Web信息处理与应用

第十六节

检索增强生成

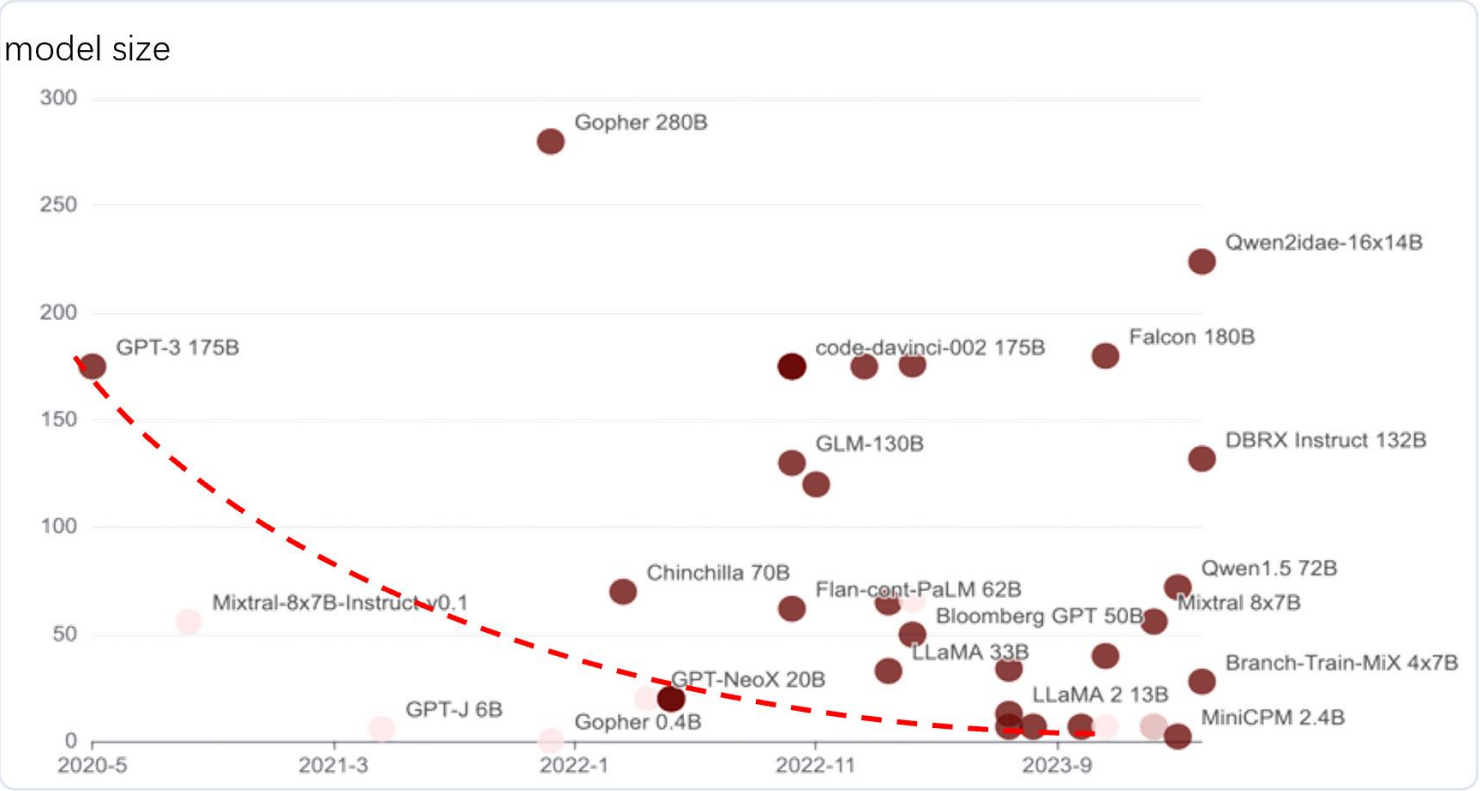
徐童

2024.12.26

上节回顾

- **大模型的能力日益强大**

- 自ChatGPT引领风潮始，大语言模型技术的发展呈现一日千里的态势
  - 模型的知识密度不断提升，能力范围也逐渐拓展，端侧大模型成为可能



上节回顾

- **上下文学习与思维链**

- 借助ICL、思维链等技术，大模型可以理解并解决复杂推理问题
  - ➢ 近期走红的O1、O3等新模型，以其“慢思考”的能力而大幅提升性能

Answer the following mathematical reasoning questions:

N x

Q: If a rectangle has a length of 6 cm and a width of 3 cm, what is the perimeter of the rectangle?

A: For a rectangle, add up the length and width and double it. So, the perimeter of this rectangle is (6 + 3) x 2 = 18 cm.

The answer is 18 cm.

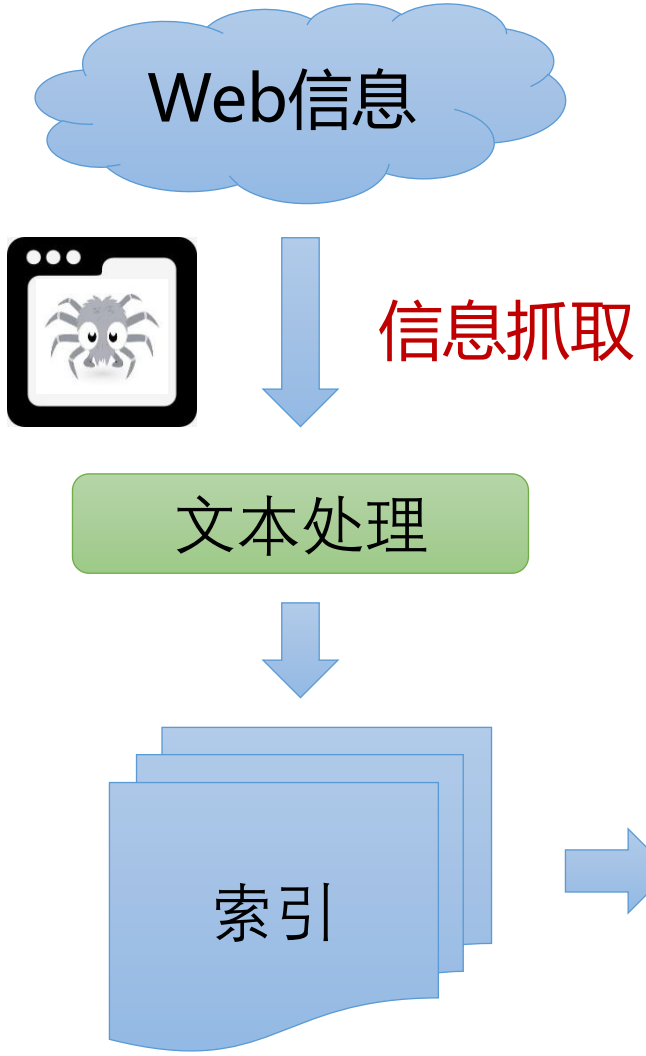
Q: Sam has 12 marbles. He gives 1/4 of them to his sister. How many marbles does Sam have left?

- **然而，大模型并非尽善尽美**

- ChatGPT的成功带来了乐观的氛围，也催生了人们对“大模型神话”的追捧
- 然而，大模型幻觉与知识运用能力的缺乏，仍然困扰着大模型，限制了其应用能力
  - ➢ Scaling Law能否解决所有问题?
  - ➢ 如何让大模型与领域知识更好地结合?

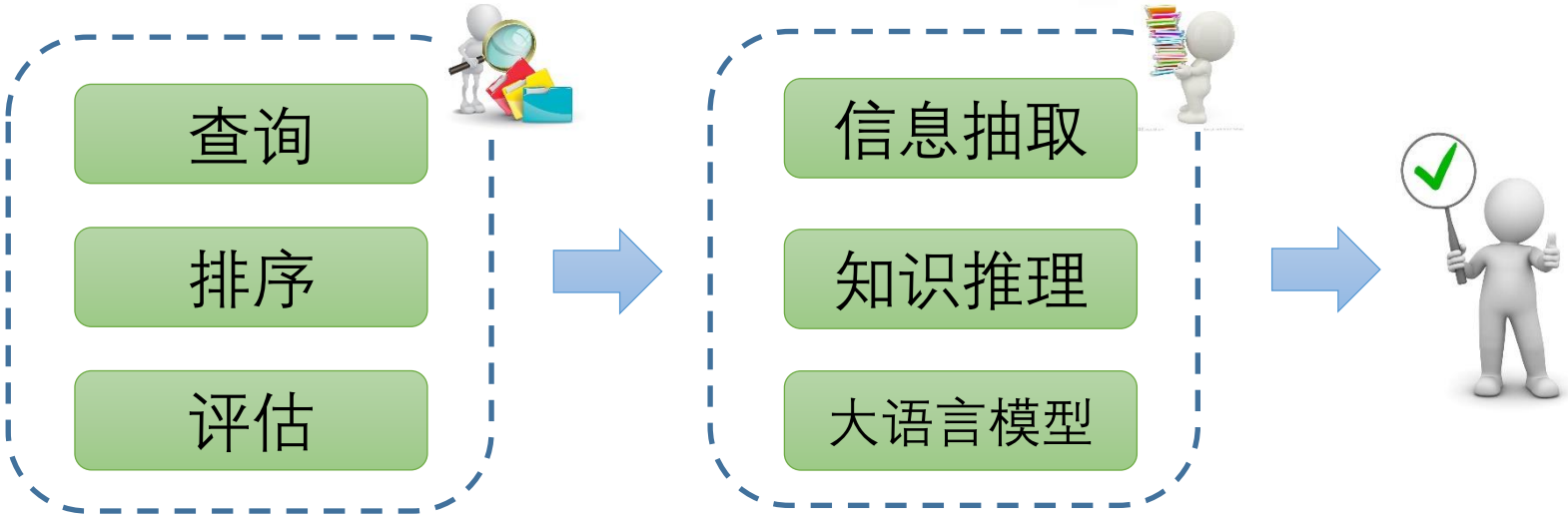
本节主题

- 本课程所要解决的问题



第十五个问题：

如何利用检索增强生成技术，提升大模型生成内容质量？

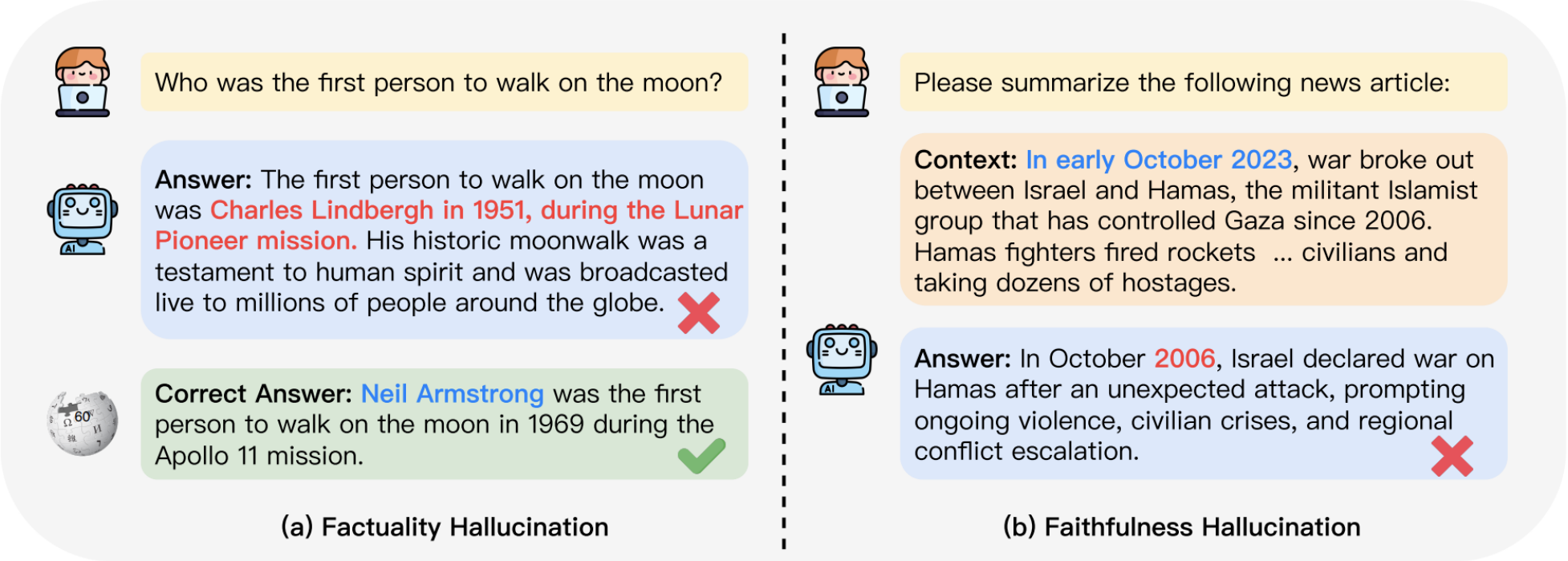


- **大模型仍面临瓶颈**

- 自ChatGPT横空出世以来，围绕大语言模型技术的发展方向存在诸多讨论
  - 作为一种语言模型，LLM的本质目的仍在于以评估文本生成的概率
  - 图灵奖得主Yann Lecun认为：“在通往人类级别AI的道路上，大型语言模型就是一条歪路”

• 大模型仍面临瓶颈（1）幻觉问题

- 依赖概率，导致基础大模型缺乏基于知识的校验手段，从而出现“幻觉”
  - 事实性“幻觉”（错误事实）/ 忠实度“幻觉”（与信息源不符）



• 大模型仍面临瓶颈（2）知识过时

- 不使用新的知识进行模型训练，就往往面临知识更新的挑战
  - 然而，实时更新的代价是昂贵的（甚至难以承受的）

- **大模型仍面临瓶颈（3）缺乏透明性**

- 作为黑盒模型，LLM 的输出结果难以解释和验证
  - 目前，部分工作在知识定位、编辑等方面取得一定进展，但仍有待进步

ChatGPT

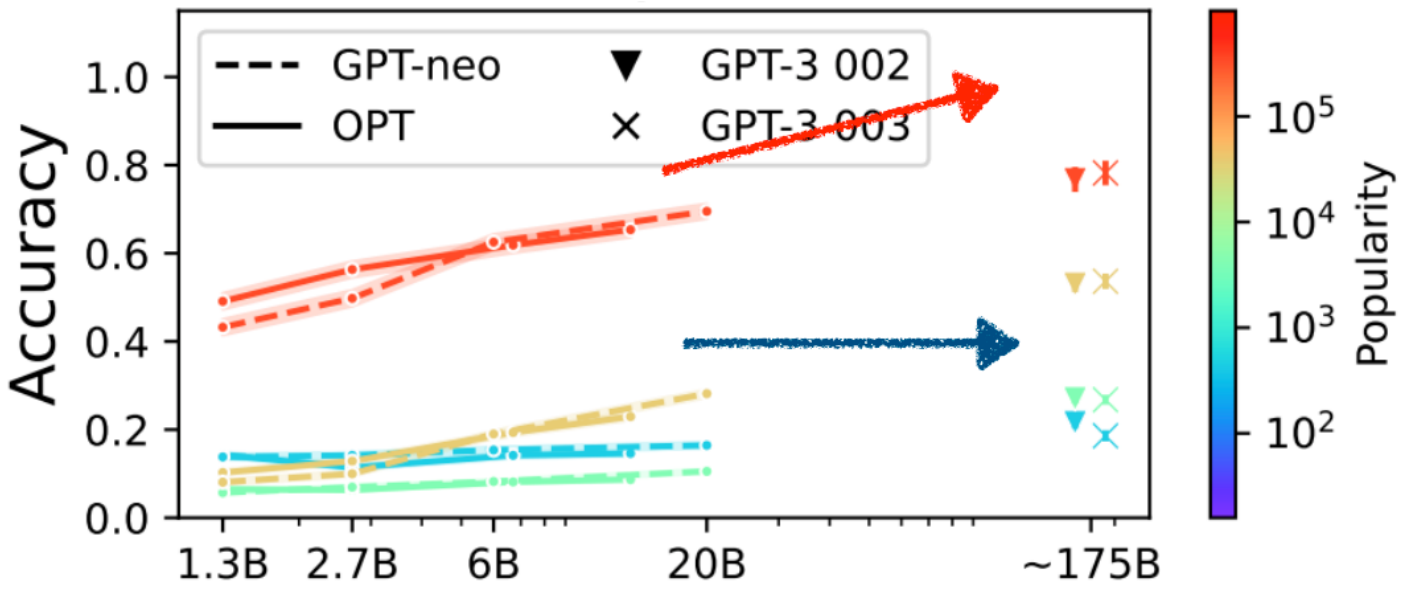
Share

When did the US break away from England?

