AINLP
我爱自然语言处理
一个有趣有AI的自然语言处理社区
长按扫码关注我们

MiniMax-M1: Scaling Test-Time Compute Efficiently with Lightning Attention

MiniMax[1]

We introduce MiniMax-M1, the world's first open-weight, large-scale hybrid-attention reasoning model. MiniMax-M1 is powered by a hybrid Mixture-of-Experts (MoE) architecture combined with a lightning attention mechanism. The model is developed based on our previous MiniMax-Text-01 model (MiniMax et al., 2025), which contains a total of 456 billion parameters with 45.9 billion parameters activated per token. The M1 model natively supports a context length of 1 million tokens, 8x the context size of DeepSeek R1. Furthermore, the lightning attention mechanism in MiniMax-M1 enables efficient scaling of test-time compute – For example, compared to DeepSeek R1, M1 consumes 25% of the FLOPs at a generation length of 100K tokens. These properties make M1 particularly suitable for complex tasks that require processing long inputs and thinking extensively. MiniMax-M1 is trained using large-scale reinforcement learning (RL) on diverse problems ranging from traditional mathematical reasoning to sandbox-based, real-world software engineering environments. In addition to the inherent efficiency advantage of lightning attention for RL training, we propose CISPO, a novel RL algorithm to further enhance RL efficiency. CISPO clips importance sampling weights rather than token updates, outperforming other competitive RL variants. Combining hybrid-attention and CISPO enables MiniMax-M1's full RL training on 512 H800 GPUs to complete in only three weeks, with a rental cost of just $534,700. We release two versions of MiniMax-M1 models with 40K and 80K thinking budgets respectively, where the 40K model represents an intermediate phase of the 80K training. Experiments on standard benchmarks show that our models are comparable or superior to strong open-weight models such as the original DeepSeek-R1 and Qwen3-235B, with particular strengths in complex software engineering, tool utilization, and long-context tasks. Through efficient scaling of test-time compute, MiniMax-M1 serves as a strong foundation for next-generation language model agents to reason and tackle real-world challenges. We publicly release MiniMax-M1 at https://github.com/MiniMax-AI/MiniMax-M1.

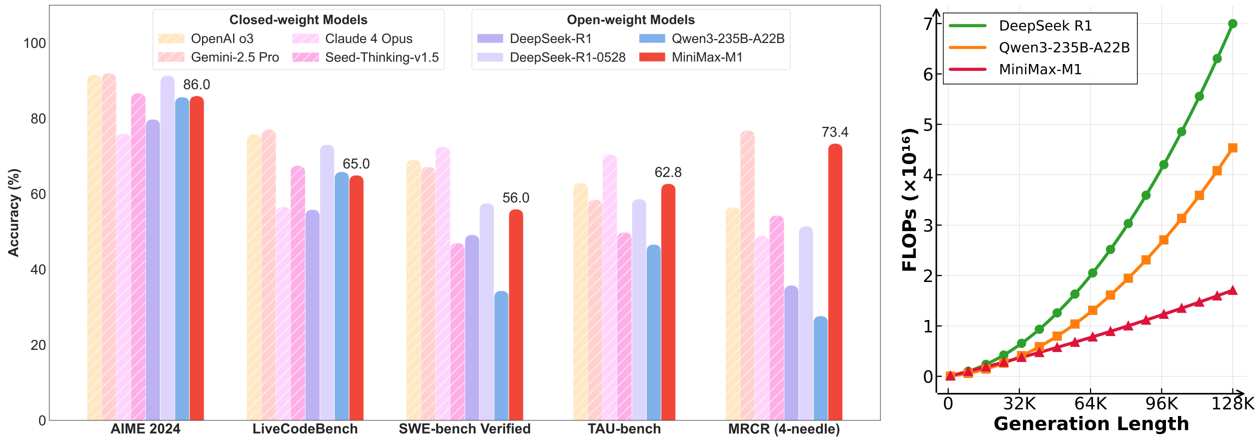

Figure 1 | **Left**: Benchmark performance comparison of leading commercial and open-weight models across competition-level mathematics, coding, software engineering, agentic tool use, and long-context understanding tasks. We use the MiniMax-M1-80k model here for MiniMax-M1. **Right**: Theoretical inference FLOPs scaling with generation length (# tokens).

[1] Please send correspondence to model@minimax.io.

MiniMax-M1：使用闪电注意力高效扩展测试时的计算

MiniMax[1]

我们介绍MiniMax-M1，世界上首个开源权重、大规模混合注意力推理模型。MiniMax-M1由混合专家（MoE）架构结合闪电注意力机制驱动。该模型基于我们之前的MiniMax-Text-01模型（MiniMax 等，2025年）开发，包含总共4560亿参数，每个标记激活45.9亿参数。M1模型原生支持100万令牌的上下文长度，是DeepSeek R1的8倍。此外，MiniMax-M1中的闪电注意力机制实现了测试时计算的高效扩展——例如，与DeepSeek R1相比，在生成长度为{v*}的情况下，M1的FLOPs仅占其25%。这些特性使得M1特别适合处理需要长输入和深入思考的复杂任务。MiniMax-M1采用大规模强化学习（RL）在多样化问题上进行训练，涵盖从传统数学推理到沙箱式的真实软件工程环境。除了闪电注意力在RL训练中的固有效率优势外，我们还提出了CISPO，一种新颖的RL算法，进一步提升RL效率。CISPO通过剪裁重要性采样权重而非令牌更新，优于其他竞争的RL变体。结合混合注意力和CISPO，使MiniMax-M1在512个H800 GPU上完成全部RL训练仅需三周，租赁成本仅为534,700美元。我们发布了两个版本的MiniMax-M1模型，分别具有40K和80K的思考预算，其中40K模型代表80K训练的中间阶段。在标准基准测试中，我们的模型表现与或优于强大的开源模型，如原始的DeepSeek-R1和Qwen3-235B，在复杂软件工程、工具利用和长上下文任务中表现尤为出色。通过高效扩展测试时的计算，MiniMax-M1为下一代语言模型代理的推理和应对现实挑战提供了坚实基础。我们在https://github.com/MiniMax-AI/MiniMax-M1公开发布MiniMax-M1。

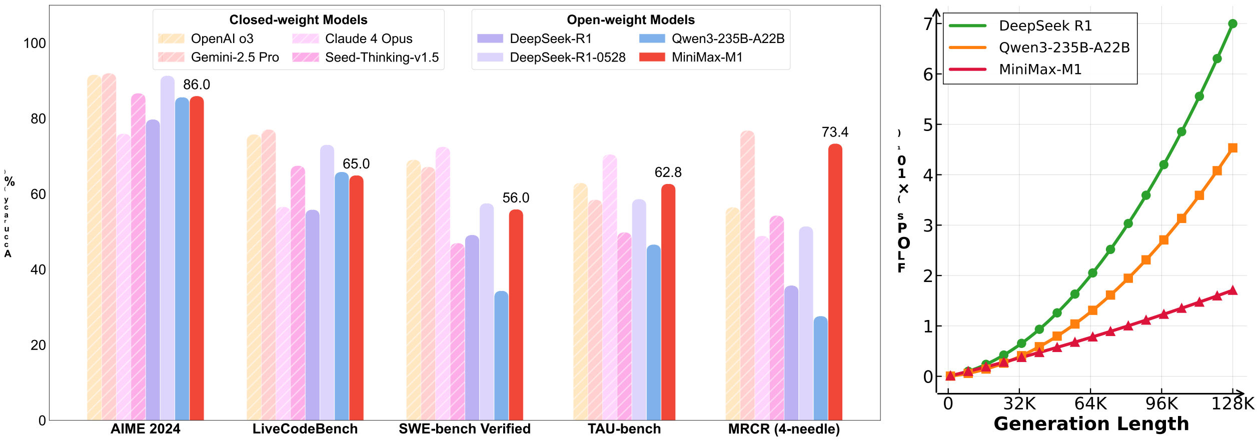

图1 | 左：在竞赛级数学、编码、软件工程、代理工具使用和长上下文理解任务中，领先的商业模型和开源模型的基准性能比较。这里我们使用MiniMax-M1的MiniMax-M1-80k模型。右：生成长度（#令牌）与理论推理FLOPs的扩展关系。

[1] Please send correspondence to model@minimax.io.

1. Introduction

Large reasoning models (LRMs), such as OpenAI o1 (OpenAI, 2024a) and DeepSeek-R1 (DeepSeek-AI et al., 2025), have demonstrated remarkable success by extending the length of reasoning through large-scale reinforcement learning (RL). In recent months, both the open-source community and commercial organizations have followed this trend, achieving significant advances on complex tasks such as Olympiad mathematics competitions and competitive programming (Anthropic, 2025; Google DeepMind, 2025; Hu et al., 2025; Kimi Team, 2025; Seed et al., 2025; Yu et al., 2025; Zeng et al., 2025). The success of LRMs has been primarily attributed to a new scaling dimension of test-time compute—As more FLOPs are dedicated to extended reasoning processes during generation, model performance shows consistent improvement, particularly for complex real-world applications (Jimenez et al., 2024; OpenAI, 2025).

However, continuously extending the reasoning process is challenging within the traditional transformer architecture (Vaswani et al., 2017), due to the inherent quadratic computational complexity of the softmax attention mechanism. While previous works have proposed various techniques to mitigate this issue—such as sparse attention (Beltagy et al., 2020; Lu et al., 2025; Yuan et al., 2025; Zaheer et al., 2020), linear attention (Arora et al., 2024; Choromanski et al., 2021; Du et al., 2025; He et al., 2024; Katharopoulos et al., 2020; Peng et al., 2024b, 2021; Qin et al., 2021, 2022a,b, 2024a,c; Shen et al., 2024; Sun et al., 2025, 2023; Zhang et al., 2024), linear attention with delta decay (Peng et al., 2025; Yang et al., 2024a,b), state space models (Dao and Gu, 2024; Glorioso et al., 2024; Gu and Dao, 2024; Gu et al., 2020, 2022, 2023; Gupta et al., 2022; Jamba Team, 2024; Ren et al., 2024), and linear RNNs (Behrouz et al., 2024; Chou et al., 2024; Chung and Ç, 2014; Hochreiter and Schmidhuber, 1997; Martin and Cundy, 2018; Peng et al., 2023, 2024a; Qin et al., 2023, 2024d; Siems et al., 2025; Sun et al., 2024; von Oswald et al., 2025)—these approaches have not been fully validated in large-scale reasoning models, and nearly all competitive LRMs to date still rely on traditional attention designs. An exception is the Hunyuan-T1 model (Tencent AI Lab, 2025) that employs the Mamba architecture (Dao and Gu, 2024; Gu and Dao, 2024). However, this model is not open-sourced and few details are disclosed. In this work, we aim to build and open-source a large reasoning model that can efficiently scale up test-time compute and compete with the state-of-the-art reasoning models.

We introduce MiniMax-M1, a reasoning model with a hybrid Mixture-of-Experts (MoE) architecture and Lightning Attention (Qin et al., 2024b), an I/O-aware implementation of a linear attention variant (Qin et al., 2022a). MiniMax-M1 is developed based on our previous MiniMax-Text-01 (MiniMax et al., 2025) model, and comprises 456 billion parameters in total, with 45.9 billion activations and 32 experts. In our attention design, a transformer block with softmax attention follows every seven transnormer blocks (Qin et al., 2022a) with lightning attention. This design theoretically enables efficient scaling of reasoning lengths to hundreds of thousands of tokens, as illustrated in Figure 1 (Right). For example, compared to DeepSeek R1, M1 consumes less than 50% of the FLOPs at a generation length of 64K tokens, and approximately 25% of the FLOPs at a length of 100K tokens. This substantial reduction in computational cost makes M1 significantly more efficient during both inference and large-scale RL training. Furthermore, owing to its lightning attention mechanism and in line with MiniMax-Text-01, our M1 model natively supports a context length of up to 1 million tokens – eight times the context size of DeepSeek R1 and an order of magnitude greater than all open-weight LRMs available to date. These features make M1 particularly well-suited for addressing complex, real-world tasks that require processing long inputs and generating extended thinking. A comparison of the maximum input and output lengths of M1 and other leading models is demonstrated in Table 1.

To develop our M1 model, we first continue pretraining MiniMax-Text-01 on 7.5T tokens from a carefully curated, reasoning-intensive corpus. Subsequently, we perform supervised fine-tuning (SFT)

1. 引言

大型推理模型（LRMs），如OpenAI o1（OpenAI，2024a）和DeepSeek-R1（DeepSeek-AI等，2025），通过扩展推理长度并结合大规模强化学习（RL），展现出了显著的成功。近年来，开源社区和商业组织都紧跟这一趋势，在奥林匹克数学竞赛和竞赛编程等复杂任务上取得了重大突破（Anthropic，2025；Google DeepMind，2025；Hu等，2025；Kimi团队，2025；Seed等，2025；Yu等，2025；Zeng等，2025）。LRMs的成功主要归因于测试时计算的新维度——随着在生成过程中投入的FLOPs增加，用于扩展推理的计算量也在不断增加，模型性能表现出持续的提升，尤其是在复杂的实际应用中（Jimenez等，2024；OpenAI，2025）。

然而，由于softmax注意力机制固有的二次计算复杂度，在传统的transformer架构（Vaswani等人，2017）中，持续扩展推理过程具有挑战性。虽然之前的工作提出了多种技术来缓解这一问题——例如稀疏注意力（Beltagy等人，2020；Lu等人，2025；Yuan等人，2025；Zaheer等人，2020）、线性注意力（Arora等人，2024；Choromanski等人，2021；Du等人，2025；He等人，2024；Katharopoulos等人，2020；Peng等人，2024b，2021；Qin等人，2021，2022a,b，2024a,c；Shen等人，2024；Sun等人，2025，2023；Zhang等人，2024）、带有delta衰减的线性注意力（Peng等人，2025；Yang等人，2024a,b）、状态空间模型（Dao和Gu，2024；Glorioso等人，2024；Gu和Dao，2024；Gu等人，2020，2022，2023；Gupta等人，2022；Jamba团队，2024；Ren等人，2024）、以及线性RNN（Behrouz等人，2024；Chou等人，2024；Chung和Ç，2014；Hochreiter和Schmidhuber，1997；Martin和Cundy，2018；Peng等人，2023，2024a；Qin等人，2023，2024d；Siems等人，2025；Sun等人，2024；von Oswald等人，2025）——这些方法尚未在大规模推理模型中得到充分验证，几乎所有具有竞争力的LRMs到目前为止仍然依赖传统的注意力设计。唯一的例外是采用Mamba架构（Dao和Gu，2024；Gu和Dao，2024）的Hunyuan-T1模型（腾讯AI实验室，2025）。然而，该模型未开源，披露的细节也很少。在本工作中，我们旨在构建并开源一个能够高效扩展测试时计算能力、与最先进推理模型竞争的大型推理模型。

我们介绍了MiniMax-M1，一种具有混合专家（MoE）架构和闪电注意力（Qin等，2024b）的推理模型，这是一种线性注意力变体（Qin等，2022a）的输入/输出感知实现。MiniMax-M1基于我们之前的MiniMax-Text-01（MiniMax等，2025）模型开发，总共包含4560亿参数，具有459亿激活和32个专家。在我们的注意力设计中，每隔七个transnormer块（Qin等，2022a）就跟随一个带有闪电注意力的softmax注意力的transformer块。这一设计在理论上实现了推理长度的高效扩展，达到数十万令牌，如图1（右）所示。例如，与DeepSeek R1相比，M1在生成长度为64K令牌时的FLOPs少于50%，在长度为100K令牌时大约只有25%的FLOPs。这种显著的计算成本降低使得M1在推理和大规模RL训练中都具有极高的效率。此外，由于其闪电注意力机制，并且与MiniMax-Text-01一致，我们的M1模型原生支持高达100万令牌的上下文长度——是DeepSeek R1上下文大小的八倍，也是迄今为止所有开源权重LRMs的十倍数量级。这些特性使M1特别适合处理需要长输入和生成长时间思考的复杂实际任务。M1与其他领先模型的最大输入和输出长度的比较见表1。

为了开发我们的M1模型，我们首先在7.5T个标记的MiniMax-Text-01上继续预训练
c经过精心策划、推理密集的语料库。随后，我们进行有监督的微调（SFT）

Table 1 | The maximum supported input length and output length (# tokens) of different reasoning models. For Claude-4 we refer to the Claude-4-Opus model. "DS-R1" represents the latest `DeepSeek-R1-0528` model.

	o3	Gemini 2.5 Pro	Claude 4	DS-R1	Qwen3-235B	MiniMax-M1-80k
Max Input	200K	1M	200K	128K	128K	1M
Max Output	100K	64K	32K	64K	32K	80K

to inject certain chain-of-thought (CoT) (Wei et al., 2022) patterns, establishing a strong foundation for reinforcement learning, the core stage of M1 development. Notably, our RL scaling with M1 is made efficient through innovations from two key perspectives: (1) We propose a novel RL algorithm, CISPO, which abandons the trust region constraint and instead clips the importance sampling weights to stabilize training. This approach always leverages all tokens for gradient computations, achieving enhanced efficiency compared to GRPO (Shao et al., 2024) and DAPO (Yu et al., 2025) empirically – For example, on a controlled study based on Qwen2.5-32B models (Qwen et al., 2025), CISPO achieves a 2x speedup compared to DAPO; (2) Although the hybrid-attention design in M1 naturally allows for efficient RL scaling, unique challenges arise when scaling RL with this architecture. For instance, we find a precision mismatch between the training and inference kernels of our architecture, which prevents reward growth during RL training. We develop targeted solutions to address these challenges and successfully scale up RL with this hybrid architecture. In the end, our efficient RL framework enables us to complete a full RL run of MiniMax-M1 within 3 weeks using 512 H800 GPUs—equivalent to a rental cost of approximately $0.53M USD.

In addition to methodological innovations, we curate a diverse set of problems and environments for RL training. Our data encompasses both verifiable and non-verifiable problems. For verifiable problems that are typically considered critical for reasoning learning, we not only include mathematical reasoning and competitive programming problems as commonly used in related works, but also leverage our previous data synthesis framework SynLogic (Liu et al., 2025a) to generate diverse logical reasoning problems spanning 41 distinct tasks. Furthermore, we construct sandboxes for complex software engineering (SE) environments derived from SWE-bench (Jimenez et al., 2024), and conduct RL on real-world SE problems with execution-based rewards to improve M1's performance in challenging SE scenarios. Our unverifiable problems span a broad range of domains such as question answering and creative writing, where we use generative reward models to provide the feedback.

