自动化学报
ACTA AUTOMATICA SINICA

高效提升多模态大语言模型推理能力的级联强化学习策略

王玮 蒲恒骏 景凌林 乔宇

A Cascade Reinforcement Learning Strategy for Efficiently Enhancing the Reasoning Ability of Multimodal Large Language Models

WANG Wei-Yun, PU Heng-Jun, JING Ling-Lin, QIAO Yu

在线阅读 View online: https://doi.org/10.16383/j.aas.c250515

高效提升多模态大语言模型推理能力的级联强化学习策略

王玮赟[1,2] 蒲恒骏[1] 景凌林[1] 乔 宇[1]

摘 要 强化学习在提升多模态大语言模型的推理能力上展现出巨大潜力, 逐渐成为模型训练过程中的关键步骤. 然而, 在线强化学习算法需要策略模型在训练过程中实时采样, 且收敛速度较慢, 因此训练成本昂贵; 离线强化学习算法虽然整体成本更低, 但是也牺牲了性能上限. 本文尝试将离线强化学习训练成本低的特性与在线强化学习性能上限高的优势相结合, 提出一种新的训练策略——级联强化学习. 这套训练策略包含离线强化学习和在线强化学习两个训练阶段, 其中离线强化学习阶段用于加速模型收敛并提升后续训练的稳定性; 在线强化学习阶段对模型进行更精细的训练, 进一步提升其性能上限. 本文通过一系列定量分析实验证明了相比单一的在线强化学习算法, 级联强化学习可以通过一半的训练成本达到更高的性能上限. 这种简单有效的训练策略将 InternVL3.5-8B-Instruct 和 InternVL3.5-241B-A28B-Instruct 在七个多模态推理评测基准上的平均准确率分别提升了 6.7% 和 6.5%, 证明了这一策略的有效性和可扩展性.

关键词 多模态大语言模型; 强化学习; 高效后训练策略; 推理能力增强

引用格式 王玮赟, 蒲恒骏, 景凌林, 乔宇. 高效提升多模态大语言模型推理能力的级联强化学习策略. 自动化学报, 2026, **52**(8): 1615−1632

DOI 10.16383/j.aas.c250515 **CSTR** 32138.14.j.aas.c250515

A Cascade Reinforcement Learning Strategy for Efficiently Enhancing the Reasoning Ability of Multimodal Large Language Models

WANG Wei-Yun[1,2] PU Heng-Jun[1] JING Ling-Lin[1] QIAO Yu[1]

Abstract Reinforcement learning has demonstrated substantial potential in enhancing the reasoning ability of multimodal large language models and has become a critical step during model training. However, online reinforcement learning algorithms require real-time sampling from the policy model during training and typically suffer from slow convergence speed, resulting in high training cost; Offline reinforcement learning algorithms offer lower overall cost but often sacrifices performance ceiling. In this paper, we propose a novel training strategy, termed cascade reinforcement learning, which integrates the low training cost of offline reinforcement learning with the superior performance potential of online reinforcement learning. This training strategy consists of two stages: An offline reinforcement learning stage that accelerates model convergence and improves subsequent training stability, followed by an online reinforcement learning stage that performs more fine-grained training to further enhance the model′s performance ceiling. Through extensive quantitative analysis, we demonstrate that our cascade reinforcement learning achieves higher performance ceiling than standalone online reinforcement learning algorithms while requiring only half of the training cost. Using this simple yet effective training strategy, we improve the average accuracy of InternVL3.5-8B-Instruct and InternVL3.5-241B-A28B-Instruct by 6.7% and 6.5%, respectively, across seven multimodal reasoning evaluation benchmarks, validating the effectiveness and scalability of the proposed strategy.

Keywords multimodal large language models; reinforcement learning; efficient post-training strategy; reasoning ability enhancement

Citation Wang Wei-Yun, Pu Heng-Jun, Jing Ling-Lin, Qiao Yu. A cascade reinforcement learning strategy for efficiently enhancing the reasoning ability of multimodal large language models. *Acta Automatica Sinica*, 2026, **52**(8): 1615−1632

随着多模态大语言模型在多模态感知和理解能力上展现出越来越强大的性能, 学界的研究热点逐渐从相对简单的多模态感知任务转向相对复杂的多模态推理任务. 与此同时, 由于在提升大语言模型的推理能力上展现出巨大的潜力, 强化学习 (reinforcement learning, RL) 逐渐成为模型训练过程中的关键步骤. 其中, 以直接偏好优化 (direct preference optimization, DPO)[1] 和混合偏好优化

收稿日期 2025-09-30 录用日期 2026-02-13
Manuscript received September 30, 2025; accepted February 13, 2026
科技创新 2030——"新一代人工智能"重大项目 (2022ZD0160102) 资助
Supported by National Key R&D Program of China (2022ZD 0160102)
本文责任编委 林宙辰
Recommended by Associate Editor LIN Zhou-Chen
1. 上海人工智能实验室 上海 200233 2. 复旦大学 上海 200438
1. Shanghai Artificial Intelligence Laboratory, Shanghai 200233
2. Fudan University, Shanghai 200438

(mixed preference optimization, MPO)[2] 为代表的离线强化学习将预先采样得到的离线轨迹整理成偏好对的形式对模型进行训练, 尽管引入的参考模型 (reference model) 增加了一定的显存压力, 但其训练时间上的成本与传统的有监督微调 (supervised fine-tuning, SFT) 是接近的. 而以群组相对优势策略优化 (group relative policy optimization, GRPO)[3] 和群组序列策略优化 (group sequence policy optimization, GSPO)[4] 为代表的在线强化学习则只需要推理问题即可进行训练, 在训练过程中模型实时生成回答, 并通过策略梯度法来根据对应的奖励值对模型参数进行更新. 然而对于多模态大语言模型而言, 由于其自回归生成的整体特性, 轨迹采样的成本是十分昂贵的, 这使得针对多模态大语言模型的在线强化学习成本也十分昂贵. 第 3.2 节将通过实验证明, 离线强化学习的训练成本更低, 收敛速度更快, 但是性能上限更低; 在线强化学习的训练成本更高, 收敛速度更慢, 但是性能上限更高. 针对这一现象, 一个关键问题是能否将离线强化学习训练成本低的特性与在线强化学习性能上限高的优势相结合, 从而以更低的成本达到更高的性能.

本文通过一系列定量实验, 从训练成本以及性能收益的角度, 分析不同离线与在线强化学习算法的组合方式. 具体而言, 以多模态思维链 (multi-modal chain-of-thought, M3CoT) 数据集[5] 的训练集和验证集为实验平台, 基于该数据集提供的 7861 条训练数据, 通过只使用其中一半的训练数据的方式来模拟有限数据下的训练场景, 分析不同的强化学习训练算法对模型性能的影响. 基于这些实验, 本文发现以下结论: 1) 对于指令微调后的模型在推理能力上的增量训练, 离线强化学习和在线强化学习都可以带来比有监督微调更高的性能收益; 2) 在有限数据场景下, 离线强化学习收敛速度更快, 取得的性能收益也高于在线强化学习, 而在全量数据场景下, 在线强化学习展现出更优的性能上限; 3) 在有限数据场景下, 在相同训练问题上先通过离线强化学习算法进行训练, 再通过在线强化学习算法进行训练可以取得和直接用在线强化学习算法训练两轮接近甚至更优的训练效果.

基于这些结论, 本文提出级联强化学习 (cascade reinforcement learning, CascadeRL) 的训练策略, 该策略将多模态大语言模型的强化学习阶段拆分成离线强化学习阶段和在线强化学习阶段. 其中, 离线强化学习阶段利用小尺寸的模型收集离线轨迹, 更大尺度的模型基于这些轨迹进行训练, 无需昂贵的采样成本, 让模型快速收敛, 取得初步的性能收益; 在线强化学习阶段则利用当前正在训练的策略模型在线收集轨迹, 实时提供奖励值来更新模型参数, 进一步提升模型性能, 带来更高的性能收益. 本文基于这种简单有效的训练策略对 InternVL3.5[6] 的指令微调版本进行增量式训练. 实验结果表明, InternVL3.5 的八种尺寸的指令微调模型在七个多模态推理评测基准上的综合性能平均提升接近 8% 的准确率, 证明了这一策略的有效性和可扩展性. 其中, InternVL3.5-8B-Instruct 和 InternVL3.5-241B-A28B-Instruct 在七个多模态推理评测基准上的平均准确率分别提升了 6.7% 和 6.5%.

除此之外, 本文对训练过程中模型关于正负样本的对数生成概率的变化情况进行分析, 从模型训练动力学的角度提供级联强化学习为什么有效的分析. 实验结果表明, 对于有监督微调, 模型关于正样本的对数生成概率上升的同时, 负样本的对数生成概率也有轻微上升. 对于在线强化学习, 负样本的对数生成概率有下降趋势. 而对于离线强化学习, 负样本的对数生成概率下降, 正样本的对数生成概率没有明显变化. 模型关于正负样本的对数生成概率的变化趋势从模型训练动力学的角度解释了强化学习以及级联强化学习为什么有效: 有监督微调让模型模仿标准答案时, 也无意间提高了和标准答案只在关键词元 (token) 上有差异的错误答案的生成概率, 而强化学习则降低了这些错误答案的生成概率, 使其推理过程中受到因个别错误词元导致的错误累积的影响更小, 并且通过对负样本的抑制实现对模型输出分布的裁剪, 提升模型输出分布的整体质量, 从而提升模型的整体性能. 级联强化学习则先对模型输出中的低质量区域进行快速修剪, 然后让模型基于更高质量的响应分布进行在线训练, 从而提升训练效率和效果.

本文的主要贡献包含以下三个方面:

1) 针对离线强化学习和在线强化学习进行一系列定量实验, 并基于这些实验总结一系列实验结论. 除此之外, 本文对训练过程中, 模型关于正负样本的对数生成概率的变化情况进行分析, 从模型训练动力学的角度提供级联强化学习为什么有效的分析.

2) 提出级联强化学习的训练策略, 该策略仅通过一半的训练成本即可取得和在线强化学习接近甚至更优的训练效果.

3) 在 InternVL3.5 的指令微调版本上进行增量式训练, 使得八种尺寸的模型在七个多模态推理评测基准上的综合性能平均提升接近 8% 的准确率, 证明了这一策略的有效性和可扩展性.

1 相关工作

1.1 多模态大语言模型

随着大语言模型[7−14] 的发展, 多模态大语言模型也取得了显著进展[15−17]. 为了能够复用在海量单模态数据上经过预训练的大语言模型和视觉基座模型, 一系列研究通过引入中间件的方式来对齐视觉与语言的表征空间, 使得视觉特征图可以被展开后以软提示词的形式输入大语言模型, 只通过较低成本的增量式训练便取得了良好的多模态性能. 此外, 还有一些工作通过引入额外的视觉信息融合层来扩展预训练的大语言模型, 使得模型内部通过不同的参数来分别处理不同模态的词元, 大大减少了用来表征每张图像所需要的视觉词元的数量, 但是额外引入的大量未训练参数也带来了额外的训练成本. 最近, 也有研究探索了无视觉编码器的架构, 这些架构使用统一的 Transformer 模型联合处理视觉和文本信息, 无需独立的视觉编码器和视觉信息融合层. 除了模型架构的探索, 学界也尝试构建高质量的训练数据来提升多模态感知、理解和推理能力. 例如 LAION-5B[18]、OmniCorpus[19] 以及 OBELICS[20] 等工作爬取了互联网上的大规模数据, 并通过一系列后处理算法对其进行了解析和清洗, 构建了超大规模的多模态预训练语料库. 而以 MMInstruct[21] 和 AllSeeing[22] 为代表的工作利用开源模型构建了一套迭代化的半自动数据标注管线, 通过较低的成本构建了一批高质量指令跟随和多模态感知数据集. 尽管这些研究取得了一系列的进展, 主流的多模态大语言模型仍依赖于预训练 + 有监督微调的训练范式, 这种范式容易受到曝光偏置的影响, 导致多模态推理能力和长序列输出能力有限. 针对这一问题, 近期的工作开始尝试引入强化学习的训练算法, 通过在模型训练过程中引入负样本的方式来进一步提升模型的训练效率和效果. 具体而言, 以 DPO[1] 和 MPO[2] 为代表的工作通过离线强化学习的方式以较低的训练成本取得了性能收益, 而以 GRPO[3] 和 GSPO[4] 为代表的工作则采用在线强化学习的方式通过更高的训练成本取得了更高的性能上限. 尽管近期的研究在基于强化学习的训练算法上做出了一系列改进, 但是目前的主流算法仍然只通过离线强化学习或在线强化学习中的一种算法进行训练. 本文通过一系列定量分析实验证明, 离线强化学习和在线强化学习并非对立的两种算法, 可以将他们结合起来, 以更低的训练成本取得更高的性能上限.

