Agentic RAG时代：DeepSeek引领推理模型升级下的知识检索增强

王昊奋

同济大学特聘研究员、OpenKG 发起人之一

2025.04.18

目录

01 知识增强大模型概述

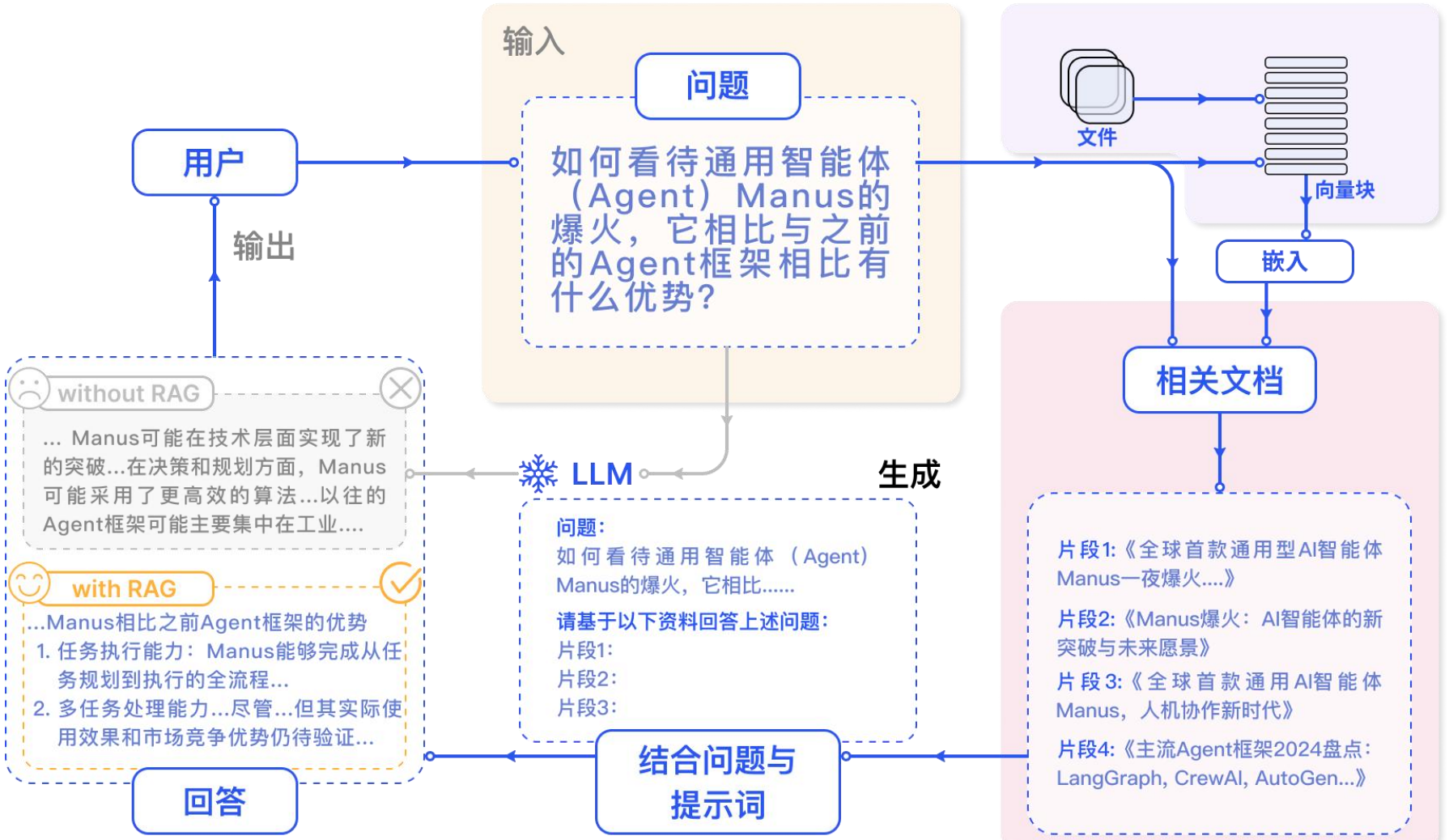
知识检索增强技术 (Retrieval-Augmented Generation, RAG)