We train two versions of MiniMax-M1 models with 40K and 80K tokens of maximum generation length respectively, which leads to two models MiniMax-M1-40k and MiniMax-M1-80k. MiniMax-M1-80k outperforms MiniMax-M1-40k on complex mathematical and coding tasks, further demonstrating the benefits of scaling test-time compute. As shown in Figure 1 (Left), MiniMax-M1 surpasses previous leading open-weight models such as the original DeepSeek-R1 and Qwen-235B overall, with particular advantages in complex software engineering, tool-using, and long-context tasks. Compared to the latest DeepSeek-R1-0528 model, MiniMax-M1 lags in mathematical and coding competitions but achieves comparable or superior performance in more realistic tool-using and long-context scenarios. Notably, MiniMax-M1 outperforms Gemini 2.5 Pro on the agentic tool use benchmark TAU-Bench (Yao et al., 2025), and surpasses OpenAI o3 and Claude 4 Opus on long-context understanding benchmarks. With efficient test-time scaling, we contend that MiniMax-M1 establishes a strong foundation for next-generation language model agents to address real-world challenges.

To facilitate collaboration and advancement in the field, we have made our models publicly available at GitHub and Hugging Face. They are now supported by both the `vLLM` and `Transformers` frameworks, with detailed deployment guides available at vLLM and Transformers respectively. This

表1 | 不同推理模型支持的最大输入长度和输出长度（# 令牌）。对于Claude-4，我们指的是Claude-4-Opus模型。“DS-R1”代表最新的 DeepSeek-R1-0528 模型。

	o3	Gemini 2.5 Pro	Claude 4	DS-R1	Qwen3-235B	MiniMax-M1-80k
Max Input	200K	1M	200K	128K	128K	1M
Max Output	100K	64K	32K	64K	32K	80K

为了注入某些链式思维（CoT）（Wei 等，2022）模式，为强化学习奠定坚实基础，这是 M1 发展的核心阶段。值得注意的是，我们通过两个关键视角的创新，使得与 M1 相关的 RL 扩展变得高效：(1) 我们提出了一种新颖的 RL 算法，CISPO，它放弃了信任区域约束，而是对重要性采样权重进行裁剪以稳定训练。这种方法始终利用所有令牌进行梯度计算，在经验上比 GRPO（Shao 等，2024）和 DAPO（Yu 等，2025）具有更高的效率——例如，在基于 Qwen2.5-32B 模型（Qwen 等，2025）的对照研究中，CISPO 比 DAPO 提升了 2 倍的速度；(2) 尽管 M1 中的混合注意力设计自然允许高效的 RL 扩展，但在用该架构进行 RL 扩展时也会出现一些独特的挑战。例如，我们发现架构的训练内核和推理内核之间存在精度不匹配，这阻碍了 RL 训练中的奖励增长。我们开发了有针对性的解决方案来应对这些挑战，并成功实现了用该混合架构进行 RL 的扩展。最终，我们的高效 RL 框架使我们能够在 3 周内使用 512 个 H800 GPU 完成 MiniMax-M1 的完整 RL 运行——这相当于租赁成本约为 0.53 百万美元美元。

除了方法创新之外，我们还策划了一套多样化的RL训练问题和环境。我们的数据涵盖了可验证和不可验证的问题。对于通常被认为对推理学习至关重要的可验证问题，我们不仅包括在相关工作中常用的数学推理和竞赛编程问题，还利用我们之前的数据合成框架SynLogic（刘等，2025a）生成了涵盖41个不同任务的多样化逻辑推理问题。此外，我们构建了基于SWE-bench（Jimenez等，2024）派生的复杂软件工程（SE）环境沙箱，并在实际的SE问题上进行RL训练，采用基于执行的奖励，以提升M1在具有挑战性的SE场景中的表现。我们的不可验证问题涵盖了问答、创意写作等广泛领域，我们使用生成奖励模型来提供反馈。

我们分别训练了两个版本的MiniMax-M1模型，最大生成长度为40K和80K的tokens，分别命名为MiniMax-M1-40k和MiniMax-M1-80k。MiniMax-M1-80k在复杂的数学和编码任务中优于MiniMax-M1-40k，进一步展示了扩展测试时计算能力的优势。如图1（左）所示，MiniMax-M1整体超越了之前的领先开源模型，如原始的DeepSeek-R1和Qwen-235B，在复杂软件工程、工具使用和长上下文任务中具有特别的优势。与最新的DeepSeek-R1-0528模型相比，MiniMax-M1在数学和编码竞赛中略逊一筹，但在更实际的工具使用和长上下文场景中表现出相当或更优的性能。值得注意的是，MiniMax-M1在agentic工具使用基准TAU-Bench（Yao等，2025）上优于Gemini 2.5 Pro，并在长上下文理解基准上超越OpenAI o3和Claude 4 Opus。通过高效的测试时扩展，我们认为MiniMax-M1为下一代语言模型代理应对现实世界挑战奠定了坚实的基础。

为了促进该领域的合作与发展，我们已将我们的模型公开发布
a可在 GitHub 和 Hugging Face 上获取。它们现在由 vLLM 和 Transformers 两个版本支持
f框架，分别提供 vLLM 和 Transformers 的详细部署指南。

enables easy integration of MiniMax-M1 into modern inference pipelines. We also provide commercial standard API at minimax.io.

2. Preparation for Scalable RL: Continual Pretraining and SFT

In this work, we focus on scaling up reinforcement learning to enhance reasoning capabilities of Minimax-Text-01. To facilitate scalable RL training, we first carry out continual pretraining of our base model to strengthen its intrinsic reasoning abilities. Subsequently, we perform a cold-start supervised fine-tuning (SFT) stage to inject specific reasoning patterns to the model, thereby providing a stronger foundation for the subsequent RL phase.

2.1. Continual Pre-Training: Foundation for RL Scaling

To enhance the reasoning and long context capabilities of the foundation model while ensuring diversity, we continue training the MiniMax-Text-01 model with additional 7.5T tokens with optimized data quality and mixture.

Training Data. We refine our pretraining Web and PDF parsing mechanisms and enhance our heuristic cleaning rules to ensure a high recall rate for mathematical and code-related data. We prioritize the extraction of natural Question-Answer (QA) pairs from a diverse range of sources, including webpages, forums, and textbooks, while strictly avoiding the use of synthetic data. Additionally, we conduct semantic deduplication on the QA data to maintain its diversity and uniqueness. Furthermore, we increase the proportion of STEM (Science, Technology, Engineering, and Mathematics), code, book, and reasoning-related data to 70%. This significantly enhances the foundation model's ability to handle complex tasks without compromising its other general capabilities.

Training Recipe. We decrease the coefficient of the MoE auxiliary loss and adjust the parallel training strategy to support a larger training micro batch size, which mitigates the detrimental effects of the auxiliary loss on overall model performance. Based on MiniMax-Text-01, we continue training with a constant learning rate of 8e-5 for 2.5T tokens, followed by a decay schedule over 5T tokens down to 8e-6.

Long Context Extension. For a hybrid-lightning architecture model with higher convergence complexity, we have observed that excessively aggressive extensions of the training length can lead to a sudden gradient explosion that may occur during the training process. This makes the optimization process extremely challenging. We attribute this to the parameter optimization of the earlier layers not keeping up with the changes in the later layers – For lightning attention, the earlier and later layers have different decay rates, which makes the earlier layers focus more on local information. We alleviate this issue by adapting a smoother extension of context length across four stages, starting from a 32K context window length and ultimately extending the training context to 1M tokens.

2.2. Supervised Fine-Tuning: Focused Alignment for Efficient RL

After continual pretraining, we conduct Supervised Fine-Tuning (SFT) to instill desired behaviors like reflection-based Chain-of-Thought (CoT) reasoning using high-quality examples, creating a strong starting point for more efficient and stable RL in the next stage. Specifically, we curate data samples with long CoT responses. These data samples cover diverse domains such as math, coding, STEM, writing, QA, and multi-turn chat. Math and coding samples account for around 60% of all the data.

使得将MiniMax-M1轻松集成到现代推理流程中。我们还在minimax.io提供商业标准API。

2. 可扩展强化学习的准备工作：持续预训练和SFT

在本工作中，我们专注于扩大强化学习的规模，以增强 Minimax-Text-01 的推理能力。为了促进可扩展的强化学习训练，我们首先对基础模型进行持续预训练，以增强其内在的推理能力。随后，我们进行冷启动的监督微调（SFT）阶段，将特定的推理模式注入模型，从而为后续的 RL 阶段提供更坚实的基础。

2.1. 持续预训练：强化学习扩展的基础

To 提升基础模型的推理能力和长上下文处理能力，同时确保
d我们继续使用额外的7.5T标记对MiniMax-Text-01模型进行训练，优化
dATA质量和混合。

训练数据。我们优化预训练的网页和PDF解析机制，并增强启发式清洗规则，以确保数学和代码相关数据的高召回率。我们优先从各种来源（包括网页、论坛和教科书）提取自然的问答（QA）对，同时严格避免使用合成数据。此外，我们对QA数据进行语义去重，以保持其多样性和唯一性。此外，我们将STEM（科学、技术、工程和数学）、代码、书籍和推理相关数据的比例提高到70%。这显著增强了基础模型处理复杂任务的能力，同时不影响其其他通用能力。

训练方案。我们降低了MoE辅助损失的系数，并调整了并行训练策略，以支持更大的微批次训练规模，从而减轻辅助损失对整体模型性能的负面影响。基于MiniMax-Text-01，我们以8e-5的恒定学习率继续训练2.5万亿个tokens，然后在5万亿个tokens上采用衰减计划，降低到8e-6。

长上下文扩展。对于具有更高收敛复杂度的混合闪电架构模型，我们观察到过度激进的训练长度扩展可能导致在训练过程中发生突发的梯度爆炸。这使得优化过程变得极其具有挑战性。我们将其归因于前几层的参数优化未能跟上后几层的变化——对于闪电注意力机制，前后几层具有不同的衰减率，这使得前几层更关注局部信息。我们通过在四个阶段中采用更平滑的上下文长度扩展来缓解这一问题，从32K的上下文窗口长度开始，最终将训练上下文扩展到1M个标记。

2.2. 有监督微调：高效强化学习的聚焦对齐

经过持续预训练后，我们进行有监督微调（SFT），以通过高质量的示例灌输期望的行为，例如基于反思的链式思维（CoT）推理，为下一阶段更高效、更稳定的强化学习奠定坚实的基础。具体而言，我们筛选具有长链式思维响应的数据样本。这些数据样本涵盖数学、编码、STEM、写作、问答和多轮对话等多个领域。数学和编码样本约占所有数据的60%。

3. Efficient RL Scaling: Algorithms and Lightning Attention

As shown in Figure 1 (Right), the M1 architecture demonstrates a clear efficiency advantage during inference. This naturally facilitates efficient RL scaling where increasingly longer responses are generated. However, as pioneers in scaling up RL with this hybrid architecture, we encounter unique challenges during the process, and the RL procedure can become unstable or even fail due to various issues. To address these difficulties, we develop targeted solutions that enable us to successfully scale up RL training for M1. In addition, we propose a new RL algorithm that achieves greater RL efficiency compared to existing methods. These dual contributions yield an efficient and scalable RL framework for training M1, where the complete training cycle requires 3 weeks on 512 H800 GPUs—equivalent to a rental cost of approximately $0.53M USD. In this section, we first provide general context on RL and present our novel RL algorithm, and then describe the specific challenges we face with the hybrid architecture, along with the solutions we devise to overcome them.

3.1. Efficient RL Scaling with CISPO

Background. For questions q from a dataset $\mathcal{D}$, we denote π as the policy model parameterized by θ, and o as the response generated by the policy. PPO (Schulman et al., 2017) adopts the following objective to optimize the policy to maximize the expected return, and a clipping operation is applied to stabilize training:

$$\mathcal{J}_{\text{PPO}}(\theta) = \mathbb{E}_{q\sim\mathcal{D}, o_i\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{|o_i|}\sum_{t=1}^{|o_i|} \min\left(r_{i,t}(\theta)\hat{A}_{i,t}, \text{clip}\left(r_{i,t}(\theta), 1-\epsilon, 1+\epsilon\right)\hat{A}_{i,t} \right) - \beta D_{KL}(\pi_\theta || \pi_{\text{ref}}) \right], \tag{1}$$

where $r_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t}|q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q,o_{i,<t})}$ is the importance sampling (IS) weight, which is used to correct the distribution during off-policy updates, because we use $\pi_{\theta_{\text{old}}}$ to collect trajectories to update the policy via multiple steps in a minibatch manner. While PPO requires a separate value model to compute the advantage $\hat{A}_{i,t}$, GRPO (Shao et al., 2024) eliminates the value model and defines the advantage as the output reward relative to other responses in the group:

$$\hat{A}_{i,t} = \frac{R_i - \text{mean}(\{R_j\}_{j=1}^G)}{\text{std}(\{R_j\}_{j=1}^G)}, \tag{2}$$

where R_i is the reward of the response, and G responses $\{o_i\}_{i=1}^G$ are sampled for each question. The reward is either from rule-based verifiers such as in mathematical problem solving, or from a reward model.

Issues of Token Clipping. In our initial experiments with the hybrid architecture under the zero-RL setting, we observed that the GRPO algorithm adversely affected training performance and failed to effectively promote the emergence of long CoT reasoning behaviors. Through a series of controlled ablation studies, we ultimately identified the undesirable clipping operation in the original PPO/GRPO loss as the primary factor contributing to degraded learning performance. Specifically, we found that tokens associated with reflective behaviors (e.g., `However`, `Recheck`, `Wait`, `Aha`), which often serve as "forks" in reasoning paths, were typically rare and assigned low probabilities by our base model. During policy updates, these tokens were likely to exhibit high $r_{i,t}$ values. As a result, these tokens were clipped out after the first on-policy update, preventing them from contributing to subsequent off-policy gradient updates. This issue was particularly pronounced in our hybrid-architecture model and further hindered the scalability of reinforcement learning. These low-probability tokens, however,

3. 高效的强化学习扩展：算法与闪电注意力

如图1（右）所示，M1架构在推理过程中展现出明显的效率优势。这自然有助于在生成越来越长的响应时实现高效的RL扩展。然而，作为采用这种混合架构进行RL扩展的先驱，我们在过程中遇到了一些独特的挑战，RL过程可能会因各种问题变得不稳定甚至失败。为了解决这些困难，我们开发了有针对性的解决方案，使我们能够成功地扩展M1的RL训练。此外，我们还提出了一种新的RL算法，其RL效率优于现有方法。这两方面的贡献共同构建了一个高效且可扩展的RL框架，用于训练M1，其中完整的训练周期在512块H800 GPU上需要3周——相当于约0.53百万美元的租赁成本。在本节中，我们首先提供关于RL的总体背景，并介绍我们的新颖RL算法，然后描述我们在混合架构中面临的具体挑战，以及我们为克服这些挑战所设计的解决方案。

3.1. 使用CISPO实现高效的RL扩展

背景。对于来自数据集$\mathcal{D}$的问题q，我们用π表示由θ参数化的策略模型，用o表示策略生成的响应。PPO（Schulman等，2017）采用以下目标来优化策略，以最大化预期回报，并应用裁剪操作以稳定训练：

$$\begin{aligned}\mathcal{J}_{\text{PPO}}(\theta) = \mathbb{E}&_{q\sim\mathcal{D}, o_i\sim\pi_{\theta_{\text{old}}}(\cdot|q)}\\ &\left[\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\min\Big(r_{i,t}(\theta)\hat{A}_{i,t}, \text{clip}\big(r_{i,t}(\theta), 1-\epsilon, 1+\epsilon\big)\hat{A}_{i,t}\Big) - \beta D_{KL}(\pi_\theta||\pi_{\text{ref}})\right],\end{aligned} \tag{1}$$

其中 $r_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t}|q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q,o_{i,<t})}$ 是重要性采样（IS）权重，用于在离策略更新过程中校正分布，因为我们使用 $\pi_{\theta_{\text{old}}}$ 来收集轨迹，通过多步的小批量方式更新策略。而 PPO 需要一个单独的值模型来计算优势$\hat{A}_{i,t}$，而 GRPO（Shao 等，2024）则省略了值模型，并将优势定义为相对于组内其他响应的输出奖励：

$$\hat{A}_{i,t} = \frac{R_i - \text{mean}(\{R_j\}_{j=1}^G)}{\text{std}(\{R_j\}_{j=1}^G)}, \tag{2}$$

w这里 R_i 是响应的奖励，且每个问题都采样了 G 个响应 $\{o_i\}_{i=1}^G$。
r奖励要么来自于基于规则的验证器，比如在数学问题求解中，要么来自于奖励
m模型。

令牌裁剪问题。在我们在零RL设置下的混合架构初步实验中，我们观察到GRPO算法对训练性能产生了不利影响，未能有效促进长链推理行为（CoT）的出现。通过一系列受控消融研究，我们最终确定了原始PPO/GRPO损失中不良的裁剪操作是导致学习性能下降的主要因素。具体而言，我们发现与反思行为相关的令牌（例如`However`、`Recheck`、`Wait`、`Aha`），这些令牌通常作为推理路径中的“分叉”，在我们的基础模型中通常较少出现且概率较低。在策略更新过程中，这些令牌很可能表现出较高的$r_{i,t}$值。因此，这些令牌在第一次策略更新后被裁剪掉，阻止它们对后续的离策略梯度更新做出贡献。这一问题在我们的混合架构模型中尤为突出，进一步阻碍了强化学习的扩展性。然而，这些低概率的令牌，

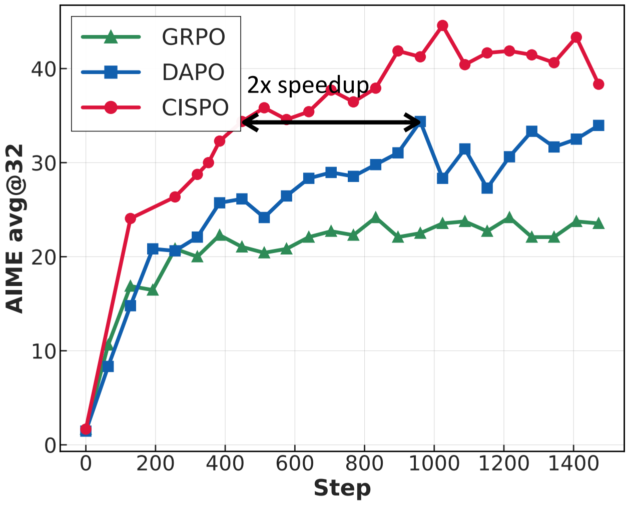

Figure 2 | Comparison of GRPO, DAPO, and our proposed CISPO on AIME 2024, based on Qwen2.5-32B-base. CISPO outperforms both GRPO and DAPO in terms of performance at the same number of training steps, and achieves comparable performance to DAPO using 50% of the training steps.

are often crucial for stabilizing entropy (Cui et al., 2025) and facilitating scalable RL (Wang et al., 2025). Although DAPO attempts to mitigate this issue by increasing the upper clipping bound (Yu et al., 2025), we found this approach to be less effective in our setup, which involved 16 rounds of off-policy updates per generation batch.

