Simple Test-Time Scaling 论文解读

2025年2月10日 1 分钟阅读

#opensource #reasoning #SFT #论文 #Finetuning

> *本文介绍了来自李飞飞团队的Simple Test-Time Scaling论文，并对其技术原理、主要贡献、论文方法、评估结果和局限性进行了详细解读。*

引言

Paper: https://arxiv.org/html/2501.19393v2
Github: https://arxiv.org/html/2501.19393v2

测试时缩放是一种用于语言建模的有前景的新方法，它利用额外的测试时计算资源来提升性能。最近，OpenAI的o1模型展现了这种能力，但没有公开其方法，这导致了许多复制工作。我们寻求实现测试时缩放和强大推理性能的最简单方法。 1.首先，我们精心整理了一个包含1000个问题及其推理轨迹的小型数据集s1K，这些问题依据我们在消融实验中验证的三个标准（难度、多样性和质量）进行挑选。 2.其次，我们开发了预算强制机制来控制测试时计算资源。具体而言，当模型试图结束思考过程时，我们会强制终止其思考过程，或者通过在模型的生成内容后多次添加"等待"来延长思考时间。这能够促使模型复查自己的答案，常常修正不正确的推理步骤。 3.在对Qwen2.5 - 32B - Instruct语言模型使用s1K数据进行监督微调并为其配备预算强制机制之后，我们的s1 - 32B模型在竞赛数学问题（如MATH和AIME24）上比o1 - preview模型的表现高出多达27%。此外，通过预算强制机制对s1 - 32B进行扩展，能够在无测试时干预的情况下超越其原有性能：在AIME24上的表现从50%提升到57%。 4. 我们的模型、数据和代码开源于https://github.com/simplescaling/s1。

全文摘要

一句话总结： 号称使用50美元微调Qwen2.5 - 32B - Instruct成类似Deepseek R1的推理性模型，来自李飞飞团队。

目录

文章信息

字数
阅读时间
发布时间
更新时间

标签

#opensource #reasoning #
#论文 #Finetuning

这篇论文介绍了一种新的语言模型——test-time scaling，并通过实验验证了其在数学问题上的优越性能。该方法使用额外的测试时间计算来提高模型的表现，同时控制计算预算以避免过拟合。作者们还开发了一个小型数据集和预算强制技术，用于训练模型并优化其表现。最终结果表明，这种方法可以在不进行干预的情况下提高模型的性能，并且可以应用于其他领域的问题解决。

论文速读

论文方法

方法描述

该论文提出了两种分类测试时间缩放方法：序列化和并行化。其中，序列化方法是指后续计算依赖于早期计算（例如长推理链），而并行化方法是指计算独立运行（例如多数投票）。本文主要关注序列化方法，并提出新的序列化缩放方法以及如何对其进行评估。

方法改进

该论文提出的改进方法是预算强制（budget forcing）。它是一种简单且直观的解码时间干预方式，在测试时间强制模型产生最大或最小数量的思考标记以控制模型在思考阶段的输出量。通过将"End-of-Thinking"标记添加到早期退出位置来实现最大限制，并通过抑制"End-of-Thinking"标记的生成来鼓励模型反思当前生成的内容。这种方法可以使模型更好地达到最佳答案。 此外，该论文还提供了基准线方法：条件长度控制和拒绝采样。条件长度控制方法告诉模型在提示中应该生成多长时间，分为三种粒度：基于令牌的控制、基于步数的控制和基于类别的控制。拒绝采样方法则是在给定计算预算的情况下，随机抽样直到生成符合预算的回答为止。

解决的问题

该论文旨在解决测试时间缩放问题，即如何在不牺牲性能的前提下，使模型能够在不同测试时间下自适应地调整其行为。为此，该论文提出了多种方法来衡量测试时间缩放的效果，包括可控性、测试时间缩放斜率和性能等指标。这些方法可以帮助研究人员选择最适合特定任务的缩放方法，并提高模型的性能和效率。

论文实验

本文介绍了作者进行的三个对比实验，分别是： 1.模型性能对比实验：该实验比较了不同模型在AIME24、MATH500和GPQA等三个领域的表现，并给出了相应的评估指标和得分。结果表明，作者提出的s1-32B模型是样本效率最高的开放数据推理模型之一，在这三个领域中都表现出色。

2.数据量、多样性和难度对模型效果的影响实验：该实验通过调整数据集中的数量、多样性和平滑度来测试这些因素对模型效果的影响。结果表明，结合高质量、难度和多样性三个标准的数据集能够提高模型的训练效果。 3.测试时间控制方法对比实验：该实验比较了不同的测试时间控制方法，包括预算强制、步长条件控制、分类条件控制和拒绝采样等。结果表明，预算强制是最有效的测试时间控制方法，可以完美控制、良好扩展并提高模型的表现。 总之，本文通过对多个实验的比较分析，展示了作者提出的s1-32B模型及其相关技术的有效性和优越性。

论文的亮点是

找到了1000个高质量的数据和推理轨迹（来自google gemini thinking），可以用它们来SFT一个不错的中小模型成一个类似Deepseek R1的推理模型

找到了一个通过延迟推理时间来获得更好推理效果的方法。

对业界的影响

使用相同的S1K数据，应该可以把一些小模型微调成O1/R1这样的DeepThinking模型。

QA

为什么只进行1,000样本的监督微调可以实现如此高的性能提升?

我们假设模型在预训练期间已经接触到了大量的推理数据，这些数据跨越了数万亿个标记。因此，我们的模型已经具备了执行推理的能力。我们的样本高效微调阶段只是激活它，并且我们在测试时间使用预算强制进一步扩展它。这类似于LIMA提出的"表面对齐假设"，其中作者发现1,000个示例就足以使模型遵循用户偏好。

详细解释什么是"预算强制技术"?

预算强制技术是一种用于控制模型思考持续时间的技术。它通过在生成过程中强行终止模型的思考过程或者通过多次添加"等待"语句来延长思考过程来实现。这种技术可以帮助模型进行双倍检查并修复错误的推理步骤，从而提高性能。

s1-32B模型是如何进行测试时间的控制的?

在训练完成后，我们通过测试时间预算强制来控制模型花费的计算量。(I) 如果模型生成的思考标记超过了期望限制，则我们强制添加一个结束思考标记来终止思考过程，并让模型开始生成答案。(II) 如果我们希望模型在某个问题上花费更多的测试时间计算，我们可以抑制结束思考标记的生成，并将"等待"附加到当前推理轨迹中，以鼓励更多探索。

该论文对将来推理模型的训练的启发是什么?

这篇论文提供了一种简单而有效的方法来实现测试时间的扩展。这种方法可以用于提高语言模型的性能，并且可以应用于其他类型的推理任务。这个方法的关键是创建一个包含高质量、多样性和难度的推理数据集，并使用测试时间预算强制来控制模型的计算量。这可以帮助模型更好地推理出正确答案并避免错误的推理步骤。这个方法也可以作为未来推理模型训练的一个重要参考点，帮助研究人员设计更有效的训练策略。

Pangu Deep Dive...

Pangu 相关论文的深度解析和...