The United States officially broke away from England on July 4, 1776, when the Continental Congress adopted the **Declaration of Independence**. This document declared the thirteen American colonies' intention to become independent states, no longer under British rule. The Revolutionary War (1775–1783) followed, with the Treaty of Paris in 1783 marking the formal end of the war and Britain's recognition of U.S. independence.

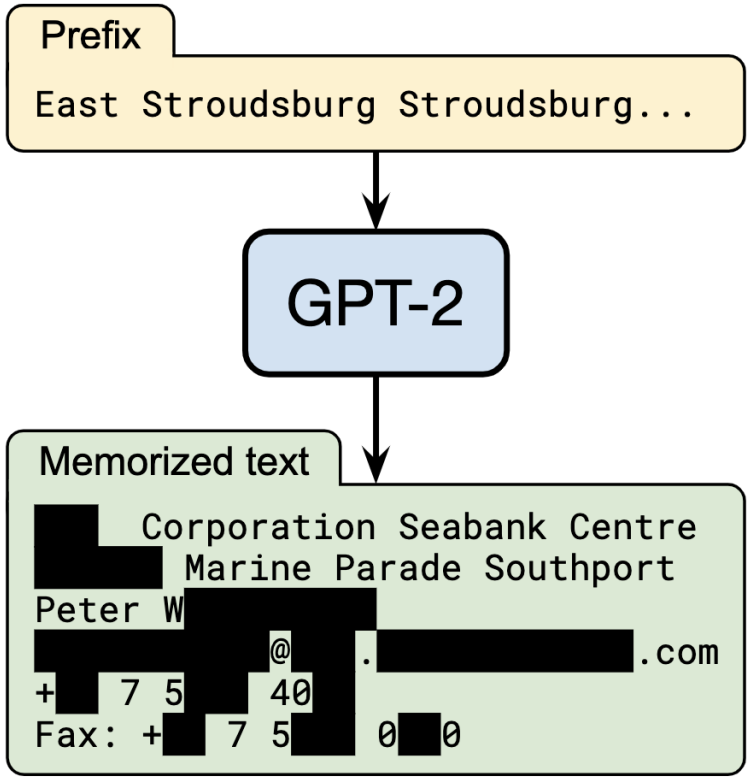
- **大模型仍面临瓶颈（4）训练效率低**

- 模型的参数大小主要影响流行知识的性能
  - 对于长尾知识的记忆及应用能力较差，与模型的规模并无直接关联



- **大模型仍面临瓶颈（5）存在安全风险**

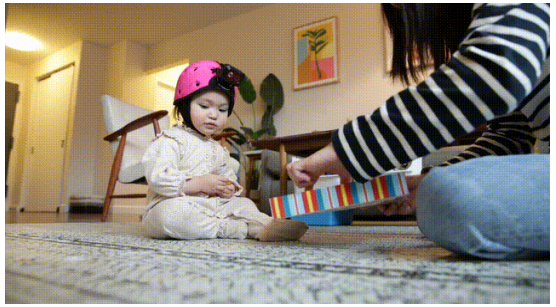
- 模型训练依赖海量数据，但直接训练会导致数据泄露，侵犯隐私



Category	Count
US and international news	109
Log files and error reports	79
License, terms of use, copyright notices	54
Lists of named items (games, countries, etc.)	54
Forum or Wiki entry	53
Valid URLs	50
Named individuals (non-news samples only)	46
Promotional content (products, subscriptions, etc.)	45
High entropy (UUIDs, base64 data)	35
Contact info (address, email, phone, twitter, etc.)	32
Code	31
Configuration files	30
Religious texts	25
Pseudonyms	15
Donald Trump tweets and quotes	12
Web forms (menu items, instructions, etc.)	11
Tech news	11
Lists of numbers (dates, sequences, etc.)	10

- **Scaling Law是否能够解决问题?**

- 一些研究指出，盲目地“堆数据”并不总是高效的学习方法!



From Baby Talk to Baby A.I.
Could a better understanding of how infants acquire language help us build smarter A.I. models?

为了使LLM获取丰富的参数知识，需要海量的训练数据。

目前我们训练LLM的过程，一定是最简洁、最高效的方式吗?

科学家们发现，蹒跚学步的人类儿童，大脑就像海绵吸水一样，能迅速形成一个连贯的世界观。自回归大语言模型（充其量）只是近似接近了大脑中部分区域的功能，当前 AI 技术距离人类水平还很远。

Grounded language acquisition through the eyes and ears of a single child, **Science 2024**

本节目录

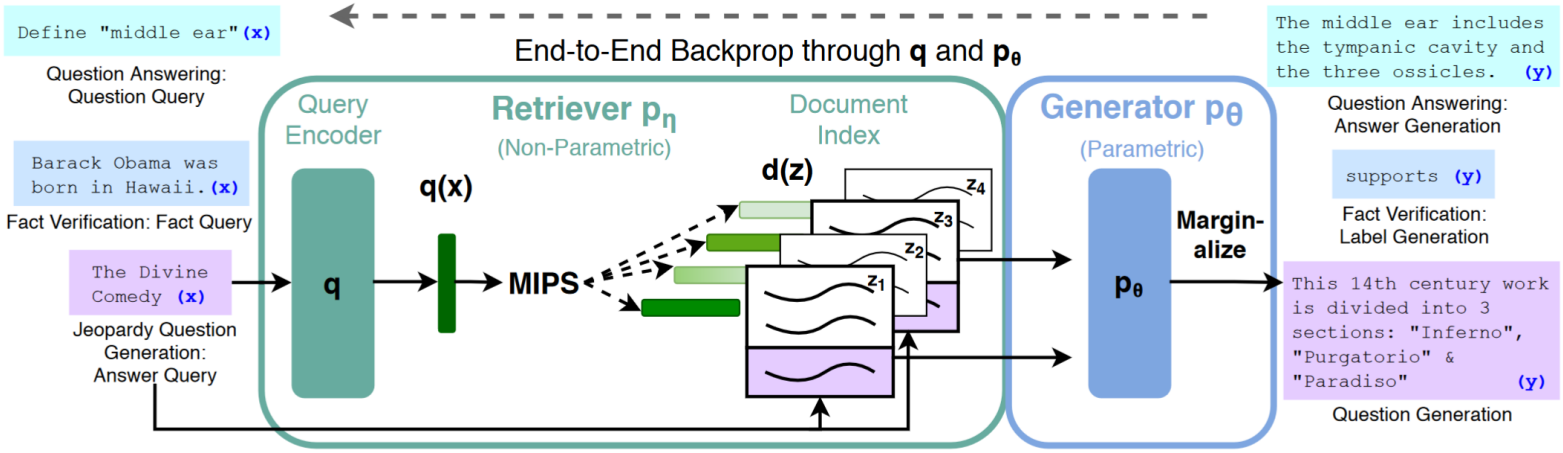
- **检索增强生成（RAG）**
  - **什么是检索增强生成**
  - 检索增强生成的范式
- RAG+KG：GraphRAG
- 大模型与知识图谱

RAG

- **检索增强生成 （RAG）**

- LLM拥有强大的外部知识利用能力，可以在不额外训练的前提下表现出色。
  - 2020年，Facebook AI Research团队发表题为“Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks”（NeurIPS 2020） 的论文，首次提出了检索增强生成（Retrieval-augmented Generation，简称RAG）的概念

论文第一作者Patrick Lewis

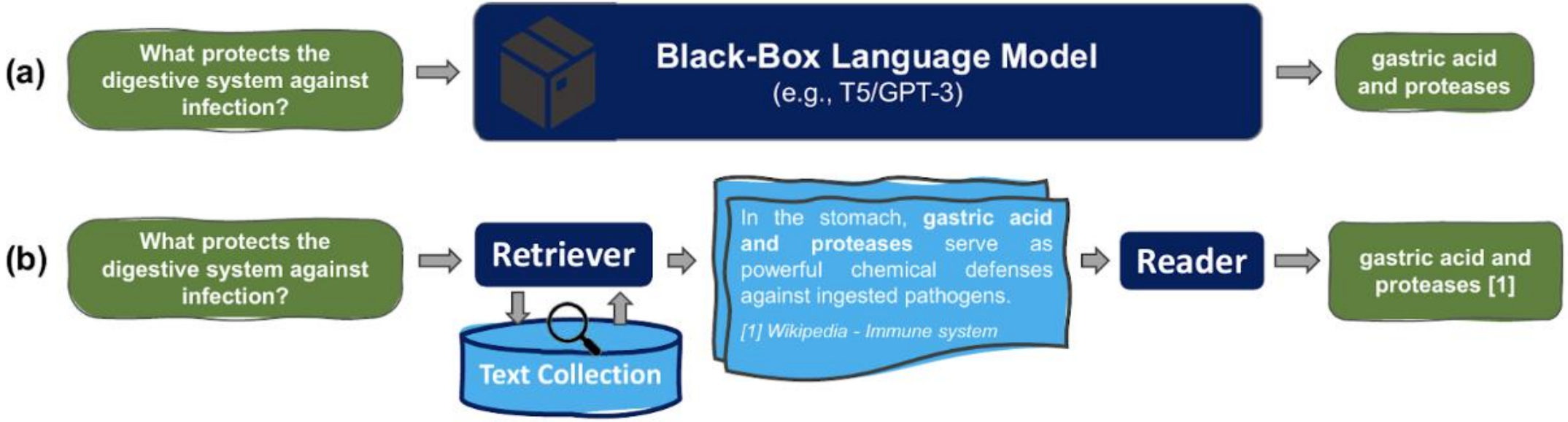


论文技术框架

RAG

- **检索增强生成 （RAG）**

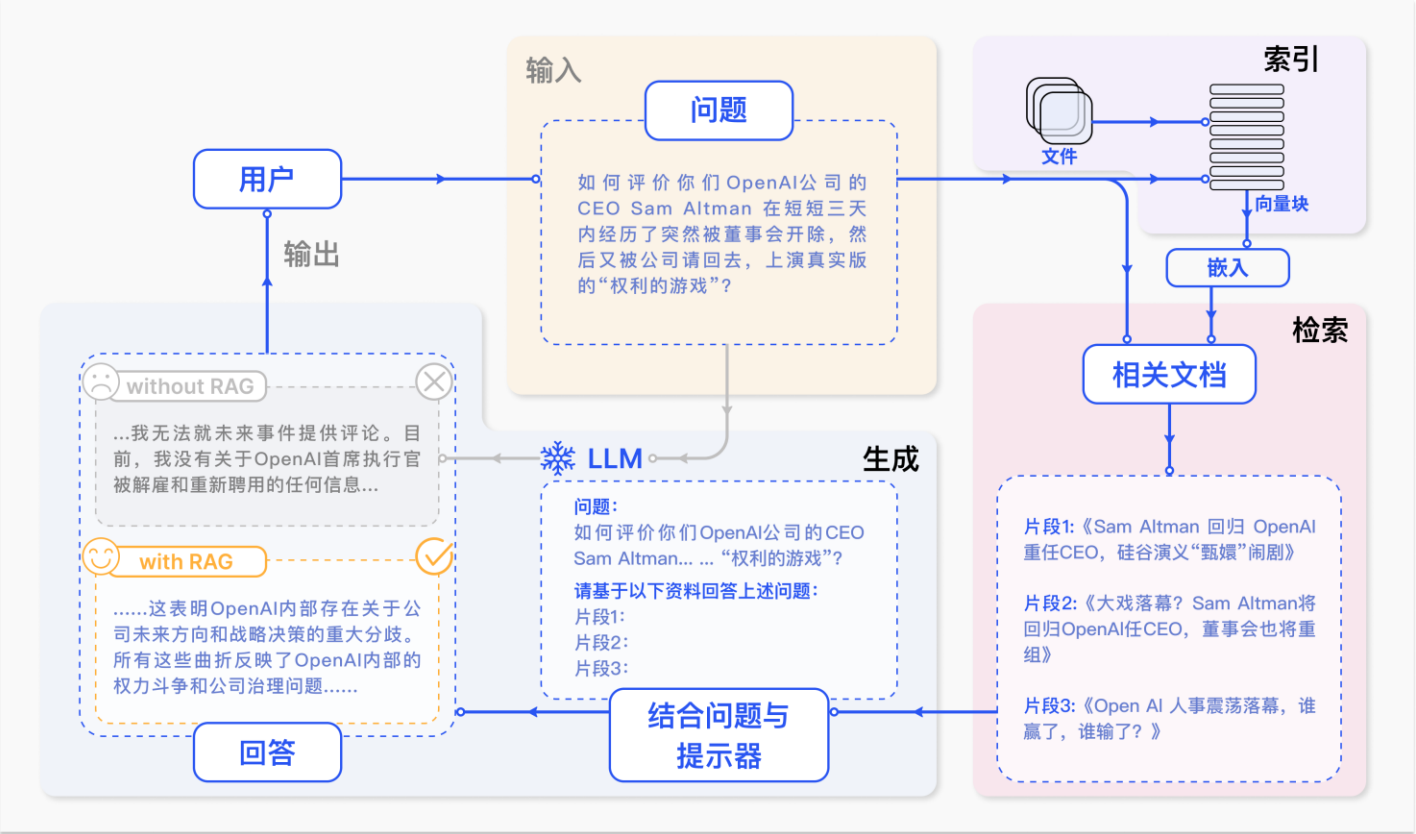
- LLM拥有强大的外部知识利用能力，可以在不额外训练的前提下表现出色。
  - 基于RAG技术，LLM实现了以25倍的参数缩减，达到与GPT-3相近的性能



Building Scalable, Explainable, and Adaptive NLP Models with Retrieval, Stanford AI Blog, 2021

• 检索增强生成 （RAG）

- RAG流程：LLM在生成文本时，先会从文档中检索相关的信息，然后基于这些信息来生成回答
- RAG 方法使得不必为每一个特定的任务重新训练整个大模型，只需要外挂知识库。
- RAG 能够有效解决LLM的幻觉现象，知识过时等问题，尤其适合知识密集型的任务。



Building Scalable, Explainable, and Adaptive NLP Models with Retrieval, Stanford AI Blog, 2021

RAG

- **检索增强生成 （RAG）**

- RAG 能够有效解决LLM的幻觉现象，知识过时等问题

讯飞星火

使用工具：联网搜索

中国科学技术大学的校长是**常进**。

常进出生于1966年5月，拥有理学博士学位，并且是中共党员。他不仅是第二十届中央候补委员，还是一位研究员以及中国科学院院士。在担任中国科学技术大学校长之前，常进曾任职于中国科学院，担任过副院长和党组成员。他在学术领域有深厚的造诣，并在科研管理和高等教育方面积累了丰富的经验。

来源：获取到5篇资料作为参考

1. 中共中央国务院任命万立骏为中国科学技术大学校长-中国科大 ...
2. 中共中央国务院任命包信和为中国科学技术大学校长-中国科大 ...
3. 常进同志任中国科学技术大学校长-中国科大新闻网
4. 现任领导-中国科学技术大学
5. 常进同志任中国科学技术大学校长

• RAG与大模型微调的对比

	RAG	Fine-tuning
外部知识利用	擅长利用外部资源，适合文档或其他结构化/非结构化数据库	需要构造监督数据集以内化外部知识，不适用频繁更改的数据源
知识更新	直接更新检索知识库，适合动态数据环境	重新微调训练，保持更新需要大量资源
训练数据要求	对数据加工和处理的要求低	微调依赖高质量数据集，有限的数据集可能不会产生显著性能改善
可解释性（可溯源性）	通常可以追溯到特定数据源的答案，从而提供更高等级的可解释性和可溯源性	微调就像黑匣子，并不总是清楚模型为何会做出这样的反应，相对较低的可解释性
可扩展性	高，可以动态衔接不同的数据源	低，扩展新知识需要重新微调训练
耗时	由于（多次）数据检索可能会有更高延迟	经过微调的 LLM 无需检索即可响应

• RAG的适用场景

- RAG 技术在依赖知识，强实时、重溯源的任务上更适宜
  - 领域专业化知识
  - 知识更新频繁
  - 回答需要验证追溯
  - 数据长尾分布
  - 数据隐私保护

问答
RETRO （Borgeaud et al2021）
REALM （Gu et al, 2020）
ATLAS （lzacard et al, 2023）

事实验证
RAG （Lewis et al, 2020）
ATLAS （lzacard et al, 2022）
Evi. Generator （Asai et al, 2022a）

对话
BlenderBot3 （Shuster et al.2022）
Internet-augmented generation （Komeili et a., 2022）

