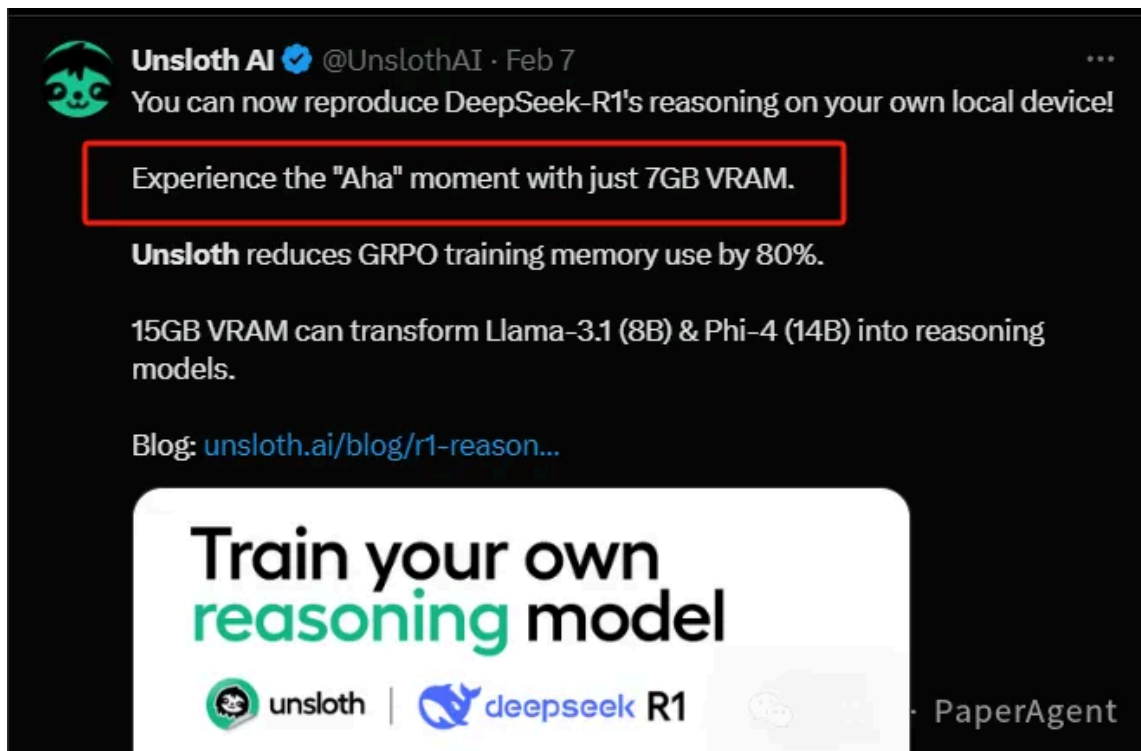


7G显存，训练自己的DeepSeek-R1，GRPO资源暴降80%

PaperAgent 2025年02月10日 16:12 湖北

DeepSeek-R1自发布以来，已出现众多开源复现（Open-R1，Tiny-Zero、simpleRL-reason、Mini-R1等等），但受限于GPU条件限制，有小伙伴可能没run起来~

此刻，不用为GPU发愁，Unsloth AI黑科技来了，**使用 Unsloth (GRPO) 训练自己的 R1 推理模型**（附代码），增强了整个 GRPO 流程，使其使用的 VRAM 比 Hugging Face + FA2 少 **80%**。可以使用 Qwen2.5 (1.5B) 在仅 **7GB VRAM** 上重现 R1-Zero 的“顿悟时刻”：



GRPO + “啊哈”时刻

DeepSeek 的研究人员在使用纯强化学习 (RL) 训练 R1-Zero 时观察到了“顿悟时刻”。该模型学会了通过重新评估其初始方法延长其思考时间，而无需任何人工指导或预定义指令。

尽管仅使用 **GRPO 训练了 100 步的 Phi-4**，但结果已经很明显了。没有 GRPO 的模型没有思考标记，而使用 GRPO 训练的模型有思考标记并且也有正确答案。



Prompt: Which is bigger? 9.11 or 9.9?

Before (Phi-4 with no reasoning)

9.11 is bigger than 9.9. When comparing decimal numbers, you look at the digits from left to right. Both numbers have the same whole number part (9), so you compare the digits in the tenths place next. In 9.11, the tenths place is 1, and in 9.9, the tenths place is 9. Since 1 is less than 9, you might initially think 9.9 is larger, but you also need to consider the hundredths place in 9.11, which is 1. When you express 9.9 as 9.90 for comparison, you see that 9.11 is greater than 9.90. Therefore, 9.11 is bigger than 9.9.

After (Phi-4 with GRPO) 💡

<reasoning>
To determine which number is bigger between 9.11 and 9.9, we should compare the two numbers digit by digit from left to right.

1. First, compare the digits in the units place:
 - Both numbers have a 9 in the units place.
2. Next, compare the digits in the tenths place:
 - The number 9.11 has a 1 in the tenths place.
 - The number 9.9 has a 9 in the tenths place.Since 1 is less than 9, the number 9.11 is less than 9.9 based on the tenths place comparison.
3. For thoroughness, consider the hundredths place:
 - The number 9.11 has a 1 in the hundredths place.
 - The number 9.9 can be written as 9.90, which has a 0 in the hundredths place.

Even if we compare the hundredths place, 1 is greater than 0, but this is irrelevant since the comparison in the tenths place already determines that 9.11 is smaller than 9.9. Therefore, 9.9 is greater than 9.11.

