arxiv.org/abs/2305.11554

ToolkenGPT: Augmenting Frozen Language Models with Massive Tools via Tool Embeddings

ToolkenGPT：通过工具的token化与向量化，提升LLM的工具调用能力

开源代码：github.com/Ber666/ToolkenGPT

Shibo Hao[1], **Tianyang Liu**[1], **Zhen Wang**[1,2], **Zhiting Hu**[1]
[1]UC San Diego, [2]Mohamed bin Zayed University of Artificial Intelligence
{s5hao, til040, zhw085, zhh019}@ucsd.edu

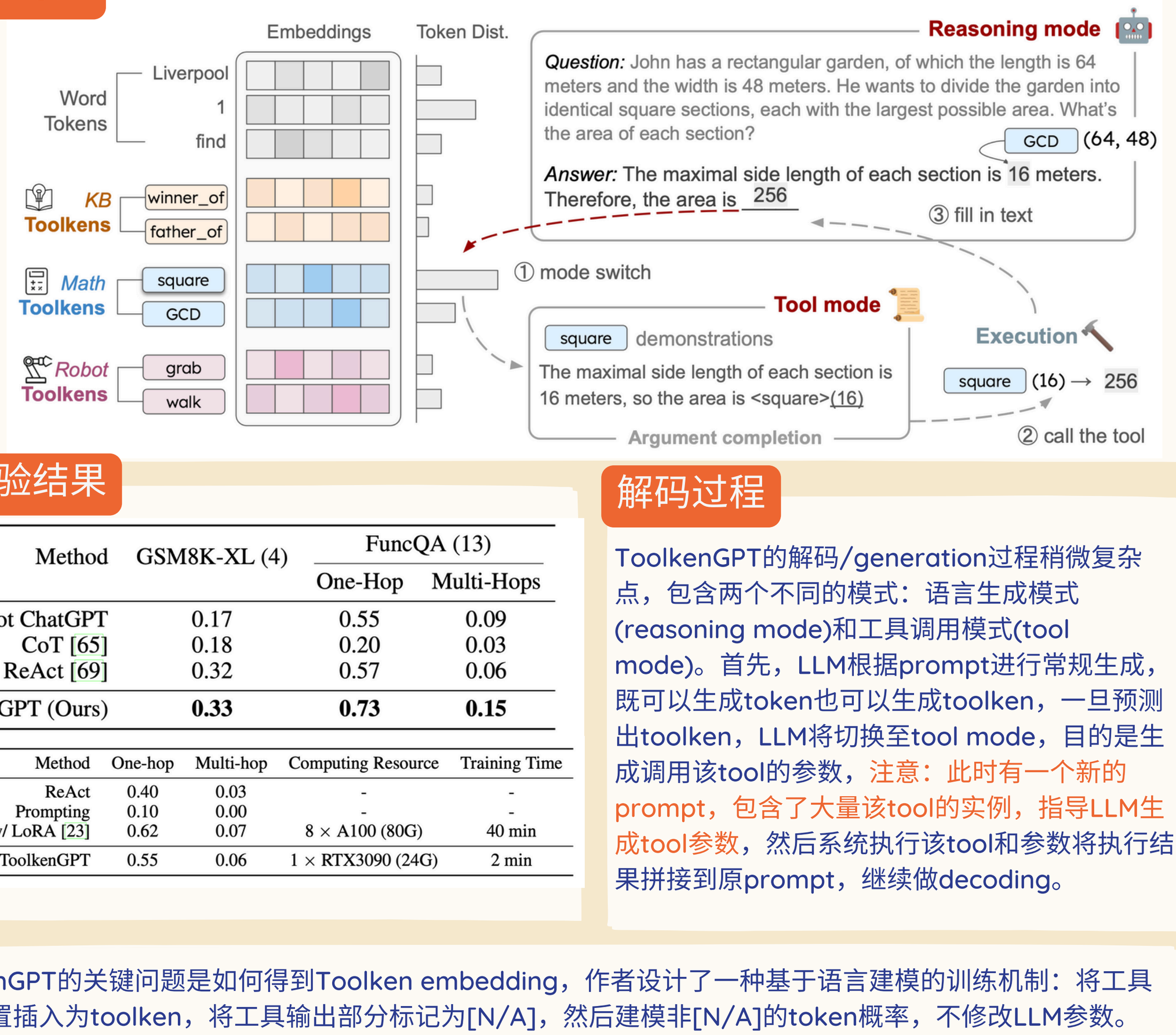
简介

本文提出了ToolkenGPT，它的核心思想是将每个工具tool表示为一个新的token(称为toolken)，然后学习一个toolken embedding，通过将toolken embedding合并到词表vocab中，ToolkenGPT可以让LLM能够像预测普通token一样预测何时用何种工具调用，从而实现统一的token/tool生成机制。ToolkenGPT无需对LLM做tuning，仅仅是扩展其词表vocab，就能快速适配新tool并且比prompt方法能支持更多的tool，在并且还支持一条数据的multi-tool调用。

背景

如何让LLM能够学会使用tool增强推理、计算与生成的事实性？现有方法主要依赖两类方案（本文是23年的论文呢）：一是通过少量示例的in-context learning，即few-shot方法，缺点是受限于上下文长度，每种tool都要有使用实例，难以支持大规模的tool调用；二是通过sft LLM，但是sft一般是针对一种或少数几种tool构建训练集训练，难以扩展到大规模tool，并且适配新tool的能力很差。
ToolkenGPT提出：只需将每个工具表示为一个单独的token(toolken)，并学习其向量表示，就能让LLM像生成普通token一样生成工具调用。

TOOLKENGPT



部分实验结果

Method	GSM8K-XL (4)	FuncQA (13)	
		One-Hop	Multi-Hops
0-shot ChatGPT	0.17	0.55	0.09
CoT [65]	0.18	0.20	0.03
ReAct [69]	0.32	0.57	0.06
ToolkenGPT (Ours)	**0.33**	**0.73**	**0.15**

Method	One-hop	Multi-hop	Computing Resource	Training Time
ReAct	0.40	0.03	-	-
Prompting	0.10	0.00	-	-
Fine-tune w/ LoRA [23]	0.62	0.07	8 × A100 (80G)	40 min
ToolkenGPT	0.55	0.06	1 × RTX3090 (24G)	2 min

解码过程

ToolkenGPT的解码/generation过程稍微复杂点，包含两个不同的模式：语言生成模式(reasoning mode)和工具调用模式(tool mode)。首先，LLM根据prompt进行常规生成，既可以生成token也可以生成toolken，一旦预测出toolken，LLM将切换至tool mode，目的是生成调用该tool的参数，注意：此时有一个新的prompt，包含了大量该tool的实例，指导LLM生成tool参数，然后系统执行该tool和参数将执行结果拼接到原prompt，继续做decoding。

思考

ToolkenGPT的关键问题是如何得到Toolken embedding，作者设计了一种基于语言建模的训练机制：将工具调用位置插入为toolken，将工具输出部分标记为[N/A]，然后建模非[N/A]的token概率，不修改LLM参数。
至于如何构建训练toolken embedding的训练集，toolken embedding是否和vocab embedding语义对齐？显然至关重要

@机器爱学习