The CISPO Algorithm. In response, we propose a new algorithm that explicitly avoids dropping tokens, even those associated with large updates, while inherently maintaining entropy within a reasonable range to ensure stable exploration. First, recall that the vanilla REINFORCE objective with corrected distribution for offline updates is:

$$\mathcal{J}_{\text{REINFORCE}}(\theta) = \mathbb{E}_{(q,a)\sim\mathcal{D},o_i\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \texttt{sg}(r_{i,t}(\theta))\hat{A}_{i,t} \log \pi_\theta(o_{i,t} \mid q, o_{i,<t}) \right], \tag{3}$$

where $\texttt{sg}(\cdot)$ denotes the stop-gradient operation. Rather than clipping the token updates as in PPO/GRPO, we instead clip the importance sampling weight in Eq. 3 to stabilize training. We term our approach CISPO (**C**lipped **IS**-weight **P**olicy **O**ptimization). Adopting the group relative advantage from GRPO and the token-level loss (Liu et al., 2025b; Yu et al., 2025), CISPO optimizes the following objective:

$$\mathcal{J}_{\text{CISPO}}(\theta) = \mathbb{E}_{(q,a)\sim\mathcal{D},\{o_i\}_{i=1}^G\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{\sum_{i=1}^G |o_i|} \sum_{i=1}^{G} \sum_{t=1}^{|o_i|} \texttt{sg}(\hat{r}_{i,t}(\theta))\hat{A}_{i,t} \log \pi_\theta(o_{i,t} \mid q, o_{i,<t}) \right], \tag{4}$$

where $\hat{r}_{i,t}(\theta)$ is the clipped IS weight:

$$\hat{r}_{i,t}(\theta) = \text{clip}\left(r_{i,t}(\theta), 1 - \epsilon_{low}^{IS}, 1 + \epsilon_{high}^{IS}\right). \tag{5}$$

We note that without weight clipping, $\mathcal{J}_{\text{CISPO}}$ reduces to the standard policy gradient objective. In our experiments, we did not impose a lower bound on the IS weight by setting ϵ_{low}^{IS} to a large value;

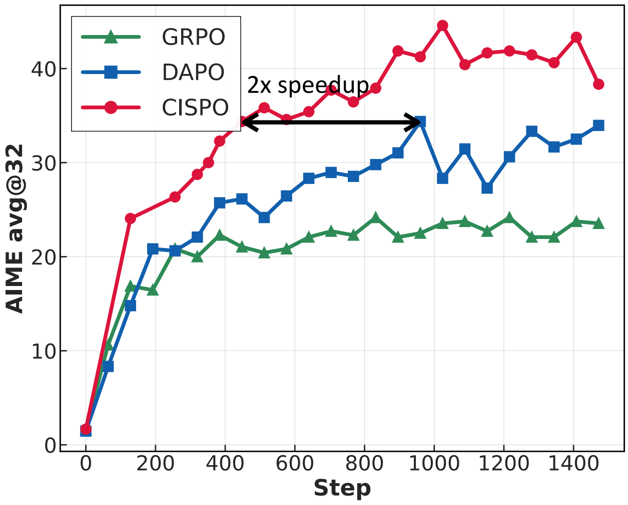

图2 | 基于Qwen2.5-32B-base的AIME 2024上GRPO、DAPO和我们提出的CISPO的比较。CISPO在相同训练步数下的性能优于GRPO和DAPO，并且在使用50%训练步数的情况下实现了与DAPO相当的性能。

通常对于稳定熵（Cui 等人，2025）和促进可扩展的强化学习（Wang 等人，2025）至关重要。虽然DAPO 试图通过增加上限裁剪界限（Yu 等人，2025）来缓解这个问题，但我们发现这种方法在我们的设置中效果较差，该设置涉及每一代批次进行 16 轮的离策略更新。

CISPO 算法。作为回应，我们提出了一种新算法，明确避免丢弃令牌，即使是那些与大更新相关的令牌，同时本质上保持熵在合理范围内，以确保稳定的探索。首先，回顾一下，用于离线更新的修正分布的 vanilla REINFORCE 目标是：

$$\mathcal{J}_{\text{REINFORCE}}(\theta) = \mathbb{E}_{(q,a)\sim\mathcal{D},o_i\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{|o_i|}\sum_{t=1}^{|o_i|} \text{sg}(r_{i,t}(\theta))\hat{A}_{i,t}\log\pi_\theta(o_{i,t}\mid q,o_{i,<t})\right], \tag{3}$$

其中 $\text{sg}(\cdot)$ 表示停止梯度操作。与在 PPO/GRPO 中对令牌更新进行裁剪不同，我们在式 3 中裁剪重要性采样权重以稳定训练。我们将此方法称为 CISPO（裁剪的 IS 权重策略优化）。借鉴 GRPO 的组相对优势和令牌级损失（Liu 等，2025b；Yu 等，2025），CISPO 优化以下目标：

$$\mathcal{J}_{\text{CISPO}}(\theta) = \mathbb{E}_{(q,a)\sim\mathcal{D},\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{\sum_{i=1}^{G}|o_i|}\sum_{i=1}^{G}\sum_{t=1}^{|o_i|} \text{sg}(\hat{r}_{i,t}(\theta))\hat{A}_{i,t}\log\pi_\theta(o_{i,t}\mid q,o_{i,<t})\right], \tag{4}$$

其中 $\hat{r}_{i,t}(\theta)$ 是裁剪后的 IS 权重：

$$\hat{r}_{i,t}(\theta) = \text{clip}\left(r_{i,t}(\theta), 1-\epsilon_{low}^{IS}, 1+\epsilon_{high}^{IS}\right). \tag{5}$$

W请注意，如果不进行权重裁剪，$\mathcal{J}_{\text{CISPO}}$ 将退化为标准的策略梯度目标。在我们的实验中，我们没有通过将 ϵ_{low}^{IS} 设置为一个较大的值来对 IS 权重施加下界；

instead, we only tuned ϵ^{IS}_{high}. Although the gradient of Eq. 4 is slightly biased due to weight clipping, this approach preserves gradient contributions from all tokens, especially in long responses. CISPO proves effective in our experiments, helping reduce variance and stabilizing RL training. In addition, we utilize the dynamic sampling and length penalty techniques from Yu et al. (2025). There is no KL penalty term in CISPO similar to other recent works (Hu et al., 2025; Yu et al., 2025).

A General Formulation. While we adopt CISPO in our experiments, here we further present a unified formulation by introducing a token-wise mask into the CISPO objective. This allows for hyperparameter tuning to control whether, and under what conditions, gradients from specific tokens should be dropped:

$$\mathcal{J}_{\text{unify}}(\theta) = \mathbb{E}_{(q,a)\sim\mathcal{D},\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{\sum_{i=1}^{G}|o_i|} \sum_{i=1}^{G}\sum_{t=1}^{|o_i|} \text{sg}(\hat{r}_{i,t}(\theta))\hat{A}_{i,t}\log\pi_{\theta}(o_{i,t} \mid q, o_{i,<t})M_{i,t} \right]. \tag{6}$$

The mask $M_{i,t}$ is equivalent to the mask implicitly defined in the PPO trust region:

$$M_{i,t} = \begin{cases} 0 & \text{if } \hat{A}_{i,t} > 0 \text{ and } r_{i,t}(\theta) > 1 + \epsilon_{\text{high}}, \\ 0 & \text{if } \hat{A}_{i,t} < 0 \text{ and } r_{i,t}(\theta) < 1 - \epsilon_{\text{low}}, \\ 1 & \text{otherwise.} \end{cases} \tag{7}$$

This unified loss formulation can flexibly represent different clipping strategies under a common framework.

Empirical Validation of CISPO. To validate the effectiveness of CISPO, we empirically compare it with DAPO and GRPO in a zero-RL training setting. Specifically, we apply different RL algorithms to train the Qwen2.5-32B-base model on the mathematical reasoning dataset from Yu et al. (2025), and report performance on the AIME 2024 benchmark. As shown in Figure 2, CISPO significantly outperforms both DAPO and GRPO with the same number of training steps. Notably, CISPO demonstrates superior training efficiency compared to other approaches; for example, it matches DAPO's performance with only 50% of the training steps.

3.2. Efficient RL Scaling with Lightning Attention – Challenges and Recipes

As shown in Figure 1 (Right), we emphasize that our hybrid attention inherently enables more efficient RL scaling compared to traditional attention designs, since rollout computation and latency are often the primary bottlenecks in RL training. However, as pioneers in conducting large-scale RL experiments with this novel architecture, we encountered unique challenges and developed targeted solutions, as we describe below.

Computational Precision Mismatch in Generation and Training. RL training is highly sensitive to computational precision. During our RL training, we observed a significant discrepancy in the probabilities of rolled-out tokens between training-mode and inference-mode, as shown in Figure 3 (Left). This discrepancy arose from a precision mismatch between the training and inference kernels. The issue was detrimental and prevented reward growth in our experiments. Interestingly, this issue did not appear in smaller, dense models with softmax attention. Through layer-by-layer analysis, we identified high-magnitude activations in the LM head at the output layer as the primary source of error. To address this, we increased the precision of the LM output head to FP32, thereby realigning the two theoretically identical probabilities, as demonstrated in Figure 3 (Right). This adjustment improved the correlation between training and inference probabilities from approximately 0.9x to

相反，我们只调优了 ϵ^{IS}_{high}。虽然公式 4 的梯度由于权重裁剪而略有偏差，但这种方法保留了所有令牌的梯度贡献，尤其是在长回复中。CISPO 在我们的实验中证明是有效的，有助于降低方差并稳定强化学习训练。此外，我们还采用了 Yu 等人（2025）提出的动态采样和长度惩罚技术。CISPO 中没有类似于其他近期工作（Hu 等人，2025；Yu 等人，2025）的 KL 惩罚项。

一般形式。在我们的实验中采用CISPO的同时，这里通过引入逐个标记的掩码到CISPO目标中，进一步提出了一个统一的公式。这允许进行超参数调节，以控制是否以及在什么条件下，应丢弃来自特定标记的梯度：

$$\mathcal{J}_{\text{unify}}(\theta) = \mathbb{E}_{(q,a)\sim\mathcal{D},\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{\sum_{i=1}^{G}|o_i|}\sum_{i=1}^{G}\sum_{t=1}^{|o_i|}\mathtt{sg}(\hat{r}_{i,t}(\theta))\hat{A}_{i,t}\log\pi_\theta(o_{i,t}\mid q,o_{i,<t})M_{i,t}\right]. \tag{6}$$

掩码 $M_{i,t}$ 等同于在 PPO 信任区域中隐式定义的掩码：

$$M_{i,t} = \begin{cases} 0 & \text{if } \hat{A}_{i,t} > 0 \text{ and } r_{i,t}(\theta) > 1+\epsilon_{\text{high}}, \\ 0 & \text{if } \hat{A}_{i,t} < 0 \text{ and } r_{i,t}(\theta) < 1-\epsilon_{\text{low}}, \\ 1 & \text{otherwise.} \end{cases} \tag{7}$$

这种统一的损失公式可以在一个共同的框架下灵活地表示不同的裁剪策略。

CISPO的经验验证。为了验证CISPO的有效性，我们在零强化学习（zero-RL）训练环境中，实证比较了它与DAPO和GRPO的性能。具体而言，我们应用不同的强化学习算法，在Yu等人（2025）提供的数学推理数据集上训练Qwen2.5-32B-base模型，并在AIME 2024基准测试中报告性能。如图2所示，CISPO在相同的训练步数下显著优于DAPO和GRPO。值得注意的是，CISPO在训练效率方面优于其他方法；例如，它只用一半的训练步数就达到了DAPO的性能。

3.2. 使用闪电注意力进行高效RL扩展——挑战与方案

如图1（右）所示，我们强调我们的混合注意力本质上比传统注意力设计更高效地实现RL的扩展，因为在RL训练中，rollout计算和延迟通常是主要的瓶颈。然而，作为采用这种新颖架构进行大规模RL实验的先驱，我们遇到了一些独特的挑战，并开发了有针对性的解决方案，具体如下。

生成与训练中的计算精度不匹配。强化学习（RL）训练对计算精度非常敏感。在我们的RL训练过程中，我们观察到训练模式和推理模式下滚动生成的标记概率存在显著差异，如图3（左）所示。这一差异源于训练内核和推理内核之间的精度不匹配。这个问题非常严重，阻碍了我们实验中的奖励增长。有趣的是，这个问题在较小的、密集的模型中使用softmax注意力时并未出现。通过逐层分析，我们发现输出层的LM头中存在高幅值激活，成为误差的主要来源。为了解决这个问题，我们将LM输出头的精度提升到FP32，从而重新对齐两个理论上相同的概率，如图3（右）所示。这一调整将训练和推理概率之间的相关性从大约0.9x提升到了{v*}。

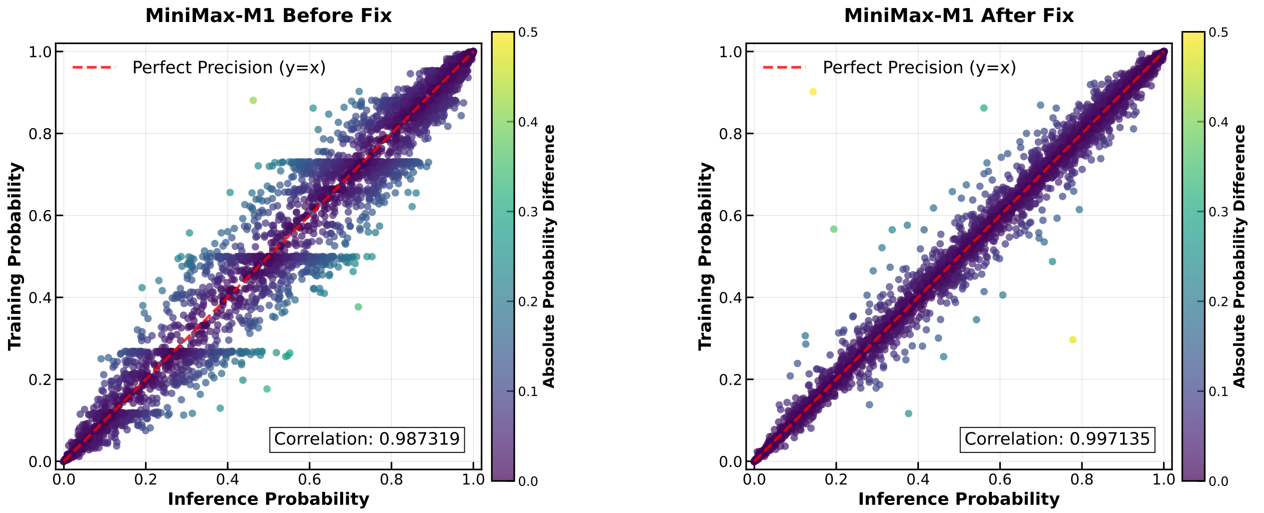

Figure 3 | Probability of tokens in training-mode code vs. probability of tokens in inference-mode code. Each point in the figures represents an individual token. The Pearson correlation coefficient is indicated in the figures. Theoretically, the two probabilities should be identical, and all the tokens should be exactly on the diagonal line. **Left:** Correlation of the M1 model before our fix; **Right:** Correlation of the M1 model after applying our fix of using FP32 precision for the LM output head.

0.99x. Notably, this correlation metric remained stable throughout training, enabling successful reward increase.

Optimizer Hyperparameter Sensitivity. We employ the AdamW (Loshchilov and Hutter, 2019) optimizer, and inappropriate configurations of β_1, β_2, and ϵ can lead to non-convergence during training. (Molybog et al., 2023). For instance, using the default configuration from VeRL (Sheng et al., 2024), where betas = (0.9, 0.999) and eps = 1e-8, can result in such issues. We have observed that the gradient magnitudes in MiniMax-M1 training span a wide range, from 1e-18 to 1e-5, with the majority of the gradients being smaller than 1e-14. Furthermore, the correlation between the gradients of adjacent iterations is weak. Based on this, we set $\beta_1 = 0.9$, $\beta_2 = 0.95$, and eps=1e-15.

Early Truncation via Repetition Detection. During RL training, we found that complex prompts could induce pathologically long and repetitive responses, whose large gradients threatened model stability. Our goal was to preemptively terminate these generation loops rather than penalize the already repetitive text. As simple string-matching is ineffective against varied repetition patterns, we developed a heuristic based on token probabilities. We observed that once a model enters a repetitive cycle, the probability for each token soars. Consequently, we implemented an early truncation rule: generation is halted if 3,000 consecutive tokens each have a probability above 0.99. This method successfully prevents model instability and improves generation throughput by eliminating these pathological, long-tail cases.

4. Scaling Reinforcement Learning with Diverse Data

In this section, we describe the data and reward we adopted for our RL stage. We incorporate a diverse set of environments in our RL training pipeline, including tasks that can be verified by rules and general tasks that need to be verified through reward models. All these environments are integrated into the RL stage using a carefully designed curriculum.