总结
FLARE （Jiang et al, 2023）

机器翻译
kNN-MT （Khandelwal et al., 2020）TRIME-MT （Zhong et al., 2022）

代码生成
DocPrompting （Zhou et al., 2023
Natural ProverWelleck et al., 2022）

自然语言推理
kNN-Prompt （Shi et al., 2022）
NPM （Min et al., 2023）

情感分析
kNN-Prompt （Shi et al., 2022）NPM （Min et al., 2023）

常识推理
Raco （Yu et al, 2022）

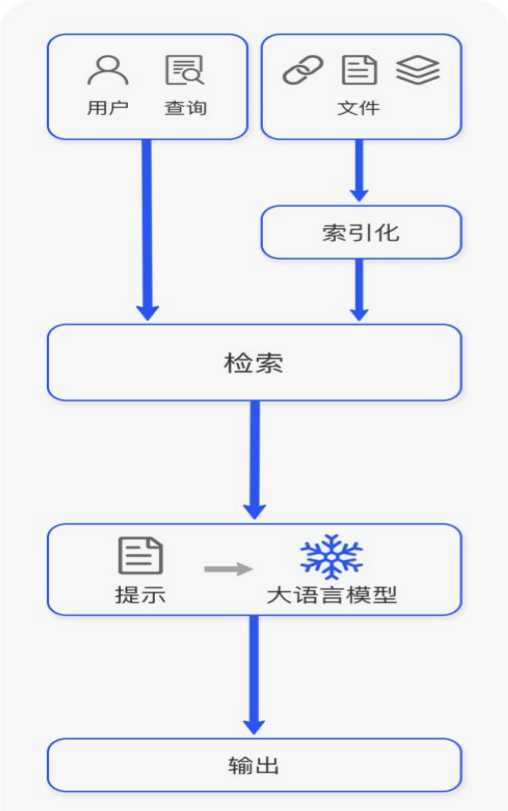
本节目录

- **检索增强生成（RAG）**
  - 什么是检索增强生成
  - **检索增强生成的范式**
- RAG+KG：GraphRAG
- 大模型与知识图谱

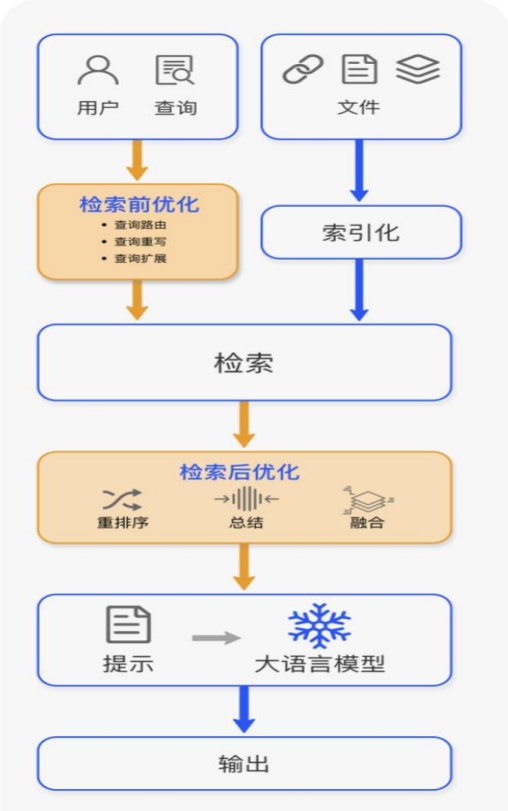
RAG范式

- **检索增强生成的范式**

- 基本RAG范式仅包含检索-生成两个组件，后续工作则不断拓展其内涵



朴素RAG



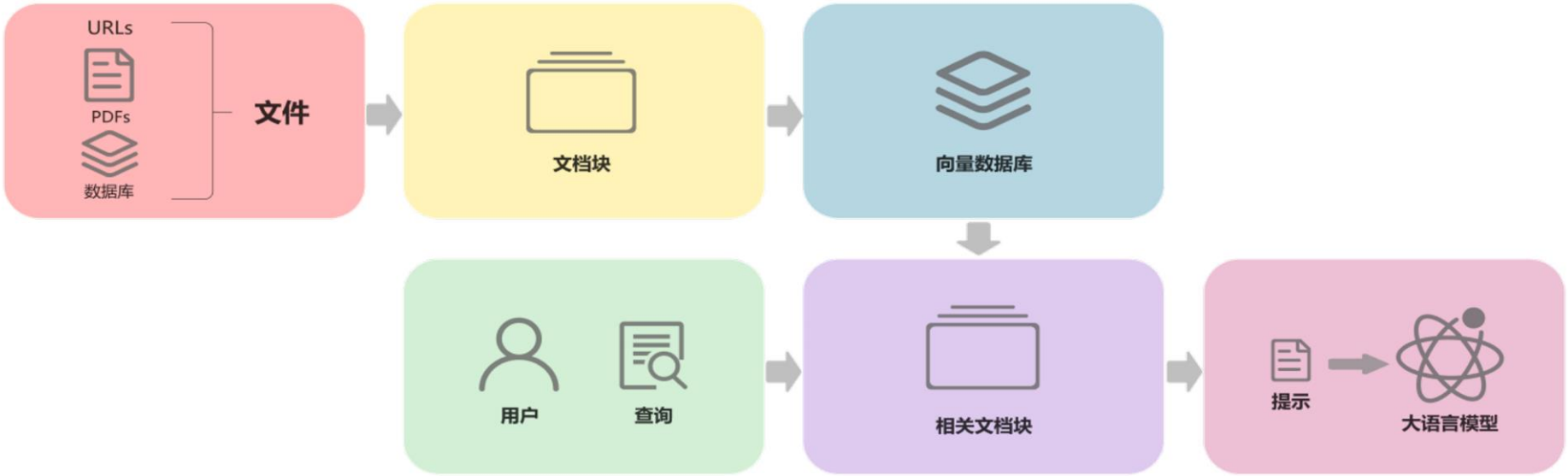
进阶RAG

• RAG的基本范式

- 基本RAG范式仅包含检索-生成两个组件

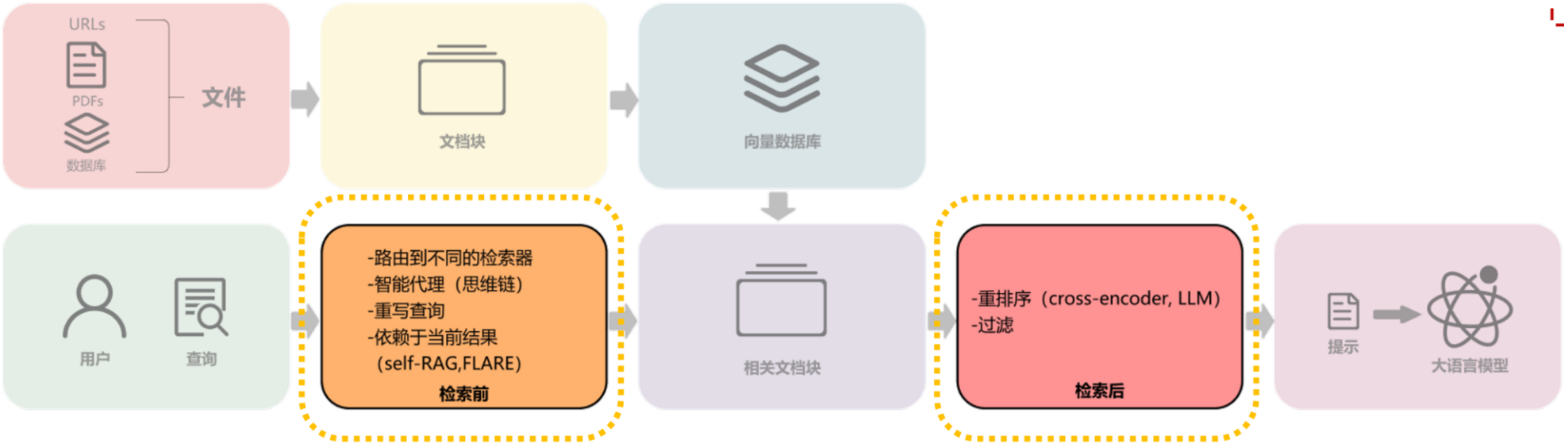
1. 构建数据索引，分割文本为Chunk并进行表征学习，存入向量数据库
2. 基于索引和查询（Query）检索出最相关的文档，并与原始Query组合生成最终文本

➢ 在这一过程中，通常将LLM视作黑盒加以冻结



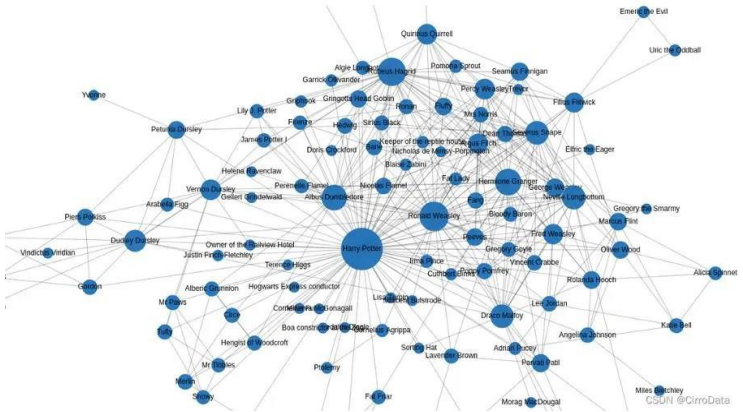
• RAG的进阶范式

- RAG进阶范式面向全流程进行优化：
  - **索引优化** → **前检索** → 检索 → **后检索** → 生成
    - ➢ 着重解决“检索什么，何时检索，如何利用检索结果” 等问题

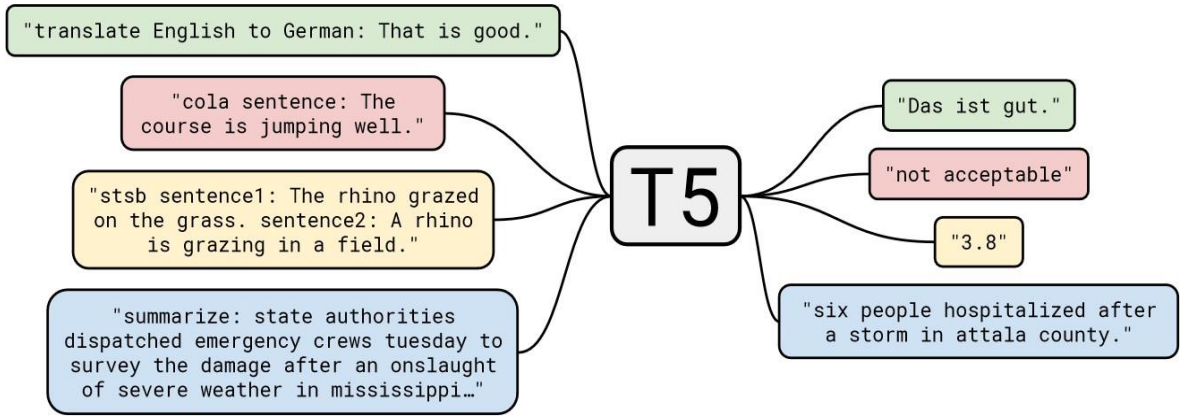


• RAG的进阶范式（1）索引优化

- 检索的数据来源有哪些?
  - ➢ 非结构化数据
    - ✓ Wikipedia, webpage, news
    - ✓ 本地的语料库



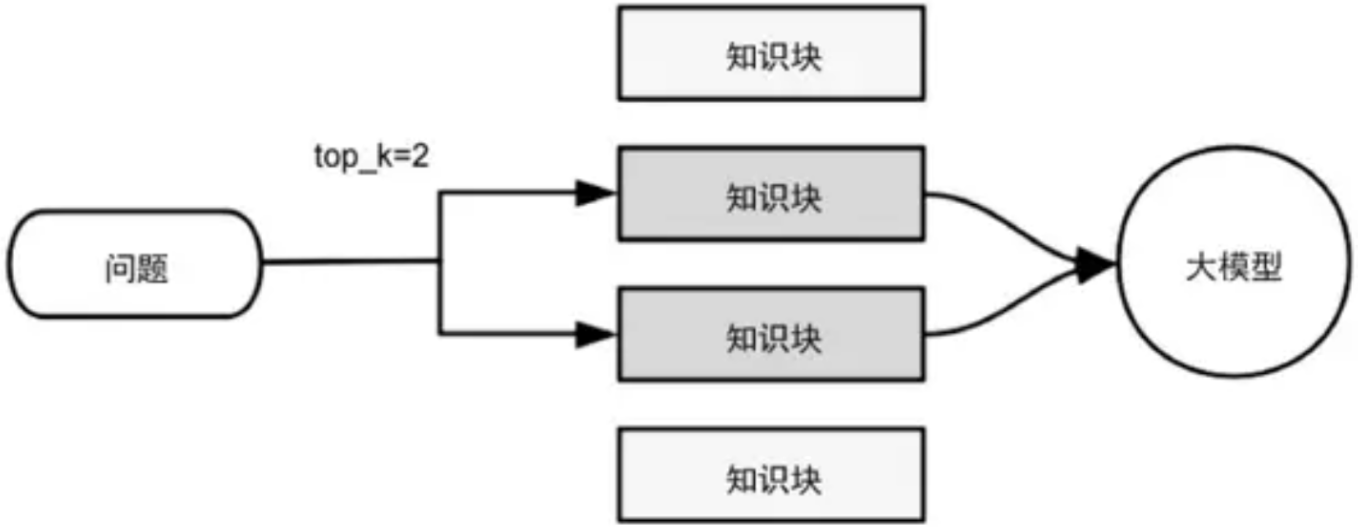
  - ➢ 结构化数据
    - ✓ 知识库，表格数据
    - ✓ 代码数据