</reasoning>

<answer>

9.9 is bigger than 9.11.

</answer>

公众号 · PaperAgent

这种魔力可以通过 GRPO 重现，GRPO 是一种 RL 算法，它不需要价值函数就能有效优化响应，这与依赖价值函数的近端策略优化 (PPO) 不同。

工作原理：

- 该模型生成了多组响应。
- 每个响应都根据正确性或由某些设定的奖励函数（而不是 LLM 奖励模型）创建的其他指标进行评分。
- 计算该组的平均分数。
- 每个回答的分数都会与组平均分数进行比较。
- 该模型得到强化，以支持得分更高的反应。

举个例子，假设我们想要一个模型来解决：

1 + 1 等于多少？ >> 思路/计算 >> 答案是 2。

2 + 2 等于多少？ >> 思路/计算 >> 答案是 4。

最初，必须收集大量数据来填补计算/思考流程。但 GRPO（DeepSeek 使用的算法）或其他 RL 算法可以引导模型自动展示推理能力并创建推理轨迹。相反，需要创建好的奖励函数或验证器。例如，如果它得到了正确的答案，就给它 1 分。如果有些单词拼写错误，就减 0.1 分。等等！可以提供很多函数来奖励这个过程。

Unsloth 中的 GRPO

等待至少 300 步，奖励才会真正增加，请使用最新版本的 vLLM。Unsloth 在 Colab 上的示例只训练了一个小时，因此结果低于标准。为了获得良好的结果，需要训练至少 **12 个小时**。

这是建议将 GRPO 应用于至少有 **1.5B 参数的模型**，以正确生成思考标记，因为较小的模型可能无法做到这一点。如果使用的是基础模型，请确保有一个聊天模板。**GRPO 的训练损失跟踪现在直接内置在 Unsloth 中，无需使用 wandb 等外部工具。**

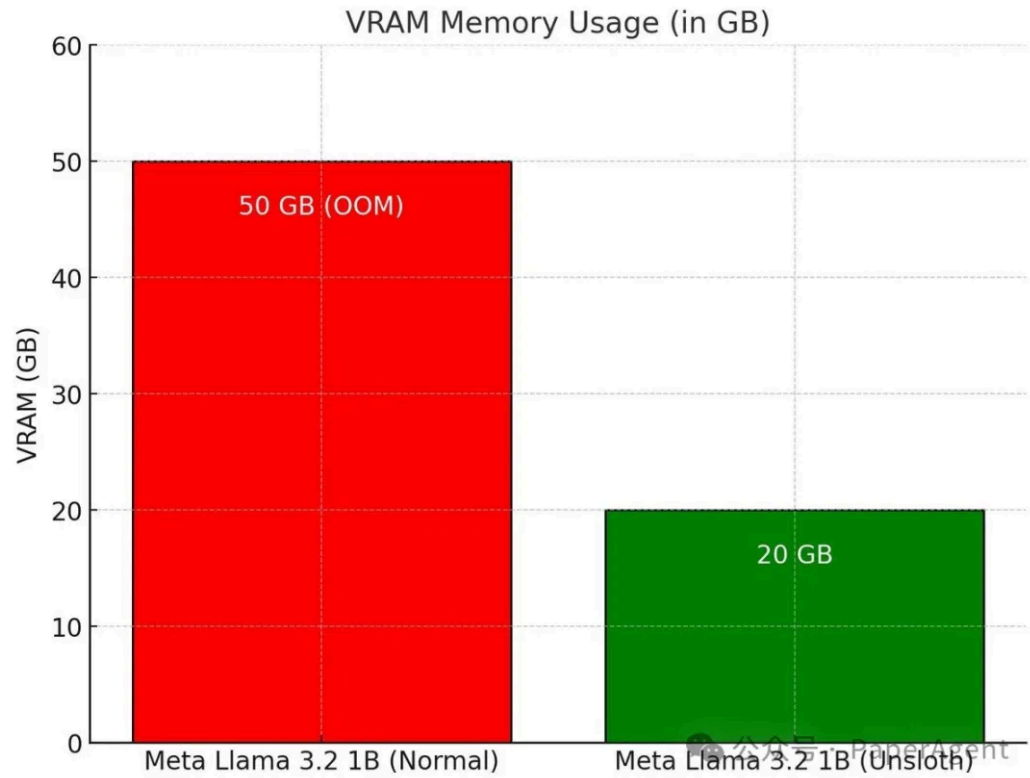


[19/100 27:23 < 2:10:29, 0.01 it/s, Epoch 0.00/1]

Step	Training Loss	reward	reward_std	completion_length	k1
1	0.000000	0.041667	0.064550	200.000000	0.000000
2	0.000000	0.125000	0.000000	200.000000	0.000000
3	0.000000	0.404500	0.960518	191.666672	0.000002
4	0.000000	0.104167	0.051031	200.000000	0.000002
5	0.000000	0.104167	0.051031	200.000000	0.000002
6	0.000000	0.044500	0.197184	200.000000	0.000002
7	0.000000	0.044167	0.145617	198.300000	0.000004

除了添加 GRPO 支持外，随后还支持在线 DPO、PPO 和 RLOO！

比较 Unsloth 的在线 DPO VRAM 消耗与标准 Hugging Face + FA2



```
1 log/r1-reasoning
2 ch.google.com/github/unslothai/notebooks/blob/main/nb/Llama3.1_(8B)-GRPO.ipynb
```