LLM的缺陷

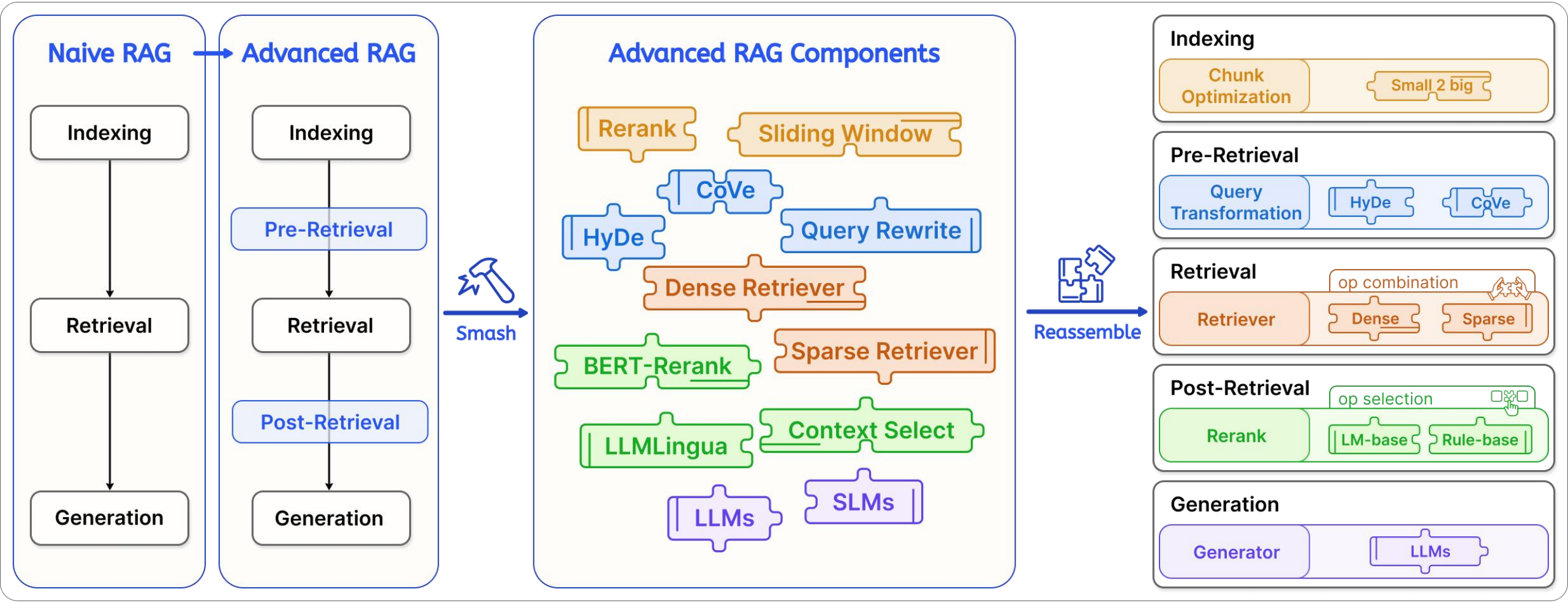
- 幻觉
- 信息过时
- 参数化知识效率低
- 缺乏专业领域的深度知识
- 推理能力弱

实际应用的需求

- 领域精准问答
- 数据频繁更新
- 生成内容可解释可溯源
- 成本可控
- 数据隐私保护



RAG 范式发展趋势


Naive RAG
Indexing
Retrieval
Generation
Advanced RAG
Indexing
Pre-Retrieval
Retrieval
Post-Retrieval
Generation
Smash
Advanced RAG Components
Rerank
Sliding Window
CoVe
HyDe
Query Rewrite
Dense Retriever
BERT-Rerank
Sparse Retriever
LLMLingua
Context Select
LLMs
SLMs
Reassemble
Indexing
Chunk Optimization
Small 2 big
Pre-Retrieval
Query Transformation
HyDe
CoVe
Retrieval
op combination
Retriever
Dense
Sparse
Post-Retrieval
op selection
Rerank
LM-base
Rule-base
Generation
Generator
LLMs