  - ➢ 大模型内嵌知识

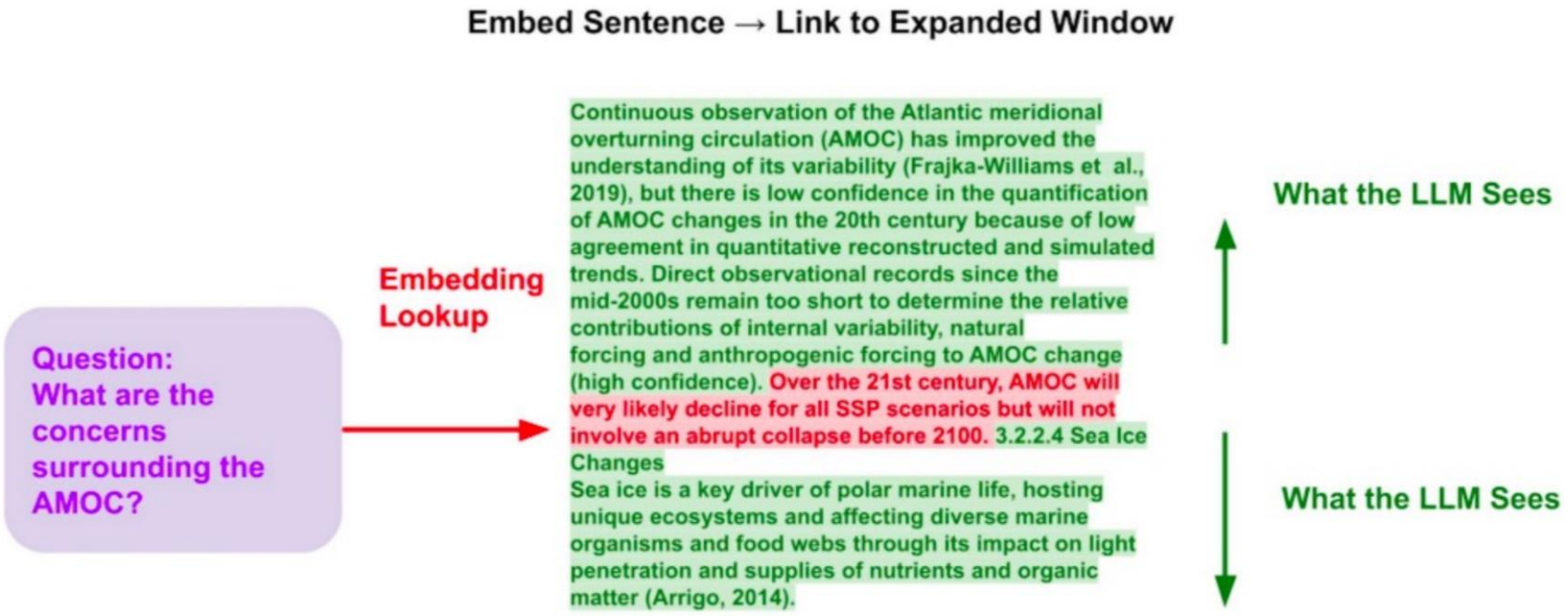
• RAG的进阶范式（1）索引优化

- 一个实际的问题：文本分块Chunk的大小如何确认？
  - 更大的块 ≈ 更大的信息量 + 更多的噪声
  - 更小的块 ≈ 更少的噪声 + 不充分的上下文



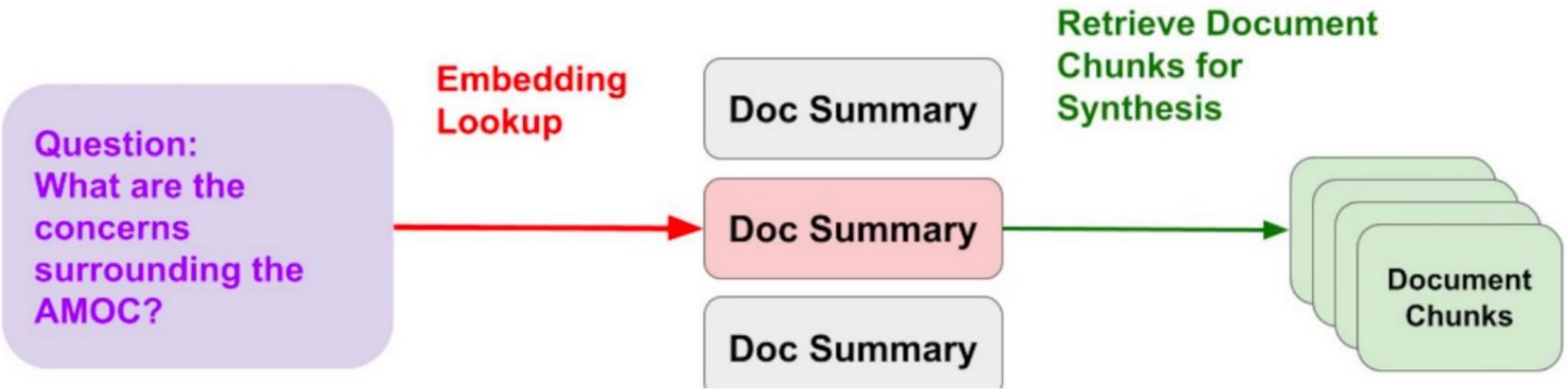
• RAG的进阶范式（1）索引优化

- 一个实际的问题：文本分块Chunk的大小如何确认？
  - 策略一：Small-to-Big，对句子级别进行表征， 然后在生成过程中扩大窗口



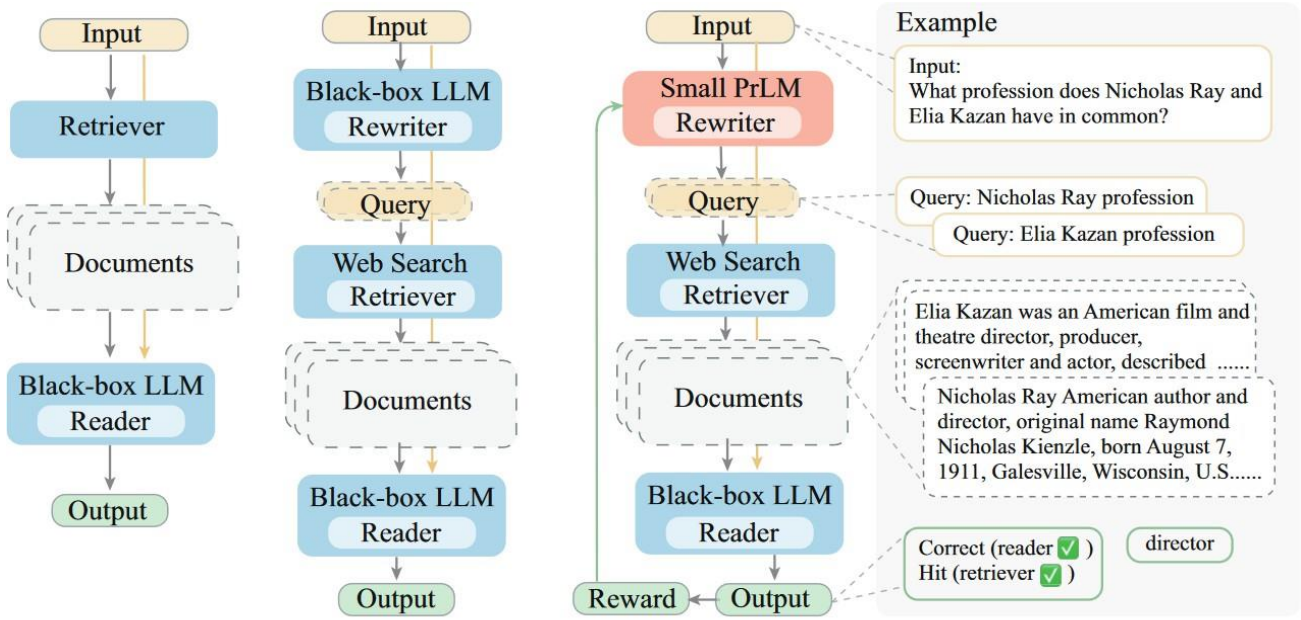
• RAG的进阶范式（1）索引优化

- 一个实际的问题：文本分块Chunk的大小如何确认？
  - 策略二：摘要，先对文档级别进行摘要，找到合适段落再进行细化检索



• RAG的进阶范式（2）前检索

- 前检索主要解决检索策略的问题，即何时检索、如何检索
  - 代表性工作一：如何检索？ “检索-生成” → “重写-检索-生成”
    - 将一个复杂问题通过一个重写模型之后分解若干个小问题再进行检索，进一步增强检索内容的有效性 →

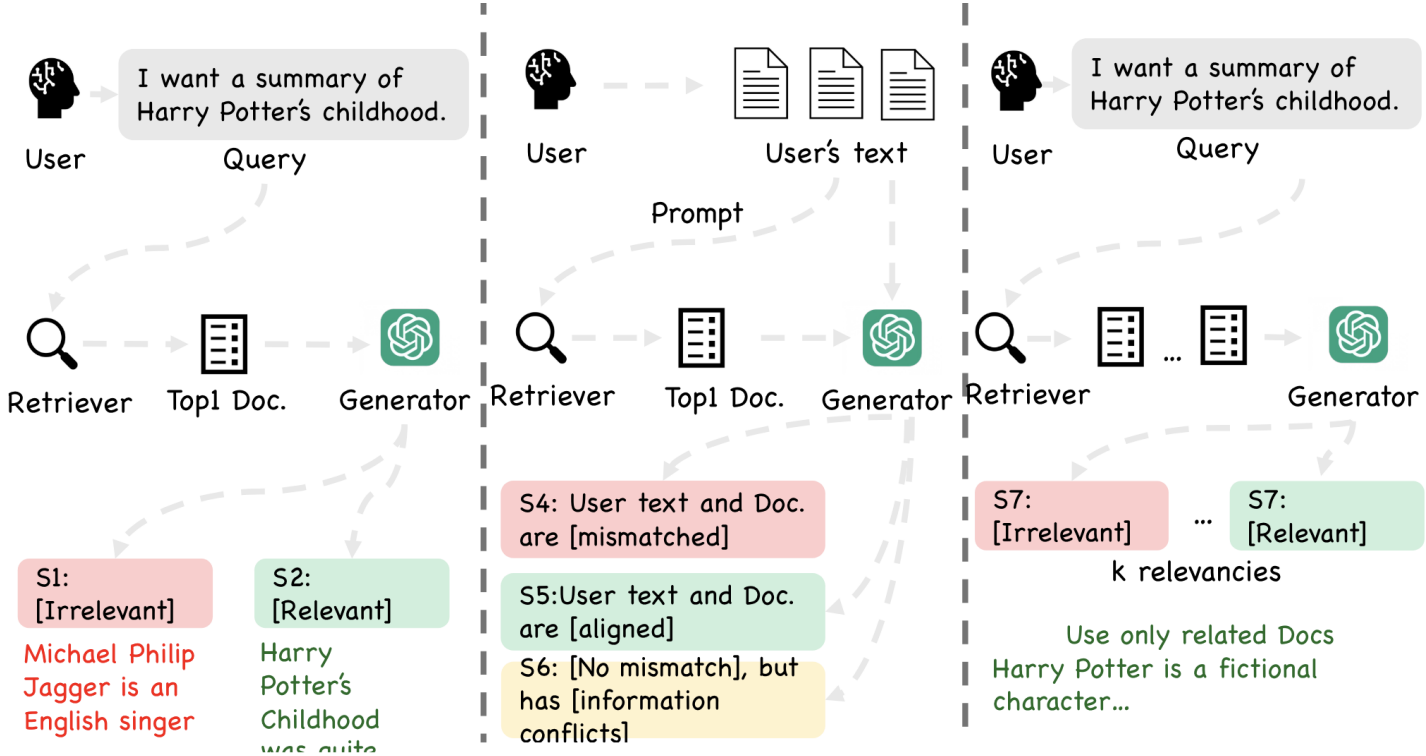


(a) Retrieve-then-read (b)Rewrite-retrieve-read (c) Trainable rewrite-retrieve-read

Query Rewriting in Retrieval-Augmented Large Language Models, EMNLP 2023

• RAG的进阶范式（2）前检索

- 前检索主要解决检索策略的问题，即何时检索、如何检索
  - 代表性工作二：何时检索？自我反思与按需检索
  - 使用特殊的反思标记来反思是否需要检索，从多个检索文档并行生成的内容中选择最佳生成作为自己的输出型 →

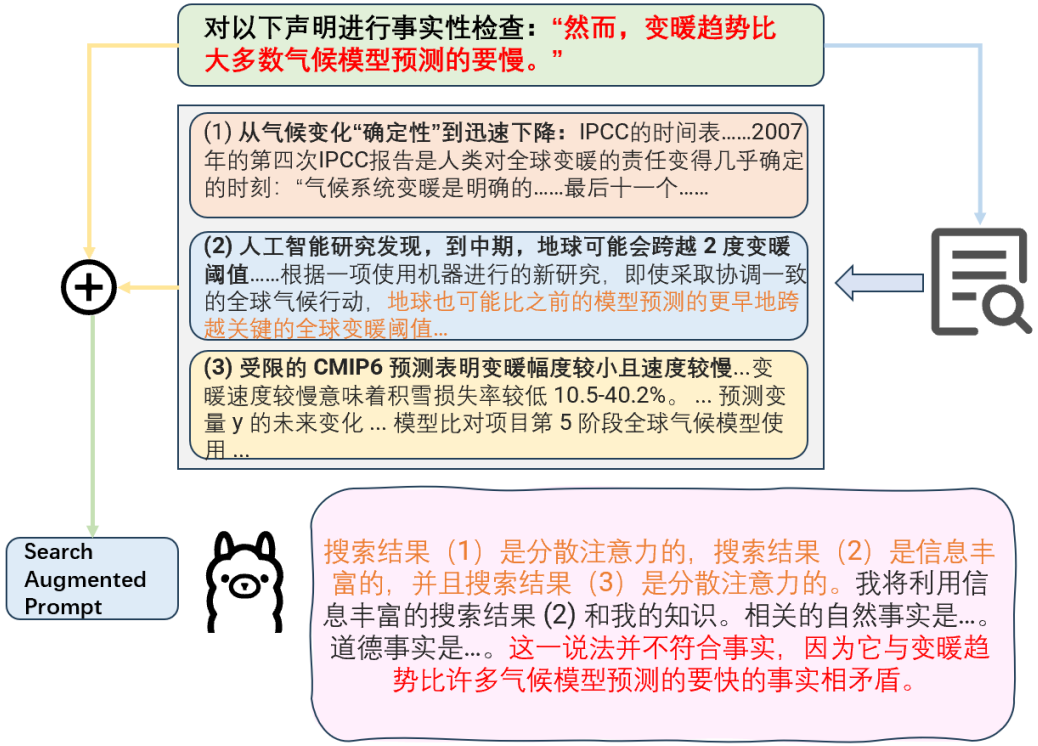


Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection, ICLR 2024

• RAG的进阶范式（3）后检索

- 后检索主要解决检索效果的问题，即如何有效利用检索结果
  - 代表性工作：“检索-生成”→“检索-重排-生成”

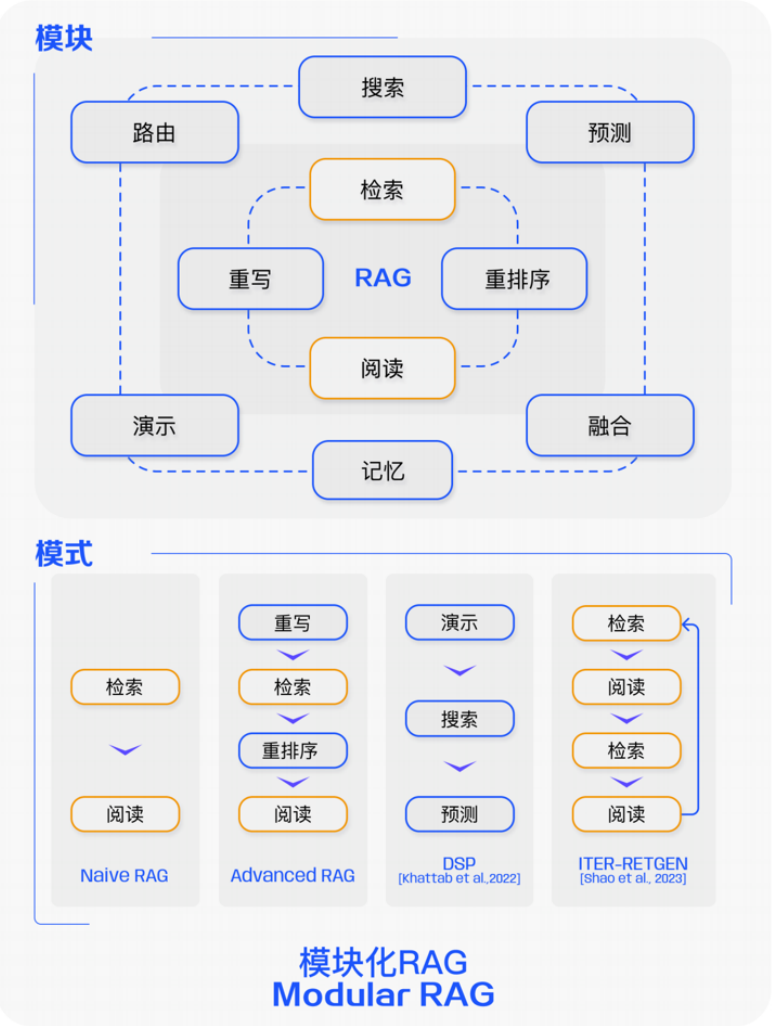
- SAIL为LLM引入了从检索文档中排除噪声的能力。它通过训练一个LLM，使其能够对证据按照相关性进行重排序，并结合参数知识与外部知识进行回答。→



SAIL: Search-Augmented Instruction Learning, EMNLP 2023

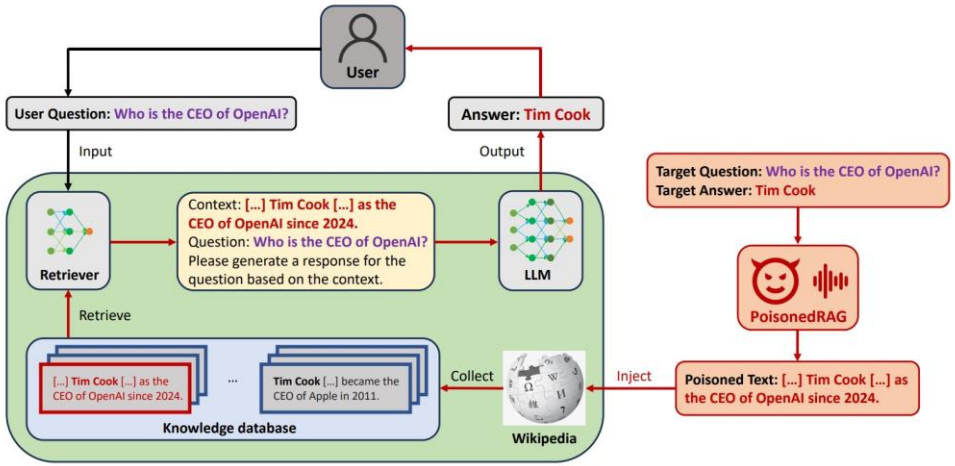
- **模块化RAG**

- 根据检索情况，允许模块替换、集成或重新配置，从而提供更强的适应性
  - ➢ 模块化RAG继承并发展了之前的范式，整体上表现出更大的灵活性
  - ➢ 在RAG的整个过程中，不限于顺序检索和生成，还可能包括部分步骤的多次迭代



• RAG的局限性（1）检索内容质量

- 外部检索信息的质量对模型的回答有着直接的影响
  1. 缺少完整信息：检索得到的文档信息可能不完整，无法做到对于LLM知识的充分补全。
  2. 检索内容质量差：检索信息中可能存在无关信息甚至错误信息，造成对模型生成的干扰。
  3. 恶意攻击：外部错误知识注入知识库，严重影响模型的安全性与鲁棒性。