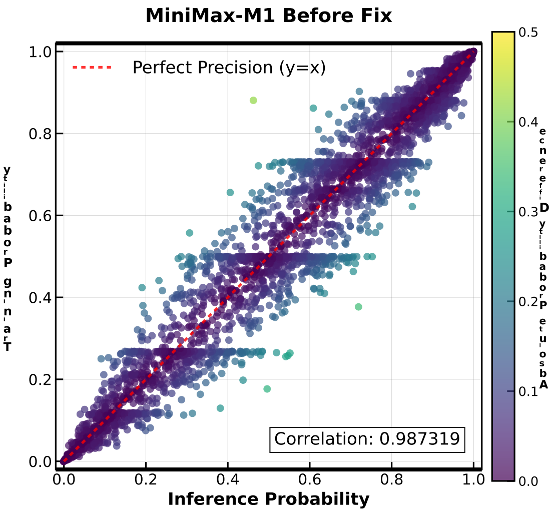

图3 | 训练模式代码中标记的概率与推理模式代码中标记的概率。图中的每个点代表一个单独的标记。图中标注了皮尔逊相关系数。理论上，这两个概率应当完全相同，所有的标记都应位于对角线上。左图：在我们修正之前的 M1 模型的相关性；右图：在应用我们使用 FP32 精度作为语言模型输出头的修正之后的 M1 模型的相关性。

0.99x. 值得注意的是，这个相关性指标在整个训练过程中保持稳定，从而实现了成功r奖励增加。

优化器超参数敏感性。我们采用AdamW（Loshchilov和Hutter，2019）优化器，不适当的β_1、β_2和ϵ配置可能导致训练过程中不收敛（Molybog等，2023）。例如，使用VeRL（Sheng等，2024）中的默认配置，其中beta = (0.9、0.999) 和eps = 1e-8，可能会出现此类问题。我们观察到MiniMax-M1训练中的梯度幅值范围很广，从1e-18到1e-5，大部分梯度小于1e-14。此外，相邻迭代的梯度相关性较弱。基于此，我们将$\beta_1 = 0.9$、$\beta_2 = 0.95$ 和eps= 1e-15。

通过重复检测实现早期截断。在强化学习训练过程中，我们发现复杂的提示可能引发病理性长且重复的响应，其巨大的梯度威胁到模型的稳定性。我们的目标是预先终止这些生成循环，而不是惩罚已经重复的文本。由于简单的字符串匹配对多样化的重复模式效果不佳，我们开发了一种基于标记概率的启发式方法。我们观察到，一旦模型进入重复循环，每个标记的概率就会飙升。因此，我们实现了一条早期截断规则：如果连续3,000个标记的概率都高于0.99，则停止生成。该方法成功防止了模型的不稳定性，并通过消除这些病理性长尾情况，提高了生成的吞吐量。

4. 使用多样化数据进行强化学习的扩展

在本节中，我们描述了用于强化学习阶段的数据和奖励。我们在强化学习训练流程中结合了多样的环境，包括可以通过规则验证的任务和需要通过奖励模型验证的通用任务。所有这些环境都通过精心设计的课程融入到强化学习阶段中。

4.1. Reasoning-Intensive Tasks with Rule-based Verification

Below, we introduce our data that can be verified by deterministic rules. For all the following tasks, we employ rule-based final correctness as the correctness reward, complemented by a format reward.

Mathematical Reasoning. Our initial mathematical dataset comprises hundreds of thousands of high-quality, competition-level problems, meticulously curated and organized from public sources and official mathematics competitions. These problems span a wide range of difficulty levels, each paired with a standard reference solution. Our data cleaning pipeline begins with the removal of incomplete samples and those exhibiting formatting or typographical errors. We subsequently apply embedding-based deduplication across the RL data sources and enforce a strict separation from the SFT dataset to avoid any overlap, as leakage from the SFT phase into the RL stage hinders exploration and undermines training effectiveness. Additionally, we employ both n-gram and embedding-based methods to eliminate potential contamination from commonly used mathematical benchmark test sets, thereby ensuring the integrity and fairness of our evaluations. We filter out samples containing multiple sub-problems, proof-based questions, and binary questions (e.g., true/false) that are susceptible to random guessing. Multiple-choice questions are reformulated into open-ended formats to better align with our reinforcement learning framework. Next, we employ our internal model to extract the final answers from the reference solution, retaining only those samples whose extracted answers can be correctly parsed by our rule-based answer checker. Finally, we use a strong reasoning model to compute the pass@10 for each question and retain only those samples with a pass rate strictly between 0 and 0.9, resulting in a curated dataset of nearly 50K high-quality mathematical samples for our RL training.

Logical Reasoning. For logical reasoning data, we carefully select 41 logical reasoning tasks requiring non-trivial reasoning ability such as cipher and Sudoku, then we implement a data synthesis framework to synthesize all the data. Concretely, we utilize our SynLogic framework (Liu et al., 2025a) to implement the data synthesis pipeline featuring task-specific data generators and rule-based task-specific verifiers, enabling automatic logical data generation. We meticulously configure the difficulty parameters during generation, ensuring the appropriate learning challenge of the generated data. Specifically, to prevent inclusion of overly difficult instances, we establish an upper difficulty bound based on the solvability limits of current strong reasoning models, requiring their pass@10 rates greater than zero. Similarly, we set a lower difficulty bound using the lowest difficulty parameters for which the MiniMax-Text-01 model achieves pass rates between 0 and 0.5. This approach ensures the data maintains a balance between difficulty and learnability. In addition, as the model capabilities improve during training, we increase the difficulty of the data in the later stages. Using this framework, we synthesize approximately 53K logical reasoning samples for RL training.

Competitive Programming. For the competitive programming problems, we collect publicly available problems from online judge platforms and popular coding websites. For problems lacking test cases, we develop an LLM-based workflow and use the MiniMax-Text-01 model to generate comprehensive test suites. Similar to our approach with mathematical reasoning datasets, we filter problems based on quality and difficulty using pass rates from model sampling, retaining moderately challenging and high-quality algorithmic problems. Through this process, we generate 30K competitive programming data samples for RL training.

Software Engineering. For the software engineering domain, inspired by SWE-bench (Jimenez et al., 2024), we construct verifiable reinforcement learning environments by leveraging real-world data from public GitHub repositories. Our dataset primarily comprises issues and pull requests (PRs) that encapsulate common software development challenges, including bug localization, code repair, and test case synthesis. To facilitate effective reinforcement learning, we develop a sophisticated containerized sandbox environment that simulates a realistic software development workflow. This

4.1. 以规则为基础的验证的推理密集型任务

B以下，我们介绍可以通过确定性规则验证的数据。对于以下所有任务，
w我们采用基于规则的最终正确性作为正确性奖励，并辅以格式奖励。

数学推理。我们的初始数学数据集包含数十万高质量、竞赛级别的问题，经过精心筛选和整理，来源于公共资源和官方数学竞赛。这些问题涵盖了广泛的难度等级，每个问题都配有标准参考解答。我们的数据清洗流程首先去除不完整的样本以及存在格式或排版错误的样本。随后，我们在强化学习数据源中应用基于嵌入的去重方法，并严格将其与SFT数据集分离，以避免任何重叠，因为SFT阶段的泄漏会阻碍探索并削弱训练效果。此外，我们还采用n-gram和嵌入式方法，消除来自常用数学基准测试集的潜在污染，从而确保评估的完整性和公平性。我们过滤掉包含多个子问题、基于证明的问题以及易于随机猜测的二元问题（例如，正确/错误）。多项选择题被重新表述为开放式问题，以更好地适应我们的强化学习框架。接下来，我们使用内部模型从参考解中提取最终答案，只保留那些其提取的答案能被我们的规则基础答案检查器正确解析的样本。最后，我们使用强大的推理模型计算每个问题的pass@10，并只保留那些通过率严格在0到0.9之间的样本，从而获得近5万高质量的数学样本，用于我们的强化学习训练。

逻辑推理。对于逻辑推理数据，我们精心选择了41个需要非平凡推理能力的任务，如密码和数独，然后我们实现了一个数据合成框架来合成所有数据。具体而言，我们利用我们的SynLogic框架（Liu等，2025a）实现了具有任务特定数据生成器和基于规则的任务特定验证器的数据合成流程，从而实现自动逻辑数据生成。在生成过程中，我们细致地配置难度参数，确保生成数据具有适当的学习挑战性。具体来说，为了防止包含过难的实例，我们根据当前强推理模型的可解性极限建立了一个上限难度，要求它们的pass@10率大于零。同样，我们使用MiniMax-Text-01模型在0到0.5之间的通过率所对应的最低难度参数设置了一个下限难度。这种方法确保数据在难度和可学习性之间保持平衡。此外，随着模型能力在训练过程中不断提升，我们在后期阶段增加数据的难度。利用这个框架，我们合成了大约53K个逻辑推理样本用于RL训练。

竞赛编程。对于竞赛编程题目，我们收集了来自在线判题平台和流行编码网站的公开题目。对于缺少测试用例的题目，我们开发了基于大型语言模型的工作流程，并使用 MiniMax-Text-01 模型生成全面的测试用例。类似于我们处理数学推理数据集的方法，我们根据模型采样的通过率对题目进行质量和难度筛选，保留中等难度和高质量的算法题。通过这一过程，我们生成了30K个竞赛编程数据样本用于强化学习训练。

软件工程。对于软件工程领域，受到SWE-bench（Jimenez等人，2024）的启发，我们通过利用来自公共GitHub仓库的真实数据构建可验证的强化学习环境。我们的数据集主要包括问题和拉取请求（PRs），这些内容反映了常见的软件开发挑战，包括漏洞定位、代码修复和测试用例合成。为了促进有效的强化学习，我们开发了一个复杂的容器化沙箱环境，模拟了一个逼真的软件开发工作流程。

environment enables the actual execution of code, providing direct and verifiable feedback on the correctness and efficacy of an agent's proposed interventions. The pass/fail status of pre-defined or newly generated test cases serves as the primary reward signal for our RL framework. A successful execution that passes all relevant test cases yields a positive reward, while compilation errors, runtime failures, or test case regressions result in a zero or negative reward, thus providing a clear signal for policy optimization. Through this process, we curate several thousand high-quality data samples. Each sample includes a problem description (e.g., bug report from an issue), the initial faulty code, and a set of associated test cases. This setup allows our RL agent to learn to accurately pinpoint bugs, propose correct code fixes, and even synthesize new, effective test cases, with performance directly verifiable through the execution within our sandboxed environment.

4.2. General Domain Tasks with Model-based Feedbacks

In this section, we further extend the RL scope to a wider array of general domain tasks. As these tasks cannot be easily verified by rules, we utilize reward models to provide the feedback.

4.2.1. Data and Reward Models

Our general RL dataset consists of a total of 25K complex samples. These can be broadly categorized into two types: samples with ground-truth answers that are verifiable but difficult to validate using rules, and samples without ground-truth answers.

Tasks with Ground Truth. This category primarily includes STEM and other factual problems where answers are objective but may have multiple valid expressions. Such diversity often renders rule-based answer checkers inaccurate. Our data cleaning process is similar to that used in mathematical reasoning, while we use our Generative Reward Model (GenRM) as a verifier, instead of relying on rule-based checkers. To evaluate consistency between ground-truth answers and model responses, we adopt a five-grade reward scale to evaluate the two components. First, we construct a human-annotated reward model benchmark, which covers a range of objective tasks across diverse knowledge and task domains, especially the pairs of model response–ground truth that rule-based checkers fail to judge accurately. Second, we evaluate the GenRM's effectiveness by comparing the Best-of-N (BoN) responses selected by GenRM against the pass@N metrics across several benchmarks. GenRM performance is assessed using its accuracy on the human-annotated benchmark and the performance gap between BoN and pass@N. These metrics guide experiments to optimize both the data distribution and the prompt design used during the GenRM training.

Tasks without Ground Truth. This category encompasses a wider range of tasks, including instruction-following, creative writing, etc. Prompts are sampled from a large pool based on our internal tagging system, ensuring a balanced training distribution across fine-grained domains. Even though these queries are typically open-ended and do not have a ground-truth answer, we seek to pair a reference answer for each query, which serves as a reference for reward model judgment. To this end, we first generate responses by various internal and external models, and then these reference answers will undergo our internal quality evaluation. During RL training, we adopt a pairwise comparison framework to evaluate model responses. Each comparison yields a score of -1, 0, or 1, indicating whether the model's output is worse than, similar to, or better than a reference answer. For instruction-following tasks with constraints particularly, we utilize both the rule-based reward to assess whether the response satisfies the constraint, and model-based reward to evaluate response's quality. As with the ground-truth setting, we first build a human-annotated benchmark, incorporating multiple blind preference judgments from reliable annotators. We then refine our scoring criteria and preference prompt to optimize accuracy as well as potential biases, which would be mentioned in §4.2.2 below.

环境使得代码的实际执行成为可能，提供了关于代理提出的干预措施的正确性和有效性的直接且可验证的反馈。预定义或新生成的测试用例的通过/未通过状态作为我们强化学习框架的主要奖励信号。成功执行并通过所有相关测试用例会产生正向奖励，而编译错误、运行时失败或测试用例回归则会导致零或负奖励，从而为策略优化提供明确的信号。通过这一过程，我们整理了数千个高质量的数据样本。每个样本包括问题描述（例如，问题报告中的错误信息）、初始的有缺陷代码以及一组相关的测试用例。这一设置使我们的强化学习代理能够学习准确定位错误、提出正确的代码修复方案，甚至合成新的有效测试用例，其性能可以通过在我们的沙箱环境中的执行直接验证。

4.2. 具有模型反馈的通用领域任务

我在本节中，我们将强化学习的范围进一步扩展到更广泛的通用领域任务。由于这些
t无法通过规则轻松验证询问，我们利用奖励模型提供反馈。

4.2.1. Data and Reward Models

O你的通用RL数据集总共包含25K个复杂样本。这些可以大致分类为
i分为两类：具有可验证但难以验证的真实答案的样本
r规则，以及没有真实答案的样本。

具有 Ground Truth 的任务。此类别主要包括 STEM 和其他事实性问题，其中答案是客观的，但可能有多种有效表达。这种多样性常常导致基于规则的答案检查器不准确。我们的数据清洗过程类似于数学推理中使用的方法，而我们使用生成奖励模型（GenRM）作为验证器，而不是依赖基于规则的检查器。为了评估 Ground Truth 答案与模型响应之间的一致性，我们采用五级奖励尺度来评估这两个组成部分。首先，我们构建了一个由人工标注的奖励模型基准，涵盖了跨越不同知识和任务领域的多种客观任务，特别是那些规则检查器无法准确判断的模型响应- Ground Truth 对。其次，我们通过比较由 GenRM 选择的最佳响应（Best-of-N，BoN）与多个基准测试中的通过率@N（pass@N）指标，评估 GenRM 的有效性。GenRM 的性能通过其在人工标注基准上的准确率以及 BoN 与 pass@N 之间的性能差距进行评估。这些指标指导实验，以优化在 GenRM 训练过程中使用的数据分布和提示设计。

没有 Ground Truth 的任务。这一类别涵盖了更广泛的任务，包括指令执行、创意写作等。提示从我们内部标签系统的庞大库中抽取，确保在细粒度领域中训练分布的平衡。尽管这些查询通常是开放式的，没有明确的正确答案，但我们会为每个查询配对一个参考答案，作为奖励模型判断的参考依据。为此，我们首先由各种内部和外部模型生成响应，然后对这些参考答案进行内部质量评估。在强化学习训练中，我们采用成对比较的框架来评估模型响应。每次比较的得分为 -1、0 或 1，表示模型输出比参考答案差、相似或更优。对于具有特别约束的指令执行任务，我们同时使用基于规则的奖励来评估响应是否满足约束，以及基于模型的奖励来评估响应的质量。与有 Ground Truth 的设置类似，我们首先建立一个由人工标注的基准，结合来自可靠标注者的多次盲偏好判断。然后，我们会优化评分标准和偏好提示，以提升准确性以及潜在偏差，相关内容将在§4.2.2中提及。

To minimize the potential biases, training data are also optimized by several methods, such as multiple-blind consistent judgment, position-switched consistent judgment, etc. Once an optimal GenRM is trained, a Swiss Round scoring system is performed across the training dataset to determine the most suitable reference answer for RL training.

4.2.2. Addressing Bias of Generative Reward Models for Long CoT

Effective general RL for complex CoT reasoning tasks is critically dependent on accurate and unbiased reward models. Assessing such CoT responses turns out to be challenging, and we found that GenRMs preferred longer outputs over potentially superior concise alternatives, irrespective of actual reasoning quality. This **length bias** is a significant issue as it may substantially misguide RL policy optimization, incentivizing verbosity without substance and inducing reward hacking. Our initial efforts to improve GenRM fidelity include standard offline strategies: (1) Diversifying training data with a wide range of response lengths, sources, and quality tiers; (2) Incorporating adversarial examples to expose vulnerabilities; and (3) Refining model architectures. However, empirical analysis revealed that purely offline evaluation and preemptive mitigation of length bias in GenRMs frequently failed to prevent length bias during RL training.

Consequently, our core strategy incorporates continuous online monitoring of length bias during RL training. Specific metrics are established to detect whether the RL policy disproportionately extends output lengths to maximize GenRMs rewards without gains in task success or reasoning depth. Upon detecting such detrimental length-seeking behavior, indicative of exploiting GenRMs length bias, immediate GenRMs recalibration is triggered. This iterative adjustment is vital to preempt reward hacking related to output length, ensuring the policy prioritized substantive capability enhancement over superficial text inflation. Complementing this adaptive approach, RL-side techniques including reward shaping, value clipping, and normalization are systematically employed. These mechanisms desensitize reward signals to extreme values from superficial characteristics (e.g., length), thereby directing policy optimization toward substantive quality and correctness of its long CoT reasoning.