1.2 强化学习

强化学习是大语言模型和多模态大语言模型后训练阶段的关键技术. 具体而言, 基于人类反馈的强化学习 (reinforcement learning from human feedback, RLHF) 使用人类偏好作为奖励信号对模型进行微调, 使模型行为与人类偏好对齐. InstructGPT[11] 使用奖励模型作为人类偏好的代理, 并通过近端策略优化 (proximal policy optimization, PPO) 算法[23] 最大化该奖励, 从而提升模型在“有用、诚实、无害”方面的表现. PPO-Max[24] 进一步探索了 PPO 的实现细节, 提出了更稳定的算法版本. 此外, DPO[1] 提出了一种基于 Bradley-Terry 模型[25] 的高效偏好优化算法, 无需显式奖励模型和在线轨迹采样, 只通过预先收集好的响应对即可进行训练, 显著降低了计算开销和训练成本. 后续研究从多个角度对该方法进行了分析与改进, 例如 BCO 算法[26] 将其进行扩展, 在训练阶段只需要单个响应及该响应的二分类好坏标签, 进一步降低了数据构建成本; MPO 算法[2] 则引入了额外的辅助损失函数, 缓解了 DPO 算法中的挤压问题. 虽然这些算法并没有引入直接的奖励值, 但是和传统的离线强化学习算法一样, 使用离线收集好的轨迹进行训练, 并针对每条轨迹有一个和奖励值方向类似的优化方向, 因此也可以被视为一种特殊的离线强化学习. 上述算法被提出的初衷在于提升模型的指令跟随能力和回答安全性的同时减少其幻觉, 因此需要引入额外的奖励模型作为人类偏好的代理. 近期随着如何提升模型推理能力这一问题得到越来越多的关注, 一系列工作开始尝试用强化学习来提升模型的这一能力. 具体而言, GRPO[3] 算法去除强化学习过程中的奖励模型, 只根据回答的正确性来为模型提供奖励值, 提升了模型的数学推理能力. 后续的 GSPO[4] 等算法针对 GRPO 的优化目标进行了一系列的改进, 进一步提升了强化学习的效率. 近期, 以 GLM-4.5V[27] 和 Keye-VL[28] 为代表的工作引入迭代式强化学习, 其目的在于通过多个阶段的强化学习, 每个阶段基于不同的训练数据提升模型不同维度的性能, 进而提升模型的性能广度. 本文探索离线强化学习与在线强化学习的结合方式, 提出级联强化学习策略, 基于同一批训练数据以更低的训练成本提升模型性能深度.

1.3 课程学习与分阶段学习

课程学习 (curriculum learning) 的核心思想是按照由易到难的顺序组织训练样本, 从而改善模型训练过程中的收敛速度与训练后的泛化性能[29−31].

在大语言模型训练领域, 以 R3[31−35] 为代表的工作通过一系列启发式规则对训练样本进行编排, 从而提升训练效率. 具体而言, R3[31] 在训练初期为模型输入当前问题的标准推理过程, 模型只需要根据推理过程预测最终答案, 随着训练过程模型能力逐渐变强, 模型采样时输入的标准推理过程也越来越短, 模型需要根据更少的标准内容来完成更多的推理, 从而实现训练过程中样本由易到难的变化, 巧妙地将课程学习的思想应用至大语言模型的训练过程中. 而 LightR1[35] 则基于 DeepSeek[36] 关于给定样本多次采样的正确率来评估样本难度, 从而区分模型在不同阶段的训练过程中使用不同难度的训练样本, 进而提升模型训练效率. 课程学习和分阶段学习的关键在于根据训练目标对数据进行筛选或重构, 形成具有不同难度层次的数据集, 并按照预定策略应用于模型训练的不同阶段. 该类方法主要通过调整数据分布来引导模型学习, 而不同阶段通常沿用相同的训练算法, 不涉及训练算法框架的改变. 本文探究的是基于给定数据, 在算法层面如何调整以最大化模型的训练效果, 并在不同的训练阶段使用不同的训练算法.

2 级联强化学习

与预训练和有监督微调相比, 本文认为强化学习的核心优势在于其能够引入负样本, 通过降低负样本的生成概率来对模型输出空间中的低质量区域进行剪枝, 进而提升整体输出质量. 作为近端策略优化算法 PPO[23] 的一种变体, 直接偏好优化算法 DPO[1] 允许模型基于已有的轨迹进行训练, 因此这种方式也可以被视为一种离线强化学习算法. 离线强化学习算法通常具有较高的训练效率, 但与在线强化学习算法相比, 其性能上限通常较低. 相反, 尽管在线强化学习算法在效果上表现更好, 但由于需要在训练过程中从策略模型实时采样轨迹, 所以计算成本高且训练耗时长.

本文提出级联强化学习, 旨在结合离线强化学习训练效率高与在线强化学习性能上限高的优势, 从而促进多模态大语言模型的后训练过程. 一个与本文提出的级联强化学习类似的概念是带级联结构的强化学习算法[37−39], 这些算法虽然面向不同的领域, 但是其核心思想都是把复杂流程拆分成若干子任务后, 由多级强化学习策略执行联合训练. 本文聚焦于大语言模型的训练, 提到的级联强化学习特指先进行离线强化学习训练, 随后进行在线强化学习的训练策略, 并且和过去这些工作相比, 本文是针对同一个任务使用不同算法进行深入训练, 并不涉及对任务的拆分.

如图 1(c) 所示, 首先使用离线强化学习算法对模型进行微调, 作为一个高效的预热阶段, 利用离线强化学习算法收敛快、训练成本低的特性, 对模型的输出分布进行一个快速但粗糙的剪枝, 取得初步的性能收益, 并为后续阶段提供更高质量的轨迹采样, 从而使下一阶段的训练能够聚焦在更高质量的分布区域, 提升训练效率. 接着采用在线强化学习算法在相同数据上进一步优化模型, 使其输出分布能够基于模型自身生成的轨迹进行进一步的精细训练, 从而取得更进一步的性能收益. 与单一的离线或在线强化学习相比, 本文提出的级联强化学习能够在降低 GPU 时间成本的情况下, 取得更进一步的性能提升.

1) 离线强化学习. 在离线强化学习阶段, 本文使用混合偏好优化算法 MPO[2] 对模型进行训练. 该阶段对模型的输出分布进行一个快速但粗糙的剪枝, 取得初步的性能收益, 并为后续阶段提供更高质量的轨迹采样, 从而使下一阶段的训练能够聚焦在更高质量的分布区域, 提升训练效率、加速收敛. 该阶段的训练数据包含三个输入: 问题 x、正样本 y_c 以及负样本 y_r. 具体而言, MPO 的训练目标是将三类损失函数结合起来: 偏好损失 $\mathcal{L}_p$、质量损失 $\mathcal{L}_q$ 以及生成损失 $\mathcal{L}_g$. 其数学表达式如下:

$$\mathcal{L}_{\text{MPO}} = w_p\mathcal{L}_p + w_q\mathcal{L}_q + w_g\mathcal{L}_g \tag{1}$$

其中, w_* 表示各个损失函数的权重. 在实践中, MPO 算法分别选择 DPO 损失[1]、BCO 损失[26] 以及 LM 损失[1] 来作为偏好损失、质量损失以及生成

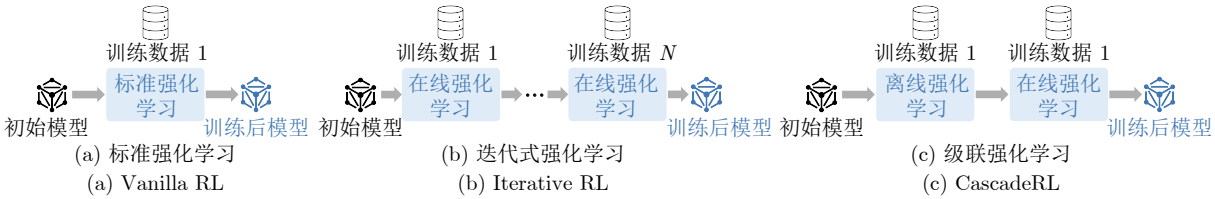


图 1 不同强化学习框架示意图

Fig. 1 Illustration of different RL frameworks

损失.

DPO 损失作为偏好损失的目的在于帮助模型学习不同响应之间的相对质量, 其损失函数定义如下:

$$\mathcal{L}_p = -\log\sigma\left(\beta\log\frac{\pi_\theta\left(y_c|x\right)}{\pi_0\left(y_c|x\right)} - \beta\log\frac{\pi_\theta\left(y_r|x\right)}{\pi_0\left(y_r|x\right)}\right) \tag{2}$$

其中, β 是 KL 惩罚系数; π_θ 是正在训练的策略模型, 从 π_0 初始化得到; σ 是 Sigmoid 函数.

除此之外, 作为质量损失的 BCO 损失旨在帮助模型学习每个响应的绝对质量, 其训练目标定义如下:

$$\mathcal{L}_q = \mathcal{L}_q^+ + \mathcal{L}_q^- \tag{3}$$

其中, $\mathcal{L}_q^+$ 和 $\mathcal{L}_q^-$ 分别表示针对正样本和负样本的损失. 这两个损失独立计算, 从而要求模型能够区分不同响应的绝对质量. 这些损失的定义如下:

$$\mathcal{L}_q^+ = -\log\sigma\left(\beta\log\frac{\pi_\theta\left(y_c|x\right)}{\pi_0\left(y_c|x\right)} - \delta\right) \tag{4}$$

$$\mathcal{L}_q^- = -\log\sigma\left(-\left(\beta\log\frac{\pi_\theta\left(y_r|x\right)}{\pi_0\left(y_r|x\right)} - \delta\right)\right) \tag{5}$$

其中, δ 表示奖励偏移, 通过历史奖励值的滑动平均计算得到, 引入的目的是提升训练稳定性.

最后, MPO 算法引入 LM 损失作为生成损失帮助模型学会正样本的生成方式, 该损失函数的定义如下:

$$\mathcal{L}_g = -\frac{\log\pi_\theta\left(y_c|x\right)}{|y_c|} \tag{6}$$

其中, $|y_c|$ 表示响应中的词元数量. 相比 DPO 算法, MPO 算法的核心优势在于引入了额外的质量损失和生成损失, 从而提升了训练的效果和稳定性. 具体而言, 如 Smaug[40] 中分析的, DPO 算法在训练过程中会同时降低模型关于正负样本的对数生成概率, 这是因为 DPO 算法只要求模型关于正负样本的对数生成概率的差值变大, 这就使得模型关于负样本的对数生成概率下降的同时, 产生了“挤压”的作用, 使得正样本的对数生成概率也同步下降, 只是下降速度更慢, 而这一问题导致 DPO 算法的训练效果和稳定性受到影响. 正如 MPO 算法的原文分析[2], 受益于额外引入的质量损失及生成损失带给正样本和负样本数据的各自明确的优化方向, 模型关于正负样本的生成概率不再受到“挤压”, 在 MPO 算法的训练过程中, 模型关于正样本的对数生成概率几乎不变, 而关于负样本的对数生成概率则持续下降, 因此取得了更好的训练效果和稳定性.