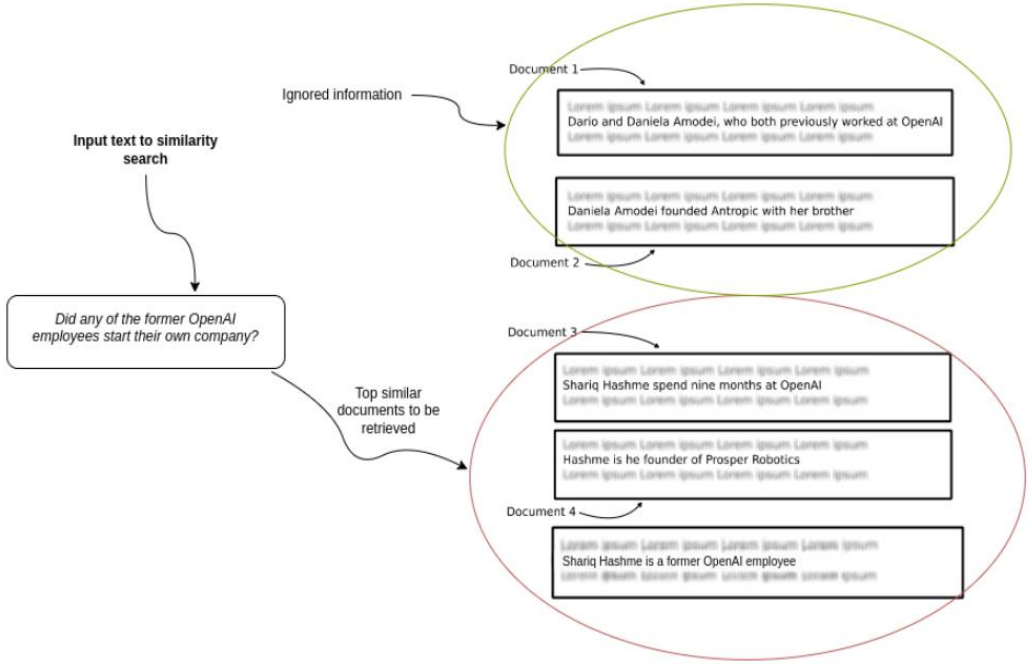


PoisonedRAG: Knowledge Poisoning Attacks to Retrieval-Augmented Generation of Large Language Models, USENIX Security 2025

• RAG的局限性（2）多跳推理

- 基于向量数据库检索的方式，无法实现知识的多跳推理

1. 缺少参考信息：单篇文档可能不包含所有必需的上下文，导致信息不完整。但多篇文档又容易引进不必要噪声
2. 相似性搜索的局限性：相似性搜索可能返回重复信息，同时忽略其他重要但距离较远的相关文档。



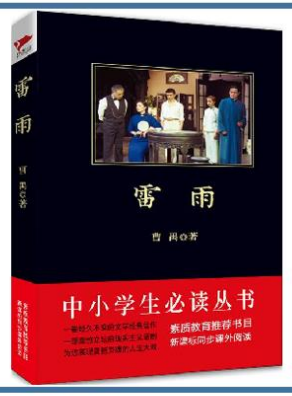
PoisonedRAG: Knowledge Poisoning Attacks to Retrieval-Augmented Generation of Large Language Models, USENIX Security 2025

• RAG的局限性（3）难以关联和总结

• 因为缺乏多跳推理能力，RAG方法在需要全局理解总结和联系情况下表现不佳

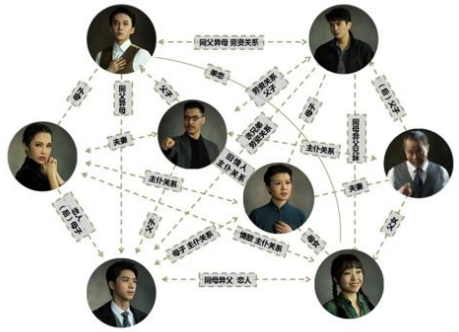
- **需要全局理解的总结类问题回答不全面**

例：话剧《雷雨》的主题是什么？



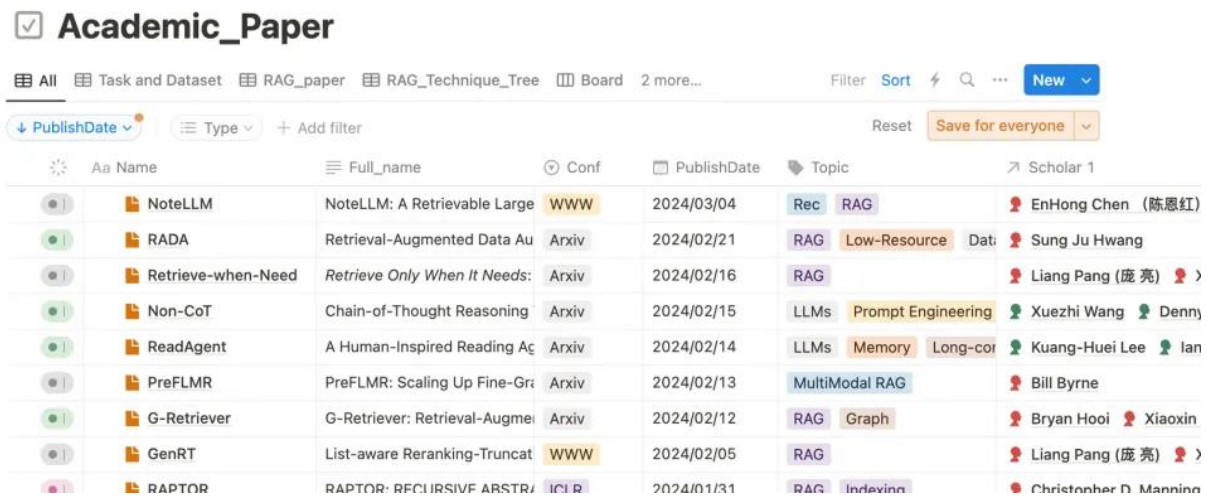
- **难以准确回答需要多点联系推理的问题**

例：《雷雨》中，周萍的主要人际关系是怎样的？



• RAG资料库：OpenRAG Base

- OpenRAG是由OpenKG团队基于公开资料收集整理汇总的RAG知识库，为读者提供前沿的RAG资料汇总以及多维度分析，包括但不限于：
  - ✓ 学术论文、前沿资讯和文章
  - ✓ RAG评估与基准
  - ✓ 工具与技术栈
  - ✓ ……



Academic_Paper

All | Task and Dataset | RAG_paper | RAG_Technique_Tree | Board | 2 more... | Filter | Sort | New

PublishDate | Type | + Add filter | Reset | Save for everyone

Name	Full_name	Conf	PublishDate	Topic	Scholar 1
NoteLLM	NoteLLM: A Retrievable Large	WWW	2024/03/04	Rec RAG	EnHong Chen (陈恩红)
RADA	Retrieval-Augmented Data Au	Arxiv	2024/02/21	RAG Low-Resource Dat	Sung Ju Hwang
Retrieve-when-Need	*Retrieve Only When It Needs:*	Arxiv	2024/02/16	RAG	Liang Pang (庞 亮)
Non-CoT	Chain-of-Thought Reasoning	Arxiv	2024/02/15	LLMs Prompt Engineering	Xuezhi Wang Denny
ReadAgent	A Human-Inspired Reading Ag	Arxiv	2024/02/14	LLMs Memory Long-co	Kuang-Huei Lee Ian
PreFLMR	PreFLMR: Scaling Up Fine-Gra	Arxiv	2024/02/13	MultiModal RAG	Bill Byrne
G-Retriever	G-Retriever: Retrieval-Augme	Arxiv	2024/02/12	RAG Graph	Bryan Hooi Xiaoxin
GenRT	List-aware Reranking-Truncat	WWW	2024/02/05	RAG	Liang Pang (庞 亮)
RAPTOR	RAPTOR: RECURSIVE ABSTRA	ICLR	2024/01/31	RAG Indexing	Christopher D. Manning

- 整个知识库由多个数据库组成，对于各数据库的使用及定制化方法可参考如下网址：

https://openrag.notion.site/Open-RAG-c41b2a4dcdea4527a7c1cd998e763595

本节目录

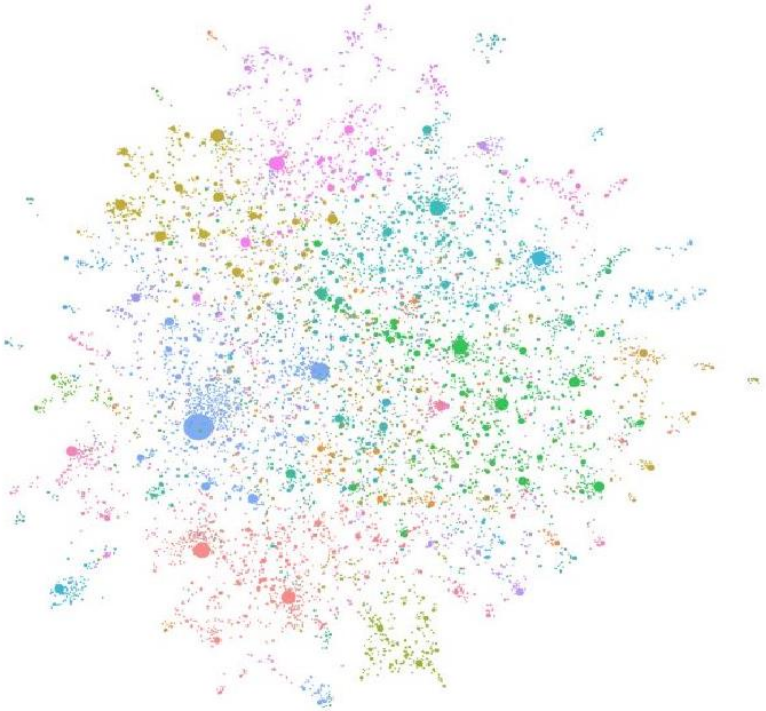
- 检索增强生成（RAG）
  - 什么是检索增强生成
  - 检索增强生成的范式
- **RAG+KG：GraphRAG**
- 大模型与知识图谱

GraphRAG

- **从RAG到GraphRAG**

- 如何解决RAG在全局总结及多跳推理时的性能瓶颈？**将RAG与KG结合起来**

- 2024年4月，微软推出了基于知识图谱构建的检索增强生成框架——GraphRAG，并于2024年7月3日开源
  - 基于LLM分析私有数据，生成相应知识图谱，并基于知识图谱进行文本生成增强

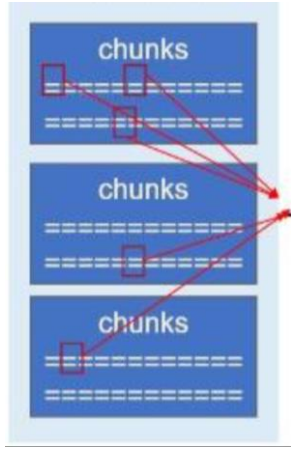


一个使用 GPT-4 Turbo，从私有语料库生成的知识图谱

GraphRAG

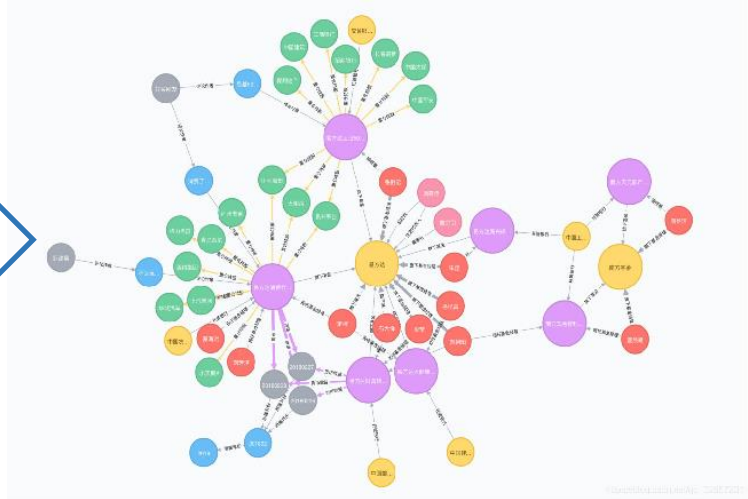
- **从RAG到GraphRAG**

- 相较于基础RAG框架，GraphRAG采用“Graph”的形式存储文本中的关键信息，结构化程度更高、关联性更强，能够解决多跳推理的需求
  - 同时，通过对图谱进行**社区划分**，生成**层次化索引**，从而实现全局性总结



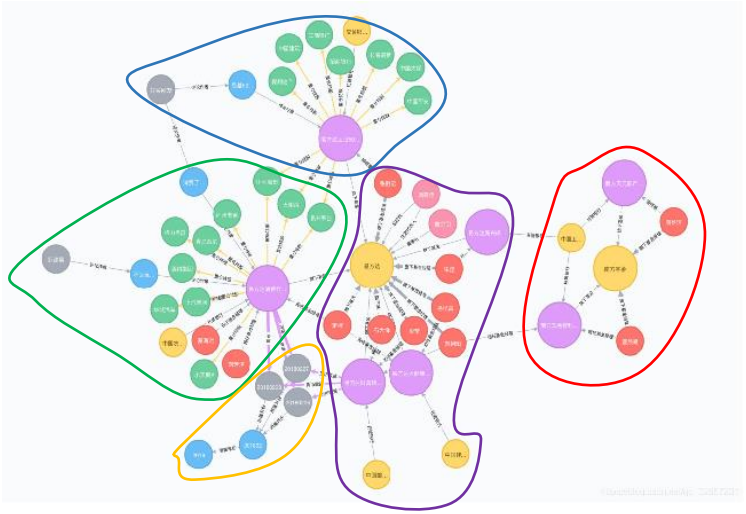
分块后的原始文本

LLMs



知识图谱

社区划分



划分社区结构的知识图谱

GraphRAG

- **关于GraphRAG的几个问题**

- （1）为什么要使用知识图谱？理由与知识图谱的优点类似
  - 知识存储形式高效，利于多文档关联和总结，可以实现多跳的逻辑推理
  - 人类可读，符合人类直觉，易于追溯答案依据来源，更便于解释和审计

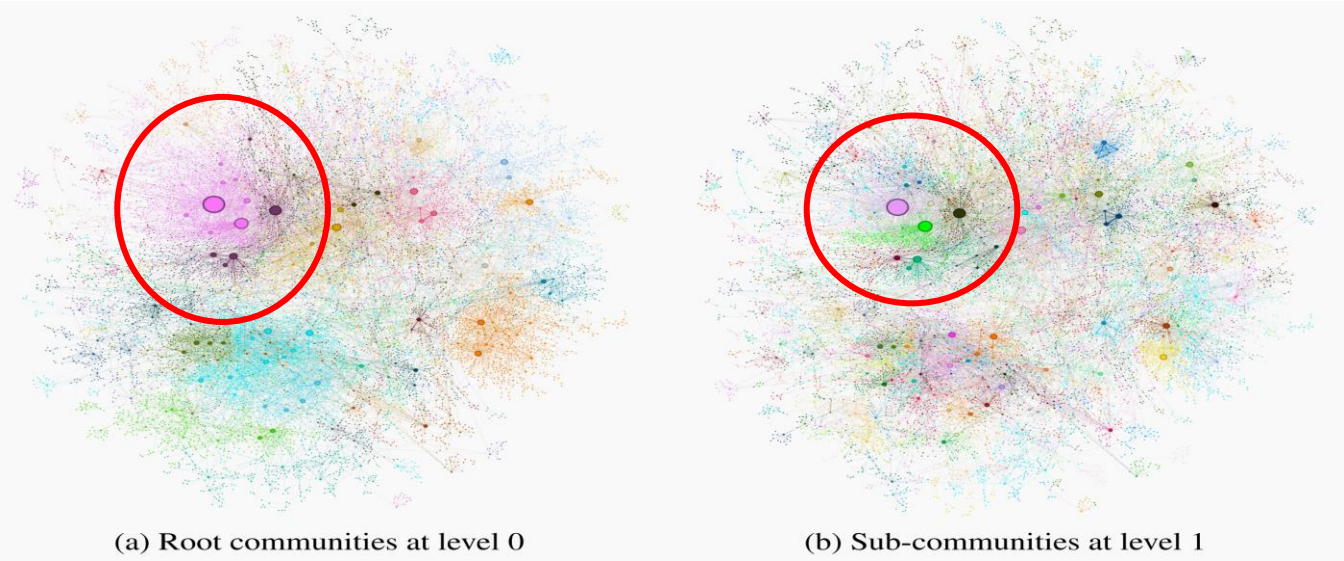
基础RAG的缺点	GraphRAG的优势
上下文窗口信息量有限，内容不完整	基于知识关联，可以有效拓展信息
即使扩充上下文窗口，信息仍可能被噪声埋没	已抽取关键信息，避免噪声干扰
简单提供原始片段，需要LLM对文本进行处理	对信息进行结构化表达，直观、易得、可解释

