/AAAI/article/view/6239。

A. Z. Broder. 关于文档的相似性与包含关系。收录于《序列的压缩与复杂性 1997》（目录编号 97TB100171）会议论文集，第 21–29 页。IEEE，1997。

M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. D. O. Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman 等人。评估在代码上训练的大型语言模型。ArXiv预印本，abs/2107.03374，2021。网址 https://arxiv.org/abs/2107.03374。

P. Clark、I. Cowhey、O. Etzioni、T. Khot、A. Sabharwal、C. Schoenick 和 O. Tafjord。你认为你已经解决了问答问题吗？试试 arc，AI2 推理挑战。ArXiv 预印本，abs/1803.05457，2018。网址 https://arxiv.org/abs/1803.05457。

K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano 等. 训练验证器以解决数学应用题。ArXiv 预印本，abs/2110.14168，2021。网址 https://arxiv.org/abs/2110.14168。

Y. N. Dauphin, A. Fan, M. Auli 和 D. Grangier。使用门控卷积网络进行语言建模。收录于 D. Precup 和 Y. W. Teh 编著的《第34届国际机器学习会议论文集》，ICML 2017，澳大利亚悉尼，新南威尔士，2017年8月6-11日，机器学习研究论文集第70卷，页码 933–941。PMLR，2017。网址 http://proceedings.mlr.press/v70/dauphin17a.html。

X. Du, Y. Yao, K. Ma, B. Wang, T. Zheng, K. Zhu, M. Liu, Y. Liang, X. Jin, Z. Wei, 等人。Supergpqa：在285个研究生学科中扩展大模型评估。ArXiv预印本，abs/2502.14739，2025。网址 https://arxiv.org/abs/2502.14739。

D. Dua, Y. Wang, P. Dasigi, G. Stanovsky, S. Singh 和 M. Gardner. DROP：一个需要对段落进行离散推理的阅读理解基准。在 J. Burstein、C. Doran 和 T. Solorio 编辑的《2019 年北美计算语言学协会会议论文集：人类语言技术》第一卷（长篇和短篇论文），第 2368–2378 页，明尼阿波利斯，明尼苏达州，2019 年。计算语言学协会。doi: 10.18653/v1/N19-1246。网址 https://aclanthology.org/N19-1246。

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 等。我们完成了 mmlu 吗？ArXiv 预印本，abs/2406.04127，2024。网址 https://arxiv.org/abs/2406.04127。

F. Gloeckle, B. Y. Idrissi, B. Rozière, D. Lopez-Paz 和 G. Synnaeve。通过多标记预测实现更好更快的大型语言模型。收录于2024年维也纳奥地利举行的第41届国际机器学习会议（ICML 2024），时间为2024年7月21日至27日。OpenReview.net，2024。网址 https://openreview.net/forum?id=pEWAcejiU2。

A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, et al. The llama 3 herd of models. ArXiv preprint, abs/2407.21783, 2024. URL https://arxiv.org/abs/2407.21783.

A. Gu, B. Rozière, H. J. Leather, A. Solar-Lezama, G. Synnaeve, and S. Wang. Cruxeval: A benchmark for code reasoning, understanding and execution. In Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024. OpenReview.net, 2024. URL https://openreview.net/forum?id=Ffpg52swvg.

D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. ArXiv preprint, abs/2501.12948, 2025. URL https://arxiv.org/abs/2501.12948.

J. He, J. Liu, C. Y. Liu, R. Yan, C. Wang, P. Cheng, X. Zhang, F. Zhang, J. Xu, W. Shen, S. Li, L. Zeng, T. Wei, C. Cheng, B. An, Y. Liu, and Y. Zhou. Skywork open reasoner series. https://capricious-hydrogen-41c.notion.site/Skywork-Open-Reaonser-Series-1d0bc9ae823a80459b46c149e4f51680, 2025. Notion Blog.

D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt. Measuring massive multitask language understanding. In 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net, 2021a. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

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, abs/2103.03874, 2021b. URL https://arxiv.org/abs/2103.03874.

C.-P. Hsieh, S. Sun, S. Kriman, S. Acharya, D. Rekesh, F. Jia, Y. Zhang, and B. Ginsburg. Ruler: What's the real context size of your long-context language models? ArXiv preprint, abs/2404.06654, 2024. URL https://arxiv.org/abs/2404.06654.

J. Hu, Y. Zhang, Q. Han, D. Jiang, X. Zhang, and H.-Y. Shum. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model. ArXiv preprint, abs/2503.24290, 2025. URL https://arxiv.org/abs/2503.24290.