4.3. Curriculum of Incorporating Diverse Data

Given that our RL data spans a wide spectrum of categories, a core challenge is training a single policy capable of excelling on both reasoning-intensive tasks and general domain tasks. To address this, our approach entails a carefully managed curriculum and dynamic weighting strategy for reasoning and general-domain tasks during the RL training process with CISPO: we start with only the reasoning-intensive tasks with rule-based reward, and then gradually mix in the general domain tasks. This ensures that the model continues to refine its verifiable skills (e.g., in math and code) while progressively enhancing its performance on a diverse spectrum of general tasks, from complex instruction following to open-ended CoT reasoning. This mixed RL training encourages the model to learn context-dependent application of its reasoning abilities—applying rigorous, step-by-step deduction for verifiable problems and more flexible, adaptive generation for general queries—all within a unified policy framework. It prevents catastrophic forgetting of specialized skills while fostering broader generalization.

5. Extending RL Scaling to Longer Thinking

Our first RL training is performed with an output length limit of 40K tokens. Given that the hybrid architecture of M1 natively supports near-linear scaling for longer sequences, as demonstrated in Figure 1 (Right), we further extend the generation length during RL training to 80K tokens. This results in a new model, which we refer to as MiniMax-M1-80k.

为了最小化潜在偏差，训练数据还通过多种方法进行优化，例如多盲一致性判断、位置交换一致性判断等。一旦训练出最优的GenRM，就会在整个训练数据集上采用瑞士轮评分系统，以确定最适合用于RL训练的参考答案。

4.2.2. Addressing Bias of Generative Reward Models for Long CoT

对于复杂的CoT推理任务，有效的通用强化学习（RL）极大依赖于准确且无偏的奖励模型。评估此类CoT响应的难度很大，我们发现GenRMs倾向于偏好较长的输出，而非潜在更优的简洁替代品，无论实际推理质量如何。这种长度偏差是一个严重的问题，因为它可能严重误导RL策略优化，促使模型产生冗长无实质内容的回答，并引发奖励操控。我们最初为提高GenRM的准确性所采取的措施包括： (1) 使用多样化的训练数据，涵盖不同长度、来源和质量层次的响应； (2) 引入对抗样本以暴露模型的脆弱性； (3) 改进模型架构。然而，实证分析显示，纯粹的离线评估和提前缓解GenRMs中的长度偏差，常常无法在RL训练过程中防止长度偏差的出现。

因此，我们的核心策略包括在强化学习（RL）训练过程中持续在线监测长度偏差。建立了特定的指标，以检测RL策略是否过度延长输出长度，以最大化GenRMs的奖励，而没有在任务成功率或推理深度方面取得提升。一旦检测到这种有害的追求长度行为，表明其在利用GenRMs的长度偏差，就会立即触发GenRMs的重新校准。这种迭代调整对于防止与输出长度相关的奖励操控至关重要，确保策略优先提升实质性能力，而非表面上的文本膨胀。为了配合这种自适应方法，还系统性地采用包括奖励塑形、值裁剪和归一化在内的RL端技术。这些机制使奖励信号对表面特征（如长度）的极端值不敏感，从而引导策略优化朝着实质性质量和其长链推理的正确性方向发展。

4.3. 融合多样数据的课程

鉴于我们的强化学习数据涵盖了广泛的类别，一个核心挑战是训练一个能够在推理密集任务和通用领域任务上都表现出色的单一策略。为了解决这个问题，我们的方法包括在使用CISPO进行强化学习训练过程中，采用精心管理的课程安排和动态加权策略，针对推理任务和通用领域任务：我们首先只使用基于规则奖励的推理密集任务，然后逐步引入通用领域任务。这确保模型在不断完善其可验证技能（例如数学和编码）的同时，逐步提升在各种通用任务上的表现，从复杂的指令执行到开放式的链式推理。这种混合的强化学习训练鼓励模型学习其推理能力的上下文依赖应用——对可验证的问题采用严格的逐步推导，对一般性查询则采用更灵活、适应性强的生成——所有这些都在一个统一的策略框架内进行。它防止了专业技能的灾难性遗忘，同时促进了更广泛的泛化能力。

5. 将强化学习扩展到更长时间的思考

我们的第一次强化学习训练在输出长度限制为40K个标记的情况下进行。鉴于M1的混合架构本身支持近线性扩展以处理更长的序列，如图1（右）所示，我们在强化学习训练中将生成长度进一步扩展到80K个标记。这产生了一个新模型，我们称之为MiniMax-M1-80k。

Data. To efficiently train our RL model for an 80K output length, we utilize our previously trained 40K model to guide the data filtering process. First, we evaluate the pass rates on the curated dataset described in §4 and remove samples that are easily solved. We then adjust the data distribution to favor more challenging examples, such as difficult mathematical and coding problems. Additionally, we downsample synthetic reasoning data after observing that it destabilizes long-context RL training. Specifically, outputs generated from this data type often become repetitive and homogenous, and continued exposure to these patterns proves detrimental to the model's overall performance.

Length Scaling Strategy. To gradually increase the output length, we employ a staged window expansion RL strategy. We begin with an output length of 40K and incrementally expand it to 48K, 56K, 64K, 72K, and ultimately 80K. This staged approach ensures training stability at each step. The transition to a subsequent length is determined by a set of empirical indicators. These include the convergence of perplexity on the generated sequences and whether the 99th percentile of the output lengths is approaching the current context window limit. These signals offer valuable insights into the model's readiness for scaling, which allows us to maintain robust training throughout the process.

Addressing Training Instability During Scaling. During the scaling process, we encountered a critical issue in the later stages of training at each length window. Specifically, the model exhibited susceptibility to pattern collapse, where the latter portions of generated sequences degraded into incoherent or garbled text. This phenomenon consistently coincided with increased perplexity, indicating compromised generation quality and stability. We identify the root cause: during output length extension, negative samples increase in length substantially faster than positive samples, frequently reaching the context window limit earlier. Consequently, disproportionately large negative gradients accumulate in the latter segments of generation sequences. This imbalance originates from the inherently unequal nature of GRPO's advantage normalization and the token-level loss we adopt. To address this, we implement three key solutions: (1) Detecting repetitive patterns (consecutive high-probability tokens) with early stopping to prevent excessive context window consumption by repetitive responses; (2) Adopting combined sample-level loss and token-level normalization to alleviate negative-positive sample imbalance and mitigate adverse effects; (3) Decreasing both the gradient clipping threshold and ϵ_{high}^{IS} to further stabilize generation.

6. Evaluations

6.1. Core Benchmarks

We conduct a comprehensive evaluation of MiniMax-M1 across several key domains: mathematics, general coding, software engineering, reasoning & knowledge, long context, agentic tool use, factuality, and general assistant ability. We evaluate all tasks using temperature 1.0 and top-p 0.95 sampling.

- **Mathematics:** To evaluate mathematical reasoning capabilities, we utilize several competition level math benchmarks, including MATH-500 (Hendrycks et al., 2021), AIME 2024, AIME 2025. For AIME evaluation, we sample 32 times and compute the average passrate as the final score.
- **General Coding:** We assess general programming proficiency using LiveCodeBench (Jain et al., 2025) and FullStackBench (Liu et al., 2024), which evaluate code generation across diverse programming tasks. For both benchmarks, we report scores as the average passrate of 16 samples.
- **Reasoning & Knowledge:** We assess domain knowledge and reasoning capabilities through GPQA-Diamond (Rein et al., 2024), MMLU-Pro (Wang et al., 2024), and the challenging HLE benchmark (Phan et al., 2025). For GPQA-Diamond, we sample 32 times and report the average passrate. For HLE evaluation, we assess the model without external tools. Additionally, we

数据。为了高效训练我们的RL模型以生成80K长度的输出，我们利用之前训练的40K模型来指导数据筛选过程。首先，我们在第§4节描述的精选数据集上评估通过率，并移除那些容易解决的样本。然后，我们调整数据分布，偏向更具挑战性的示例，例如困难的数学和编码问题。此外，在观察到合成推理数据会导致长上下文RL训练不稳定后，我们对其进行抽样降低。具体来说，从这种数据类型生成的输出常常变得重复且同质，持续暴露于这些模式中会对模型的整体性能产生不利影响。

长度缩放策略。为了逐步增加输出长度，我们采用分阶段的窗口扩展强化学习策略。我们从40K的输出长度开始，逐步扩展到48K、56K、64K、72K，最终达到80K。这种分阶段的方法确保了每一步的训练稳定性。向后续长度的过渡由一组经验指标决定。这些指标包括生成序列的困惑度收敛情况，以及输出长度的第99百分位是否接近当前的上下文窗口限制。这些信号为模型的扩展准备提供了宝贵的见解，使我们能够在整个过程中保持稳健的训练。

解决扩展过程中训练不稳定的问题。在扩展过程中，我们在每个长度窗口的后期训练中遇到了一个关键问题。具体而言，模型表现出对模式崩溃的易感性，即生成序列的后段退化为不连贯或乱码的文本。这一现象与困惑度的增加一致，表明生成质量和稳定性受到影响。我们识别出根本原因：在输出长度扩展过程中，负样本的长度增长远快于正样本，常常提前达到上下文窗口的限制。因此，在生成序列的后段会积累大量不成比例的负梯度。这种不平衡源自于GRPO优势归一化的固有不平等以及我们采用的基于标记的损失。为了解决这一问题，我们实施了三项关键措施：(1) 通过检测重复模式（连续高概率标记）并提前停止，防止重复响应过度占用上下文窗口；(2) 采用结合样本级损失和标记级归一化的方法，缓解正负样本不平衡，减轻不良影响；(3) 降低梯度裁剪阈值和ϵ_{high}^{IS}，以进一步稳定生成。

6. 评估

6.1. 核心基准测试

W我们对MiniMax-M1在多个关键领域进行了全面评估：数学，
g一般编码、软件工程、推理与知识、长上下文、代理工具使用、事实性，
a我们评估所有任务时，使用温度1.0和top-p 0.95采样，以衡量一般助手能力。

- 数学：为了评估数学推理能力，我们使用了多个竞赛级别的数学基准，包括 MATH-500（Hendrycks 等，2021）、AIME 2024 和 AIME 2025。对于 AIME 评估，我们抽样 32 次并计算平均通过率作为最终得分。
- 一般编码：我们使用 LiveCodeBench（Jain 等，2025）和 FullStackBench（Liu 等，2024）评估一般编程能力，这些基准测试评估在各种编程任务中的代码生成。对于这两个基准，我们都以 16 个样本的平均通过率作为得分。
- 推理与知识：我们通过 GPQA-Diamond（Rein 等，2024）、MMLU-Pro（Wang 等，2024）以及具有挑战性的 HLE 基准（Phan 等，2025）来评估领域知识和推理能力。对于 GPQA-Diamond，我们采样 32 次并报告平均通过率。对于 HLE 评估，我们在没有外部工具的情况下评估模型。此外，我们

Table 2 | **Performance of MiniMax-M1 on core benchmarks.**

Tasks	**Leading Close-Weights Models**				**Open-Weights Models**			**Our Models**	
	OpenAI-o3	**Gemini 2.5 Pro (06-05)**	**Claude 4 Opus**	**Seed-Thinking-v1.5**	**DeepSeek-R1**	**DeepSeek-R1-0528**	**Qwen3-235B-A22B**	**MiniMax-M1-40k**	**MiniMax-M1-80k**
Extended Thinking	*100K*	*64K*	*64K*	*32K*	*32K*	*64K*	*32K*	*40K*	*80K*
Mathematics									
AIME 2024	91.6	92.0	76.0	86.7	79.8	91.4	85.7	83.3	86.0
AIME 2025	88.9	88.0	75.5	74.0	70.0	87.5	81.5	74.6	76.9
MATH-500	98.1	98.8	98.2	96.7	97.3	98.0	96.2	96.0	96.8
General Coding									
LiveCodeBench *(24/8~25/5)*	75.8	77.1	56.6	67.5	55.9	73.1	65.9	62.3	65.0
FullStackBench	69.3	–	70.3	69.9	70.1	69.4	62.9	67.6	68.3
Reasoning & Knowledge									
GPQA Diamond	83.3	86.4	79.6	77.3	71.5	81.0	71.1	69.2	70.0
HLE *(no tools)*	20.3	21.6	10.7	8.2	8.6*	17.7*	7.6*	7.2*	8.4*
ZebraLogic	95.8	91.6	95.1	84.4	78.7	95.1	80.3	80.1	86.8
MMLU-Pro	85.0	86.0	85.0	87.0	84.0	85.0	83.0	80.6	81.1
Software Engineering									
SWE-bench Verified	69.1	67.2	72.5	47.0	49.2	57.6	34.4	55.6	56.0
Long Context									
OpenAI-MRCR *(128k)*	56.5	76.8	48.9	54.3	35.8	51.5	27.7	76.1	73.4
OpenAI-MRCR *(1M)*	–	58.8	–	–	–	–	–	58.6	56.2
LongBench-v2	58.8	65.0	55.6	52.5	58.3	52.1	50.1	61.0	61.5
Agentic Tool Use									
TAU-bench *(airline)*	52.0	50.0	59.6	44.0	–	53.5	34.7	60.0	62.0
TAU-bench *(retail)*	73.9	67.0	81.4	55.7	–	63.9	58.6	67.8	63.5
Factuality									
SimpleQA	49.4	54.0	–	12.9	30.1	27.8	11.0	17.9	18.5
General Assistant									
MultiChallenge	56.5	51.8	45.8	43.0	40.7	45.0	40.0	44.7	44.7

* conducted on the text-only HLE subset.

measure logical reasoning ability using ZebraLogic (Lin et al., 2025).

- **Software Engineering:** We evaluate software engineering capabilities using SWE-bench Verified (Jimenez et al., 2024), which measures the ability to resolve real-world GitHub issues. We report results derived from the Agentless scaffold (Xia et al., 2024). Departing from the original pipeline, our methodology employs a two-stage localization process (without any embedding-based retrieval mechanisms): initial coarse-grained file localization followed by fine-grained localization to specific files and code elements.
- **Long Context:** We evaluate long context understanding using OpenAI-MRCR (OpenAI, 2024b), which tests retrieval and disambiguation of multiple similar items within extended contexts, and LongBench-v2 (Bai et al., 2024), a challenging benchmark with 503 multiple-choice questions

表2 | MiniMax-M1 在核心基准测试中的性能。

Tasks	**Leading Close-Weights Models**				**Open-Weights Models**			**Our Models**	
	OpenAI-o3	**Gemini 2.5 Pro (06-05)**	**Claude 4 Opus**	**Seed-Thinking-v1.5**	**DeepSeek-R1**	**DeepSeek-R1-0528**	**Qwen3-235B-A22B**	**MiniMax-M1-40k**	**MiniMax-M1-80k**
Extended Thinking	*100K*	*64K*	*64K*	*32K*	*32K*	*64K*	*32K*	*40K*	*80K*
Mathematics									
AIME 2024	91.6	92.0	76.0	86.7	79.8	91.4	85.7	83.3	86.0
AIME 2025	88.9	88.0	75.5	74.0	70.0	87.5	81.5	74.6	76.9
MATH-500	98.1	98.8	98.2	96.7	97.3	98.0	96.2	96.0	96.8
General Coding									
LiveCodeBench *(24/8~25/5)*	75.8	77.1	56.6	67.5	55.9	73.1	65.9	62.3	65.0
FullStackBench	69.3	–	70.3	69.9	70.1	69.4	62.9	67.6	68.3
Reasoning & Knowledge									
GPQA Diamond	83.3	86.4	79.6	77.3	71.5	81.0	71.1	69.2	70.0
HLE *(no tools)*	20.3	21.6	10.7	8.2	8.6*	17.7*	7.6*	7.2*	8.4*
ZebraLogic	95.8	91.6	95.1	84.4	78.7	95.1	80.3	80.1	86.8
MMLU-Pro	85.0	86.0	85.0	87.0	84.0	85.0	83.0	80.6	81.1
Software Engineering									
SWE-bench Verified	69.1	67.2	72.5	47.0	49.2	57.6	34.4	55.6	56.0
Long Context									
OpenAI-MRCR *(128k)*	56.5	76.8	48.9	54.3	35.8	51.5	27.7	76.1	73.4
OpenAI-MRCR *(1M)*	–	58.8	–	–	–	–	–	58.6	56.2
LongBench-v2	58.8	65.0	55.6	52.5	58.3	52.1	50.1	61.0	61.5
Agentic Tool Use									
TAU-bench *(airline)*	52.0	50.0	59.6	44.0	–	53.5	34.7	60.0	62.0
TAU-bench *(retail)*	73.9	67.0	81.4	55.7	–	63.9	58.6	67.8	63.5
Factuality									
SimpleQA	49.4	54.0	–	12.9	30.1	27.8	11.0	17.9	18.5
General Assistant									
MultiChallenge	56.5	51.8	45.8	43.0	40.7	45.0	40.0	44.7	44.7

* 在仅文本的HLE子集上进行。

使用 ZebraLogic (Lin et al., 2025) 测量逻辑推理能力。

- 软件工程：我们使用经过验证的SWE-bench（Jimenez 等，2024）评估软件工程能力，该工具衡量解决实际GitHub问题的能力。我们报告基于无代理脚手架（Xia 等，2024）得出的结果。不同于原始流程，我们的方法采用两阶段定位过程（不使用任何基于嵌入的检索机制）：首先进行粗粒度的文件定位，然后进行细粒度的定位，锁定到特定文件和代码元素。

- 长上下文：我们使用OpenAI-MRCR（OpenAI，2024b）评估长上下文理解能力，该方法测试在扩展上下文中检索和消歧多个相似项的能力，以及LongBench-v2（Bai等，2024），这是一个包含503个多项选择题的具有挑战性的基准测试

across contexts ranging from 8k to 2M words.

- **Agentic Tool Use:** We assess tool use capabilities through TAU-bench (Yao et al., 2025), which emulates dynamic conversations where agents must utilize API tools while adhering to domain-specific policy guidelines. We evaluate TAU-bench with GPT-4.1 as user model, a general system prompt[2] and without any custom tools. The maximum number of interaction steps is 40.
- **Factuality:** To measure factuality of LLMs, we utilize SimpleQA (Wei et al., 2024), an adversarially-collected benchmark of fact-seeking questions with single, indisputable answers.
- **General Assistant:** We evaluate general assistant capabilities using MultiChallenge (Sirdeshmukh et al., 2025), which assesses LLMs on conducting realistic multi-turn conversations with human users. We report our scores judged by GPT-4o.