GraphRAG

- **关于GraphRAG的几个问题**

- （2）为什么要进行社区划分?
  - 可以在不同的信息粒度上，对原始数据进行语义聚合和层次化分析
  - 帮助LLMs在有限的上下文窗口中把握重点内容，避免冗余噪声

Level 0
较粗粒度

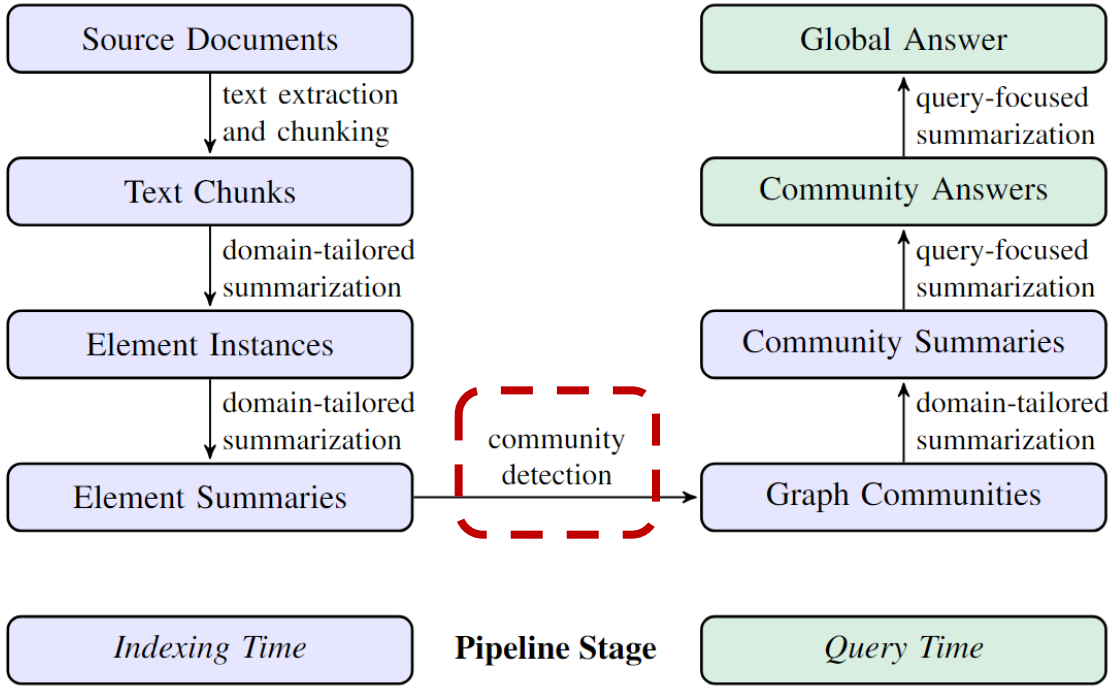


Level 1
较细粒度

GraphRAG

- **GraphRAG流程**

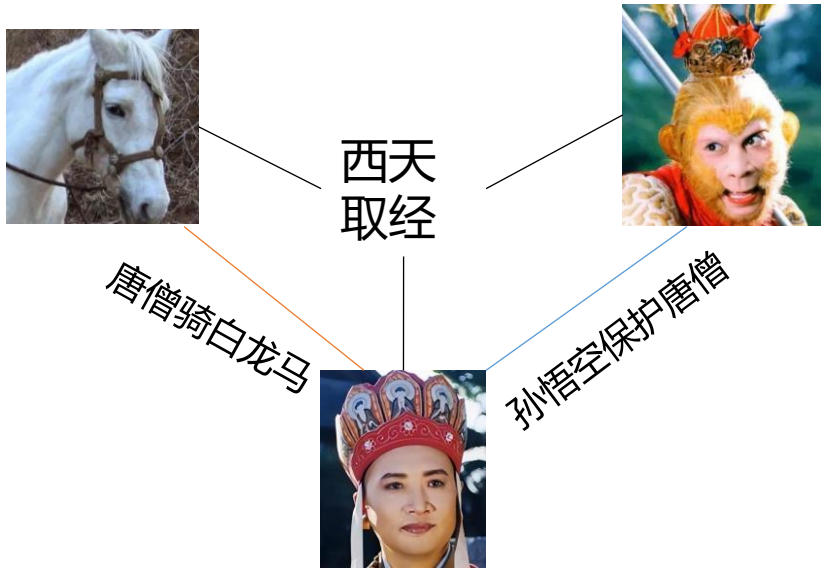
- 与基础RAG类似，GraphRAG同样采用“索引-查询”框架，但多了一步“划分”
  - 索引（Indexing）：面向私域文本构建知识图谱
  - 划分（Community Detection）基于图聚类划分社团，并为社区生成相应摘要以便查询
  - 查询（Query）：用户提出查询时，每个社团生成一个部分回答，然后将所有回答总结为最终回答



GraphRAG

- **GraphRAG流程（1）索引阶段**

- 我们以西游记为例，看看GraphRAG在每个阶段会做些什么
  - 在索引阶段，GraphRAG的主要目的是从文本中抽取关键信息
    - ✓ 获取结构化知识（实体、关系）
    - ✓ 形成实体/关系的文本描述
    - ✓ 多次迭代以确保信息完整性



局部知识图谱实例

GraphRAG

- **GraphRAG流程（2）社团划分阶段**

- 我们以西游记为例，看看GraphRAG在每个阶段会做些什么
  - 在社团划分阶段，GraphRAG首先需要获得图谱社区索引
    - ✓ 通常情况下，采用图聚类算法（如Leiden算法）获取社区
    - ✓ 在获取社区的同时，可以估计出每个节点的重要性（主角/配角）

角色社区

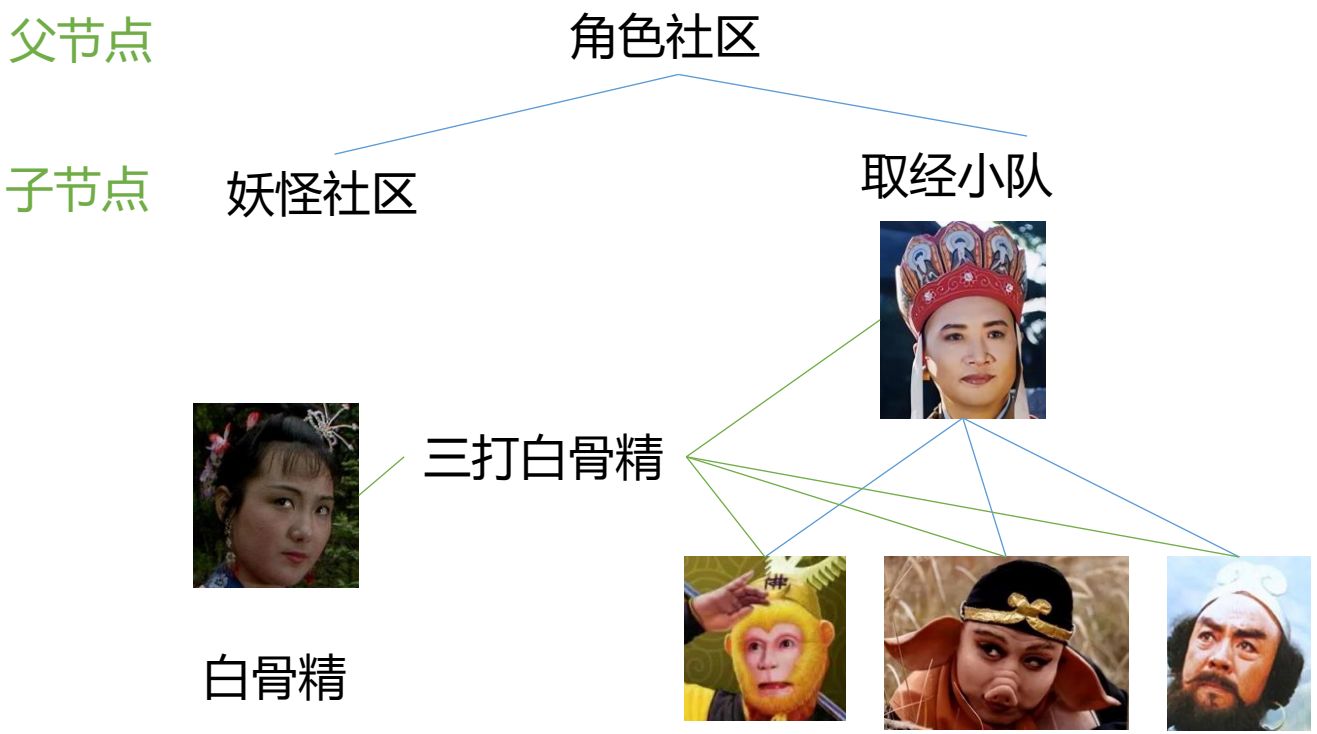
事件社区

三打白骨精
大闹天宫
真假美猴王
西天取经

GraphRAG

- **GraphRAG流程（2）社团划分阶段**

- 我们以西游记为例，看看GraphRAG在每个阶段会做些什么
  - 在图聚类过程中，社区之间天然地形成了层次关系，并对部分社区进行了细分
    - 例如，取经小队和白骨精都属于角色社区，但是他们又分属不同的子社团



- **GraphRAG流程（2）社团划分阶段**

- 我们以西游记为例，看看GraphRAG在每个阶段会做些什么
  - 最后，为不同层次的社区形成相应的摘要，以便于后续检索操作 →

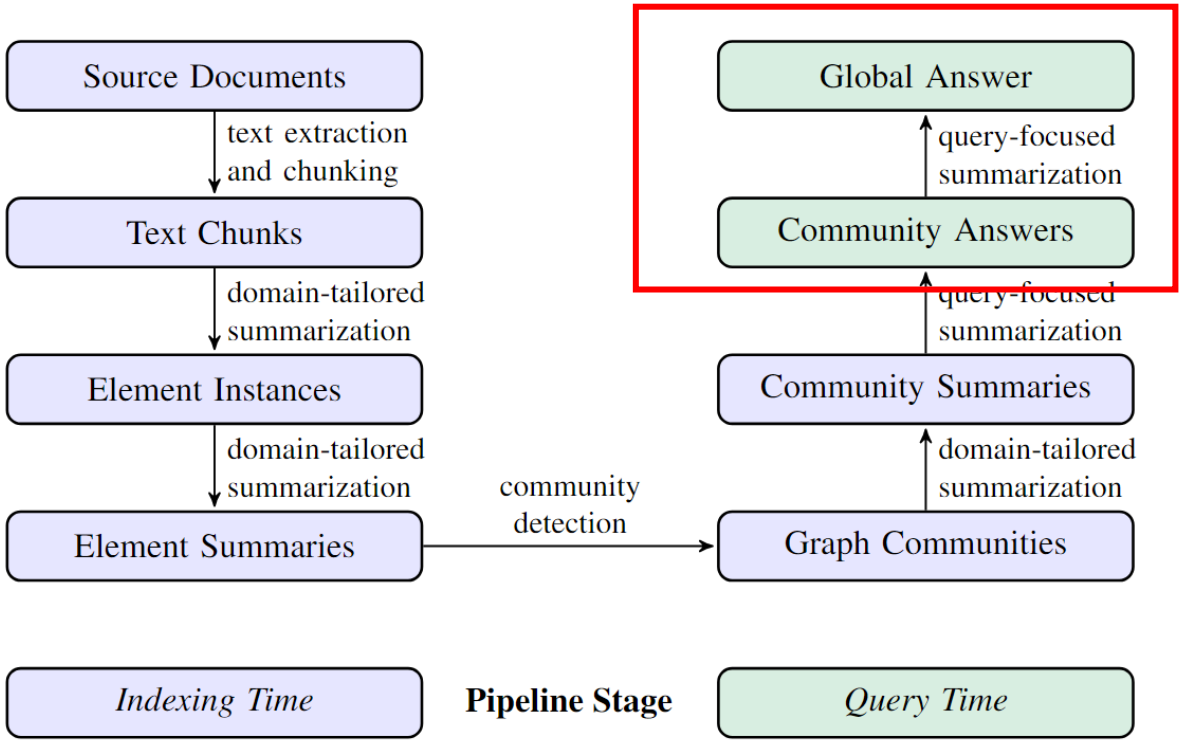
例如：

- 角色社区摘要：孙悟空是《西游记》中的主要角色之一，拥有强大的法力和聪明才智。唐僧是西天取经的领导者，猪八戒和沙僧是他们的徒弟
- 事件社区摘要：西天取经是《西游记》中的主要情节，讲述了唐僧和他的徒弟们去西天取得真经的故事。三打白骨精和大闹天宫是其中的重要事件

GraphRAG

- **GraphRAG流程（3）检索阶段**

- 我们以西游记为例，看看GraphRAG在每个阶段会做些什么
  - 在检索阶段，GraphRAG将查询/问题并行提交各个社区，各个社区生成相应答案
    - 在生成答案的同时，根据社区摘要衡量答案的相关性并排序
    - 如果相关性低甚至为0，则抛弃
  - 最后，总结生成全局答案

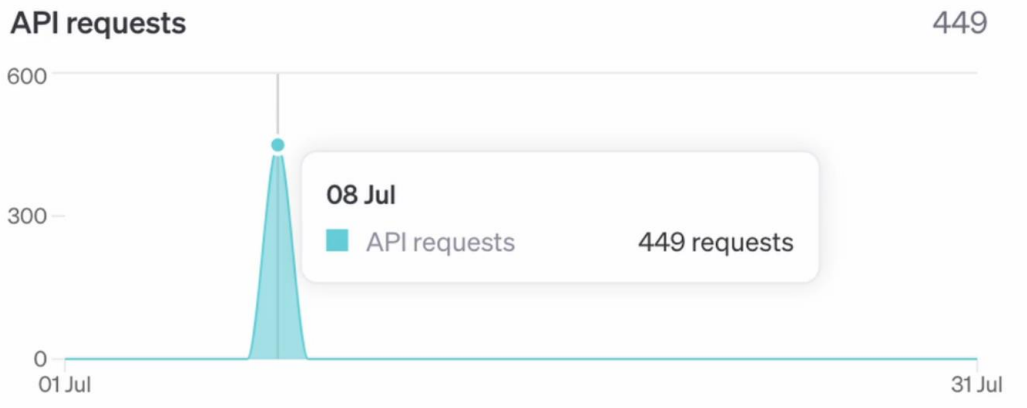


GraphRAG

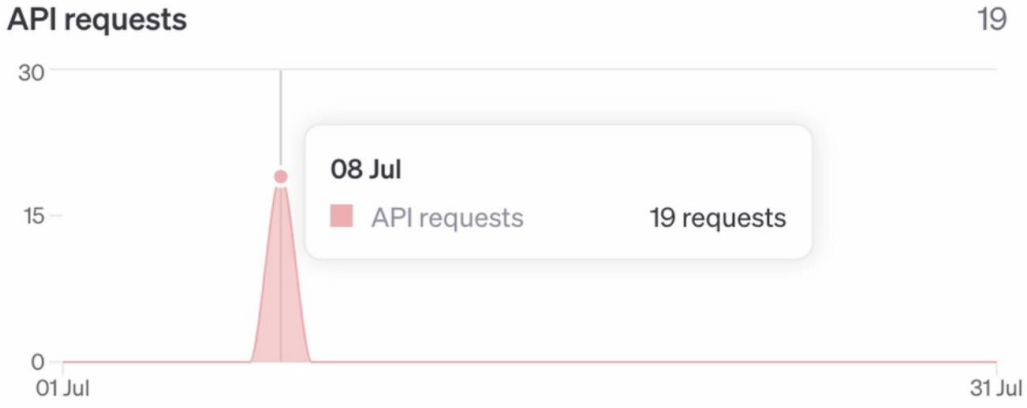
- **GraphRAG的局限性：低速 & 冗余计算**

- 以查尔斯·狄更斯的《圣诞颂歌》为测试文档，使用的大模型为GTP-4 Turbo
  - 分别利用GraphRAG和Naïve RAG创建索引以及提出两个问题

GraphRAG：449次API，共660s



Naïve RAG：19次API，共7s



  - GraphRAG在构建知识图谱的时候需要花费大量的时间
  - 对于某些简单提问时，并不需要构建完整的知识图谱，这会造成大量的冗余计算

数据来源：GraphRAG：很强，但贵得离谱

GraphRAG

- **GraphRAG的局限性：高昂的成本**

- 以查尔斯·狄更斯的《圣诞颂歌》为测试文档，使用的大模型为GTP-4 Turbo
  - 分别利用GraphRAG和Naïve RAG创建索引以及提出两个问题

GraphRAG花费11.29美元

Naïve RAG花费0.02美元

  - GraphRAG创建索引成本为9.56美元，平均每个问题1美元，相当于一个问题一杯瑞幸

数据来源：GraphRAG：很强，但贵得离谱

GraphRAG

• GraphRAG的局限性：精度-效率的平衡

- GraphRAG在回答问题时，需要人为指定在图谱中查询哪一个层级的社区
- 如何确定合适的社区层级，以平衡大模型生成答案的细节性、概括性，以及回答的质量和资源消耗，也是需要考虑的问题