Y. Huang, Y. Bai, Z. Zhu, J. Zhang, J. Zhang, T. Su, J. Liu, C. Lv, Y. Zhang, J. Lei, Y. Fu, M. Sun, and J. He. C-eval: A multi-level multi-discipline chinese evaluation suite for foundation models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/c6ec1844bec96d6d32ae95ae694e23d8-Abstract-Datasets_and_Benchmarks.html.

IOI. International olympiad in informatics, 2024. URL https://ioinformatics.org/.

N. Jain, K. Han, A. Gu, W.-D. Li, F. Yan, T. Zhang, S. Wang, A. Solar-Lezama, K. Sen, and I. Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. ArXiv preprint, abs/2403.07974, 2024. URL https://arxiv.org/abs/2403.07974.

M. Joshi, E. Choi, D. Weld, and L. Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In R. Barzilay and M.-Y. Kan, editors, Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1601–1611, Vancouver, Canada, 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1147. URL https://aclanthology.org/P17-1147.

A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan 等人。Llama 3 模型群。ArXiv 预印本，abs/2407.21783，2024。网址 https://arxiv.org/abs/2407.21783。

A. Gu, B. Rozière, H. J. Leather, A. Solar-Lezama, G. Synnaeve, 和 S. Wang. Cruxeval：一个用于代码推理、理解和执行的基准。在2024年维也纳奥地利举行的第41届国际机器学习会议（ICML 2024），2024年7月21-27日。OpenReview.net，2024。网址 https://openreview.net/forum?id=Ffpg52swvg。

D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi, 等. Deepseek-r1：通过强化学习激励大模型的推理能力。ArXiv预印本，abs/2501.12948，2025。网址 https://arxiv.org/abs/2501.12948。

J. He, J. Liu, C. Y. Liu, R. Yan, C. Wang, P. Cheng, X. Zhang, F. Zhang, J. Xu, W. Shen, S. Li, L. Zeng, T. Wei, C. Cheng, B. An, Y. Liu, 和 Y. Zhou。Skywork 开放推理器系列。https://capricious-hydrogen-41c.notion.site/Skywork-Open-Reaonser-Series-1d0bc9ae823a80459b46c149e4f51680，2025。Notion 博客。

D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song 和 J. Steinhardt. 测量大规模多任务语言理解。在第九届国际学习表征会议（ICLR 2021），虚拟会议，奥地利，2021年5月3-7日。OpenReview.net，2021a。网址 https://openreview.net/forum?id=d7KBjmI3GmQ。

D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang, D. Song, 和 J. Steinhardt. 使用数学数据集衡量数学问题解决能力。ArXiv预印本，abs/2103.03874，2021b。网址 https://arxiv.org/abs/2103.03874。

C.-P. Hsieh, S. Sun, S. Kriman, S. Acharya, D. Rekesh, F. Jia, Y. Zhang 和 B. Ginsburg. Ruler：你的长上下文语言模型的实际上下文大小是多少？ArXiv预印本，abs/2404.06654，2024。URL https://arxiv.org/abs/2404.06654。

J. Hu, Y. Zhang, Q. Han, D. Jiang, X. Zhang, 和 H.-Y. Shum. Open-reasoner-zero：一种在基础模型上扩展强化学习的开源方法。ArXiv预印本，abs/2503.24290，2025。网址 https://arxiv.org/abs/2503.24290。

Y. Huang, Y. Bai, Z. Zhu, J. Zhang, J. Zhang, T. Su, J. Liu, C. Lv, Y. Zhang, J. Lei, Y. Fu, M. Sun, 和 J. He. C-eval：一个多层次、多学科的中文基础模型评估套件。在 A. Oh、T. Naumann、A. Globerson、K. Saenko、M. Hardt 和 S. Levine 编辑的《神经信息处理系统进展36：2023年神经信息处理系统年度会议》，NeurIPS 2023，路易斯安那州新奥尔良，美国，2023年12月10日至16日，2023。网址 http://papers.nips.cc/paper_files/paper/2023/hash/c6ec1844bec96d6d32ae95ae694e23d8-Abstract-Datasets_and_Benchmarks.html。

IOI. 国际信息学奥林匹克，2024年。网址 https://ioinformatics.org/。

N. Jain, K. Han, A. Gu, W.-D. Li, F. Yan, T. Zhang, S. Wang, A. Solar-Lezama, K. Sen 和 I. Stoica。Livecodebench：对大型语言模型进行整体且无污染的代码评估。ArXiv预印本，abs/2403.07974，2024。网址 https://arxiv.org/abs/2403.07974。