Results on Math, Coding, and other General Tasks. Table 2 presents our model's performance compared to state-of-the-art large reasoning models. In mathematical reasoning, the MiniMax-M1 models demonstrate strong performance across multiple benchmarks, achieving results comparable to the close-weight model Seed-Thinking-v1.5 (Seed et al., 2025). Notably, MiniMax-M1-80k achieves 86.0% on AIME 2024, placing it second among open-weight models and trailing only the latest DeepSeek-R1-0528 model. For general coding, MiniMax-M1-80k matches Qwen3-235B on LiveCodeBench while outperforming it on FullStackBench, demonstrating robust capabilities among leading open-weight models. On reasoning & knowledge benchmarks, MiniMax-M1-80k similarly trails DeepSeek-R1-0528 but achieves competitive performance against other top open-weight models. On the factuality benchmark SimpleQA, Minimax-M1 models underperform DeepSeek-R1 while outperforming all other open-weight models and Seed-Thinking-v1.5. On MultiChallenge, both MiniMax models perform comparably to DeepSeek-R1-0528 and Claude 4 Optus, with inferior results only to o3 and Gemini-2.5-Pro.

Highlights in Complex Scenarios: Software Engineering, Long Context, and Tool use. Benefiting from our execution-based, software engineering environments during RL, MiniMax-M1-40k and MiniMax-M1-80k achieve strong scores of 55.6% and 56.0% on SWE-bench verified respectively. These results are slightly inferior to DeepSeek-R1-0528's 57.6% and significantly surpass other open-weights models. Leveraging its 1M context window, the M1 models significantly outperform all other open-weight models in long-context understanding. They even surpass OpenAI o3 and Claude 4 Opus, ranking second globally and trailing only Gemini 2.5 Pro by a small margin. In agentic tool-use scenarios (TAU-bench), MiniMax-M1-40k surpasses all open-weight models and even Gemini-2.5 Pro. Moreover, MiniMax-M1-80k consistently outperforms MiniMax-M1-40k across most benchmarks, confirming the benefits of scaling test-time compute.

6.2. Effect of RL Scaling

To investigate the effect of RL scaling, we track performance and response length throughout training. Figure 4 presents three representative examples from AIME 2024, AIME 2025, and LiveCodeBench v5, respectively. We observe consistent improvements in both model performance and response length during training. Notably, average response lengths on AIME and LiveCodeBench exceed 20,000 tokens, with AIME 2024 accuracy showing substantial gains from 68% to 80%. Crucially, the strong correlation between accuracy gains and increased response length in these visualizations underscores the importance of extending RL scaling to facilitate more extensive reasoning processes.

[2] "In each round, you need to carefully examine the tools provided to you to determine if any can be used. You must adhere to all of the policies. Pay attention to the details in the terms. Solutions for most situations can be found within these policies."

涵盖从8千到2百万词的各种语境。

- 主动工具使用：我们通过 TAU-bench（Yao 等，2025）评估工具使用能力，该基准模拟动态对话，代理在遵守特定领域政策指南的同时必须利用 API 工具。我们使用 GPT-4.1 作为用户模型、一个通用系统提示[2]，并且不使用任何自定义工具来评估 TAU-bench。最大交互步骤数为 40。
- 事实性：为了衡量大型语言模型（LLMs）的事实性，我们采用 SimpleQA（Wei 等人，2024），这是一个通过对抗方式收集的事实查询问题基准，包含单一且无可争辩的答案。
- 通用助手：我们使用 MultiChallenge（Sirdeshmukh 等，2025）评估通用助手的能力，该评估测试大规模语言模型（LLMs）与人类用户进行逼真的多轮对话的能力。我们报告由 GPT-4o 评判的分数。

在数学、编码和其他通用任务上的表现。表2展示了我们的模型与最先进的大型推理模型的性能对比。在数学推理方面，MiniMax-M1模型在多个基准测试中表现出色，取得了与接近权重模型Seed-Thinking-v1.5（Seed等，2025）相当的结果。值得注意的是，MiniMax-M1-80k在AIME 2024上的得分为86.0%，在开源模型中排名第二，仅次于最新的DeepSeek-R1-0528模型。在通用编码方面，MiniMax-M1-80k在LiveCodeBench上与Qwen3-235B持平，而在FullStackBench上优于它，展示了在领先的开源模型中的强大能力。在推理与知识基准测试中，MiniMax-M1-80k同样略逊于DeepSeek-R1-0528，但在与其他顶级开源模型的竞争中表现出色。在事实性基准SimpleQA上，Minimax-M1模型的表现不及DeepSeek-R1，但优于所有其他开源模型和Seed-Thinking-v1.5。在MultiChallenge测试中，两个MiniMax模型的表现与DeepSeek-R1-0528和Claude 4 Optus相当，只有o3和Gemini-2.5-Pro的结果略优。

复杂场景中的亮点：软件工程、长上下文和工具使用。在强化学习期间，受益于我们基于执行的软件工程环境，MiniMax-M1-40k 和 MiniMax-M1-80k 在 SWE-bench 上的验证得分分别达到了55.6%和56.0%，表现强劲。这些结果略逊于 DeepSeek-R1-0528 的57.6%，但显著优于其他开源模型。利用其1M的上下文窗口，M1模型在长上下文理解方面显著优于所有其他开源模型。它们甚至超越了OpenAI o3和Claude 4 Opus，在全球排名第二，仅次于Gemini 2.5 Pro，差距很小。在代理工具使用场景（TAU-bench）中，MiniMax-M1-40k 超过了所有开源模型，甚至超过了Gemini-2.5 Pro。此外，MiniMax-M1-80k在大多数基准测试中持续优于MiniMax-M1-40k，验证了扩展测试时计算能力的优势。

6.2. RL 缩放的影响

为了研究RL缩放的效果，我们在整个训练过程中跟踪性能和响应长度。图4展示了来自AIME 2024、AIME 2025和LiveCodeBench v5的三个具有代表性的示例。我们观察到在训练过程中，模型性能和响应长度都持续改善。值得注意的是，AIME和LiveCodeBench上的平均响应长度都超过了20,000个标记，而AIME 2024的准确率从68%显著提升到80%。关键是，这些可视化中准确率的提升与响应长度的增加之间的强相关性，强调了扩展RL缩放以促进更广泛推理过程的重要性。

[2] "In each round, you need to carefully examine the tools provided to you to determine if any can be used. You must adhere to all of the policies. Pay attention to the details in the terms. Solutions for most situations can be found within these policies."

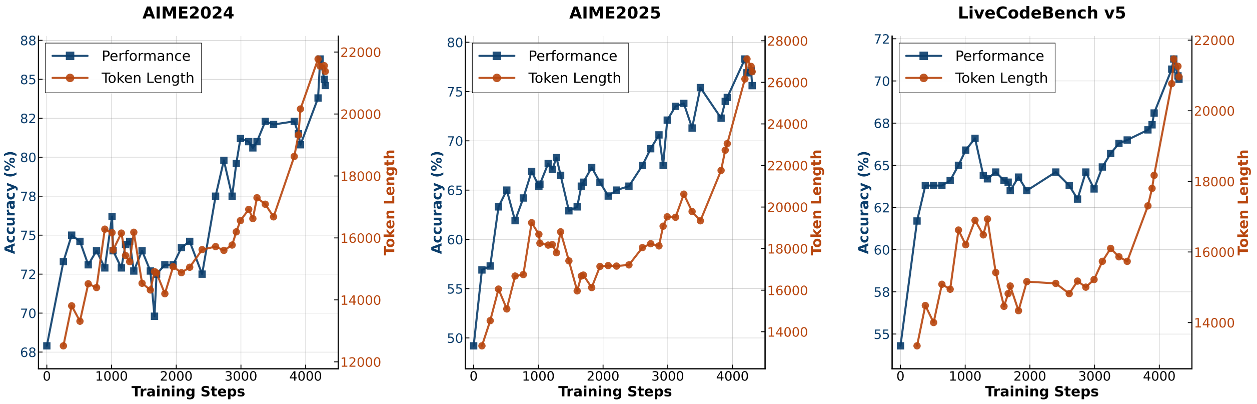

Figure 4 | Accuracy and generation length versus RL training steps for MiniMax-M1.

7. Conclusion and Future work

In this work, we introduce and release MiniMax-M1, the world's first open-weight, large-scale reasoning model featuring a lightning attention mechanism. This efficient attention design enables MiniMax-M1 to natively support inputs of up to 1M tokens and generation lengths of 80K tokens—both significantly exceeding capabilities of other open-weight models. These capabilities render MiniMax-M1 uniquely suited for complex, realistic scenarios requiring long context and extended reasoning, properties empirically validated by its strong performance on software engineering, agentic tool use, and long-context understanding benchmarks. Beyond the inherent efficiency advantages of lightning attention for RL training, this work contributes a novel RL algorithm, CISPO, to accelerate training. Combining architectural advantages with CISPO, we efficiently trained MiniMax-M1, with complete RL training completed in three weeks using 512 H800 GPUs. Across comprehensive evaluations, MiniMax-M1 ranks among the world's best open-weight models alongside DeepSeek-R1 and Qwen3-235B.

Looking forward, as test-time compute continuously scales to power increasingly complex scenarios, we foresee significant potential for such efficient architectures in addressing real-world challenges. These include automating company workflows (Xu et al., 2025) and conducting scientific research (OpenAI, 2025; Si et al., 2024). Real-world applications particularly demand LRMs that function as agents interacting with environments, tools, computers, or other agents—requiring reasoning across dozens to hundreds of turns while integrating long-context information from diverse sources. We envision MiniMax-M1 serving as a strong foundation for such applications with unique advantages, and we are fully dedicated to further evolving MiniMax-M1 toward this goal.

References

Anthropic. Claude 3.7 sonnet and claude code. https://www.anthropic.com/news/claude-3-7-sonnet, 2025. Blog post, February 24, 2025.

Simran Arora, Sabri Eyuboglu, Michael Zhang, Aman Timalsina, Silas Alberti, Dylan Zinsley, James Zou, Atri Rudra, and Ré. Simple linear attention language models balance the recall-throughput tradeoff. *arXiv preprint arXiv:2402.18668*, 2024.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. LongBench. *arXiv preprint arXiv:2412.15204*, 2024.

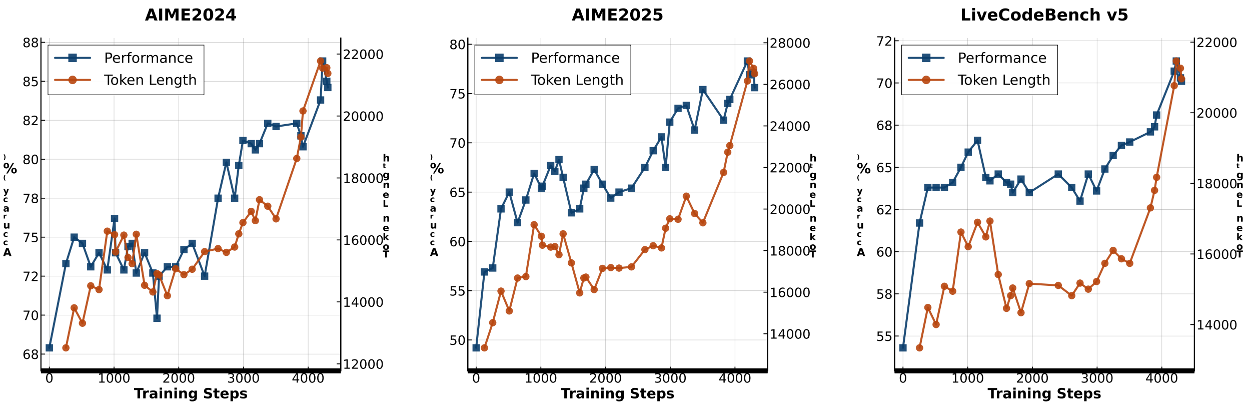

图4 | MiniMax-M1的准确率和生成长度与RL训练步骤的关系。

7. 结论与未来工作

在本工作中，我们引入并发布了MiniMax-M1，世界上首个具有闪电注意力机制的开源大规模推理模型。这种高效的注意力设计使得MiniMax-M1能够原生支持高达1M个令牌的输入和80K个令牌的生成长度——这两个指标都显著超出其他开源模型的能力。这些能力使得MiniMax-M1在需要长上下文和扩展推理的复杂、真实场景中具有独特优势，其在软件工程、代理工具使用和长上下文理解基准测试中的出色表现验证了这一点。除了闪电注意力在强化学习训练中的固有效率优势外，本工作还贡献了一种新颖的强化学习算法CISPO，以加快训练过程。结合架构优势与CISPO，我们高效地训练了MiniMax-M1，在使用512个H800 GPU的情况下，三周内完成了全部强化学习训练。在全面评估中，MiniMax-M1与DeepSeek-R1和Qwen3-235B一同跻身世界顶尖的开源模型行列。

展望未来，随着测试时的计算能力不断提升以应对日益复杂的场景，我们预见到这种高效架构在解决实际问题方面具有巨大潜力。这些应用包括自动化公司工作流程（Xu 等，2025）和进行科学研究（OpenAI，2025；Si 等，2024）。实际应用特别需要作为代理与环境、工具、计算机或其他代理交互的LRMs——这要求在数十到数百轮的推理中，整合来自多源的长上下文信息。我们设想MiniMax-M1将作为此类应用的坚实基础，具有独特优势，我们也将全力以赴，进一步发展MiniMax-M1以实现这一目标。

参考文献

Anthropic。Claude 3.7 十四行诗与 Claude 代码。`https://www.anthropic.com/news/claude-3-7-sonnet`，2025年。博客文章，2025年2月24日。

SImran Arora、Sabri Eyuboglu、Michael Zhang、Aman Timalsina、Silas Alberti、Dylan Zinsley、James Zou、Atri Rudra 和 Ré。简单线性注意力语言模型平衡了召回率与吞吐量的权衡。*arXiv preprint arXiv:2402.18668*，2024。

Y牛市白、屈青图、张佳杰、彭浩、王晓智、吕鑫、曹树林、许嘉正、侯磊、董玉晓、唐杰、李娟子。LongBench。*arXiv preprint arXiv:2412.15204*，2024。

Ali Behrouz, Peilin Zhong, and Vahab Mirrokni. Titans: Learning to memorize at test time. *arXiv preprint arXiv:2501.00663*, 2024.

Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

Krzysztof Marcin Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Quincy Davis, Afroz Mohiuddin, Lukasz Kaiser, David Benjamin Belanger, Lucy J Colwell, and Adrian Weller. Rethinking attention with Performers. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=Ua6zuk0WRH.

Yuhong Chou, Man Yao, Kexin Wang, Yuqi Pan, Rui-Jie Zhu, Jibin Wu, Yiran Zhong, Yu Qiao, Bo Xu, and Guoqi Li. Metala: Unified optimal linear approximation to softmax attention map. *Advances in Neural Information Processing Systems*, 37:71034–71067, 2024.

Junyoung Chung and Ç. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, Zhiyuan Liu, Hao Peng, Lei Bai, Wanli Ouyang, Yu Cheng, Bowen Zhou, and Ning Ding. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*, 2025.

Tri Dao and Albert Gu. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality. *arXiv preprint arXiv:2405.21060*, 2024.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Jusen Du, Weigao Sun, Disen Lan, Jiaxi Hu, and Yu Cheng. Mom: Linear sequence modeling with mixture-of-memories. *arXiv preprint arXiv:2502.13685*, 2025.

Paolo Glorioso, Quentin Anthony, Yury Tokpanov, James Whittington, Jonathan Pilault, Adam Ibrahim, and Beren Millidge. Zamba: A compact 7b SSM. *arXiv preprint arXiv:2405.16712*, 2024.

Google DeepMind. Gemini pro. https://deepmind.google/models/gemini/pro/, 2025. Web page, accessed 2025.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=tEYskw1VY2.

Albert Gu, Tri Dao, Stefano Ermon, Atri Rudra, and Christopher Ré. Hippo: Recurrent memory with optimal polynomial projections. *Advances in neural information processing systems*, 33:1474–1487, 2020.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=uYLFoz1vlAC.

艾利·贝鲁兹、钟佩琳和米拉克尼·瓦哈布。泰坦：学习在测试时记忆。*arXiv preprint arXiv:2501.00663*，2024年。伊兹·贝尔塔吉、马修·E·彼得斯和阿尔曼·科汉。Longformer：长文档变换器。*arXiv preprint arXiv:2004.05150*，2020年。克日什托夫·马尔钦·乔罗曼斯基、瓦列里·利科舍尔斯托夫、戴维·多汉、宋星友、安德雷娅·加恩、塔马斯·萨洛斯、彼得·霍金斯、贾里德·昆西·戴维斯、阿夫鲁兹·莫希丁、卢卡什·凯撒、戴维·本杰明·贝兰热、露西·J·科韦尔和阿德里安·韦勒。用Performers重新思考注意力机制。在*International Conference on Learning Representations*，2021年。网址 https://openreview.net/forum?id= Ua6zuk0WRH。周宇宏、姚曼、王可欣、潘宇奇、朱瑞杰、吴吉斌、钟逸然、乔宇、徐博和李国奇。Metala：对softmax注意力图的统一最优线性逼近。*Advances in Neural Information Processing Systems*，37:71034–71067，2024年。

Junyoung Chung 和 Ç. 关于门控循环神经网络在序列建模中的经验评估。*arXiv preprint arXiv:1412.3555*，2014。

甘曲崔、张宇辰、陈嘉成、袁立凡、王志、左宇新、李浩展、范宇辰、陈华钰、陈维泽、刘志远、彭浩、白磊、欧阳万里、程宇、周博文、丁宁。强化学习的熵机制用于推理语言模型。*arXiv preprint arXiv:2505.22617*，2025年。Tri Dao 和 Albert Gu。Transformers 是 ssms：通过结构化状态空间对偶实现的广义模型和高效算法。*arXiv preprint arXiv:2405.21060*，2024年。DeepSeek-AI，郭达雅、杨德建、张浩伟、宋俊孝、张若瑜、徐润新、朱启浩、马世荣、王佩怡等。Deepseek-r1：通过强化学习激励LLMs的推理能力。*arXiv preprint arXiv:2501.12948*，2025年。杜俊森、孙伟高、蓝迪森、胡佳熙、程宇。Mom：基于记忆混合的线性序列建模。*arXiv preprint arXiv:2502.13685*，2025年。Paolo Glorioso、Quentin Anthony、Yury Tokpanov、James Whittington、Jonathan Pilault、Adam Ibrahim 和 Beren Millidge。Zamba：一个紧凑的7b SSM。*arXiv preprint arXiv:2405.16712*，2024年。谷歌 DeepMind。Gemini pro。https://deepmind.google/models/gemini/pro/，2025年。网页，访问于2025年。Albert Gu 和 Tri Dao。Mamba：通过选择性状态空间实现线性时间序列建模。在*First Conference on Language Modeling*，2024年。网址 https://openreview.net/forum?id= tEYskw1VY2。Albert Gu、Tri Dao、Stefano Ermon、Atri Rudra 和 Christopher Ré。Hippo：具有最优多项式投影的循环记忆。*Advances in neural information processing systems*，2020年第33卷第1474-1487页。Albert Gu、Karan Goel 和 Christopher Ré。高效建模长序列的结构化状态空间。在*The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*。OpenReview.net，2022年。网址 https://openreview.net/forum?id= uYLFoz1vlAC。

Albert Gu, Isys Johnson, Aman Timalsina, Atri Rudra, and Christopher Re. How to train your HIPPO: State space models with generalized orthogonal basis projections. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=klK17OQ3KB.