C0~C3是在全局询问阶段
用于生成中间答案的社区层级

Podcast transcripts
播客记录

	SS	TS	C0	C1	C2	C3
SS	50	17	28	25	22	21
TS	83	50	50	48	43	44
C0	72	50	50	53	50	49
C1	75	52	47	50	52	50
C2	78	57	50	48	50	52
C3	79	56	51	50	48	50

Comprehensiveness
综合性

	SS	TS	C0	C1	C2	C3
SS	50	18	23	25	19	19
TS	82	50	50	50	43	46
C0	77	50	50	50	46	44
C1	75	50	50	50	44	45
C2	81	57	54	56	50	48
C3	81	54	56	55	52	50

Diversity
多样性

	SS	TS	C0	C1	C2	C3
SS	50	42	57	52	49	51
TS	58	50	59	55	52	51
C0	43	41	50	49	47	48
C1	48	45	51	50	49	50
C2	51	48	53	51	50	51
C3	49	49	52	50	49	50

Empowerment
帮助性

	SS	TS	C0	C1	C2	C3
SS	50	56	65	60	60	60
TS	44	50	55	52	51	52
C0	35	45	50	47	48	48
C1	40	48	53	50	50	50
C2	40	49	52	50	50	50
C3	40	48	52	50	50	50

Directness

News articles
新闻文章

	SS	TS	C0	C1	C2	C3
SS	50	20	28	25	21	21
TS	80	50	44	41	38	36
C0	72	56	50	52	54	52
C1	75	59	48	50	58	55
C2	79	62	46	42	50	59
C3	79	64	48	45	41	50

Comprehensiveness

	SS	TS	C0	C1	C2	C3
SS	50	33	38	35	29	31
TS	67	50	53	45	44	40
C0	62	47	50	40	41	41
C1	65	55	60	50	50	50
C2	71	56	59	50	50	51
C3	69	60	59	50	49	50

Diversity

	SS	TS	C0	C1	C2	C3
SS	50	47	57	49	50	50
TS	53	50	58	50	50	48
C0	43	42	50	42	45	44
C1	51	50	58	50	52	51
C2	50	50	55	48	50	50
C3	50	52	56	49	50	50

Empowerment
帮助性

	SS	TS	C0	C1	C2	C3
SS	50	54	59	55	55	54
TS	46	50	55	53	52	52
C0	41	45	50	48	48	47
C1	45	47	52	50	49	49
C2	45	48	52	51	50	49
C3	46	48	53	51	51	50

Directness

- **GraphRAG的升级版本**

- GraphRAG并不适用于所有RAG场景，与基本RAG框架的结合才是更优解
  - 2024年11月，微软再次推出GraphRAG的升级版本——LazyGraphRAG
  - 在面对局部的、侧重事实型的问题时，LazyGraphRAG仅采用传统RAG框架进行文本生成
  - 只有在涉及全局查询与理解请求时，才使用GraphRAG
  - **成本最多可降至1/1000！**

GraphRAG

- **如果想实践一下GraphRAG……**

- GraphRAG （Microsoft）
  - 很快将包括LazyGraphRAG版本
  - https://github.com/microsoft/graphrag
- Neo4j+GraphRAG
  - https://github.com/neo4j/neo4j
- OpenSPG
  - https://github.com/OpenSPG

Overview

The GraphRAG project is a data pipeline and transformation suite that is designed to extract meaningful, structured data from unstructured text using the power of LLMs.

To learn more about GraphRAG and how it can be used to enhance your LLMs ability to reason about your private data, please visit the Microsoft Research Blog Post.

OpenSPG is a knowledge graph engine developed by Ant Group in collaboration with OpenKG, based on the SPG (Semantic-enhanced Programmable Graph) framework, which is a summary of Ant Group's years of experience in constructing and applying diverse domain knowledge graphs in the financial scenarios.

本节目录

- 检索增强生成（RAG）
  - 什么是检索增强生成
  - 检索增强生成的范式
- RAG+KG：GraphRAG
- **大模型与知识图谱**

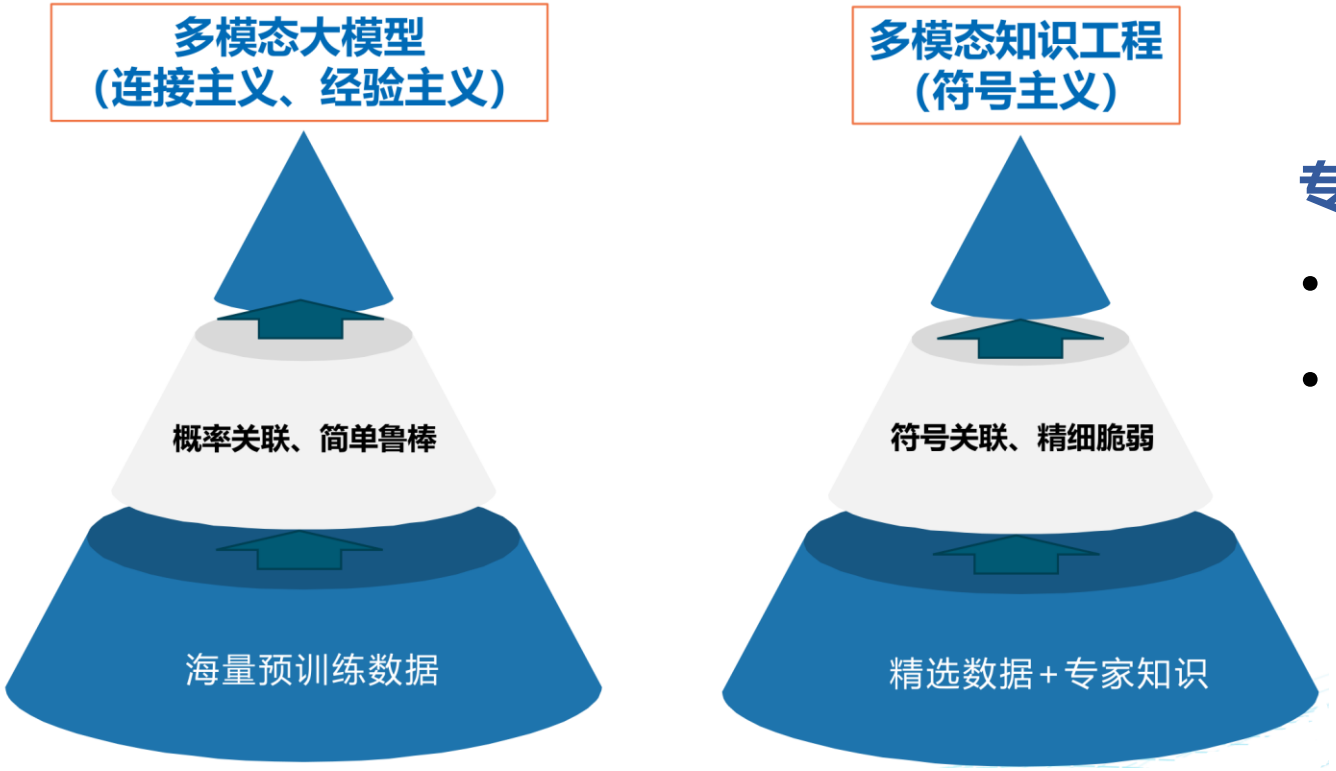
大模型+知识

- **知识图谱是否依然有其价值?**

- 由GraphRAG管窥一豹，在大模型时代，知识工程 / 知识图谱依然不可或缺

统计学习

- 端到端、干预少
- “数”尽其用



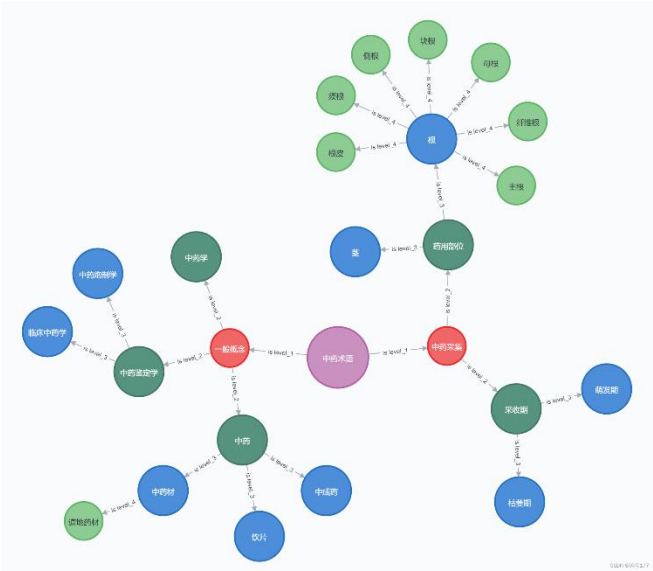
专家系统、知识图谱

- 易推理、可干预、可解释
- 信息损失

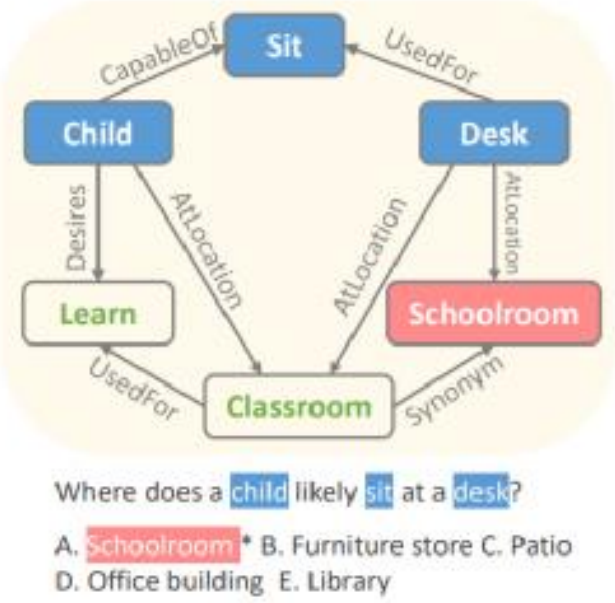
大模型+知识

- **LLM可以为知识图谱做些什么?**

- GraphRAG提供了从私域文本构建KG的傻瓜版本，但仍有诸多改进空间
  - 面临的问题：1）知识从何而来；2）如何更深入地利用



如何有效抽取与整理结构化知识?

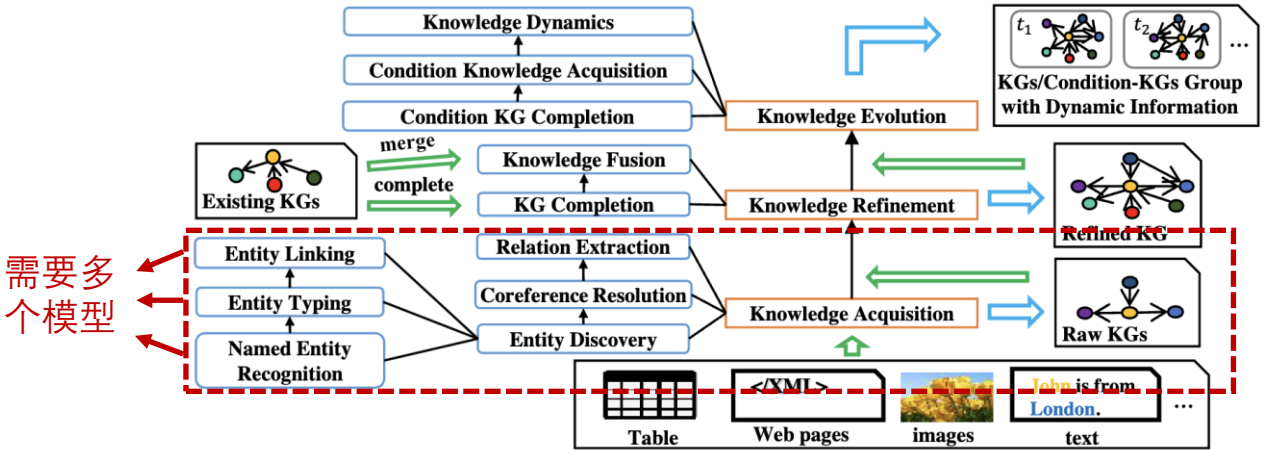


如何充分利用复杂结构的知识?

大模型+知识

大模型如何服务知识图谱

- 借助大语言模型强大的文本解析能力，可以从无到有地构建高质量图谱
- 知识图谱构建为什么可以，且需要使用大模型？
  - 传统方法依赖大量训练数据，并且很难统一多种任务
  - LLMs拥有丰富的内生知识，可以在多种任务上表现优异