RAG 的典型范式（Naive RAG）

步骤1：构建数据索引：

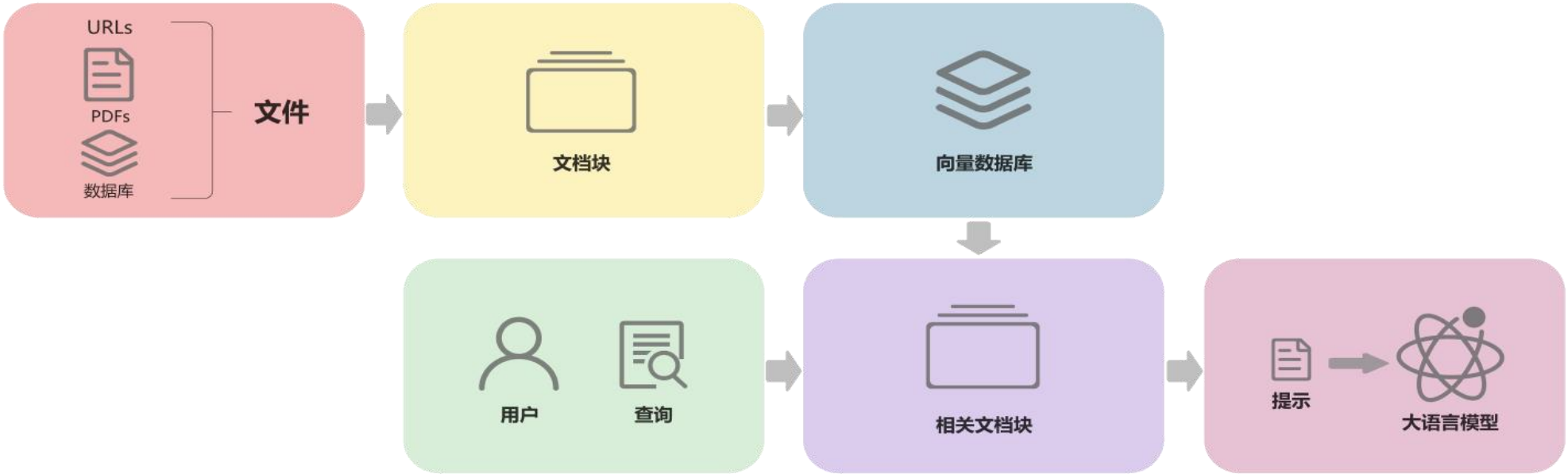
1. 将文档分割成均匀的块，每个块是一段原始文本
2. 利用编码模型为每个文本块生成Embedding
3. 将每个块的Embedding存储到向量数据库中