2) 在线强化学习. 在在线学习阶段, 本文使用 GSPO 算法[4] 进行训练, 因为该算法在稠密模型 (dense model) 上的性能和 GRPO[3] 接近而在混合专家 (mixture-of-experts, MoE) 模型上展现出更好的性能. 在训练过程中, 本文去掉了参考模型的 KL 约束, 因为早期实验结果表明, 引入参考模型的约束没有性能收益. 在训练过程中, 模型首先针对每个输入的问题随机采样 16 个响应, 每个响应根据格式 (正确 0.1 分, 错误 0 分) 和正确性 (正确 0.9 分, 错误 0 分) 获得结果奖励. 训练时, 使用策略梯度法进行梯度估计, 优势函数基于同一问题不同响应的结果奖励归一化得到:

$$\widehat{A}_i = \frac{r\left(x,\ y_i\right) - \text{mean}\left(\left\{r\left(x,\ y_i\right)\right\}_{i=1}^G\right)}{\text{std}\left(\left\{r\left(x,\ y_i\right)\right\}_{i=1}^G\right)} \tag{7}$$

其中, y_i 是针对问题 x 生成的第 i 个响应; G 是针对该问题生成的响应总数; $r\left(x,\ y_i\right)$ 是该响应的结果奖励; $\text{mean}\left(\cdot\right)$ 和 $\text{std}\left(\cdot\right)$ 表示均值和方差.

给定优势函数, GSPO 的优化目标如下 (这里省略了词元裁剪以简化公式):

$$\mathcal{L}_{\text{GSPO}}(\theta) = \frac{1}{G}\sum_{i=1}^{G}\widehat{A}_i\left(\frac{\pi_\theta\left(y_i|x\right)}{\pi_{\theta_{\text{old}}}\left(y_i|x\right)}\right)^{\frac{1}{|y_i|}} \tag{8}$$

其中, $\pi_\theta\left(y_i|x\right)$ 表示对于策略模型 θ 的响应 y_i 的生成概率; $|y_i|$ 表示响应中的词元数量; θ_{old} 表示更新前的策略模型. 相比 GRPO, GSPO 算法的主要改进是在词元层面进行归一化, 使得损失函数在响应层面计算, 而不是词元层面计算, 这一改进削弱了损失函数对每一个词元的敏感性, 从而提升了混合专家模型在强化学习过程中的稳定性. 具体而言, 混合专家模型在在线强化学习的连续参数更新阶段, 可能随着参数更新, 改变对部分词元的路由结果, 使得计算损失时的模型状态和响应采样时的模型状态出现偏差, 此时如果在词元层面计算损失函数, 会导致模型的训练稳定性受到影响.

第 3.2 节中还将对比 GRPO 与 GSPO 在级联强化学习中的训练效果, 因此这里也一并给出 GRPO 的优化目标:

$$\mathcal{L}_{\text{GRPO}}(\theta) = \frac{1}{G}\sum_{i=1}^{G}\frac{1}{|y_i|}\sum_{t=1}^{|y_i|}\widehat{A}_i\frac{\pi_\theta\left(y_{i,\,t}|x,\ y_{i,\,<t}\right)}{\pi_{\theta_{\text{old}}}\left(y_{i,\,t}|x,\ y_{i,\,<t}\right)} \tag{9}$$

其中, $y_{i,\,t}$ 表示响应 y_i 中的第 t 个词元; $y_{i,\,<t}$ 表示响应 y_i 中的前 $t-1$ 个词元; $\pi_\theta\left(y_{i,\,t}|x,\ y_{i,\,<t}\right)$ 表示对于策略模型 θ, 其中的词元 $y_{i,\,t}$ 的生成概率.

需要强调的是, 虽然本文在介绍级联强化学习策略的过程中提到离线阶段使用 MPO 算法, 在线阶段使用 GSPO 算法, 但这种训练策略本质上不和

任何训练算法绑定, 使用者可以根据实际场景切换更适合的算法. 本文第 3.2 节的实验表明, 尽管不同的级联强化学习实例化方案存在一定性能差异, 但是整体上都可以用更低的成本取得和单一的在线强化学习方案接近甚至更优的性能.

与使用单一强化学习范式直接训练的模型相比, 级联强化学习具有以下优势: a) 更好的训练稳定性: 在离线强化学习阶段, 轨迹收集与模型参数更新的过程是解耦的, 有效缓解了诸如奖励入侵 (reward hacking) 等问题. 本文在在线强化学习阶段通过实验观察到, 更强大的模型表现出更稳定和健壮的训练动态. 因此, 离线强化学习阶段获得的性能增益可以进一步提升下一阶段训练的稳定性, 降低了模型对算法的敏感性. b) 更高的训练效率: 在离线强化学习阶段, 生成的离线轨迹可以在不同模型的训练过程中共享, 从而均摊在线强化学习过程中昂贵的轨迹采样成本. 同时, 经过离线强化学习训练的模型在下一阶段的在线训练中可以用更少的训练步数达到更好的性能, 从而进一步降低训练成本. 值得注意的是, 在一次模型的发布过程中, 往往需要进行多次训练迭代才能最终定版, 且对于以 Qwen 系列[41] 和 InternVL 系列[6] 为代表的模型, 在一次发布中会涉及多个模型尺寸, 因此本文对离线强化学习数据构建阶段的成本均摊的考虑方式是具备现实意义的. 除此之外, 抛开成本均摊带来的效率收益, 本文提出的级联强化学习策略仍然可以带来更高的收益, 且训练成本加上数据构建的成本和单纯使用在线强化学习进行长轮次训练的成本接近. c) 更高的性能上限: 如第 3.3.3 节所示, 通过 MPO 微调后的模型在随后的在线强化学习阶段中, 仅需更少的训练步骤即可达到更高的性能上限.

3) 与已有方法的对比. 值得注意的是, 目前已经有一部分多模态大模型在训练流程中引入了和级联强化学习类似的迭代式强化学习策略. 然而, 如图 1, 迭代式强化学习和级联强化学习是不同的两种训练策略. 其中, 以 GLM-4.5V[27] 和 Keye-VL[28] 为代表的工作虽然引入了迭代式强化学习, 但是其目的在于通过多个阶段的强化学习, 每个阶段基于不同的训练数据提升模型不同维度的性能, 其核心动机在于提升模型基于不同数据的性能广度. 而本文提出的级联强化学习的目的是在同一批训练数据的基础上通过离线强化学习和在线强化学习相结合的方式以更低的训练成本达到更好的训练效果, 其核心动机在于提升模型基于同一批数据的性能深度. 在方法层面, 这些工作在每个阶段都使用相同的在线或离线强化学习算法, 并没有探索将不同的强化学习算法结合的方式. 同时需要强调的是, 本文算法和迭代式强化学习的策略并不冲突, 该策略中的强化学习算法可以替换为级联强化学习来进一步提升训练效率.

3 实验结果

3.1 训练细节

本节提供更多的训练细节. 具体而言, 级联强化学习包含离线强化学习和在线强化学习两个阶段. 给定一组训练数据, 先使用离线强化学习算法在这组数据上完整训练一个轮次, 随后在此基础上使用在线强化学习算法完整训练第二个轮次, 训练时只继承上一阶段的模型权重, 不需要继承优化器和学习率调度器的状态. 在离线强化学习算法训练前, 需要先用初始化模型采样响应来完成离线训练数据的构建, 然而实践中, 不同模型的训练可以复用这一批训练数据, 因此引入的数据成本可以忽略不计.

在离线强化学习阶段, 本文选用 MPO 算法[2], 其中相对质量损失的权重设置为 0.8, 质量损失的权重设置为 0.2, 生成损失的权重设置为 1.0. 在训练时, 本实验采用 AdamW 优化器, 学习率设置为 2×10^{-7}, 使用余弦学习率调度策略, 预热比例 (warmup ratio) 为 0.03. 训练过程中使用 FSDP (fully sharded data parallel)、CPU 卸载 (CPU offload)、梯度检查点 (gradient checkpointing) 等技术来节省显存消耗. 单卡训练的批大小 (batch size) 设置为 1, 通过梯度累积实现等效的全局批大小 (global batch size) 为 256. 模型的最大序列长度设置为 16 384 个词元.

在在线强化学习阶段, 本文选用 GSPO 算法[4]. 在训练时, 策略网络使用 AdamW 优化器进行训练, 学习率设置为 1×10^{-6}, 使用余弦学习率调度策略, 预热比例为 0.03. 训练过程中使用 FSDP、CPU 卸载、梯度检查点等技术来节省显存消耗. 值得注意的是, 本实验中未引入 KL 散度进行约束, 对应的 KL 散度的系数均设置为 0. 在响应采样阶段, 单次采样的批大小为 512 个问题, 且针对每个 prompt 生成 16 条候选响应, 总计生成 8 192 条响应, 每条至多 32 768 个词元, 用于构成组内样本. 在策略更新阶段, 迷你批大小 (mini-batch size) 设置为 512. 为提高硬件利用率并适应不同样本长度, 模型计算对数概率阶段启用了动态批大小机制. 对于超过长度限制的样本不做特殊处理, 直接将奖励值设置为 0. 计算奖励值时, 答案正确得 0.9 分, 格

式正确得 0.1 分.

3.2　分析性实验

本节将基于 M3CoT[5] 的训练集和验证集进行一系列分析性实验, 对有监督学习、离线强化学习以及在线强化学习在模型训练中的效果进行对比, 并基于这些实验结果总结出一系列结论. 首先在第 3.2.1 节中介绍实验设置, 随后在第 3.2.2 节中提供具体的实验结果和实验分析, 最后在第 3.2.3 节中分析不同训练算法训练过程中模型关于正负样本的对数生成概率的变化情况, 从模型训练动力学的角度对不同训练算法进行分析.

3.2.1　实验设置

实验将围绕 M3CoT 训练集中的 7 861 条数据以及验证集中的 1 108 条数据展开. 为了节省实验成本, 在 InternVL3.5-4B-Instruct[6] 上进行训练. M3CoT 是一个针对多模态推理的数据集, 包含 K12 难度的学科、数学以及视觉常识推理数据. 在实验过程中, 将使用全部训练数据的实验称为全量数据设置, 只使用一半训练数据 (~ 3 930 条数据) 的实验称为有限数据设置. 由于有监督学习、离线强化学习以及在线强化学习使用的训练数据输入并不相同, 为了保证不同实验结果之间的可比性, 通过控制模型训练过程中涉及的响应数量来对齐训练的样本量, 并使用相同的学习率 (1×10^{-6}) 进行训练. 具体而言, 在有限数据设置下, 在线强化学习阶段基于 ~ 3 930 条多模态问题进行训练, 每个问题采样 16 条回答, 因此每个轮次 (episode) 总共在 62.9 K 条响应上得到了监督信号. 而离线强化学习阶段每个样本包含两条响应, 所以基于约 32.2 K 条样本训练. 类似地, 有监督微调阶段每个样本只包含一条响应, 所以在 59.2 K 条样本上进行模型训练. 在全量数据设置下, 本文通过类似的方法来对齐训练样本量. 在构建离线强化学习和有监督微调的训练数据时, 使用 InternVL3.5-4B-Instruct 在全量数据上采样轨迹, 每个问题和在线强化学习阶段一样采样 16 条回答. 在构建离线强化学习所需要的偏好对样本时, 将回答正确的样本作为正样本, 回答错误的样本作为负样本. 对于有监督微调所需要的训练数据, 将轨迹中回答正确的内容作为训练样本. 为了消除训练问题质量对实验结果的影响, 提前下采样了 ~ 3 930 条数据, 所有有限数据设置下的实验均基于这些数据构建训练样本. 需要注意的是, 由于并不能保证模型针对每个问题的正负样本数都是均衡的, 所以构建得到的数据集数量并不能完全对齐. 除此之外, 为了证明实验结论的一般性, 使用两种离线强化学习 (DPO[1] 和 MPO[2]) 以及两种在线强化学习 (GRPO[3] 和 GSPO[4]) 算法进行分析性实验.