M. Joshi, E. Choi, D. Weld 和 L. Zettlemoyer. TriviaQA：一个大规模远程监督的阅读理解挑战数据集。在 R. Barzilay 和 M.-Y. Kan 编者的《第55届计算语言学协会年会论文集（第1卷：长论文）》中，第 1601–1611 页，加拿大温哥华，2017年。计算语言学协会。doi：10.18653/v1/P17-1147。网址 https://aclanthology.org/P17-1147。

T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, J. Devlin, K. Lee, K. Toutanova, L. Jones, M. Kelcey, M.-W. Chang, A. M. Dai, J. Uszkoreit, Q. Le, and S. Petrov. Natural questions: A benchmark for question answering research. Transactions of the Association for Computational Linguistics, 7:452–466, 2019. doi: 10.1162/tacl_a_00276. URL https://aclanthology.org/Q19-1026.

W. Kwon, Z. Li, S. Zhuang, Y. Sheng, L. Zheng, C. H. Yu, J. Gonzalez, H. Zhang, and I. Stoica. Efficient memory management for large language model serving with pagedattention. In Proceedings of the 29th Symposium on Operating Systems Principles, pages 611–626, 2023.

G. Lai, Q. Xie, H. Liu, Y. Yang, and E. Hovy. RACE: Large-scale ReAding comprehension dataset from examinations. In M. Palmer, R. Hwa, and S. Riedel, editors, Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 785–794, Copenhagen, Denmark, 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1082. URL https://aclanthology.org/D17-1082.

Y. Leviathan, M. Kalman, and Y. Matias. Fast inference from transformers via speculative decoding. 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 19274–19286. PMLR, 2023. URL https://proceedings.mlr.press/v202/leviathan23a.html.

H. Li, Y. Zhang, F. Koto, Y. Yang, H. Zhao, Y. Gong, N. Duan, and T. Baldwin. Cmmlu: Measuring massive multitask language understanding in chinese. ArXiv preprint, abs/2306.09212, 2023. URL https://arxiv.org/abs/2306.09212.

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. In The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024. OpenReview.net, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

A. Liu, B. Feng, B. Xue, B. Wang, B. Wu, C. Lu, C. Zhao, C. Deng, C. Zhang, C. Ruan, et al. Deepseek-v3 technical report. ArXiv preprint, abs/2412.19437, 2024a. URL https://arxiv.org/abs/2412.19437.

J. Liu, C. S. Xia, Y. Wang, and L. Zhang. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/43e9d647ccd3e4b7b5baab53f0368686-Abstract-Conference.html.

J. Liu, D. Zhu, Z. Bai, Y. He, H. Liao, H. Que, Z. Wang, C. Zhang, G. Zhang, J. Zhang, et al. A comprehensive survey on long context language modeling. ArXiv preprint, abs/2503.17407, 2025. URL https://arxiv.org/abs/2503.17407.

Y. Liu, R. Jin, L. Shi, Z. Yao, and D. Xiong. Finemath: A fine-grained mathematical evaluation benchmark for chinese large language models. ArXiv preprint, abs/2403.07747, 2024b. URL https://arxiv.org/abs/2403.07747.

I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. OpenReview.net, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, J. Devlin, K. Lee, K. Toutanova, L. Jones, M. Kelcey, M.-W. Chang, A. M. Dai, J. Uszkoreit, Q. Le, 和 S. Petrov. 自然问题：一个用于问答研究的基准。计算语言学协会交易，7:452–466，2019年。doi: 10.1162/tacl_a_00276。网址 https://aclanthology.org/Q19-1026。

W. Kwon, Z. Li, S. Zhuang, Y. Sheng, L. Zheng, C. H. Yu, J. Gonzalez, H. Zhang, 和 I. Stoica. 用于大规模语言模型服务的高效内存管理与分页注意机制。在第29届操作系统原理研讨会论文集，页码 611–626，2023年。

G. Lai, Q. Xie, H. Liu, Y. Yang, 和 E. Hovy. RACE：来自考试的大规模阅读理解数据集。收录于 M. Palmer、R. Hwa 和 S. Riedel 编著的《2017 年自然语言处理实证方法会议论文集》，第 785–794 页，丹麦哥本哈根，2017 年。计算语言学协会。doi：10.18653/v1/D17-1082。网址 https://aclanthology.org/D17-1082。

Y. Leviathan, M. Kalman 和 Y. Matias。通过推测解码实现变换器的快速推理。在 A. Krause、E. Brunskill、K. Cho、B. Engelhardt、S. Sabato 和 J. Scarlett 编著的《国际机器学习会议》（ICML 2023），2023 年 7 月 23-29 日，夏威夷檀香山，美国，机器学习研究论文集第 202 卷，第 19274–19286 页。PMLR，2023。网址 https://proceedings.mlr.press/v202/leviathan23a.html。