步骤2：检索

通过向量相似度检索和问题最相关的k个文档。

步骤3：生成

原始Query与检索得到的文本组合起来输入大语言模型，得到最终的回答



朴素RAG
Naive RAG

RAG进阶
Advanced RAG

模块化RAG
Modular RAG

RAG 的典型范式（Advanced RAG）

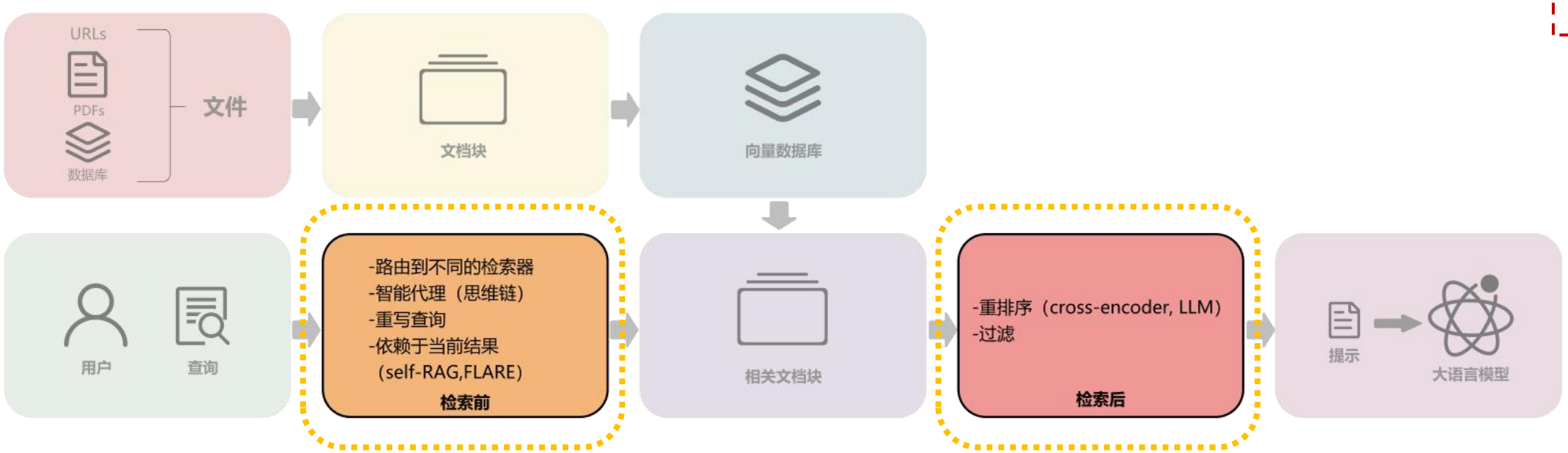
索引优化 → 前检索 → 检索 → 后检索 → 生成

- 索引优化：滑动窗口、细粒度分割、元数据
- 前检索模块：检索路由、摘要、重写、置信度判断
- 后检索模块：重排序、检索内容过滤

朴素RAG
Naive RAG

RAG进阶
Advanced RAG

模块化RAG
Modular RAG



RAG 的典型范式演化（Modular RAG）

Naive RAG

索引 → 检索 → 生成

Advanced RAG

索引 → 检索前优化 → 检索 → 检索后优化 → 生成

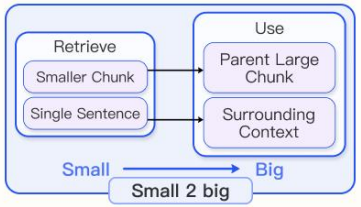
查询重写　假设文档嵌入

重排序　过滤

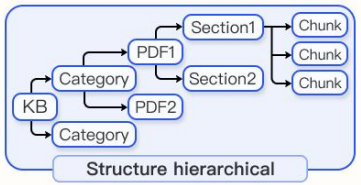
Modular RAG

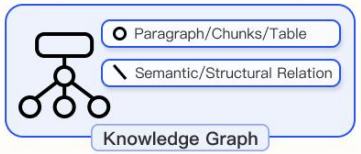