3.2.2　实验分析

1) 强化学习算法与有监督微调的训练效果对比. 表 1 提供了不同数据量下, 有监督微调、离线强化学习以及在线强化学习在 M3CoT 上的训练响应数量、显卡小时数以及得分 (正确率). 其中, Half 表示有限数据设置下 (即只使用一半的训练样本量) 训练一轮, Half × 2 表示有限数据设置下训练两轮, Full 表示全量数据设置下训练一轮. 实验结果表明, 在有限数据设置只训练一轮的情况下, 所有算法都能为模型带来性能提升, 其中离线强化学习算法带来的性能收益都高于有监督微调和在线强化学习算法, 且此时在线强化学习带来的性能收益略低于有监督微调带来的性能收益. 而在有限数据设置训练两轮的实验中, 所有算法都相比只训练一轮带来了进一步的性能提升, 其中在线强化学习算法的整体收益最高, 超过了另外两种算法, 这表明在线强化学习的收敛速度慢于另外两种训练算法. 对于全量数据的实验, 不同训练算法带来的性能趋势与有限数据训练两轮的实验中的趋势相同. 然而值得注意的是, 在相同数据量下, 在线强化学习在全量数据

表 1　不同算法的训练结果对比

Table 1　Comparison of training results for different algorithms

模型	训练响应数量	显卡小时数	得分
基线模型			
InternVL3.5-4B-Instruct	—	—	57.5
有监督微调			
SFT (Half)	~ 59.2 K	5.3	64.4
SFT (Half × 2)	~ 118.4 K	8.0	66.6
SFT (Full)	~ 124.3 K	8.0	67.0
离线强化学习算法			
DPO (Half)	~ 64.5 K	4.5	67.3
DPO (Half × 2)	~ 129.1 K	8.5	70.2
DPO (Full)	~ 128.2 K	8.5	69.5
MPO (Half)	~ 64.5 K	4.6	68.0
MPO (Half × 2)	~ 129.1 K	8.5	70.8
MPO (Full)	~ 128.2 K	8.5	70.8
在线强化学习算法			
GRPO (Half)	~ 62.9 K	65.2	63.9
GRPO (Half × 2)	~ 125.8 K	141.4	69.8
GRPO (Full)	~ 125.8 K	144.4	71.5
GSPO (Half)	~ 62.9 K	65.4	63.4
GSPO (Half × 2)	~ 125.8 K	159.3	67.3
GSPO (Full)	~ 125.8 K	168.6	72.0

训练一轮的性能 (经 GSPO 算法训练后取得 72.0) 优于有限数据训练两轮的性能 (经 GSPO算法训练后取得 67.3), 而离线强化学习和有监督微调的额外收益则没有那么明确, 这表明, 在时间和数据预算充足的情况下, 在线强化学习有着更高的性能收益.

从训练成本的角度来说, 离线强化学习和有监督微调在相同数据量上的训练成本基本相同, 在全量数据设置下, 一轮的耗时均为 8 ~ 9 显卡小时数, 而在线强化学习的成本则更高, 一轮的耗时为 140 ~ 160 显卡小时数. 这表明, 在相同的数据量上进行训练, 尽管在线强化学习有着更高的性能收益, 其训练代价也更加昂贵. 值得注意的是, 离线强化学习需要提前离线采样轨迹用于训练, 这一部分的时间成本约为 127 显卡小时数, 然而这部分数据在构建完成后, 可以支撑全部的离线强化学习实验, 即 DPO、MPO 以及后续的级联强化学习的训练, 因此这部分成本被这些实验完全均摊, 而对于在线强化学习来说, 这部分成本是每个实验都需要承担的.

2) 不同强化学习算法的训练效果对比. 表 1 提供了两种离线强化学习 (DPO 和 MPO) 以及两种在线强化学习 (GRPO 和 GSPO) 的实验结果. 这些结果表明, 尽管不同算法的优化目标有所差异, 但是对于同类算法而言, 其性能变化趋势以及训练成本是接近的. 例如, 对于离线强化学习算法而言, 模型收敛速度较快, 增加训练响应数量可以带来进一步收益, 但是有限数据训练两轮和全量数据训练一轮的性能差距不大, 表明其性能上限容易饱和; 对于在线强化学习算法, GSPO 和 GRPO 都收敛速度慢, 有限数据下训练一轮的性能甚至不如有监督微调, 但是上限高, 经过更长轮次的训练后, 性能优于另外两种训练算法.

3) 相同数据成本下级联强化学习的训练效果分析. 表 2 提供了使用级联强化学习与仅使用离线/在线强化学习算法的训练结果对比. 实验结果表明, 级联强化学习可以通过 68.3 显卡小时数的训练成本取得和训练成本 168.6 显卡小时数的在线强化学习算法 (GSPO) 相近甚至略优的性能. 值得注意的是, 在级联强化学习阶段, 离线强化学习和在线强化学习的两个子阶段都是在有限数据设置下基于同一组题目训练一轮. 而这样取得的训练结果超过了 GSPO 算法直接训练在有限数据设置下训练两轮, 并且达到了 GSPO 算法在全量数据下训练一轮的性能, 且时间成本降低 (从 168.6 显卡小时数降低至 68.3 显卡小时数). 这一结果表明, 级联强化学习不仅可以加速收敛, 保持性能上限, 还可以降低对训练数据的多样性要求, 用更低的数据成本达到更好的模型性能. 这一特性在复杂场景中尤为关键.

表 2 相同数据成本下不同算法的训练结果
Table 2 Training results of different algorithms under the same data cost

模型	训练响应数量	显卡小时数	得分
SFT (Half × 2)	~ 118.4 K	8.0	66.6
SFT (Full)	~ 124.3 K	8.0	67.0
MPO (Half × 2)	~ 129.1 K	8.5	70.8
MPO (Full)	~ 128.2 K	8.5	70.8
GSPO (Half × 2)	~ 125.8 K	159.3	67.3
GSPO (Full)	~ 125.8 K	168.6	72.0
MPO → GSPO	~ 127.4 K	68.3	72.2

注:"→"表示先基于 MPO (Half) 训练一轮, 再基于 GSPO (Half) 续训一轮.

4) 不同级联强化学习实例的训练效果对比. 表 3 对比了不同的级联强化学习的实例化方案, 使用不同的离线强化学习和在线强化学习算法进行不同顺序的组合, 并提供了先做增量式有监督微调, 再做在线强化学习的实验结果作为额外的基线. 默认情况下, 在每个阶段都基于有限数据设置训练一个轮次. 实验结果表明, 先离线强化学习后在线强化学习的训练范式普遍优于先在线强化学习后离线强化学习的训练范式. 除此之外, 离线强化学习阶段使用 MPO 算法的实验结果均优于 DPO 算法, 而在线强化学习阶段使用 GSPO 算法在多数情况下性能更优.

表 3 基于不同级联强化学习实例的训练结果
Table 3 Training results based on different CascadeRL instances

模型	训练响应数量	显卡小时数	得分
SFT → GRPO	~ 122.1 K	66.0	70.3
SFT → GSPO	~ 122.1 K	75.4	71.3
GRPO → DPO	~ 127.4 K	73.2	70.1
GRPO → MPO	~ 127.4 K	73.2	70.4
GSPO → DPO	~ 127.4 K	73.4	68.1
GSPO → MPO	~ 127.4 K	73.4	70.0
DPO → GRPO	~ 127.4 K	63.2	69.9
DPO → GSPO	~ 127.4 K	67.8	71.6
MPO → GRPO	~ 127.4 K	68.1	71.5
MPO → GSPO	~ 127.4 K	68.3	72.2
MPO (Full) → GRPO	~ 191.1 K	70.4	72.3
MPO (Full) → GSPO	~ 191.1 K	72.1	72.1

5) 不同强化学习范式对领域外泛化能力的影响. 表 4 提供了模型在 M3CoT 上训练完成后, 在其他推理类评测基准上进行评测的结果 (得分 (正确

率)). 虽然 M3CoT是一个多模态推理数据集, 但其中的题目主要围绕 K12 难度的学科和数学推理, 而 MMMU[42] 是研究生难度的学科推理基准, MathVerse[43] 是竞赛级别的数学推理基准, 因此和 M3CoT 的数据分布有较大差异, 可以作为领域外泛化能力的评测基准. 实验结果表明, 基于有监督微调训练得到的模型虽然在领域内基准取得了性能提升, 但是领域外基准的评测性能都出现了下降. 而基于强化学习算法训练的模型则没有下降或者有一定的性能提升. 其中, 基于本文提出的级联强化学习训练得到的模型整体的性能提升幅度最大, 表明级联强化学习策略相比标准强化学习具有更好的领域外泛化能力.

表 4　不同算法训练后的模型在域外评测集的性能
Table 4　Performance of models trained with different algorithms on out-of-domain evaluation sets

模型	M3CoT	MMMU	MathVerse
InternVL3.5-4B-Instruct	57.5	62.3	42.4
SFT (Full)	67.0	59.6	39.8
MPO (Full)	70.8	60.7	44.2
GSPO (Full)	72.0	63.3	43.5
GSPO → MPO	70.0	60.4	43.5
MPO → GSPO	72.2	62.8	44.8

6) 不同超参数对训练效果的影响. 表 5 基于离线强化学习算法 MPO 和在线强化学习算法 GSPO 比较了每个问题采样不同的响应数量时的模型训练效果 (得分 (正确率)). 实验结果表明, 两种算法对超参数设置的趋势相同, 当每个问题采样 4 条响应时, 模型的性能提升都非常有限, 本文认为这是因为此时模型的 Pass@N 性能, 即尝试回答 N 次时至少一次回答正确的概率不够高, 限制了模型训练时的性能上限, 从而降低了其训练效果. 而当每个问题采样的响应数量增加到 8 或 16 时, 模型的验证集性能都取得进一步提升, 然而当采样 32 条响应时, 已经出现边界效应, GSPO 算法的训练效果甚至开始下降, 本文认为这是模型开始向自身输出中的某些特殊模式过拟合导致的. 考虑到采样 32 条响应时昂贵的实验成本以及其潜在的过拟合风险, 本文将选择在每个问题采样 16 条响应的设置下进行后续实验. 除此之外, 表 5 也提供了不同 KL 散度约束的模型训练效果, 其中 coef 表示 KL 散度损失的权重系数, coef = 0 时表示没有 KL 散度损失. 结果表明, 是否引入 KL 散度对模型训练效果影响不大, 在设置 KL 散度约束强度较大时, 甚至还会限制模型的探索空间, 进而轻微减弱训练效果. 表 6 中提供了不同离线和在线强化学习比例下的训练效果. 默认设置为 1.0 : 1.0, 即在两种算法下都基于有限数据训练完整的一个轮次, 而 1.5 : 0.5 则表示离线阶段基于相同数据训练 1.5 个轮次, 在线阶段训练 0.5 个轮次. 本节控制比值的和为 2.0 来保证不同比例下模型训练过程中涉及的响应数量是几乎相同的, 从而保证对比的公平性. 结果表明, 增加离线阶段的训练比例, 在同一组数据上增加训练轮次 (1.5 : 0.5), 相比原始级联强化学习方案 (1.0 : 1.0) 的性能更差, 而增加在线阶段的训练比例可以为模型带来少量收益. 考虑到性能收益不高, 本文选择 1.0 : 1.0 的比例来维持方法简洁性.