H. Li, Y. Zhang, F. Koto, Y. Yang, H. Zhao, Y. Gong, N. Duan, 和 T. Baldwin. Cmmlu：衡量中文大规模多任务语言理解。ArXiv预印本，abs/2306.09212，2023。网址 https://arxiv.org/abs/2306.09212。

H. Lightman, V. Kosaraju, Y. Burda, H. Edwards, B. Baker, T. Lee, J. Leike, J. Schulman, I. Sutskever 和 K. Cobbe。让我们一步步验证。在第十二届国际学习表征会议（ICLR 2024），奥地利维也纳，2024年5月7-11日。OpenReview.net，2024。网址 https://openreview.net/forum?id=v8L0pN6EOi。

A. Liu, B. Feng, B. Xue, B. Wang, B. Wu, C. Lu, C. Zhao, C. Deng, C. Zhang, C. Ruan 等。Deepseek-v3 技术报告。ArXiv 预印本，abs/2412.19437，2024a。网址 https://arxiv.org/abs/2412.19437。

J. Liu, C. S. Xia, Y. Wang, 和 L. Zhang。你的 chatgpt 生成的代码真的正确吗？对大型语言模型在代码生成方面的严格评估。在 A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt 和 S. Levine 编著，《神经信息处理系统进展36：2023 年神经信息处理系统年度会议》，NeurIPS 2023，地点：新奥尔良，路易斯安那州，美国，2023年12月10日至16日，2023。网址 http://papers.nips.cc/paper_files/paper/2023/hash/43e9d647ccd3e4b7b5baab53f0368686-Abstract-Conference.html。

J. Liu, D. Zhu, Z. Bai, Y. He, H. Liao, H. Que, Z. Wang, C. Zhang, G. Zhang, J. Zhang 等人。关于长上下文语言建模的全面综述。ArXiv 预印本，abs/2503.17407，2025。网址 https://arxiv.org/abs/2503.17407。

Y. Liu, R. Jin, L. Shi, Z. Yao, 和 D. Xiong. Finemath：一个面向中文大型语言模型的细粒度数学评估基准。ArXiv预印本，abs/2403.07747，2024b。网址 https://arxiv.org/abs/2403.07747。

I. Loshchilov 和 F. Hutter。解耦的权重衰减正则化。在第七届国际学习表示会议（ICLR 2019），2019年5月6-9日，新奥尔良，路易斯安那州，美国。OpenReview.net，2019。网址 https://openreview.net/forum?id=Bkg6RiCqY7。

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

MAA. American invitational mathematics examination - aime. In American Invitational Mathematics Examination - AIME, 2025. URL https://maa.org/math-competitions/american-invitational-mathematics-examination-aime.

P. Moritz, R. Nishihara, S. Wang, A. Tumanov, R. Liaw, E. Liang, M. Elibol, Z. Yang, W. Paul, M. I. Jordan, et al. Ray: A distributed framework for emerging {AI} applications. In 13th USENIX symposium on operating systems design and implementation (OSDI 18), pages 561–577, 2018.

OpenAI. Learning to reason with llms, 2024. URL https://openai.com/index/learning-to-reason-with-llms/.

K. Paster, M. D. Santos, Z. Azerbayev, and J. Ba. Openwebmath: An open dataset of high-quality mathematical web text. In The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024. OpenReview.net, 2024. URL https://openreview.net/forum?id=jKHmjlpViu.

G. Penedo, Q. Malartic, D. Hesslow, R. Cojocaru, A. Cappelli, H. Alobeidli, B. Pannier, E. Almazrouei, and J. Launay. The refinedweb dataset for falcon llm: outperforming curated corpora with web data, and web data only. ArXiv preprint, abs/2306.01116, 2023. URL https://arxiv.org/abs/2306.01116.

G. Penedo, H. Kydlícek, L. B. Allal, A. Lozhkov, M. Mitchell, C. A. Raffel, L. von Werra, and T. Wolf. The fineweb datasets: Decanting the web for the finest text data at scale. In A. Globersons, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. M. Tomczak, and C. Zhang, editors, Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/370df50ccfdf8bde18f8f9c2d9151bda-Abstract-Datasets_and_Benchmarks_Track.html.

A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, et al. Improving language understanding by generative pre-training. OpenAI, 2018.

D. Rein, B. L. Hou, A. C. Stickland, J. Petty, R. Y. Pang, J. Dirani, J. Michael, and S. R. Bowman. Gpqa: A graduate-level google-proof q&a benchmark. In First Conference on Language Modeling, 2024.