索引

向量块优化



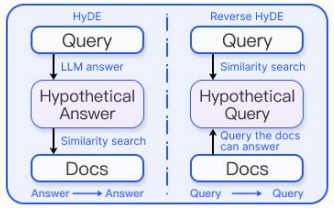
结构化组织



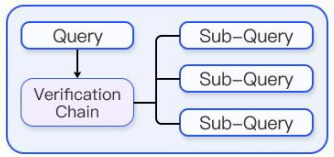


检索前优化

查询转换



查询扩展



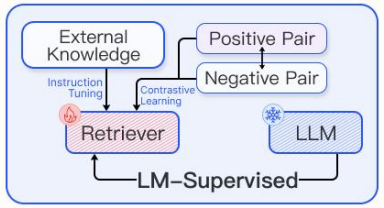
查询结构化

Text-to-Cypher

Text-to-SQL

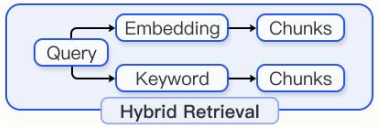
检索

检索器优化



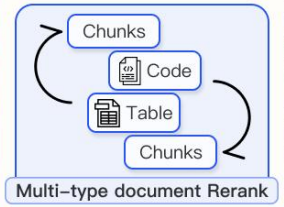
检索器源

检索器选择

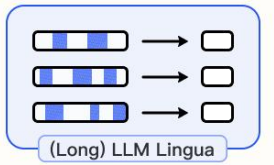


检索后优化

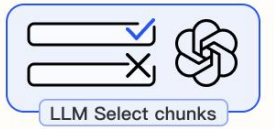
重排序



压缩

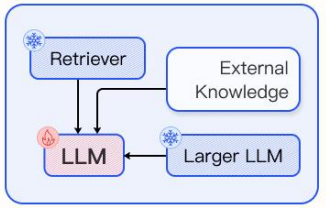


选择

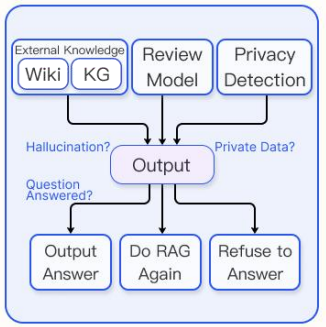


生成

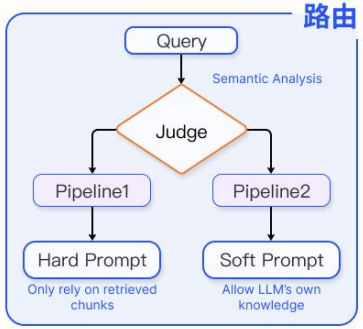
生成器微调



验证

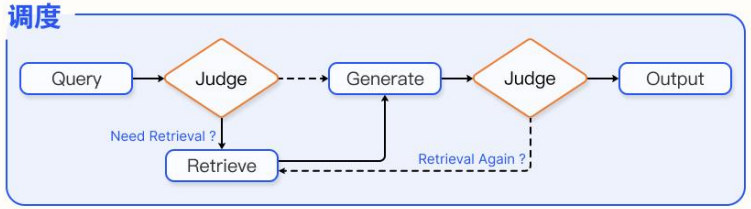