表 5　不同超参数设置对训练结果的影响
Table 5　Impact of different hyperparameter settings on training results

实验设置	MPO	GSPO
InternVL3.5-4B-Instruct	57.5	57.5
每个问题采样的响应数量		
每个问题采样 4 条响应	63.4	60.1
每个问题采样 8 条响应	67.6	63.7
每个问题采样 16 条响应	70.8	72.0
每个问题采样 32 条响应	71.2	71.7
KL 散度约束强度		
coef = 0	—	71.7
coef = 0.001	—	71.7
coef = 0.100	—	71.8
coef = 1.000	—	71.1

表 6　不同比例离线和在线强化学习的训练结果
Table 6　Training results of different offline and online reinforcement learning ratios

实验设置	显卡小时数	GSPO
InternVL3.5-4B-Instruct	—	57.5
2.0 : 0	~ 8.5 K	70.8
1.5 : 0.5	~ 48.1 K	71.4
1.0 : 1.0	~ 68.3 K	72.2
0.5 : 1.5	~ 130.2 K	72.3
0 : 2.0	~ 168.6 K	72.0

3.2.3　训练过程分析

本节用第 3.2.2 节中基于 InternVL3.5-4B-Instruct 采样的正负响应数据进行实验, 根据模型关于这些正负样本的对数生成概率的变化情况来分析不同的训练算法对模型的影响. 图 2 中展示了使用不同算法训练 InternVL3.5-4B-Instruct 的过程中, 策略模型关于起始模型的正负样本的对数生成概率

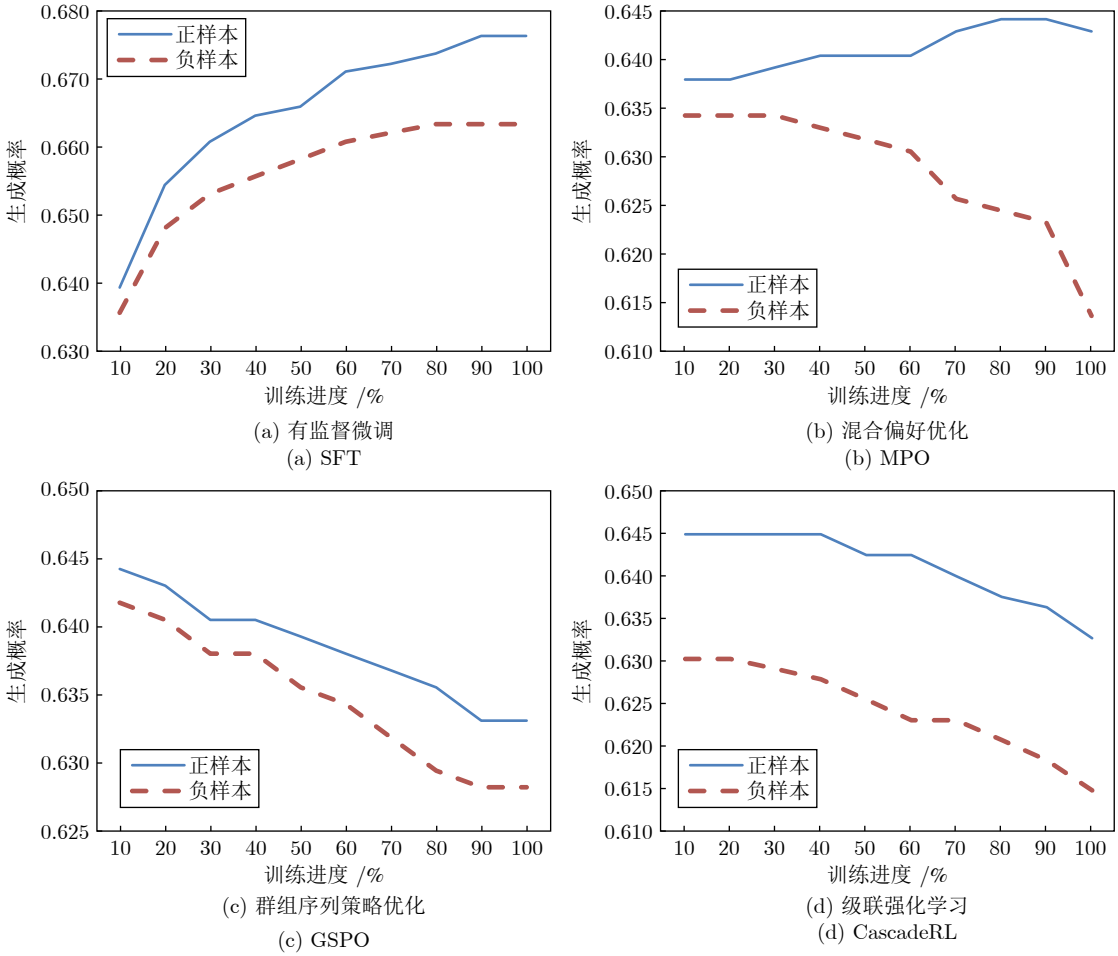


图 2 模型训练过程中关于起始模型的正负响应的生成概率

Fig. 2 Generation probabilities of positive and negative responses relative to the initial model during model training

的变化情况. 其中, 级联强化学习展示的训练曲线是离线阶段结束后的部分, 其离线阶段曲线与 MPO 一致. 具体而言, 使用第 3.2.1 节中提到的全量数据训练一轮的设置进行 SFT、MPO 以及 GSPO 的训练, 通过其中提到的有限数据设置下基于离线强化学习算法 MPO 和在线强化学习算法 GSPO 分别进行一轮训练的方式进行级联强化学习的训练. 在绘制 SFT、MPO 以及 GSPO 的训练过程中模型关于正负样本的对数生成概率曲线时, 根据训练进度的百分比 (每 10% 保存一个数据点) 来绘制. 而在绘制级联强化学习的对数生成概率曲线时, 绘制的是其在线强化学习阶段的曲线, 这是因为级联强化学习的离线强化学习阶段的对数生成概率曲线和 MPO 的对数生成概率曲线的前半段几乎完全相同. 需要提醒的是, MPO 基于全量数据训练, 而级联强化学习则基于全量数据的一半, 先使用 MPO 训练, 再使用 GSPO 在完全相同的这一半数据上进行第二轮的训练.

如图 2 所示, 有监督微调会导致模型关于正负样本的对数生成概率同步增加, 本文认为这是因为有监督微调的过程中只有正样本导致的, 使得模型的输出空间在训练过程中覆盖了正样本的同时, 也同步地覆盖了正样本附近和正样本形式上类似但是关键词元上不同的负样本的生成概率, 这种输出空间中同时覆盖正负样本的行为导致了模型在域外性能的下降.

对于强化学习算法而言, 负样本的对数生成概率均呈下降趋势. 对于在线强化学习算法 GSPO 而言, 正样本的对数生成概率也是下降趋势, 我们认为这是因为模型在训练过程中基于在线采样的响应进行训练, 因此升高的是训练过程中模型生成的响应的对数生成概率, 这部分概率的升高挤压了其他响应的对数生成概率, 导致了其他响应中的正样本的对数生成概率下降. 但是对于离线强化学习算法 MPO 而言, 正样本的对数生成概率呈上升趋势, 这是因为该算法在训练过程中引入了针对正样本的生

成损失. 无论哪种算法, 负样本的对数生成概率的下降趋势表明, 强化学习算法中引入的负样本为模型带来了分布裁剪的效果, 通过降低负样本生成的概率, 提高模型生成的响应分布的整体质量.

对于本文提出的级联强化学习, 其正样本和负样本的对数生成概率变化趋势与单一强化学习范式的变化趋势类似. 但值得注意的是, 相比直接进行在线强化学习, 级联强化学习在进入在线阶段后, 起点的正负样本的对数生成概率的差值更大, 说明此时在采样过程中得到的主要是正样本区域附近的响应, 使得模型在训练过程中, 能够直接在更高质量的采样分布上进行训练, 这一现象解释了离线阶段是如何加速在线阶段的收敛速度. 整体而言, 级联强化学习的训练流程可以看成是先基于离线强化学习算法快速修剪模型分布, 随后在更高质量的模型分布上训练以更低的成本取得更好的训练效果.

3.3 大规模实验

本节将对本文提出的级联强化学习策略进行更大规模的实验验证, 证明在更强的数据设置以及更广的模型尺寸下, 级联强化学习仍能带来性能收益.

3.3.1 实验设置

本节将基于 InternVL3.5 开源的全部八个指令微调后的模型进行后训练, 用于证明级联强化学习在不同规模的模型上的可扩展性. 需要注意的是, InternVL3.5 开源的模型中包含了稠密模型以及混合专家模型, 因此这一部分实验也可以证明本文提出的级联强化学习在不同模型架构上的有效性. 其中, 离线强化学习阶段使用的训练算法是 MPO[2], 训练数据使用的是 MMPR-v1.2[2], 包含了约 200 K 的偏好数据对. 本文直接复用了其中提供的轨迹, 没有单独采样, 从而节省了轨迹采样阶段的成本. 在线强化学习阶段使用的训练算法是 GSPO[4], 训练数据使用的是基于 MMPR-v1.2 过滤得到的 MMPR-Tiny. 具体而言, 本文用 MMPR-v1.2 中提供的轨迹及其正确性标签进行过滤, 只保留其中正确率在 0.2 到 0.8 之间的推理题目, 最终过滤得到约 70 K 的推理题目用于在线强化学习阶段的训练.

在评测基准方面, 在七个多模态推理评测基准上评测训练后的模型性能, 并和近期的一系列开源模型[27−28, 44−49] 和闭源模型[50−53] 进行性能对比. 其中, MMMU[42] 是一个多学科推理评测基准; MathVista[54]、MathVision[55]、MathVerse[43]、DynaMath[56] 以及 WeMath[57] 是数学推理评测基准; LogicVista[58] 是逻辑推理评测基准. DynaMath 评测的是该基准中提出的最坏情况准确率 (worst case accuracy), WeMath 评测的是该基准中提出的严格得分 (strict score), 其他评测基准使用的是准确率 (accuracy). 此外, MMMU 测的是 val 子集, MathVista 测的是 mini 子集, MathVerse 测的是 vision-only 子集.

3.3.2 主要结果

1) 横向对比. 表 7 提供了基于级联强化学习训练得到的 InternVL3.5-CascadeRL 和其他开源及闭源模型的横向对比. 性能指标采用的是模型在七个多模态推理评测基准 (MMMU、MathVista、MathVision、MathVerse、DynaMath、WeMath 以及 LogicVista) 上的均分. 部分结果来自 OpenCompass[59]. 实验结果表明 InternVL3.5-CascadeRL 在各个评测基准上都达到了当前开源模型中的领先性能. 与 InternVL3 相比, InternVL3.5 在所有模型规模上相较于同级别模型的推理性能提升超过 10 个点. 在大尺寸模型上, InternVL3.5-241B-A28B-CascadeRL 在七个多模态推理评测基准上的平均得分为 66.9, 超过了更大尺寸的 Step3-321B-A38B 的 64.4. 在中等尺寸的模型上, InternVL3.5-30B-A3B-CascadeRL 和 InternVL3.5-14B-CascadeRL 在 MMMU 上的得分分别为 75.6 和 73.3, 均超过了更大尺寸的 InternVL3-78B 的 72.2. 在轻量级模型中, InternVL3.5 同样取得了相较于开源基线模型的性能提升. 其中, 4B 和 2B 尺寸的模型都超越其他基线模型 10 个点以上的性能. 这些提升主要得益于级联强化学习, 展示了其在推理任务上的有效性和可扩展能力.