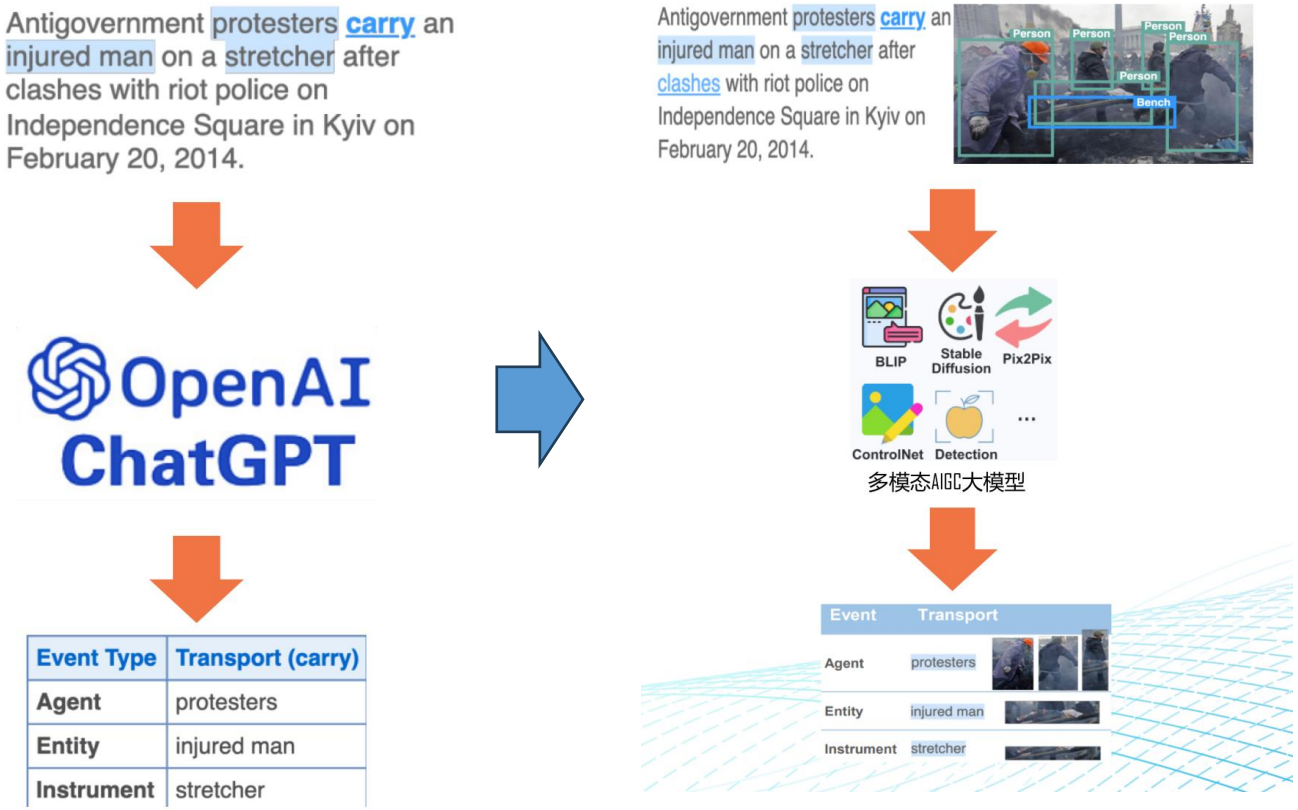


知识图谱构建的一般范式

大模型+知识

- **大模型如何服务知识图谱**

- 尤其面临跨模态知识学习任务时，大模型有“见多识广”的优势，泛化性更强

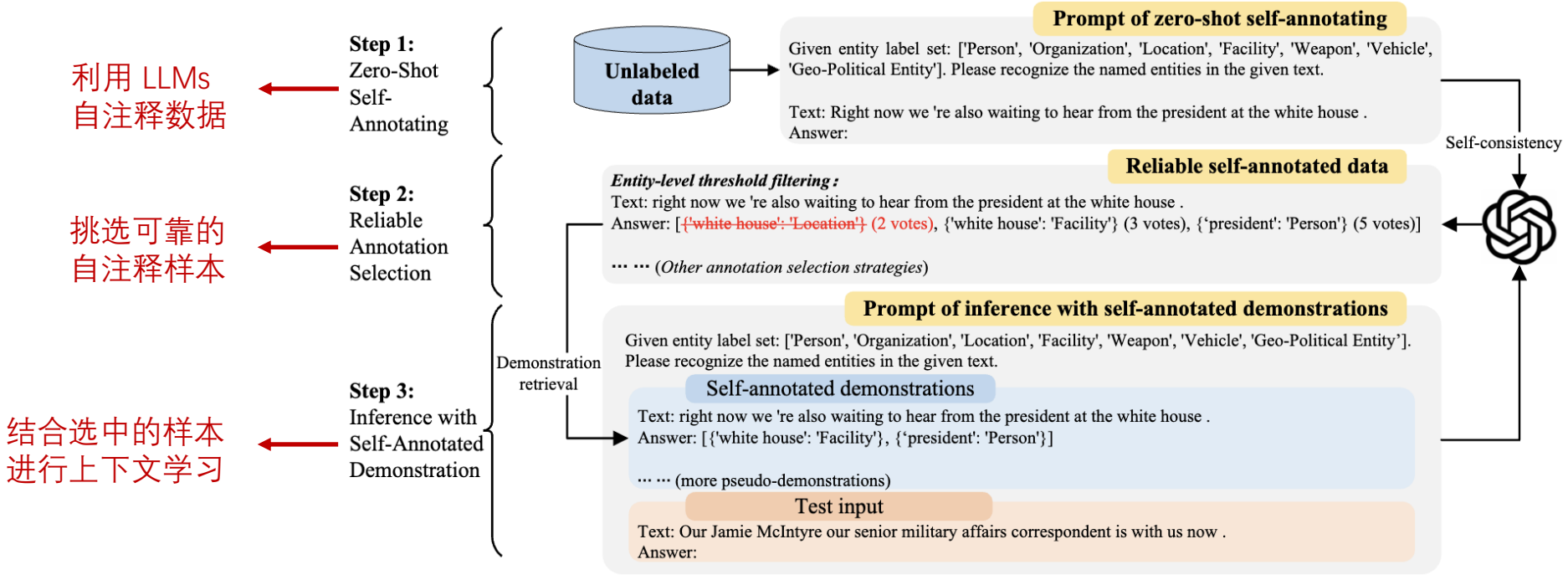


抽取文本知识 →
（三元组）

← 抽取多模态知识

大模型如何服务知识图谱

- 利用Prompt Engineering的思路，由大模型自主编写获取实体、关系的模版
  - 大模型举一反三的能力，使其能够很好实现知识的“零样本”学习

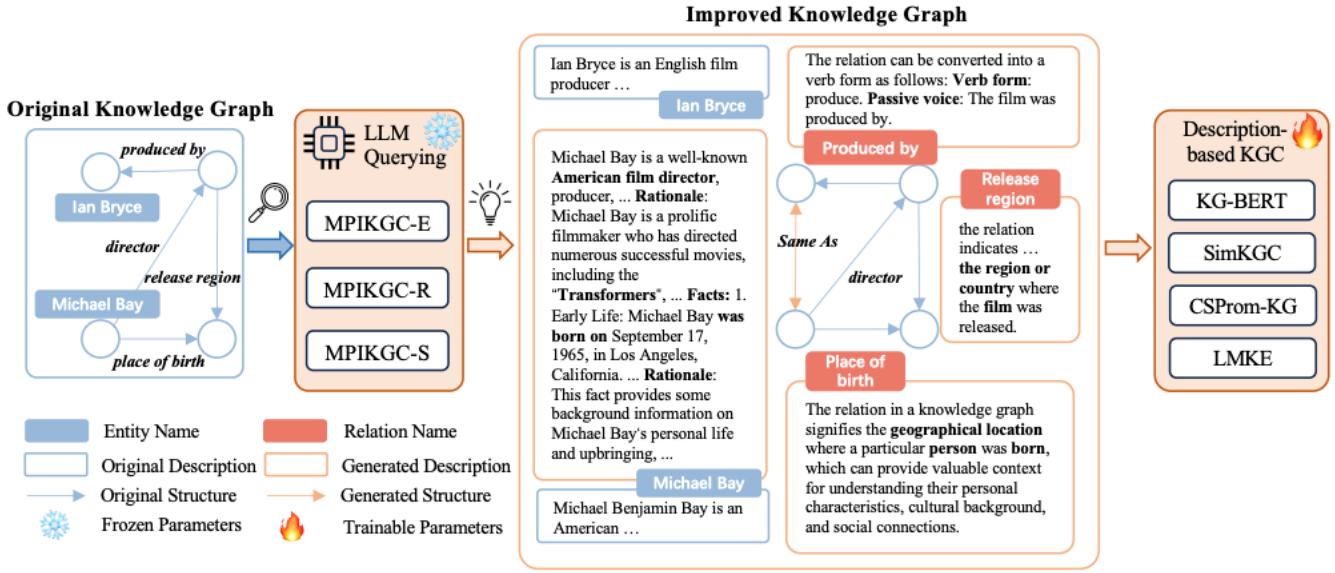


Self-Improving for Zero-Shot Named Entity Recognition with Large Language Models. NAACL 2024

大模型+知识

- **大模型如何服务知识图谱**

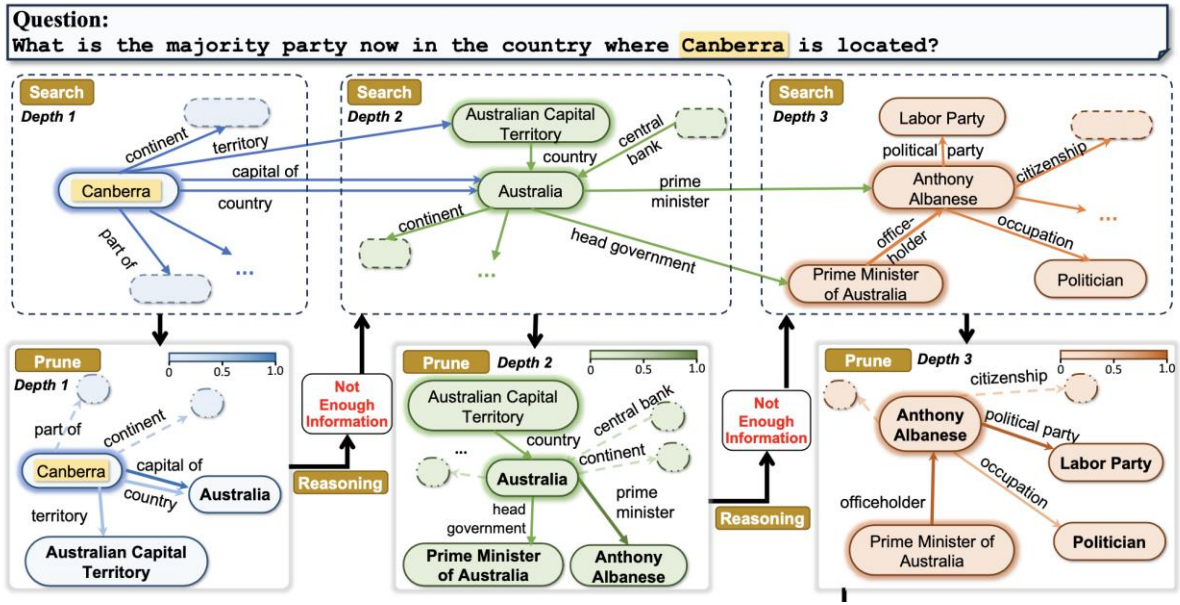
- 同时，大模型丰富的内生知识，还可以增强实体/关系的语义，提升抽取效果
  - 在不改变知识抽取模型本身的前提下，获得更准确、更完整的知识图谱
    - MPIKGC-E：利用LLMs的**推理**能力，**扩展实体描述**
    - MPIKGC-R：利用LLMs的**解释**能力，**理解关系含义**
    - MPIKGC-S：利用LLMs的**总结**能力，**增强结构信息**



Multi-perspective Improvement of Knowledge Graph Completion with Large Language Models, LREC-COLING 2024

大模型如何服务知识图谱

- 此外，大模型还可以辅助推理过程，提升推理效率与精度
  - 知识图谱路径推理的一大挑战，在于检索过程如何进行语义理解和路径选择
  - 利用大模型强大的语言理解能力，通过直接询问大模型实现路径搜索、裁剪、推理

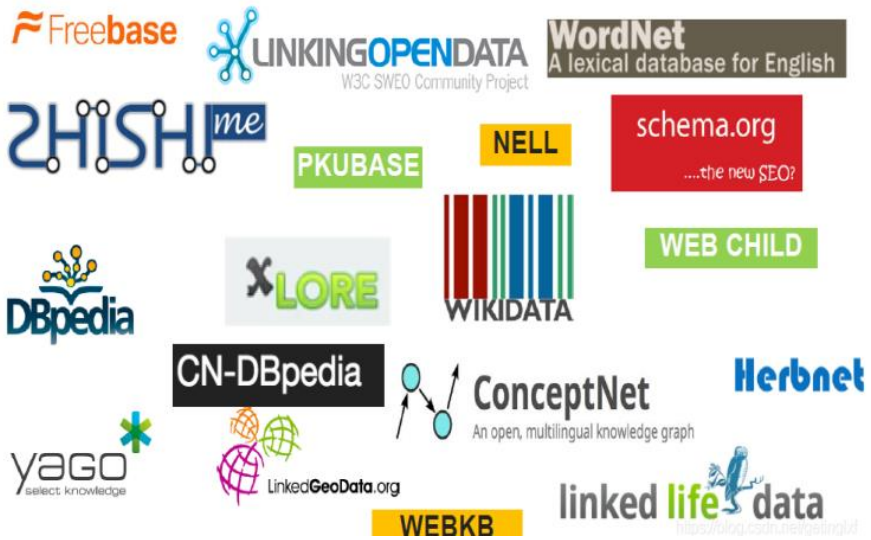


Think-on-Graph: Deep and Responsible Reasoning of Large Language Model on Knowledge Graph, ICLR 2024.

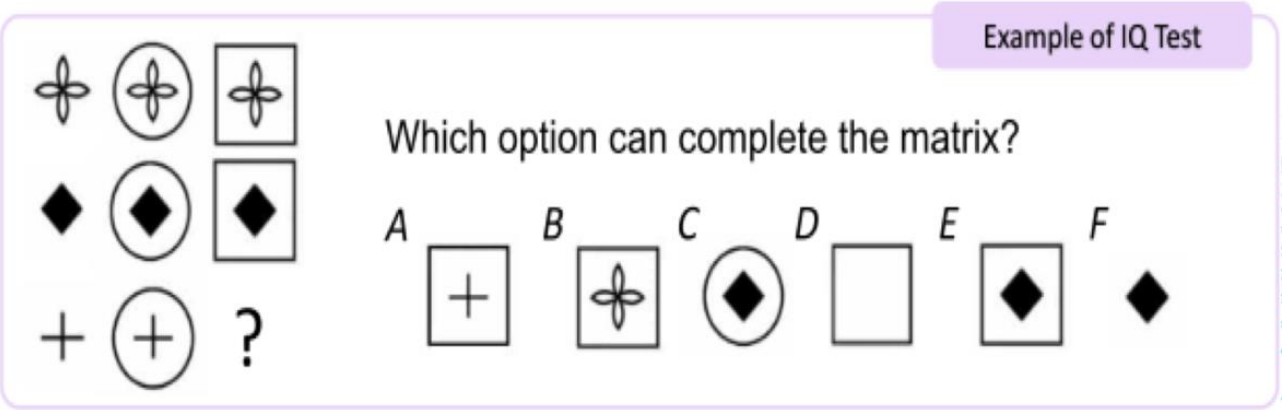
大模型+知识

- **知识图谱辅助大模型**

- 基于知识图谱中的知识构建测试集，可对大模型的能力进行各方面评估



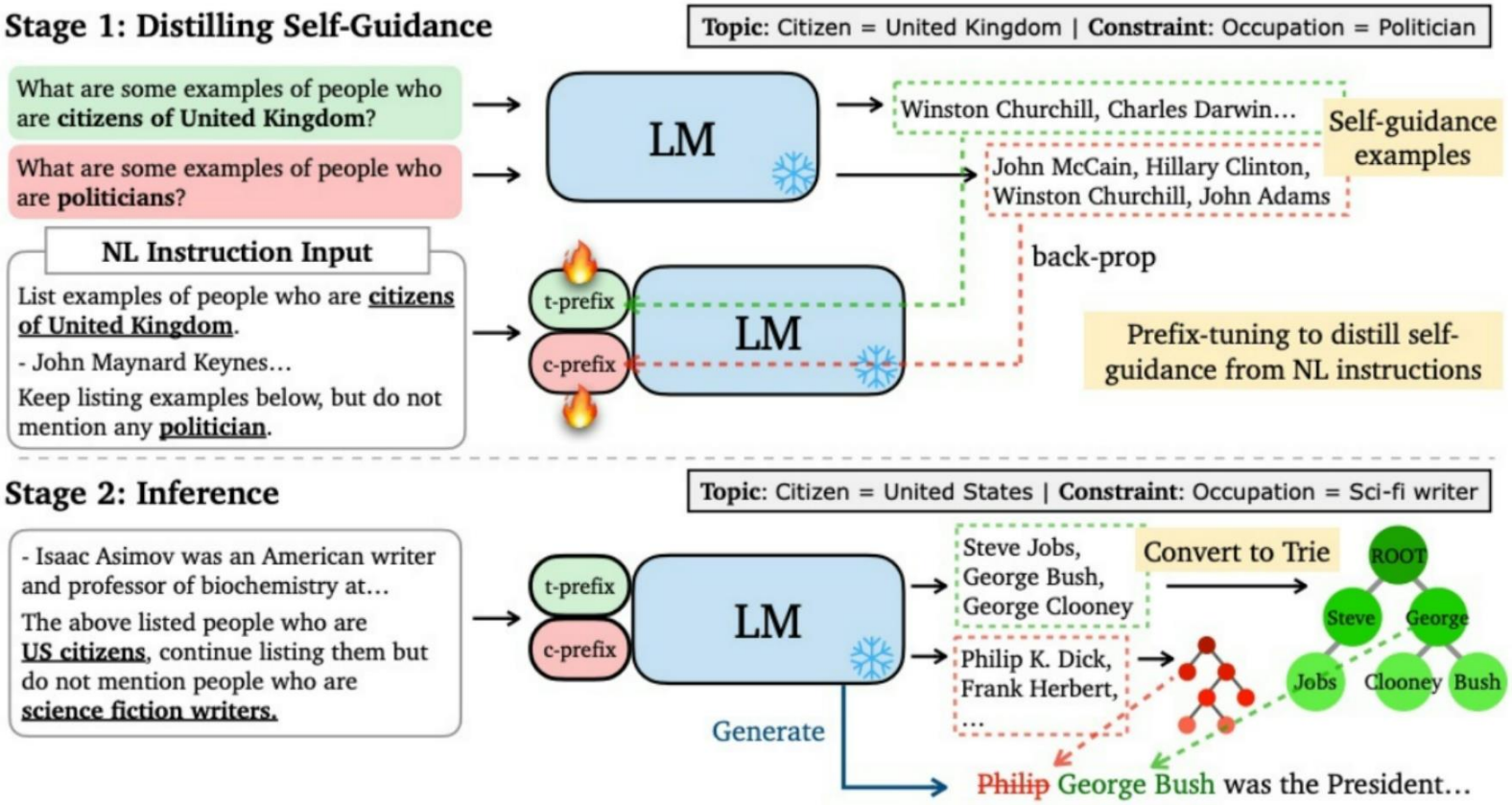
知识图谱用于事实性检验



大模型推理能力评估

- **知识图谱辅助大模型**

- 基于知识图谱约束，实现面向大模型的可控内容生成



大模型+知识

- **未来：大模型与知识图谱的双向奔赴**

- 知识工程与大模型技术优势互补，相辅相成，实现可靠、可控的AIGC

大模型技术

优势：见多识广、生成能力与泛化性强

缺陷：可解释性与可靠性差、逻辑性弱

知识图谱技术

优势：专业性强、可解释性好泛化性强

缺陷：泛化性弱、推广能力较为有限

本章小结

大语言模型与知识

- 检索增强生成（RAG）
  - 什么是检索增强生成
  - 检索增强生成的范式
- RAG+KG：GraphRAG
- 大模型与知识图谱

注：本节部分内容来自复旦大学魏忠钰、李直旭教授、同济大学王昊奋教授等同仁的相关报告，在此表示感谢