路由

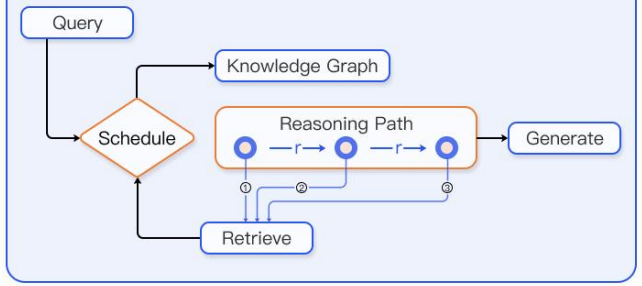


编排

调度

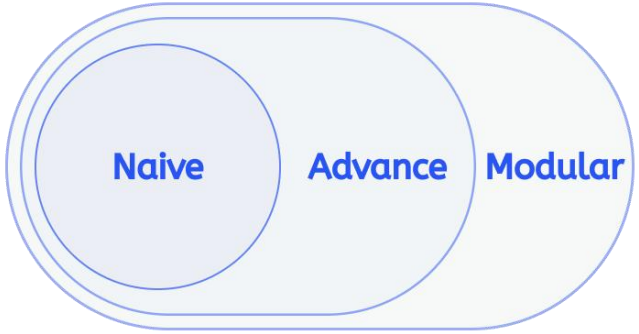


知识指南



Legend

- RAG中的关键模块
- 新模块
- 进程控制



RAG 的典型范式演化（Modular RAG）

Modular RAG

三层架构

Module

6 大主要模块：
RAG的核心流程

Sub-Module

14 个子模块：
流程中的具体功能

Operator

40+ 算子：
特定功能的具体实现

统一的RAG研究范式

对之前范式的继承与发展

RAG Flow

不同模块以及模块中不同算子的选择和编排构成了RAG Flow，从而识别出典型的RAG Flow 模式。

3 Tuning Stage

- Retriever FT
- Generator FT
- Dual FT

4 Inference Stage

- Sequential
- Conditional
- Brach
- Loop

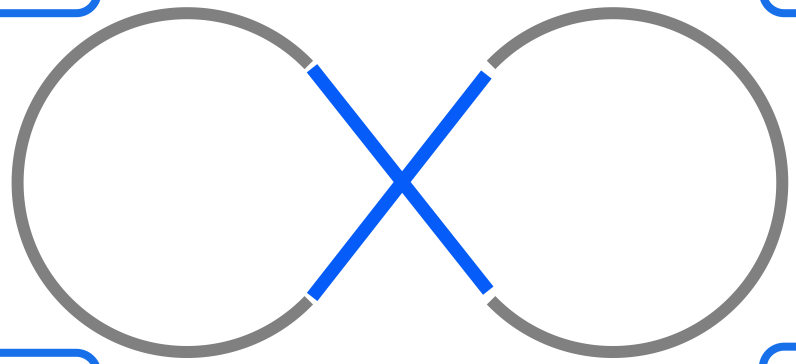
模块化RAG下的机遇

研究视角

新模块

新算子

新编排方式



场景适配

性能优化

应用扩展

应用视角