K. Sakaguchi, R. L. Bras, C. Bhagavatula, and Y. Choi. Winogrande: An adversarial winograd schema challenge at scale. In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, pages 8732–8740. AAAI Press, 2020. URL https://aaai.org/ojs/index.php/AAAI/article/view/6399.

B. Seed, Y. Yuan, Y. Yue, M. Wang, X. Zuo, J. Chen, L. Yan, W. Xu, C. Zhang, X. Liu, et al. Seed-thinking-v1. 5: Advancing superb reasoning models with reinforcement learning. ArXiv preprint, abs/2504.13914, 2025. URL https://arxiv.org/abs/2504.13914.

Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, X. Bi, H. Zhang, M. Zhang, Y. Li, Y. Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. ArXiv preprint, abs/2402.03300, 2024. URL https://arxiv.org/abs/2402.03300.

MAA。美国邀请数学考试 - AIME。在2024年的美国邀请数学考试 - AIME中。网址 https://maa.org/math-competitions/american-invitational-mathematics-examination-aime。

MAA。美国邀请数学考试 - AIME。在2025年的美国邀请数学考试 - AIME中。网址 https://maa.org/math-competitions/american-invitational-mathematics-examination-aime。

P. Moritz, R. Nishihara, S. Wang, A. Tumanov, R. Liaw, E. Liang, M. Elibol, Z. Yang, W. Paul, M. I. Jordan 等. Ray：一个用于新兴 {AI} 应用的分布式框架。收录于第 13 届 USENIX 操作系统设计与实现研讨会（OSDI 18），第 561–577 页，2018 年。

OpenAI. 使用LLMs进行推理学习，2024。网址 https://openai.com/index/learning-to-reason-with-llms/。K. Paster, M. D. Santos, Z. Azerbayev 和 J. Ba。Openwebmath：一个高质量数学网页文本的开源数据集。在第十二届学习表示国际会议（ICLR 2024），奥地利维也纳，2024年5月7-11日。OpenReview.net，2024。网址 https://openreview.net/forum?id=jKHmjlpViu。G. Penedo, Q. Malartic, D. Hesslow, R. Cojocaru, A. Cappelli, H. Alobeidli, B. Pannier, E. Almazrouei 和 J. Launay。Falcon LLM的refinedweb数据集：用网页数据超越策划语料库，仅用网页数据。ArXiv预印本，abs/2306.01116，2023。网址 https://arxiv.org/abs/2306.01116。G. Penedo, H. Kydlícek, L. B. Allal, A. Lozhkov, M. Mitchell, C. A. Raffel, L. von Werra 和 T. Wolf。Fineweb数据集：在大规模中提取网页以获得最优文本数据。在A. Globersons, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. M. Tomczak 和 C. Zhang 编辑的《神经信息处理系统进展38：2024年神经信息处理系统年度会议（NeurIPS 2024）》，加拿大温哥华，2024年12月10-15日。网址 http://papers.nips.cc/paper_files/paper/2024/hash/370df50ccfdf8bde18f8f9c2d9151bda-Abstract-Datasets_and_Benchmarks_Track.html。A. Radford, K. Narasimhan, T. Salimans, I. Sutskever 等。通过生成式预训练提升语言理解能力。OpenAI，2018。D. Rein, B. L. Hou, A. C. Stickland, J. Petty, R. Y. Pang, J. Dirani, J. Michael 和 S. R. Bowman。Gpqa：一个面向研究生的谷歌级问答基准测试。在2024年第一届语言建模会议。K. Sakaguchi, R. L. Bras, C. Bhagavatula 和 Y. Choi。Winogrande：大规模的对抗性Winograd模式挑战。在第34届人工智能协会会议（AAAI 2020）、第32届人工智能创新应用会议（IAAI 2020）、第10届人工智能教育进展研讨会（EAAI 2020），美国纽约，2020年2月7-12日，第8732-8740页。AAAI出版社，2020。网址 https://aaai.org/ojs/index.php/AAAI/article/view/6399。B. Seed, Y. Yuan, Y. Yue, M. Wang, X. Zuo, J. Chen, L. Yan, W. Xu, C. Zhang, X. Liu 等。Seed-thinking-v1.5：通过强化学习推动卓越推理模型。ArXiv预印本，abs/2504.13914，2025。网址 https://arxiv.org/abs/2504.13914。Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, X. Bi, H. Zhang, M. Zhang, Y. Li, Y. Wu 等。Deepseek-math：推动开放语言模型中数学推理的极限。ArXiv预印本，abs/2402.03300，2024。网址 https://arxiv.org/abs/2402.03300。

G. Sheng, C. Zhang, Z. Ye, X. Wu, W. Zhang, R. Zhang, Y. Peng, H. Lin, and C. Wu. Hybridflow: A flexible and efficient rlhf framework. ArXiv preprint, abs/2409.19256, 2024. URL https://arxiv.org/abs/2409.19256.