2) 纵向对比. 图 3 提供了 InternVL3.5-Instruct 模型在不同训练阶段之后的纵向性能对比. 其中, 柱状图高度表示模型在 MMMU、MathVista、MathVision、MathVerse、DynaMath、WeMath 以及 LogicVista 上的均分. 本文提出的级联强化学习包含离线强化学习和在线强化学习两个阶段, 第一阶段结束后的模型记作 InternVL3.5-MPO, 第二阶段结束后的模型记作 InternVL3.5-CascadeRL. 实验结果表明, InternVL3.5 的性能在经过 MPO 阶段后仍能进一步提升, 在推理任务上平均增幅可达 +3.5%. 而在 MPO 训练后的模型基础上, 级联强化学习可以进一步为所有尺寸和所有架构的模型提供额外的性能收益. 例如, InternVL3.5-2B-CascadeRL 在推理任务上的平均性能提升达 12.2%, 相较于指令微调模型具有性能优势. 在更大规模的模型上也可以观察到类似的效果, 例如 InternVL3.5-241B-A28B-CascadeRL 的综合推理性能提升了 6.5%. 这些消融实验验证了级联强化学习的有效性和可扩展性. 尽管算法在更大规模的模型上的收益

表 7 不同模型的多模态推理性能对比

Table 7 Comparison of multimodal reasoning performance across different models

模型	MMMU (val)	MathVista (mini)	MathVision	MathVerse (vision-only)	DynaMath	WeMath	LogicVista	平均分
InternVL3-1B[60]	43.4	45.8	18.8	18.7	5.8	13.4	29.8	25.1
InternVL3.5-1B-CascadeRL	44.2	59.3	27.3	37.8	17.2	21.5	29.3	33.8
Ovis-2B[61]	45.6	64.1	17.7	29.4	10.0	9.9	34.7	30.2
Qwen2.5-VL-3B[62]	51.2	61.2	21.9	31.2	13.2	22.9	40.3	34.6
InternVL3-2B[60]	48.6	57.0	21.7	25.3	14.6	22.4	36.9	32.4
InternVL3.5-2B-CascadeRL	59.0	71.8	42.8	53.4	31.5	48.5	47.7	50.7
Ovis-4B[61]	49.0	69.6	21.5	38.5	18.0	16.9	35.3	35.5
MiniCPM-V-4-4B[63]	51.2	66.9	20.7	18.3	14.2	32.7	30.6	33.5
InternVL2.5-4B[64]	51.8	64.1	18.4	27.7	15.2	21.2	34.2	33.2
InternVL3.5-4B-CascadeRL	66.6	77.1	54.4	61.7	35.7	50.1	56.4	57.4
MiniCPM-o2.6[63]	50.9	73.3	21.7	35.0	10.4	25.2	36.0	36.1
Ovis-8B[61]	57.4	71.8	25.9	42.3	20.4	27.2	39.4	40.6
Qwen2.5-VL-8B[62]	55.0	67.8	25.4	41.1	21.0	35.2	44.1	41.4
MiMo-VL-RL-8B[44]	66.7	81.5	60.4	71.5	45.9	66.3	61.4	64.8
Keye-VL-8B[28]	71.4	80.7	50.8	54.8	37.3	60.7	54.8	58.6
GLM-4.1V-9B[27]	68.0	80.7	54.4	68.4	42.5	63.8	60.4	62.6
InternVL3-8B[60]	62.7	71.6	29.3	39.8	25.5	37.1	44.1	44.3
InternVL3.5-8B-CascadeRL	73.4	78.4	56.8	61.5	37.7	57.0	57.3	60.3
Gemma-3-12B[65]	55.2	56.1	30.3	21.1	20.8	33.6	41.2	36.9
Ovis2-16B[61]	60.7	73.7	30.1	45.8	26.3	45.0	47.4	47.0
InternVL3-14B[60]	67.1	75.1	37.2	44.4	31.3	43.0	51.2	49.9
InternVL3.5-14B-CascadeRL	73.3	80.5	59.9	62.8	38.7	58.7	60.2	62.0
Kimi-VL-A3B-2506[66]	64.0	80.1	54.4	54.6	28.1	42.0	51.4	53.5
InternVL3.5-30B-A3B-CascadeRL	75.6	80.9	55.7	60.4	36.5	48.4	55.7	59.0
Gemma-3-27B[65]	64.9	59.8	39.8	34.0	28.5	37.9	47.3	44.6
Ovis2-34B[61]	66.7	76.1	31.9	50.1	27.5	51.9	49.9	50.6
Qwen2.5-VL-32B[62]	70.2	74.8	38.1	57.6	35.1	46.5	52.6	53.6
Skywork-R1V3-38B[45]	76.0	77.1	52.6	59.6	35.1	56.5	59.7	59.5
InternVL3-38B[60]	70.1	75.1	34.2	48.2	35.3	48.6	58.4	52.8
InternVL3.5-38B-CascadeRL	76.9	81.9	63.7	67.6	41.7	64.8	65.3	66.0
GPT-5-nano-20250807[50]	72.6	73.1	59.7	66.6	47.9	59.4	57.5	62.4
GPT-5-20250807[50]	81.8	81.9	72.0	81.2	60.9	71.1	70.0	74.1
Claude-3.7-Sonnet[51]	75.0	66.8	41.9	46.7	39.7	49.3	58.2	53.9
Gemini-2.0-Pro[52]	69.9	71.3	48.1	67.3	43.3	56.5	53.2	58.5
Gemini-2.5-Pro[52]	74.7	80.9	69.1	76.9	56.3	78.0	73.8	72.8
Doubao-1.5-Pro[53]	73.8	78.6	51.5	64.7	44.9	65.7	64.2	63.3
GLM-4.5V[27]	75.4	84.6	65.6	72.1	53.9	68.8	62.4	69.0
QvQ-72B-Preview[46]	70.3	70.3	34.9	48.2	30.7	39.0	58.2	50.2
Qwen2.5-VL-72B[62]	68.2	74.2	39.3	47.3	35.9	49.1	55.7	52.8
Step3-321B-A38B[67]	74.2	79.2	64.8	62.7	50.1	59.8	60.2	64.4
InternVL3-78B[60]	72.2	79.0	43.1	51.0	35.1	46.1	55.9	54.6
InternVL3.5-241B-A28B-CascadeRL	77.7	82.7	63.9	68.5	46.5	62.3	66.7	66.9

(241B-A28B 模型 +6.5%, 38B 模型 +8.3%, 14B 模型 +7.1%) 低于更小规模的模型 (2B 模型 +12.2%), 但是本文认为这是一个合理的结果, 因为更大规模的模型训练更充分, 在相同数据上的损失更低, 因

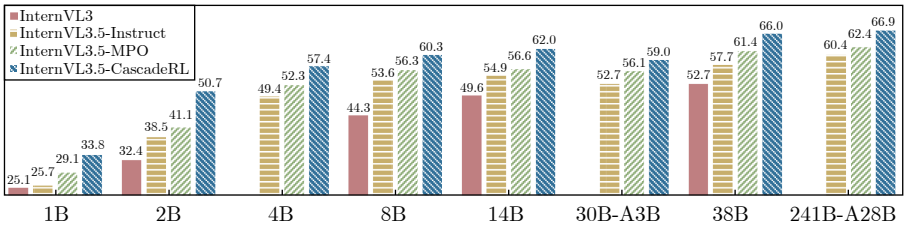


图 3　级联强化学习各阶段对模型推理性能的影响

Fig. 3　Impact of different CascadeRL stages on model reasoning performance

此熵值也更低, 在增加了对训练数据拟合程度和响应确定性的同时, 也一定程度上损失了输出多样性, 使得强化学习阶段的探索空间减小, 同时模型本身基础性能更高, 因此上升空间更少, 进而导致性能收益更少. 表 8 进一步提供了模型在每个评测基准上的性能得分, 实验结果表明模型在学科推理、数学推理以及逻辑推理的评测基准上都取得了性能收益. 值得注意的是, 尽管 MMPR-v1.2 数据中并不包含研究生难度的推理题目, 但是仍然在 MMMU 这个研究生难度的评测基准上取得了性能提升, 本文认为这是因为模型在强化学习的训练过程中, 提升了每一步的推理准确性, 从而在更难的评测基准上泛化出了更好的推理性能. 第 4 节将通过模型的输出样例具体地展示模型推理思维链中每一步的准确性提升.

3.3.3　消融实验

表 9 展示了级联强化学习与 MPO 和 GSPO 在效率与效果上的对比. 对于 MPO 和级联强化学

表 8　模型在不同训练阶段结束后的多模态推理性能对比

Table 8　Comparison of multimodal reasoning performance after different training stages

模型	MMMU (val)	MathVista (mini)	MathVision	MathVerse (vision-only)	DynaMath	WeMath	LogicVista	平均分
InternVL3.5-1B-Instruct	37.2	48.6	15.8	27.0	8.4	13.9	29.1	25.7
+ MPO	40.3	50.5	22.0	32.1	9.0	16.8	32.7	29.1
+ CascadeRL	44.2	59.3	27.3	37.8	17.2	21.5	29.3	33.8
InternVL3.5-2B-Instruct	53.0	60.8	27.0	39.6	19.8	28.1	41.2	38.5
+ MPO	54.3	62.6	34.2	46.4	21.0	28.1	40.9	41.1
+ CascadeRL	59.0	71.8	42.8	53.4	31.5	48.5	47.7	50.7
InternVL3.5-4B-Instruct	64.3	71.4	40.5	50.0	30.7	35.6	53.5	49.4
+ MPO	65.4	71.7	48.0	54.9	30.7	39.8	55.9	52.3
+ CascadeRL	66.6	77.1	54.4	61.7	35.7	50.1	56.4	57.4
InternVL3.5-8B-Instruct	68.1	74.2	46.4	55.8	30.7	46.0	53.9	53.6
+ MPO	71.2	75.9	52.6	54.8	33.1	47.7	58.6	56.3
+ CascadeRL	73.4	78.4	56.8	61.5	37.7	57.0	57.3	60.3
InternVL3.5-14B-Instruct	71.8	73.4	48.7	55.5	31.9	45.7	57.5	54.9
+ MPO	73.3	74.0	53.0	57.5	32.3	45.2	60.9	56.6
+ CascadeRL	73.3	80.5	59.9	62.8	38.7	58.7	60.2	62.0
InternVL3.5-30B-A3B-Instruct	72.3	73.3	45.1	50.4	31.9	39.7	56.4	52.7
+ MPO	71.7	75.3	50.7	58.5	32.9	43.7	59.7	56.1
+ CascadeRL	75.6	80.9	55.7	60.4	36.5	48.4	55.7	59.0
InternVL3.5-38B-Instruct	73.9	75.9	58.2	59.0	29.7	47.5	60.0	57.7
+ MPO	76.9	80.5	56.3	59.4	36.9	55.6	64.2	61.4
+ CascadeRL	76.9	81.9	63.7	67.6	41.7	64.8	65.3	66.0
InternVL3.5-241B-A28B-Instruct	76.2	80.1	55.6	61.7	36.5	49.7	63.3	60.4
+ MPO	76.0	82.2	55.3	64.1	38.3	51.3	69.4	62.4
+ CascadeRL	77.7	82.7	63.9	68.5	46.5	62.3	66.7	66.9

表 9 离线强化学习 (MPO)、在线强化学习 (GSPO) 以及 CascadeRL 的训练效率及性能对比

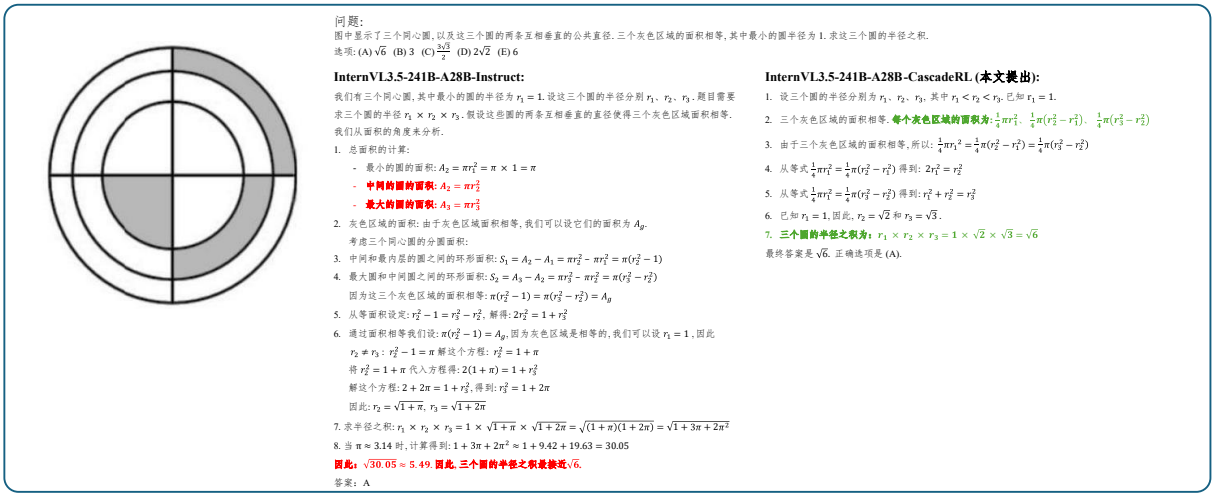
Table 9 Comparison of training efficiency and performance among offline RL (MPO), online RL (GSPO), and CascadeRL

模型	显卡小时数	MMMU (val)	MathVista (mini)	MathVision	MathVerse (vision-only)	DynaMath	WeMath	LogicVista	均分
InternVL3.5-8B-Instruct	—	68.1	74.2	46.4	55.8	30.7	46.0	53.9	53.6
+ MPO	~ 0.3 K	71.2	75.9	52.6	54.8	33.1	47.7	58.6	56.3
+ GSPO (1 episode)	~ 5.5 K	73.8	77.9	51.6	58.8	35.1	48.9	54.8	57.3
+ GSPO (2 episodes)	~ 11.0 K	72.0	78.1	51.6	58.5	35.7	54.1	57.0	58.2
+ CascadeRL	~ 5.8 K	73.4	78.4	56.8	61.5	37.7	57.0	57.3	60.3

习, 报告的是训练一个轮次后的结果; 对于 GSPO, 报告的是训练一个和两个轮次后的结果. 实验表明, MPO 仅需极少的显卡小时数即可取得 2.7% 的均分提升, 体现了离线强化学习收敛速度快的特性; GSPO 尽管能够带来更大幅度的性能提升, 但代价是更加昂贵的计算资源成本 (从 ~ 0.3 K 显卡小时数增加到 ~ 5.5 K 显卡小时数). 进一步增加 GSPO 的训练成本可以带来更多的性能收益, 使得模型的均分从 57.3 提升到 58.2, 体现了在线强化学习算法性能上限高但是训练成本昂贵的特性. 相比之下, 级联强化学习在仅需 GSPO 一半显卡小时数的情况下, 均分达到了 60.3, 实现了 6.7% 的均分提升, 兼具了收敛速度快、性能上限高的特性. 相比第 3.2 节中的实验, 这部分实验结果证明了在更大规模的训练设置下, 级联强化学习仍然可以以更低的成本带来更高的收益.