Ankit Gupta, Albert Gu, and Jonathan Berant. Diagonal state spaces are as effective as structured state spaces. In *NeurIPS*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/9156b0f6dfa9bbd18c79cc459ef5d61c-Abstract-Conference.html.

Zhihao He, Hang Yu, Zi Gong, Shizhan Liu, Jianguo Li, and Weiyao Lin. Rodimus*: Breaking the accuracy-efficiency trade-off with efficient attentions. *arXiv preprint arXiv:2410.06577*, 2024.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model. *arXiv preprint arXiv:2503.24290*, 2025.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. In *The Thirteenth International Conference on Learning Representations*, 2025.

Jamba Team. Jamba-1.5: Hybrid T. *arXiv preprint arXiv:2408.12570*, 2024.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-bench: Can language models resolve real-world github issues? In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=VTF8yNQM66.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning*, pages 5156–5165. PMLR, 2020.

Kimi Team. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.

Bill Yuchen Lin, Ronan Le Bras, Kyle Richardson, Ashish Sabharwal, Radha Poovendran, Peter Clark, and Yejin Choi. Zebralogic: On the scaling limits of llms for logical reasoning. *arXiv preprint arXiv:2502.01100*, 2025.

Junteng Liu, Yuanxiang Fan, Zhuo Jiang, Han Ding, Yongyi Hu, Chi Zhang, Yiqi Shi, Shitong Weng, Aili Chen, Shiqi Chen, Yunan Huang, Mozhi Zhang, Pengyu Zhao, Junjie Yan, and Junxian He. Synlogic: Synthesizing verifiable reasoning data at scale for learning logical reasoning and beyond. *arXiv preprint arXiv:2505.19641*, 2025a.

Siyao Liu, He Zhu, Jerry Liu, Shulin Xin, Aoyan Li, Rui Long, Li Chen, Jack Yang, Jinxiang Xia, Z. Y. Peng, Shukai Liu, Zhaoxiang Zhang, Ge Zhang, Wenhao Huang, Kai Shen, and Liang Xiang. Fullstack bench: Evaluating llms as full stack coders. *arXiv preprint arXiv:2412.00535*, 2024.

阿尔伯特·顾、伊斯斯·约翰逊、阿曼·蒂马尔西纳、特里·鲁德拉和克里斯托弗·雷。如何训练你的HIPPO：具有广义正交基投影的状态空间模型。在 *International Conference on Learning Representations*，2023。网址 `https://openreview.net/forum?id=klK17OQ3KB`。安奇特·古普塔、阿尔伯特·顾和乔纳森·贝兰特。对角线状态空间与结构化状态空间一样有效。在 *NeurIPS*，2022。网址 `http://papers.nips.cc/paper_files/paper/2022/hash/9156b0f6dfa9bbd18c79cc459ef5d61c-Abstract-Conference.html`。何志豪、余航、龚子、刘世展、李建国和林伟耀。Rodimus*：用高效注意力打破准确性与效率的权衡。*arXiv preprint arXiv:2410.06577*，2024。丹·亨德里克斯、科林·伯恩斯、索拉夫·卡达瓦斯、阿库尔·阿罗拉、史蒂文·巴萨特、埃里克·唐、宋晓东和雅各布·斯坦哈特。用数学数据集衡量数学问题解决能力。*arXiv preprint arXiv:2103.03874*，2021。赫尔穆特·霍赫赖特和尤尔根·施米德胡伯。长短期记忆。*Neural computation*，1997年第9卷第8期：1735–1780。胡景成、张银敏、韩奇、江大新、张翔宇和沈香耀。Open-reasoner-zero：一种在基础模型上扩展强化学习的开源方法。*arXiv preprint arXiv:2503.24290*，2025。纳曼·贾因、韩金、古斯·古、李文丁、闫凡佳、张天俊、王思达、阿曼多·索拉尔-莱萨玛、塞尼克·塞尔、伊恩·斯托伊卡。Livecodebench：用于代码的大型语言模型的整体和无污染评估。在 *The Thirteenth International Conference on Learning Representations*，2025。Jamba团队。Jamba-1.5：混合T。*arXiv preprint arXiv:2408.12570*，2024。卡洛斯·E·希门尼斯、约翰·杨、亚历山大·韦蒂格、姚顺宇、裴克新、奥菲尔·普雷斯和卡尔蒂克·纳拉西姆汉。SWE-bench：语言模型能解决真实世界的GitHub问题吗？在 *International Conference on Learning Representations*，2024。网址 `https://openreview.net/forum?id=VTF8yNQM66`。安吉洛斯·卡萨罗普洛斯、阿普尔夫·维亚斯、尼古拉斯·帕帕斯和弗朗索瓦·弗勒雷。变换器是RNN：具有线性注意力的快速自回归变换器。在 *International Conference on Machine Learning*，第5156–5165页。PMLR，2020。Kimi团队。Kimi k1.5：用LLMs扩展强化学习。*arXiv preprint arXiv:2501.12599*，2025。比尔·余辰·林、罗南·勒布拉斯、凯尔·理查森、阿希什·萨巴尔瓦尔、拉达·普文德兰、彼得·克拉克和邹艺津。Zebralogic：关于LLMs在逻辑推理中的扩展极限。*arXiv preprint arXiv:2502.01100*，2025。刘俊腾、范远翔、江卓、丁汉、胡永义、张驰、史一奇、翁世通、陈爱丽、陈诗奇、黄云楠、张墨之、赵鹏宇、闫俊杰和何俊贤。Synlogic：大规模合成可验证推理数据，用于学习逻辑推理及其扩展。*arXiv preprint arXiv:2505.19641*，2025a。刘思尧、朱赫、刘杰、辛书林、李奥彦、隆瑞、陈力、杨杰、夏金祥、彭子阳、刘书凯、张昭翔、张戈、黄文浩、沈凯、向亮。Fullstack bench：评估LLMs作为全栈编码器。*arXiv preprint arXiv:2412.00535*，2024。

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Enzhe Lu, Zhejun Jiang, Jingyuan Liu, Yulun Du, Tao Jiang, Chao Hong, Shaowei Liu, Weiran He, Enming Yuan, Yuzhi Wang, et al. Moba: Mixture of block attention for long-context llms. *arXiv preprint arXiv:2502.13189*, 2025.

Eric Martin and Chris Cundy. Parallelizing linear recurrent neural nets over sequence length. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=HyUNwulC-.

MiniMax, Aonian Li, Bangwei Gong, Bo Yang, Boji Shan, Chang Liu, Cheng Zhu, Chunhao Zhang, Congchao Guo, Da Chen, Dong Li, et al. Minimax-01: Scaling foundation models with lightning attention. *arXiv preprint arXiv:2501.08313*, 2025.

Igor Molybog, Peter Albert, Moya Chen, Zachary DeVito, David Esiobu, Naman Goyal, Punit Singh Koura, Sharan Narang, Andrew Poulton, Ruan Silva, Binh Tang, Diana Liskovich, Puxin Xu, Yuchen Zhang, Melanie Kambadur, Stephen Roller, and Susan Zhang. A theory on adam instability in large-scale machine learning. *arXiv preprint arXiv:2304.09871*, 2023.

OpenAI. Introducing openai o1. https://openai.com/o1/, 2024a. Web page, accessed 2024.

OpenAI. Openai mrcr dataset. https://huggingface.co/datasets/openai/mrcr, 2024b. Accessed: 2025-06-15.

OpenAI. Introducing deep research, 2025. URL https://openai.com/index/introducing-deep-research/.

Bo Peng, Eric Alcaide, Quentin Gregory Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Nguyen Chung, Leon Derczynski, et al. Rwkv: Reinventing rnns for the transformer era. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

Bo Peng, Daniel Goldstein, Quentin Anthony, Alon Albalak, Eric Alcaide, Stella Biderman, Eugene Cheah, Teddy Ferdinan, Haowen Hou, and Przemysł Kazienko. Eagle and finch: Rwkv with matrix-valued states and dynamic recurrence. *arXiv preprint arXiv:2404.05892*, 2024a.

Bo Peng, Daniel Goldstein, Quentin Anthony, Alon Albalak, Eric Alcaide, Stella Biderman, Eugene Cheah, Teddy Ferdinan, Haowen Hou, and Przemysł Kazienko. Eagle and finch: Rwkv with matrix-valued states and dynamic recurrence. *arXiv preprint arXiv:2404.05892*, 2024b.

Bo Peng, Ruichong Zhang, Daniel Goldstein, Eric Alcaide, Xingjian Du, Haowen Hou, Jiaju Lin, Jiaxing Liu, Janna Lu, William Merrill, et al. Rwkv-7. *arXiv preprint arXiv:2503.14456*, 2025.

Hao Peng, Nikolaos Pappas, Dani Yogatama, Roy Schwartz, Noah Smith, and Lingpeng Kong. Random feature attention. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=QtTKTdVrFBB.

刘子辰，陈长宇，李文俊，齐鹏辉，庞天宇，杜超，李伟顺，林敏。理解 r1-zero 类训练：一个关键视角。*arXiv preprint arXiv:2503.20783*，2025b。伊利亚·洛什奇洛夫和弗兰克·胡特尔。解耦权重衰减正则化。在 *International Conference on Learning Representations*，2019。卢恩哲，姜哲俊，刘靖远，杜玉伦，姜涛，洪超，刘少伟，何伟然，袁恩明，王玉芝等。Moba：用于长上下文大型语言模型的块注意力混合。*arXiv preprint arXiv:2502.13189*，2025。埃里克·马丁和克里斯·坎迪。在序列长度上并行化线性递归神经网络。在 *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*。OpenReview.net，2018。网址 https://openreview.net/forum?id=HyUNwulC-。

MiniMax，奥尼安·李，龚邦伟，杨博，山博济，刘畅，朱成，张春浩，郭聪超，陈达，李东等。Minimax-01：用闪电注意力扩展基础模型。*arXiv preprint arXiv:2501.08313*，2025。

伊戈尔·莫利博格、彼得·阿尔伯特、莫雅·陈、扎查里·德维托、多伊德·埃西奥布、纳曼·戈亚尔、普尼特·辛格·库拉、沙兰·纳朗、安德鲁·普尔顿、阮·席尔瓦、平·唐、迪安娜·利斯科维奇、浦新、张宇辰、梅拉妮·坎巴杜尔、斯蒂芬·罗勒和张苏珊。关于大规模机器学习中阿达姆不稳定性的理论。*arXiv preprint arXiv:2304.09871*，2023。

OpenAI。介绍 openai o1。https://openai.com/o1/，2024a。网页，访问于2024年。

OpenAI。Openai mrcr 数据集。https://huggingface.co/datasets/openai/mrcr，2024b。访问日期：2025-06-15。OpenAI。引入深度研究，2025。网址 https://openai.com/index/introducing-deep-research/。彭博，埃里克·阿尔凯德，昆汀·格雷戈里·安东尼，阿隆·阿尔巴拉克，塞缪尔·阿尔卡迪诺，斯特拉·比德曼，曹欢奇，程鑫，阮明俊，莱昂·德尔克斯基 等。Rwkv：为变换器时代重新发明 RNNs。在 *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*，2023。彭博，丹尼尔·戈德斯坦，昆汀·安东尼，阿隆·阿尔巴拉克，埃里克·阿尔凯德，斯特拉·比德曼，尤金·蔡，泰迪·费迪南，侯浩文，Przemys Kazienko。鹰与雀：具有矩阵值状态和动态递归的 Rwkv。*arXiv preprint arXiv:2404.05892*，2024a。彭博，丹尼尔·戈德斯坦，昆汀·安东尼，阿隆·阿尔巴拉克，埃里克·阿尔凯德，斯特拉·比德曼，尤金·蔡，泰迪·费迪南，侯浩文，Przemys Kazienko。鹰与雀：具有矩阵值状态和动态递归的 Rwkv。*arXiv preprint arXiv:2404.05892*，2024b。彭博，张瑞冲，丹尼尔·戈德斯坦，埃里克·阿尔凯德，杜兴建，侯浩文，林佳炬，刘嘉兴，卢贾娜，威廉·梅里尔 等。Rwkv-7。*arXiv preprint arXiv:2503.14456*，2025。郝鹏，尼古拉斯·帕帕斯，Dani Yogatama，Roy Schwartz，Noah Smith，和孔灵鹏。随机特征注意力。在 *International Conference on Learning Representations*，2021。网址 https://openreview.net/forum?id=QtTKTdVrFBB。

Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, et al. Humanity's last exam. *arXiv preprint arXiv:2501.14249*, 2025.

Zhen Qin, Weixuan Sun, Hui Deng, Dongxu Li, Yunshen Wei, Baohong Lv, Junjie Yan, Lingpeng Kong, and Yiran Zhong. cosformer: Rethinking softmax in attention. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

Zhen Qin, Xiaodong Han, Weixuan Sun, Dongxu Li, Lingpeng Kong, Nick Barnes, and Yiran Zhong. The devil in linear transformer. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7025–7041, 2022a.

Zhen Qin, Weixuan Sun, Hui Deng, Dongxu Li, Yunshen Wei, Baohong Lv, Junjie Yan, Lingpeng Kong, and Yiran Zhong. cosFormer: Rethinking softmax in attention. In *International Conference on Learning Representations*, 2022b. URL https://openreview.net/forum?id=Bl8CQrx2Up4.

Zhen Qin, Songlin Yang, and Yiran Zhong. Hierarchically gated recurrent neural network for sequence modeling. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 33202–33221, 2023.

Zhen Qin, Yuxin Mao, Xuyang Shen, Dong Li, Jing Zhang, Yuchao Dai, and Yiran Zhong. You only scan once: Efficient multi-dimension sequential modeling with lightnet. *arXiv preprint arXiv:2405.21022*, 2024a.

Zhen Qin, Weigao Sun, Dong Li, Xuyang Shen, Weixuan Sun, and Yiran Zhong. Lightning attention-2: A free lunch for handling unlimited sequence lengths in large language models. *arXiv preprint arXiv:2401.04658*, 2024b.

Zhen Qin, Weigao Sun, Dong Li, Xuyang Shen, Weixuan Sun, and Yiran Zhong. Various lengths, constant speed: Efficient language modeling with lightning attention. In *International conference on machine learning*, pages 41517–41535. PMLR, 2024c.

Zhen Qin, Songlin Yang, Weixuan Sun, Xuyang Shen, Dong Li, Weigao Sun, and Yiran Zhong. HGRN2. *arXiv preprint arXiv:2404.07904*, 2024d.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024.

Liliang Ren, Yang Liu, Yadong Lu, Yelong Shen, Chen Liang, and Weizhu Chen. Samba: Simple hybrid state space models for efficient unlimited context language modeling. *arXiv preprint arXiv:2406.07522*, 2024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

朗范，艾丽斯·加蒂，韩子文，李纳森·李，胡约瑟芬，张休，陈博 Calvin Zhang，穆罕默德·沙阿班，约翰·林，肖恩·史伊，等。人类的最后考验。*arXiv preprint arXiv:2501.14249*，2025年。秦振，孙伟轩，邓辉，李东旭，韦云深，吕宝红，闫俊杰，孔灵鹏，中怡然。cosformer：重新思考注意力中的softmax。在*Proceedings of the International Conference on Learning Representations (ICLR)*，2021年。秦振，韩晓东，孙伟轩，李东旭，孔灵鹏，Nick Barnes，中怡然。线性变换器中的魔鬼。在*Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*，第7025–7041页，2022年a。秦振，孙伟轩，邓辉，李东旭，韦云深，吕宝红，闫俊杰，孔灵鹏，中怡然。cosFormer：重新思考注意力中的softmax。在*International Conference on Learning Representations*，2022年b。网址 `https://openreview.net/forum?id=Bl8CQrx2Up4`。秦振，杨松林，中怡然。层级门控循环神经网络用于序列建模。在*Proceedings of the 37th International Conference on Neural Information Processing Systems*，第33202–33221页，2023年。秦振，毛玉欣，沈旭阳，李东，张静，戴玉超，中怡然。你只需扫描一次：轻网高效多维序列建模。*arXiv preprint arXiv:2405.21022*，2024年a。秦振，孙伟高，李东旭，沈旭阳，孙伟轩，中怡然。Lightning attention-2：处理大语言模型中无限序列长度的免费方案。*arXiv preprint arXiv:2401.04658*，2024年b。秦振，孙伟高，李东旭，沈旭阳，孙伟轩，中怡然。各种长度，恒定速度：利用Lightning attention实现高效语言建模。在*International conference on machine learning*，第41517–41535页。PMLR，2024年c。秦振，杨松林，孙伟轩，沈旭阳，李东，孙伟高，中怡然。HGRN2。*arXiv preprint arXiv:2404.07904*，2024年d。Qwen，：，安阳，杨宝松，张贝辰，惠碧源，郑博，俞炳，李成远，刘大恒，黄飞，韦浩然，林欢，杨建，涂建宏，张建新，杨建新，杨佳熙，周景仁，林俊阳，邓凯，卢克明，杨科勤，余柯，李梅，薛明锋，张佩，朱秦，门锐，林润基，李天浩，唐天翼，夏婷玉，任兴章，任轩成，张杨，苏杨，张怡昌，Wan Yu，刘玉琼，崔泽宇，张振儒，邱子涵。Qwen2.5技术报告。*arXiv preprint arXiv:2412.15115*，2025年。David Rein，Betty Li Hou，Asa Cooper Stickland，Jackson Petty，Richard Yuanzhe Pang，Julien Dirani，Julian Michael，和Samuel R Bowman。Gpqa：一个研究生级别的谷歌防作弊问答基准。在*First Conference on Language Modeling*，2024年。任亮，刘洋，陆雅东，沈叶龙，梁辰，陈伟柱。Samba：用于高效无限上下文语言建模的简单双模状态空间模型。*arXiv preprint arXiv:2406.07522*，2024年。John Schulman，Filip Wolski，Prafulla Dhariwal，Alec Radford，Oleg Klimov。近端策略优化算法。*arXiv preprint arXiv:1707.06347*，2017年。

ByteDance Seed, Jiaze Chen, Tiantian Fan, Xin Liu, Lingjun Liu, Zhiqi Lin, Mingxuan Wang, Chengyi Wang, Xiangpeng Wei, Wenyuan Xu, et al. Seed1. 5-thinking: Advancing superb reasoning models with reinforcement learning. *arXiv preprint arXiv:2504.13914*, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, et al. DeepSeekMath. *arXiv preprint arXiv:2402.03300*, 2024.