J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu. Roformer: Enhanced transformer with rotary position embedding. Neurocomputing, 568:127063, 2024.

M. Suzgun, N. Scales, N. Schärli, S. Gehrmann, Y. Tay, H. W. Chung, A. Chowdhery, Q. Le, E. Chi, D. Zhou, and J. Wei. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In A. Rogers, J. Boyd-Graber, and N. Okazaki, editors, Findings of the Association for Computational Linguistics: ACL 2023, pages 13003–13051, Toronto, Canada, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.824. URL https://aclanthology.org/2023.findings-acl.824.

G. Team. Gemma 2: Improving open language models at a practical size, 2024. URL https://arxiv.org/abs/2408.00118.

K. Team, A. Du, B. Gao, B. Xing, C. Jiang, C. Chen, C. Li, C. Xiao, C. Du, C. Liao, et al. Kimi k1. 5: Scaling reinforcement learning with llms. ArXiv preprint, abs/2501.12599, 2025. URL https://arxiv.org/abs/2501.12599.

H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. ArXiv preprint, abs/2307.09288, 2023. URL https://arxiv.org/abs/2307.09288.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, editors, Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, pages 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

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: A more robust and challenging multi-task language understanding benchmark. In A. Globersons, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. M. Tomczak, and C. Zhang, editors, Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/ad236edc564f3e3156e1b2feafb99a24-Abstract-Datasets_and_Benchmarks_Track.html.

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, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.257. URL https://aclanthology.org/2023.findings-emnlp.257.

A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, et al. Qwen2. 5 technical report. ArXiv preprint, abs/2412.15115, 2024. URL https://arxiv.org/abs/2412.15115.

Q. Yu, Z. Zhang, R. Zhu, Y. Yuan, X. Zuo, Y. Yue, T. Fan, G. Liu, L. Liu, X. Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. ArXiv preprint, abs/2503.14476, 2025. URL https://arxiv.org/abs/2503.14476.

G. Sheng, C. Zhang, Z. Ye, X. Wu, W. Zhang, R. Zhang, Y. Peng, H. Lin, 和 C. Wu. Hybridflow：一种灵活高效的RLHF框架。ArXiv预印本，abs/2409.19256，2024。网址 https://arxiv.org/abs/2409.19256.

J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, 和 Y. Liu. Roformer：带有旋转位置嵌入的增强型变换器。神经计算，568：127063，2024。

M. Suzgun, N. Scales, N. Schärli, S. Gehrmann, Y. Tay, H. W. Chung, A. Chowdhery, Q. Le, E. Chi, D. Zhou 和 J. Wei. 挑战性 BIG-bench 任务以及链式思考是否能解决它们。在 A. Rogers、J. Boyd-Graber 和 N. Okazaki 编者的《计算语言学协会发现：ACL 2023》一书中，第 13003–13051 页，多伦多，加拿大，2023 年。计算语言学协会。doi: 10.18653/v1/2023.f indings-acl.824。网址 https://aclanthology.org/2023.findings-acl.824。

G. 团队。Gemma 2：在实际规模下改进开放式语言模型，2024。网址 https://arxiv.org/abs/2408.00118。

K. 团队，A. 杜，B. 高，B. 邢，C. 江，C. 陈，C. 李，C. 萧，C. 杜，C. 廖，等。Kimi k1. 5：使用大规模语言模型扩展强化学习。ArXiv预印本，abs/2501.12599，2025。网址 https://arxiv.org/abs/2501.12599。

H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale 等. Llama 2：开源基础模型和微调聊天模型。ArXiv 预印本，abs/2307.09288，2023。网址 https://arxiv.org/abs/2307.09288。

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, 和 I. Polosukhin。注意力机制就是你所需要的一切。在 I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan 和 R. Garnett 编者的《神经信息处理系统进展30：2017年神经信息处理系统年度会议》，2017年12月4-9日，加利福尼亚州长滩，美国，页码 5998–6008，2017。网址 https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html。