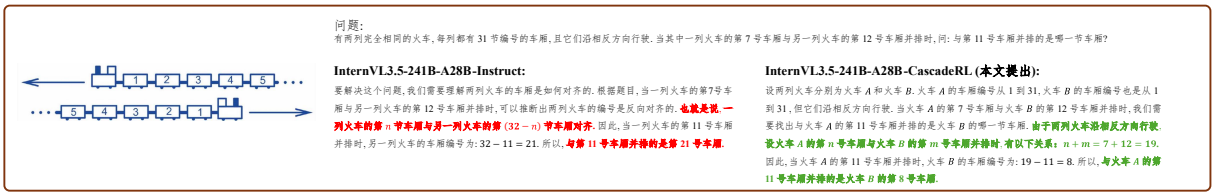
4 定性分析

本节提供了模型基于级联强化学习训练前后的输出来直观地展示模型训练前后推理能力的变化情况. 图 4(a) 中给出了模型训练前后都能回答正确的题目. 这个例子表明, 训练后的模型给出的思维链变得精简的同时也更加可靠. 这种收益来自模型训练过程中针对同一问题进行多次采样的操作, 对于训练前的模型, 虽然模型偶尔也能基于不可靠的思维链猜出正确答案, 但是这些不可靠的思维链更可



(a) 训练前后均能回答正确的输出样例

(a) Examples of outputs answered correctly both before and after training



(b) 训练后才能回答正确的输出样例

(b) Examples of outputs answered correctly only after training

图 4 模型训练前后的输出结果对比

Fig. 4 Comparison of model output results before and after training

能导出错误答案, 从而在训练过程中受到更多惩罚, 而可靠的思维链更可能导出正确答案, 从而在训练过程中受到更多奖励, 因此在训练过程中, 模型的输出分布逐渐向更可靠的思维链的方向收敛. 图 4(b) 给出了模型训练后才能回答正确的题目. 这个例子表明, 受益于推理过程中生成的更高质量的思维链, 模型的推理性能也得到了提升.

5　实验结果稳定性验证

本节对第 3.2.3 节中的实验结果的稳定性进行分析. 具体而言, 本节将第 3.2.3 节中的实验重复五次, 并根据五次实验的均值和标准差绘制包含了置信区间的模型训练过程中正负响应的生成概率变化曲线. 图 5 展示了这一生成概率在训练过程中的变化情况, 其中实线部分表示关于正样本的生成概率, 虚线部分表示关于负样本的生成概率, 阴影部分表示二者基于各自标准差计算得到的范围. 实验结果表明, 尽管训练结果具有一定的随机性, 但是模型训练过程中, 不同算法关于正负样本生成概率的变化趋势没有明显变化, 这一结果证明了第 3.2.3 节中分析结果的可靠性. 值得注意的是, 实验结果表明, 基于在线强化学习算法的训练曲线随训练进度增加, 关于正负样本生成概率的标准差增加速度高于离线强化学习算法的增加速度, 这是因为在线算法的训练过程中涉及针对模型响应的在线随机采样, 训练的随机性更强, 带来了更大的标准差; 离线算法的训练样本固定, 随机性只来自数据顺序和训练过程中浮点数计算的精度误差, 标准差更小. 这里的随机性也一定程度上解释了在线强化学习算法收敛速度慢的原因, 即训练过程中随机性大, 收敛方向不明确, 因此需要更多训练轮次来减少误差.

6　结论

本文提出了一种新的强化学习策略——级联强化学习. 该策略将离线强化学习和在线强化学习相结合, 在保留了后者性能上限高的优势的前提下, 进一步吸收了前者收敛速度快的特性. 实验结果表明, 级联强化学习可以通过仅在线强化学习一半的训练成本便可取得更高的性能上限. 除此之外, 本文分析了训练过程中, 模型关于正负样本的对数生成概率的变化情况, 从模型训练动力学的角度对有监督微调和各类强化学习方案进行了对比, 提供了级联强化学习为什么有效的分析. 分析结果表明, 级联强化学习保持了在线强化学习训练趋势的同时, 进一步扩大了正负样本间生成概率的差距, 进而达到加速收敛的效果. 基于这种简单有效的训练策略, 本文将 InternVL3.5-8B-Instruct 和 InternVL3.5-241B-A28B-Instruct 在七个多模态推理评测基准上的综合准确率分别提升了 6.7% 和 6.5%, 证明了这一策略的有效性和可扩展性.

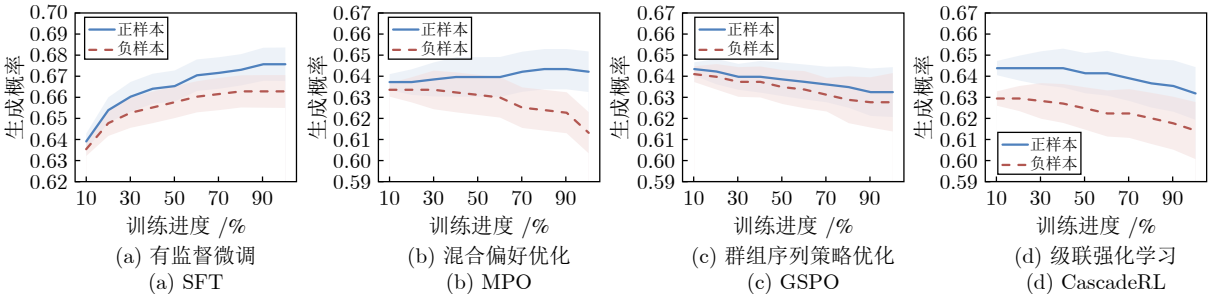


图 5　不同算法训练过程中模型关于起始模型的正负响应的生成概率

Fig. 5　Generation probabilities of positive and negative responses relative to the initial model during training with different algorithms

参考文献

1　Rafailov R, Sharma A, Mitchell E, Manning S, Ermon C D, Finn C. Direct preference optimization: Your language model is secretly a reward model. In: Proceedings of the 37th International Conference on Neural Information Processing Systems. New Orleans, USA: Curran Associates Inc., 2023. Article No. 2338

2　Wang W Y, Chen Z, Wang W H, Cao Y, Liu Y Z, Gao Z W, et al. Enhancing the reasoning ability of multimodal large language models via mixed preference optimization. arXiv preprint arXiv: 2411.10442, 2024.

3　Shao Z H, Wang P Y, Zhu Q H, Xu R X, Song J X, Bi X, et al. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. arXiv preprint arXiv: 2402.03300, 2024.

4　Zheng C J, Liu S X, Li M Z, Chen X H, Yu B W, Gao C, et al. Group sequence policy optimization. arXiv preprint arXiv: 2507.18071, 2025.

5　Chen Q G, Qin L B, Zhang J, Chen Z, Xu X, Che W X. M3CoT: A novel benchmark for multi-domain multi-step multi-modal chain-of-thought. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics.

Bangkok, Thailand: ACL, 2024. 8199−8221

6 Wang W Y, Gao Z W, Gu L X, Pu H J, Cui L, Wei X G, et al. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. arXiv preprint arXiv: 2508.18265, 2025.

7 Touvron H, Lavril T, Izacard G, Martinet X, Lachaux M A, Lacroix T, et al. LLaMA: Open and efficient foundation language models. arXiv preprint arXiv: 2302.13971, 2023.

8 Cai Z, Cao M S, Chen H J, Chen K, Chen K Y, Chen X, et al. InternLM2 technical report. arXiv preprint arXiv: 2403.17297, 2024.

9 Yang A, Li A F, Yang B S, Zhang B C, Hui B Y, Zheng B, et al. Qwen3 technical report. arXiv preprint arXiv: 2505.09388, 2025.

10 OpenAI. Introducing gpt-oss [Online], available: https://openai.com/index/introducing-gpt-oss/, August 5, 2025

11 Ouyang L, Wu J, Jiang X, Almeida D, Wainwright C L, Mishkin P, et al. Training language models to follow instructions with human feedback. In: Proceedings of the 36th International Conference on Neural Information Processing Systems. New Orleans, USA: Curran Associates Inc., 2022. Article No. 2011

12 Xu Zheng-Fei, Xin Xin. An empirical study of Chinese entity linking based on large language model. *Acta Automatica Sinica*, 2025, **51**(2): 327−342
(徐正斐, 辛欣. 基于大语言模型的中文实体链接实证研究. 自动化学报, 2025, **51**(2): 327−342)

13 Qin Long, Wu Wan-Sen, Liu Dan, Hu Yue, Yin Quan-Jun, Yang Dong-Sheng, et al. Autonomous planning and processing framework for complex tasks based on large language models. *Acta Automatica Sinica*, 2024, **50**(4): 862−872
(秦龙, 武万森, 刘丹, 胡越, 尹全军, 阳东升, 等. 基于大语言模型的复杂任务自主规划处理框架. 自动化学报, 2024, **50**(4): 862−872)

14 Chen Chu-Yan, Liu Ye-Xu, Jia Wei-Chen, He Yu-Tong, Yuan Kun, Wang Li-Wei. An effective algorithm for LLMs training with one-bit communication compression. *Acta Automatica Sinica*, DOI: 10.16383/j.aas.c250087
(陈楚岩, 刘烨谞, 贾维宸, 何雨桐, 袁坤, 王立威. 一种基于单比特通信压缩的大模型训练方法研究. 自动化学报, DOI: 10.16383/j.aas.c250087)

15 Yan H, Gao Y, Fei C Y, Yang X P, Qiu X P. Data and model architecture in base model training. In: Proceedings of the 22nd Chinese National Conference on Computational Linguistics (Volume 2: Frontier Forum). Harbin, China: Chinese Information Processing Society of China, 2023. 1−15

16 Li Z J, Zhang J W, Wang Y, Du M F, Liu Q W, Wang D Y, et al. From multi-modal pre-training to multi-modal large language models: An overview of architectures, training. In: Proceedings of the 23rd Chinese National Conference on Computational Linguistics (Volume 2: Frontier Forum). Taiyuan, China: Chinese Information Processing Society of China, 2024. 1−33

17 Zheng Yi-Ning, Yu Zhen, Li Bu-Fan, Yang Jie, Yin Lin-Qi, Yin Zhang-Yue, et al. Survey of tool use in large language models. *Acta Automatica Sinica*, 2025, **51**(11): 2371−2386
(郑逸宁, 余镇, 李不凡, 杨捷, 殷林琪, 印张悦, 等. 大语言模型的工具使用综述. 自动化学报, 2025, **51**(11): 2371−2386)

18 Schuhmann C, Beaumont R, Vencu R, Gordon C, Wightman R, Cherti M, et al. LAION-5B: An open large-scale dataset for training next generation image-text models. In: Proceedings of the 36th International Conference on Neural Information Processing Systems. New Orleans, USA: Curran Associates Inc., 2022. Article No. 1833

19 Li Q Y, Chen Z, Wang W Y, Wang W H, Ye S L, Jin Z J, et al. OmniCorpus: An unified multimodal corpus of 10 billion-level images interleaved with text. arXiv preprint arXiv: 2406.08418, 2024.

