ZeroSearch: Incentivize the Search Capability of LLMs without Searching

如何接近零API调用成本，用RLVR训练LLM使用搜索引擎？

Hao Sun, Zile Qiao*, Jiayan Guo*, Xuanbo Fan, Yingyan Hou
Yong Jiang, Pengjun Xie, Yan Zhang*, Fei Huang, Jingren Zhou
Tongyi Lab, Alibaba Group

🔗 Homepage 🤗 Model Datasets Code

简介

想教llm学会使用搜索引擎，训练阶段一定要调用search API吗？本文提出的ZeroSearch 换了个思路，对一个llm做sft，让其模拟Google search，然后在RLVR训练阶段用它代替真正的search tool，这样在RLVR训练阶段做到了0成本调用search API并且生成文档质量可控。

NOTE 1: 为了SFT LLM，需要创建训练集，这是需要调用SEARCH API的，所以整套方案是接近零成本

NOTE 2: 在INFERENCE阶段，还是要用真实SEARCH TOOL的

背景

当提升llm能力的重要路径是让它学会使用搜索引擎，这样在推理时面对知识盲区能够主动发起检索。如何让llm学会使用搜索引擎，RLVR是目前学术界比较主流的做法。作者认为RLVR训练过程中存在两个问题：

1. 文档质量不可控：搜索引擎返回的检索结果可能含有很多噪声，影响模型学习；
2. API 调用成本过高：这一点很真实，search api真的太贵了，比如Bocha web search，1k次价格高达36元，ai search更是贵到60元

为此，本文提出了zerosearch，训练阶段用llm模拟搜索引擎，做到0成本的serch API调用。

实验设置

SFT QWEN2.5 INSTRUCT 模拟搜索引擎
单跳/多跳问答任务

- 框架：verl 实验对象：Qwen2.5 Base/Instruct和Llama 3.2 Base/Instruct
- 强化学习算法：本文是一种训练流程，不局限于某种RL算法
- reward function：F1 score，并且不需要format reward。作者发现如果用EM作为score，会reward hacking, policy倾向于生成过长的答案，以此增加覆盖正确答案的概率

SFT TEMPLATE

> You are the Google search engine.
> Given a query, you need to generate five [useful / noisy] documents for the query.
> The user is trying to answer the question: [question] whose answer is [ground truth].
> Each document should contain about 30 words, and these documents should contain [useful / noisy] information.
> Query: [query]
> [Useful / Noisy] Output:

如何构造sft llm训练集：论文中的实验用的是问答数据集，因此作者将RLVR阶段训练集的query和answer拿过来，然后用prompt的方法指导llm调用搜索引擎对query生成推理path和推理答案。提取出query和搜索返回的文档，类似于(q1, d1, d2, d3, d4, d4)，然后用llm判断每一个文档是否对回答query有作用，有用的标记为useful，否则标记为noisy。这样就有sft训练数据了，再结合上面的prompt template对llm sft。
sft后的llm既可以生成useful文档也可以生成noisy文档，在RLVR阶段，作者采用课程学习策略，让sft llm逐渐生成越来越多的noisy 文档，做到生成文档质量可控，并且主要为了强化policy的推理和search调用能力

部分实验结果

Method	Single-Hop QA			Multi-Hop QA				
	NQ	TriviaQA	PopQA	HotpotQA	2Wiki	Musique	Bamboogle	Avg.
Qwen-2.5-7B-Base/Instruct								
Direct Answer	11.60	35.60	1.20	16.40	22.20	4.80	14.40	15.17
CoT	12.80	35.60	3.80	16.20	22.60	6.60	24.00	17.37
RAG	27.40	58.20	17.80	25.80	23.20	9.40	16.80	25.51
RA-Agent	21.20	40.20	8.80	19.60	19.60	7.60	28.00	20.71
Search-o1	19.40	40.60	11.40	17.00	27.00	8.60	30.40	22.06
R1-base	27.60	47.40	27.40	21.00	29.20	9.80	27.78	27.17
R1-instruct	27.00	45.80	24.20	21.60	27.80	8.40	25.00	25.69
Search-R1-base	43.40	61.40	54.60	31.20	**37.20**	18.20	30.56	39.51
Search-R1-inst	42.40	63.40	51.60	32.80	33.20	17.40	26.39	38.17
ZeroSearch-base	42.40	**66.40**	**60.40**	32.00	34.00	18.00	**33.33**	**40.93**
ZeroSearch-inst	**43.60**	65.20	48.80	**34.60**	35.20	**18.40**	27.78	39.08

思考

本文的实验基于14B的SFT LLM就能取得非常好的效果，再来思考下，这种模拟搜索引擎的方法有没有什么弊端，首先，我个人认为LLM是没有办法完全代替搜索引擎的，因为KNOWLEDGE CUTOFF的存在，搜索引擎永远拥有更多的新知识。
此外，实验中的SFT和RLVR阶段使用的是高度重合的QUERY，因此所需的外部知识实际上已通过搜索引擎提前被覆盖，包含在了SFT LLM中。在RLVR 训练时，SFT LLM就能提供足够的POLICY所需的外部知识。
还是那句话，SEARCH API真的太贵了，本文做了一次非常好的尝试。