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, 和 W. Chen. Mmlu-pro：一个更稳健、更具挑战性的多任务语言理解基准。在 A. Globersons、L. Mackey、D. Belgrave、A. Fan、U. Paquet、J. M. Tomczak 和 C. Zhang 编者，神经信息处理系统进展 38：2024 年神经信息处理系统年度会议，NeurIPS 2024，加拿大不列颠哥伦比亚省温哥华，2024 年 12 月 10-15 日，2024。网址 http://papers.nips.cc/paper_files/paper/2024/hash/ad236edc564f3e3156e1b2feafb99a24-Abstract-Datasets_and_Benchmarks_Track.html。

H. Xia, T. Ge, P. Wang, S.-Q. Chen, F. Wei, 和 Z. Sui. 投机解码：利用投机执行加速 seq2seq 生成。在 H. Bouamor、J. Pino 和 K. Bali 编著，《计算语言学协会会议论文集：EMNLP 2023》，第 3909–3925 页，新加坡，2023年。计算语言学协会。doi: 10.18653/v1/2023.findings-emnlp.257。网址 https://aclanthology.org/2023.findings-emnlp.257。

A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei 等. Qwen2. 5 技术报告. ArXiv 预印本, abs/2412.15115, 2024. URL https://arxiv.org/abs/ 2412.15115.

Q. Yu, Z. Zhang, R. Zhu, Y. Yuan, X. Zuo, Y. Yue, T. Fan, G. Liu, L. Liu, X. Liu, 等. Dapo：一个大规模开源的LLM强化学习系统。ArXiv预印本，abs/2503.14476，2025。网址 https://arxiv.org/abs/2503.14476。

Y. Yue, Z. Chen, R. Lu, A. Zhao, Z. Wang, Y. Yue, S. Song, and G. Huang. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model?, 2025. URL https://arxiv.org/abs/2504.13837.

R. Zellers, A. Holtzman, Y. Bisk, A. Farhadi, and Y. Choi. HellaSwag: Can a machine really finish your sentence? In A. Korhonen, D. Traum, and L. Màrquez, editors, Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 4791–4800, Florence, Italy, 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472.

B. Zhang and R. Sennrich. Root mean square layer normalization. In H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, editors, Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, pages 12360–12371, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/1e8a19426224ca89e83cef47f1e7f53b-Abstract.html.

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. In K. Duh, H. Gomez, and S. Bethard, editors, Findings of the Association for Computational Linguistics: NAACL 2024, pages 2299–2314, Mexico City, Mexico, 2024a. Association for Computational Linguistics. URL https://aclanthology.org/2024.findings-naacl.149.

Y. Zhong, Z. Zhang, B. Wu, S. Liu, Y. Chen, C. Wan, H. Hu, L. Xia, R. Ming, Y. Zhu, et al. Rlhfuse: Efficient rlhf training for large language models with inter-and intra-stage fusion. ArXiv preprint, abs/2409.13221, 2024b. URL https://arxiv.org/abs/2409.13221.

F. Zhou, Z. Wang, N. Ranjan, Z. Cheng, L. Tang, G. He, Z. Liu, and E. P. Xing. Megamath: Pushing the limits of open math corpora. ArXiv preprint, abs/2504.02807, 2025. URL https://arxiv.org/abs/2504.02807.

J. Zhou, T. Lu, S. Mishra, S. Brahma, S. Basu, Y. Luan, D. Zhou, and L. Hou. Instruction-following evaluation for large language models, 2023. URL https://arxiv.org/abs/2311.07911.

Q. Zhu, D. Guo, Z. Shao, D. Yang, P. Wang, R. Xu, Y. Wu, Y. Li, H. Gao, S. Ma, et al. Deepseek-coder-v2: Breaking the barrier of closed-source models in code intelligence. ArXiv preprint, abs/2406.11931, 2024. URL https://arxiv.org/abs/2406.11931.

Y. Yue, Z. Chen, R. Lu, A. Zhao, Z. Wang, Y. Yue, S. Song, 和 G. Huang. 强化学习是否真正激励了大规模语言模型（llms）超越基础模型的推理能力？, 2025. URL https://arxiv.org/abs/2504.13837.

R. Zellers, A. Holtzman, Y. Bisk, A. Farhadi 和 Y. Choi. HellaSwag：机器真的能完成你的句子吗？收录于 A. Korhonen、D. Traum 和 L. Màrquez 编著的《第57届计算语言学协会年会论文集》，第4791–4800 页，意大利佛罗伦萨，2019 年。计算语言学协会。doi: 10.18653/v1/P19-1472。网址 https://aclanthology.org/P19-1472。

B. Zhang 和 R. Sennrich. 均方根层归一化。在 H. M. Wallach、H. Larochelle、A. Beygelzimer、F. d’ Alché-Buc、E. B. Fox 和 R. Garnett 编著的《神经信息处理系统进展 32：2019 年神经信息处理系统年度会议（NeurIPS 2019）》，2019 年 12 月 8-14 日，加拿大不列颠哥伦比亚省温哥华，页码 12360–12371，2019。网址 https://proceedings.neurips.cc/paper/2019/hash/1e8a19426224ca89e83cef47f1e7f53b-Abstract.html。