20 Laurençon H, Saulnier L, Tronchon L, Bekman S, Singh A, Lozhkov A, et al. OBELICS: An open web-scale filtered dataset of interleaved image-text documents. In: Proceedings of the 37th International Conference on Neural Information Processing Systems. New Orleans, USA: Curran Associates Inc., 2023. Article No. 3138

21 Liu Y Z, Cao Y, Gao Z W, Wang W Y, Chen Z, Wang W H, et al. MMInstruct: A high-quality multi-modal instruction tuning dataset with extensive diversity. arXiv preprint arXiv: 2407.15838, 2024.

22 Wang W Y, Shi M, Li Q Y, Wang W H, Huang Z H, Xing L J, et al. The all-seeing project: Towards panoptic visual recognition and understanding of the open world. In: Proceedings of the 12th International Conference on Learning Representations. Vienna, Austria: OpenReview.net, 2024.

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

24 Zheng R, Dou S H, Gao S Y, Hua Y, Shen W, Wang B H, et al. Secrets of RLHF in large language models part I: PPO. arXiv preprint arXiv: 2307.04964, 2023.

25 Bradley R A, Terry M E. Rank analysis of incomplete block designs: I. The method of paired comparisons. *Biometrika*, 1952, **39**(3−4): 324−345

26 Jung S, Han G, Nam D W, On K W. Binary classifier optimization for large language model alignment. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics. Vienna, Austria: ACL, 2025. 1858−1872

27 Hong W Y, Yu W M, Gu X T, Wang G, Gan G B, Tang H M, et al. GLM-4.5V and GLM-4.1V-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. arXiv preprint arXiv: 2507.01006, 2025.

28 Keye T K, Yang B, Wen B, Liu C Y, Chu C L, Song C R, et al. Kwai keye-VL technical report. arXiv preprint arXiv: 2507.01949, 2025.

29 Bengio Y, Louradour J, Collobert R, Weston J. Curriculum learning. In: Proceedings of the 26th Annual International Conference on Machine Learning. Montreal, Canada: ACM, 2009. 41−48

30 Wang X, Chen Y D, Zhu W W. A survey on curriculum learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022, **44**(9): 4555−4576

31 Xi Z H, Chen W X, Hong B Y, Jin S J, Zheng R, He W, et al. Training large language models for reasoning through reverse curriculum reinforcement learning. In: Proceedings of the 41st International Conference on Machine Learning. Vienna, Austria: JMLR.org, 2024. Article No. 2217

32 Wu J X, Zhang D M, Zhong S L, Qiao H. Trajectory-based split hindsight reverse curriculum learning. In: Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Prague, Czech Republic: IEEE, 2021. 3971−3978

33 Qu Y X, Zhang T J, Garg N M, Kumar A. RECURSIVE INTROSPECTION: Teaching language model agents how to self-improve. In: Proceedings of the 38th International Conference on

Neural Information Processing Systems. Vancouver, Canada: Curran Associates Inc., 2024. Article No. 1754

34 Wang Z T, Cui G F, Li Y J, Wan K, Zhao W T. DUMP: Automated distribution-level curriculum learning for RL-based LLM post-training. arXiv preprint arXiv: 2504.09710, 2025.

35 Wen L, Cai Y K, Xiao F R, He X, An Q, Duan Z Y, et al. Light-R1: Curriculum SFT, DPO and RL for long COT from scratch and beyond. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track). Vienna, Austria: ACL, 2025. 318−327

36 Bi X, Chen D L, Chen G T, Chen S H, Dai D M, Deng C Q, et al. DeepSeek LLM: Scaling open-source language models with longtermism. arXiv preprint arXiv: 2401.02954, 2024.

37 Sun C N, Jung G Y, Tran T X, Pompili D. Cascade reinforcement learning with state space factorization for O-RAN-based traffic steering. In: Proceedings of the 21st Annual IEEE International Conference on Sensing, Communication, and Networking (SECON). Phoenix, USA: IEEE, 2024. 1−9

38 Castillo G A, Weng B W, Zhang W, Hereid A. Reinforcement learning-based cascade motion policy design for robust 3D bipedal locomotion. *IEEE Access*, 2022, **10**: 20135−20148

39 Xu J, Qiao J B, Sun Q L, Shen K Y. A deep reinforcement learning framework for cascade reservoir operations under runoff uncertainty. *Water*, 2025, **17**(15): Article No. 2324

40 Pal A, Karkhanis D, Dooley S, Roberts M, Naidu S, White C. Smaug: Fixing failure modes of preference optimisation with DPO-positive. arXiv preprint arXiv: 2402.13228, 2024.

41 Bai J Z, Bai S, Yang S S, Wang S J, Tan S N, Wang P, et al. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. arXiv preprint arXiv: 2308.12966, 2023.

42 Yue X, Ni Y S, Zhang K, Zheng T Y, Liu R Q, Zhang G, et al. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. arXiv preprint arXiv: 2311.16502, 2023.

43 Zhang R R, Jiang D Z, Zhang Y C, Lin H K, Guo Z Y, Qiu P S, et al. MathVerse: Does your multi-modal LLM truly see the diagrams in visual math problems? arXiv preprint arXiv: 2403.14624, 2024.

44 Yue Z H, Lin Z R, Song Y F, Wang W K, Ren S H, Gu S H, et al. Mimo-VL technical report. arXiv preprint arXiv: 2506.03569, 2025.

45 Shen W, Pei J B, Peng Y, Song X C, Liu Y, Peng J, et al. Skywork-R1V3 technical report. arXiv preprint arXiv: 2507.06167, 2025.

46 Qwen Team. QVQ: To see the world with wisdom [Online], available: https://qwen.ai/blog?id=qvq-72b-preview, December 25, 2024

47 Qwen Team. Qwen2.5: A party of foundation models! [Online], available: https://qwenlm.github.io/blog/qwen2.5/, September 19, 2024

48 Mesnard T, Hardin C, Dadashi R, Bhupatiraju S, Pathak S, Sifre L, et al. Gemma: Open models based on Gemini research and technology. arXiv preprint arXiv: 2403.08295, 2024.

49 StepFun Research Team. Step-1v: A hundred billion parameter multimodal large model [Online], available: https://platform.stepfun.com, March 23, 2024

50 OpenAI. Gpt-5 system card [Online], available: https://cdn.openai.com/pdf/8124a3ceab78-4f06-96eb-49ea29ffb52f/gpt5-system-card-aug7.pdf, August 7, 2025

51 Anthropic. The Claude 3 model family: Opus, sonnet, haiku [Online], available: https://www-cdn.anthropic.com/de8ba9b01c9ab7cbabf5c33b80b7bbc618857627/Model_Card_Claude_3.pdf, March 4, 2024

52 Pichai S, Hassabis D, Kavukcuoglu K. Introducing Gemini 2.0: Our new AI model for the agentic era [Online], https://blog.google/technology/google-deepmind/google-gemini-ai-update-december-2024/, December 11, 2024

53 Guo D, Wu F M, Zhu F D, Leng F X, Shi G, Chen H B, et al. Seed1.5-VL technical report. arXiv preprint arXiv: 2505.07062, 2025.

54 Lu P, Bansal H, Xia T, Liu J C, Li C Y, Hajishirzi H, et al. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. arXiv preprint arXiv: 2310.02255, 2023.

55 Wang K, Pan J T, Shi W K, Lu Z M, Zhan M J, Li H S. Measuring multimodal mathematical reasoning with MATH-vision dataset. arXiv preprint arXiv: 2402.14804, 2024.

56 Zou C K, Guo X G, Yang R, Zhang J Y, Hu B, Zhang H. DynaMath: A dynamic visual benchmark for evaluating mathematical reasoning robustness of vision language models. arXiv preprint arXiv: 2411.00836, 2024.

57 Qiao R Q, Tan Q N, Dong G T, Wu M H, Sun C, Song X S, et al. We-Math: Does your large multimodal model achieve human-like mathematical reasoning? arXiv preprint arXiv: 2407.01284, 2024.

58 Xiao Y J, Sun E, Liu T Y, Wang W. LogicVista: Multimodal LLM logical reasoning benchmark in visual contexts. arXiv preprint arXiv: 2407.04973, 2024.

59 OpenCompass Contributors. OpenCompass: A universal evaluation platform for foundation models [Online], available: https://github.com/open-compass/opencompass, July 4, 2023

60 Zhu J G, Wang W Y, Chen Z, Liu Z Y, Ye S L, Gu L X, et al. InternVL3: Exploring advanced training and test-time recipes for open-source multimodal models. arXiv preprint arXiv: 2504.10479, 2025.

61 Lu S Y, Li Y, Chen Q G, Xu Z, Luo W H, Zhang K F, et al. Ovis: Structural embedding alignment for multimodal large language model. arXiv preprint arXiv: 2405.20797, 2024.

62 Bai S, Chen K Q, Liu X J, Wang J L, Ge W B, Song S B, et al. Qwen2.5-VL technical report. arXiv preprint arXiv: 2502.13923, 2025.

63 Yao Y, Yu T Y, Zhang A, Wang C Y, Cui J B, Zhu H J, et al. MiniCPM-V: A GPT-4V level MLLM on your phone. arXiv preprint arXiv: 2408.01800, 2024.

64 Chen Z, Wang W Y, Cao Y, Liu Y Z, Gao Z W, Cui E F, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. arXiv preprint arXiv: 2412.05271, 2024.

65 Team G, Kamath A, Ferret J, Pathak S, Vieillard N, Merhej R, et al. Gemma 3 technical report. arXiv preprint arXiv: 2503.19786, 2025.

66 Team K, Du A G, Yin B H, Xing B W, Qu B W, Wang B W, et al. Kimi-Vl technical report. arXiv preprint arXiv: 2504.07491, 2025.

67 StepFun Team. Step3: Cost-effective multimodal intelligence [Online], available: https://stepfun.ai/research/en/step3, July 25, 2025

王玮赟 复旦大学博士研究生. 2022 年获得哈尔滨工业大学计算机科学与技术专业学士学位. 主要研究方向为多模态大语言模型.
E-mail: wywang22@m.fudan.edu.cn
(**WANG Wei-Yun** Ph.D. candidate at Fudan University. He received his bachelor degree in computer science and technology from Harbin Institute of Technology in 2022. His main research interest is multimodal large language models.)

蒲恒骏 上海人工智能实验室见习研究员. 2024 年获得清华大学软件工程专业学士学位. 主要研究方向为多模态大语言模型.
E-mail: puhengjun@pjlab.org.cn
(**PU Heng-Jun** Research intern at Shanghai Artificial Intelligence Laboratory. He received his bachelor degree in software engineering from Tsinghua University in 2024. His main research interest is multimodal large language models.)

景凌林 上海人工智能实验室青年研究员. 2025 年获得英国拉夫堡大学计算机科学与技术专业博士学位. 主要研究方向为视频理解, 多模态大语言模型.
E-mail: jinglinglin@pjlab.org.cn
(**JING Ling-Lin** Young researcher at Shanghai Artificial Intelligence Laboratory. He received his Ph.D. degree in computer science and technology from Loughborough University, United Kingdom in 2025. His research interests include video understanding and multimodal large language models.)

乔 宇 上海人工智能实验室教授. 主要研究方向为计算机视觉及多模态大语言模型. 本文通信作者.
E-mail: qiaoyu@pjlab.org.cn
(**QIAO Yu** Professor at Shanghai Artificial Intelligence Laboratory. His research interests include computer vision and multimodal large language models. Corresponding author of this paper.)