Xuyang Shen, Dong Li, Ruitao Leng, Zhen Qin, Weigao Sun, and Yiran Zhong. Scaling laws for linear complexity language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 16377–16426, 2024.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv:2409.19256*, 2024.

Chenglei Si, Diyi Yang, and Tatsunori Hashimoto. Can llms generate novel research ideas? a large-scale human study with 100+ nlp researchers. *arXiv preprint arXiv:2409.04109*, 2024.

Julien Siems, Timur Carstensen, Arber Zela, Frank Hutter, Massimiliano Pontil, and Riccardo Grazzi. Deltaproduct: Improving state-tracking in linear rnns via householder products. *arXiv preprint arXiv:2502.10297*, 2025.

Ved Sirdeshmukh, Kaustubh Deshpande, Johannes Mols, Lifeng Jin, Ed-Yeremai Cardona, Dean Lee, Jeremy Kritz, Willow Primack, Summer Yue, and Chen Xing. Multichallenge: A realistic multi-turn conversation evaluation benchmark challenging to frontier llms. *arXiv preprint arXiv:2501.17399*, 2025.

Weigao Sun, Disen Lan, Tong Zhu, Xiaoye Qu, and Yu Cheng. Linear-moe: Linear sequence modeling meets mixture-of-experts. *arXiv preprint arXiv:2503.05447*, 2025.

Yu Sun, Xinhao Li, Karan Dalal, Jiarui Xu, Arjun Vikram, Genghan Zhang, Yann Dubois, Xinlei Chen, Xiaolong Wang, Sanmi Koyejo, et al. Learning to (learn at test time): Rnns with expressive hidden states. *arXiv preprint arXiv:2407.04620*, 2024.

Yutao Sun, Li Dong, Shaohan Huang, Shuming Ma, Yuqing Xia, Jilong Xue, Jianyong Wang, and Furu Wei. Retentive network: A successor to transformer for large language models. *arXiv preprint arXiv:2307.08621*, 2023.

Tencent AI Lab. Hunyuan-t1: Reasoning efficiency redefined. https://llm.hunyuan.tencent.com/#/Blog/hy-t1/, 2025. Accessed: 2025-06-15.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Johannes von Oswald, Nino Scherrer, Seijin Kobayashi, Luca Versari, Songlin Yang, Maximilian Schlegel, Kaitlin Maile, Yanick Schimpf, Oliver Sieberling, Alexander Meulemans, et al. Mesanet: Sequence modeling by locally optimal test-time training. *arXiv preprint arXiv:2506.05233*, 2025.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yu, Gao Huang, and Junyang Lin. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning. *arXiv preprint arXiv:2506.01939*, 2025.

字节跳动种子，陈嘉泽，范甜甜，刘欣，刘灵君，林志奇，王明轩，王成一，韦翔鹏，许文远，等。Seed1。5-思维：用强化学习推进卓越推理模型。*arXiv preprint arXiv:2504.13914*，2025年。邵志宏，王佩怡，朱启浩，许润新，宋俊骁，毕晓，张浩伟，张明川，李YK，吴Y，等。DeepSeekMath。*arXiv preprint arXiv:2402.03300*，2024年。沈旭阳，李东，冷瑞涛，秦震，孙伟高，中怡然。线性复杂度语言模型的缩放定律。在*Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*，第16377–16426页，2024年。盛光明，张驰，叶子林峰，吴希彬，张旺，张儒，彭阳华，林海宾，吴川。Hybridflow：一个灵活高效的RLHF框架。*arXiv preprint arXiv:2409.19256*，2024年。司成雷，杨迪一，桥本达典。LLMs能生成新颖的研究想法吗？一项涉及100+名NLP研究人员的大规模人类研究。*arXiv preprint arXiv:2409.04109*，2024年。

Julien Siems, Timur Carstensen, Arber Zela, Frank Hutter, Massimiliano Pontil 和 Riccardo Grazzi。Delta product：通过Householder乘积改善线性RNN中的状态跟踪。*arXiv preprint arXiv:2502.10297*，2025年。

由Sirdeshmukh Ved、Kaustubh Deshpande、Johannes Mols、Lifeng Jin、Ed-Yeremai Cardona、Dean Lee、Jeremy Kritz、Willow Primack、Summer Yue和Chen Xing撰写。多挑战：一个逼真的多轮对话评估基准，挑战前沿的llms。*arXiv preprint arXiv:2501.17399*，2025年。Sun Weigao、Lan Disen、Zhu Tong、Qu Xiaoye和Cheng Yu。Linear-moe：线性序列建模结合专家混合模型。*arXiv preprint arXiv:2503.05447*，2025年。Sun Yu、Li Xinhao、Dalal Karan、Xu Jiarui、Vikram Arjun、Zhang Genghan、Dubois Yann、Chen Xinlei、Wang Xiaolong、Koyejo Sanmi等。学会（在测试时学习）：具有表现力隐藏状态的Rnns。*arXiv preprint arXiv:2407.04620*，2024年。Sun Yutao、Dong Li、Huang Shaohan、Ma Shuming、Xia Yuqing、Xue Jilong、Wang Jianyong和Wei Furu。保留网络：大型语言模型的Transformer继任者。*arXiv preprint arXiv:2307.08621*，2023年。腾讯AI实验室。Hunyuan-t1：重新定义推理效率。`https://llm.hunyuan.tencent. com/#/Blog/hy-t1/`，2025年。查阅日期：2025-06-15。Vaswani Ashish、Shazeer Noam、Parmar Niki、Uszkoreit Jakob、Jones Llion、Gomez Aidan N、Kaiser ukasz和Polosukhin Illia。注意力机制就是你所需要的一切。*Advances in neural information processing systems*，2017年6月30日。von Oswald Johannes、Scherrer Nino、Kobayashi Seijin、Versari Luca、Yang Songlin、Schlegel Maximilian、Maile Kaitlin、Schimpf Yanick、Sieberling Oliver、Meulemans Alexander等。Mesanet：通过局部最优的测试时训练进行序列建模。*arXiv preprint arXiv:2506.05233*，2025年。Wang Shenzhi、Yu Le、Gao Chang、Zheng Chujie、Liu Shixuan、Lu Rui、Dang Kai、Chen Xionghui、Yang Jianxin、Zhang Zhenru、Liu Yuqiong、Yang An、Zhao Andrew、Yue Yang、Song Shiji、Yu Bowen、Huang Gao和Lin Junyang。超越80/20规则：高熵少数令牌推动有效的强化学习以实现llm推理。*arXiv preprint arXiv:2506.01939*，2025年。

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*, 2024.

Chunqiu Steven Xia, Yinlin Deng, Soren Dunn, and Lingming Zhang. Agentless: Demystifying llm-based software engineering agents. *arXiv preprint arXiv:2407.01489*, 2024.

Frank F. Xu, Yufan Song, Boxuan Li, Yuxuan Tang, Kritanjali Jain, Mengxue Bao, Zora Z. Wang, Xuhui Zhou, Zhitong Guo, Murong Cao, Mingyang Yang, Hao Yang Lu, Amaad Martin, Zhe Su, Leander Maben, Raj Mehta, Wayne Chi, Lawrence Jang, Yiqing Xie, Shuyan Zhou, and Graham Neubig. Theagentcompany: Benchmarking llm agents on consequential real world tasks. *arXiv preprint arXiv:2412.14161*, 2025.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. *arXiv preprint arXiv:2312.06635*, 2024a.

Songlin Yang, Bailin Wang, Yu Zhang, Yikang Shen, and Yoon Kim. Parallelizing linear transformers with the delta rule over sequence length. *arXiv preprint arXiv:2406.06484*, 2024b.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik R Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. In *The Thirteenth International Conference on Learning Representations*, 2025.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

Jingyang Yuan, Huazuo Gao, Damai Dai, Junyu Luo, Liang Zhao, Zhengyan Zhang, Zhenda Xie, YX Wei, Lean Wang, Zhiping Xiao, et al. Native sparse attention: Hardware-aligned and natively trainable sparse attention. *arXiv preprint arXiv:2502.11089*, 2025.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. Big Bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297, 2020.

Weihao Zeng, Yuzhen Huang, Qian Liu, Wei Liu, Keqing He, Zejun Ma, and Junxian He. Simplerl-zoo: Investigating and taming zero reinforcement learning for open base models in the wild. *arXiv preprint arXiv:2503.18892*, 2025.

Yu Zhang, Songlin Yang, Rui-Jie Zhu, Yue Zhang, Leyang Cui, Yiqiao Wang, Bolun Wang, Freda Shi, Bailin Wang, Wei Bi, et al. Gated slot attention for efficient linear-time sequence modeling. *Advances in Neural Information Processing Systems*, 37:116870–116898, 2024.

Y乌博 王学光 马学光 张歌 倪元胜 Abhranil Chandra 谷世光 任伟明 Arulraj Aaran 何轩 江子彦 等。Mmlu-pro：一个更强大且具有挑战性的多任务语言理解基准。在 *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*，2024。

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, 等人。链式思维提示引发大型语言模型的推理。*Advances in neural information processing systems*，35:24824–24837，2022年。

Jason Wei、Nguyen Karina、Hyung Won Chung、Yunxin Joy Jiao、Spencer Papay、Amelia Glaese、John Schulman 以及 William Fedus。衡量大型语言模型中的短形式事实性。*arXiv preprint arXiv:2411.04368*，2024。

春秋 Steven Xia, Yinlin Deng, Soren Dunn, 和 Lingming Zhang. 无代理：揭示基于 llm 的软件工程代理的神秘面纱。*arXiv preprint arXiv:2407.01489*，2024。

Frank F. Xu, Yufan Song, Boxuan Li, Yuxuan Tang, Kritanjali Jain, Mengxue Bao, Zora Z. Wang, Xuhui Zhou, Zhitong Guo, Murong Cao, Mingyang Yang, Hao Yang Lu, Amaad Martin, Zhe Su, Leander Maben, Raj Mehta, Wayne Chi, Lawrence Jang, Yiqing Xie, Shuyan Zhou, 和 Graham Neubig。Theagentcompany：在重要的现实世界任务中对LLM代理进行基准测试。*arXiv preprint arXiv:2412.14161*，2025年。

杨松林，王百林，沈亦康，潘拉姆斯瓦尔，金允。具有硬件高效训练的门控线性注意力变换器。*arXiv preprint arXiv:2312.06635*，2024a。

S杨昂林、王百林、张瑜、沈亦康、金尧。基于序列长度的delta规则并行线性变换器。*arXiv preprint arXiv:2406.06484*，2024b。

S候瑜耀、诺亚·辛恩、佩德拉姆·拉扎维和卡尔蒂克·R纳拉西姆汉。τ-基准：一个用于现实世界领域中工具-代理-用户交互的基准。在*The Thirteenth International Conference on Learning Representations*，2025年。

余启英，张正，朱若飞，袁玉凤，左晓辰，岳玉，戴伟南，范甜甜，刘高宏，刘灵君，刘新，林海滨，林志奇，马博乐，盛光明，童玉轩，张驰，张莫凡，张旺，朱航，朱金华，陈嘉泽，陈江杰，王成义，余洪利，宋玉轩，韦向鹏，周浩，刘晶晶，马伟颖，张雅琴，严琳，乔木，吴永辉，王明轩。Dapo：一个大规模的开源LLM强化学习系统。*arXiv preprint arXiv:2503.14476*，2025年。

J袁英阳，高佐高，大迈戴，罗俊宇，赵亮，张正彦，谢振达，韦YX，王 Lean，肖志平，等。本地稀疏注意：硬件对齐且本地可训练的稀疏注意。*arXiv preprint arXiv:2502.11089*，2025。

MAnzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang 等。Big Bird：用于更长序列的变换器。*Advances in neural information processing systems*，33：17283–17297，2020。

Weihao Zeng, Yuzhen Huang, Qian Liu, Wei Liu, Keqing He, Zejun Ma, 和 Junxian He. Simplerl-zoo：在野外研究和驯服用于开放基础模型的零强化学习。*arXiv preprint arXiv:2503.18892*，2025年。

Yu Zhang, Songlin Yang, Rui-Jie Zhu, Yue Zhang, Leyang Cui, Yiqiao Wang, Bolun Wang, Freda Shi, Bailin Wang, Wei Bi, 等。门控槽注意力用于高效线性时间序列建模。*Advances in Neural Information Processing Systems*，37：116870–116898，2024。

A. Contributors

The contributors to the report are listed in alphabetical order as follows:

Aili Chen, Aonian Li, Bangwei Gong, Binyang Jiang, Bo Fei, Bo Yang, Boji Shan, Changqing Yu, Chao Wang, Cheng Zhu, Chengjun Xiao, Chengyu Du, Chi Zhang, Chu Qiao, Chunhao Zhang, Chunhui Du, Congchao Guo, Da Chen, Deming Ding, Dianjun Sun, Dong Li, Enwei Jiao, Haigang Zhou, Haimo Zhang, Han Ding, Haohai Sun, Haoyu Feng, Huaiguang Cai, Haichao Zhu, Jian Sun, Jiaqi Zhuang, Jiaren Cai, Jiayuan Song, Jin Zhu, Jingyang Li, Jinhao Tian, Jinli Liu, Junhao Xu, Junjie Yan, Junteng Liu, Junxian He, Kaiyi Feng, Ke Yang, Kecheng Xiao, Le Han, Leyang Wang, Lianfei Yu, Liheng Feng, Lin Li, Lin Zheng, Linge Du, Lingyu Yang, Lunbin Zeng, Minghui Yu, Mingliang Tao, Mingyuan Chi, Mozhi Zhang, Mujie Lin, Nan Hu, Nongyu Di, Peng Gao, Pengfei Li, Pengyu Zhao, Qibing Ren, Qidi Xu, Qile Li, Qin Wang, Rong Tian, Ruitao Leng, Shaoxiang Chen, Shaoyu Chen, Shengmin Shi, Shitong Weng, Shuchang Guan, Shuqi Yu, Sichen Li, Songquan Zhu, Tengfei Li, Tianchi Cai, Tianrun Liang, Weiyu Cheng, Weize Kong, Wenkai Li, Xiancai Chen, Xiangjun Song, Xiao Luo, Xiao Su, Xiaobo Li, Xiaodong Han, Xinzhu Hou, Xuan Lu, Xun Zou, Xuyang Shen, Yan Gong, Yan Ma, Yang Wang, Yiqi Shi, Yiran Zhong, Yonghong Duan, Yongxiang Fu, Yongyi Hu, Yu Gao, Yuanxiang Fan, Yufeng Yang, Yuhao Li, Yulin Hu, Yunan Huang, Yunji Li, Yunzhi Xu, Yuxin Mao, Yuxuan Shi, Yuze Wenren, Zehan Li, Zelin Li, Zhanxu Tian, Zhengmao Zhu, Zhenhua Fan, Zhenzhen Wu, Zhichao Xu, Zhihang Yu, Zhiheng Lyu, Zhuo Jiang, Zibo Gao, Zijia Wu, Zijian Song, Zijun Sun

A. 贡献者

报告的贡献者按字母顺序列出如下：

陈爱丽，李奥年，龚邦伟，江彬阳，费博，杨博，山博济，余长青，王超，朱成，肖成军，杜成宇，张驰，乔楚，张春浩，杜春晖，郭聪超，陈大，丁德明，孙殿军，李东，焦恩伟，周海港，张海墨，丁汉，孙浩海，冯浩宇，蔡怀光，朱海超，孙健，庄佳琪，宋嘉远，朱金，李晶阳，田金豪，刘金利，许俊豪，闫俊杰，刘俊腾，何俊贤，冯凯一，杨科，萧克成，韩乐，王乐阳，余连飞，冯立恒，李林，郑林，杜灵格，杨凌宇，曾伦彬，余明辉，陶明亮，池明远，张墨智，林杰林，胡南，狄浓雨，高鹏，李鹏飞，赵鹏宇，任启兵，徐启迪，李启乐，王秦，田荣，冷瑞涛，陈少祥，陈少瑜，史胜敏，翁世通，关树昌，余书奇，李思辰，朱松全，李腾飞，蔡天驰，梁天润，程伟宇，孔维泽，李文凯，陈贤才，宋翔军，罗晓，苏晓，李晓波，韩晓东，侯新竹，卢轩，邹迅，沈旭阳，龚彦，马彦，王杨，史一奇，中伊然，段永红，傅永翔，李瑜，范源翔，杨玉峰，李宇浩，胡玉林，黄云南，李云志，徐昱志，毛宇轩，史宇宸，文泽泽，李泽涵，李展昊，朱振豪，范振华，吴振珍，许志超，余志航，吕志恒，姜卓，高子博，吴子嘉，宋子俊