W. Zhong, R. Cui, Y. Guo, Y. Liang, S. Lu, Y. Wang, A. Saied, W. Chen, 和 N. Duan. AGIEval：一个以人为本的基础模型评估基准。在 K. Duh、H. Gomez 和 S. Bethard 编辑的《计算语言学协会会议成果：NAACL 2024》，第 2299–2314 页，墨西哥城，墨西哥，2024a。计算语言学协会。网址 https://aclanthology.org/2024.findings-naacl.149。

Y. Zhong, Z. Zhang, B. Wu, S. Liu, Y. Chen, C. Wan, H. Hu, L. Xia, R. Ming, Y. Zhu, 等. Rlhfuse：用于大型语言模型的高效RLHF训练，结合阶段间和阶段内融合。ArXiv预印本，abs/2409.13221，2024b。网址 https://arxiv.org/abs/2409.13221。

F. Zhou, Z. Wang, N. Ranjan, Z. Cheng, L. Tang, G. He, Z. Liu, 和 E. P. Xing. Megamath：推动开放数学语料库的极限。ArXiv预印本，abs/2504.02807，2025。网址 https://arxiv.org/abs/2504.02807。

J. Zhou, T. Lu, S. Mishra, S. Brahma, S. Basu, Y. Luan, D. Zhou, 和 L. Hou. 大型语言模型的指令遵循评估，2023。网址 https://arxiv.org/abs/2311.07911。

Q. Zhu, D. Guo, Z. Shao, D. Yang, P. Wang, R. Xu, Y. Wu, Y. Li, H. Gao, S. Ma, 等. Deepseek-coder-v2：打破封闭源模型在代码智能中的壁垒。ArXiv预印本，abs/2406.11931，2024。网址 https://arxiv.org/abs/2406.11931。

A Contributions and Acknowledgments

We would like to express our sincere gratitude to all contributors, including those not listed in the paper, for their invaluable support and efforts. Authors within each role are listed alphabetically by their first name.

Core Contributors
Bingquan Xia
Bowen Shen
Cici
Dawei Zhu
Di Zhang
Gang Wang
Hailin Zhang
Huaqiu Liu
Jiebao Xiao
Jinhao Dong
Liang Zhao
Peidian Li
Peng Wang
Shihua Yu
Shimao Chen
Weikun Wang
Wenhan Ma
Xiangwei Deng
Yi Huang
Yifan Song
Zihan Jiang

Contributors
Bowen Ye
Can Cai
Chenhong He
Dong Zhang
Duo Zhang
Guoan Wang
Hao Tian
Haochen Zhao
Heng Qu
Hongshen Xu
Jun Shi
Kainan Bao
Kai Fang
Kang Zhou
Kangyang Zhou
Lei Li
Menghang Zhu
Nuo Chen
Qiantong Wang
Shaohui Liu
Shicheng Li
Shuhao Gu
Shuhuai Ren
Shuo Liu
Sirui Deng
Weiji Zhuang
Weiwei Lv
Wenyu Yang
Xin Zhang
Xing Yong
Xing Zhang
Xingchen Song
Xinzhe Xu
Xu Wang
Yihan Yan
Yu Tu
Yuanyuan Tian
Yudong Wang
Yue Yu
Zhenru Lin
Zhichao Song
Zihao Yue

A 贡献与致谢

我们衷心感谢所有贡献者，包括未在论文中列出的人员，感谢他们宝贵的支持和努力。每个角色中的作者按名字的字母顺序排列。

核心贡献者：夏冰泉、沈博文、谢茨、朱大伟、张迪、王刚、张海林、刘华秋、肖杰宝、董金浩、赵亮、李佩甸、王鹏、于世华、陈诗茂、王维坤、王文汉、马翔伟、邓毅、黄一凡、宋子涵、江志涵

许宏深 施俊 赛康南 宝开 方康周康阳 周磊 李梦航 朱诺 陈倩彤 王少辉 刘世成 李书豪 顾书怀 任硕 邵伟 武炜 吴文宇 杨欣张兴 永兴 张星辰 宋新哲 许旭王怡涵 严瑜 图媛媛 田玉东 王跃 玉 珍如 林志超 宋子豪 岳

贡献者
鲍文 叶灿 彩
陈宏 何东 张
多 张国安 王
浩 天浩辰 赵
恒 曲