AI (RAG) 研究者

算法工程师

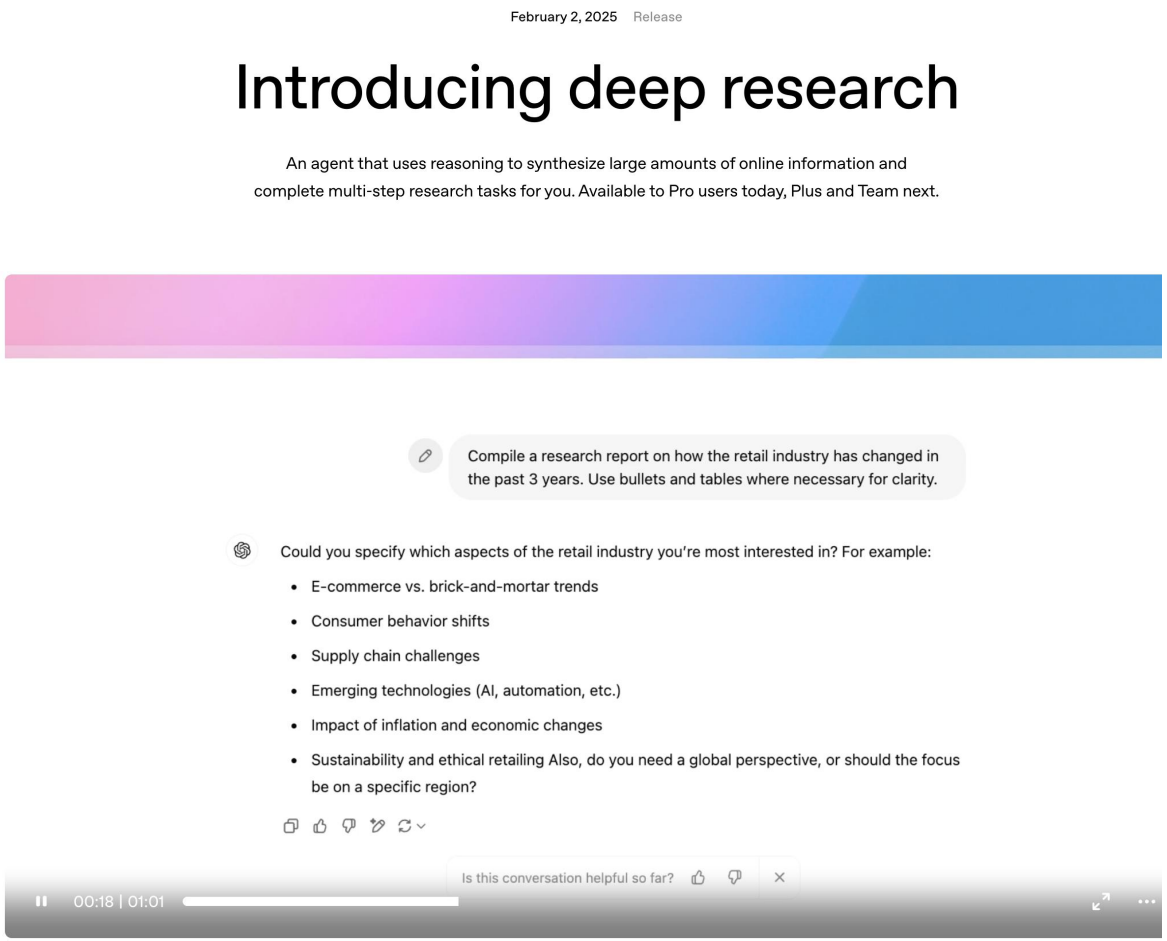
业务应用开发工程师

业务专家

02 RAG 新趋势

2-1. Agentic RAG 崛起 | 2-2. Graph与RAG 协同 | 2-3. 如何在RAG中注入推理能力

Agentic RAG 序幕拉开

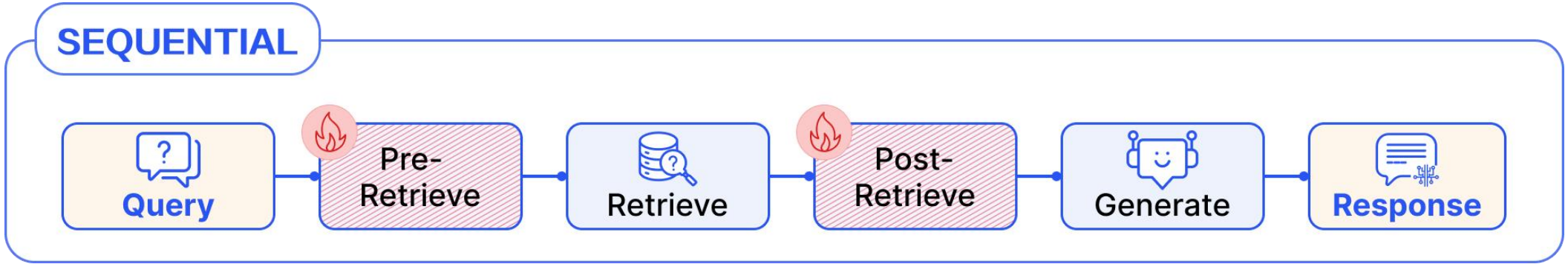


不同模块和和算子的编排就组成了RAG Flow，也拉开了Agentic RAG序幕。

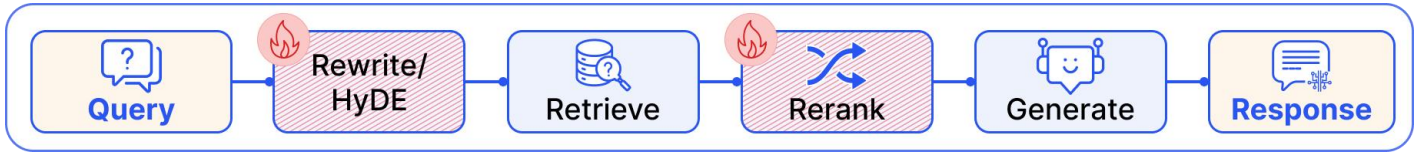
Sequential 线性 RAG Flow

Sequential Flow Pattern

各个模块线性的组织，经典的Naive RAG和基础的Advanced RAG均为线性结构。

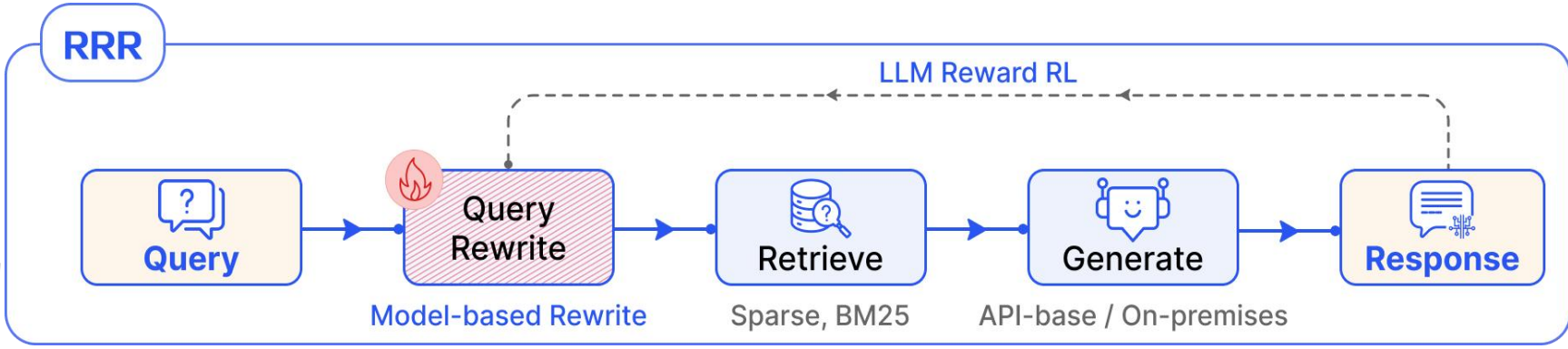


常在检索前增加Query Rewrite，在检索后增加Rerank算子提升整体效果。
例如QAnything



Rewrite-Retrieve-Read

- Query Rewrite一个小型的可训练LM
- 通过最终LLM的输出结果作为奖励
- 在强化学习的背景下，重写器优化被形式化为一个马尔科夫决策过程

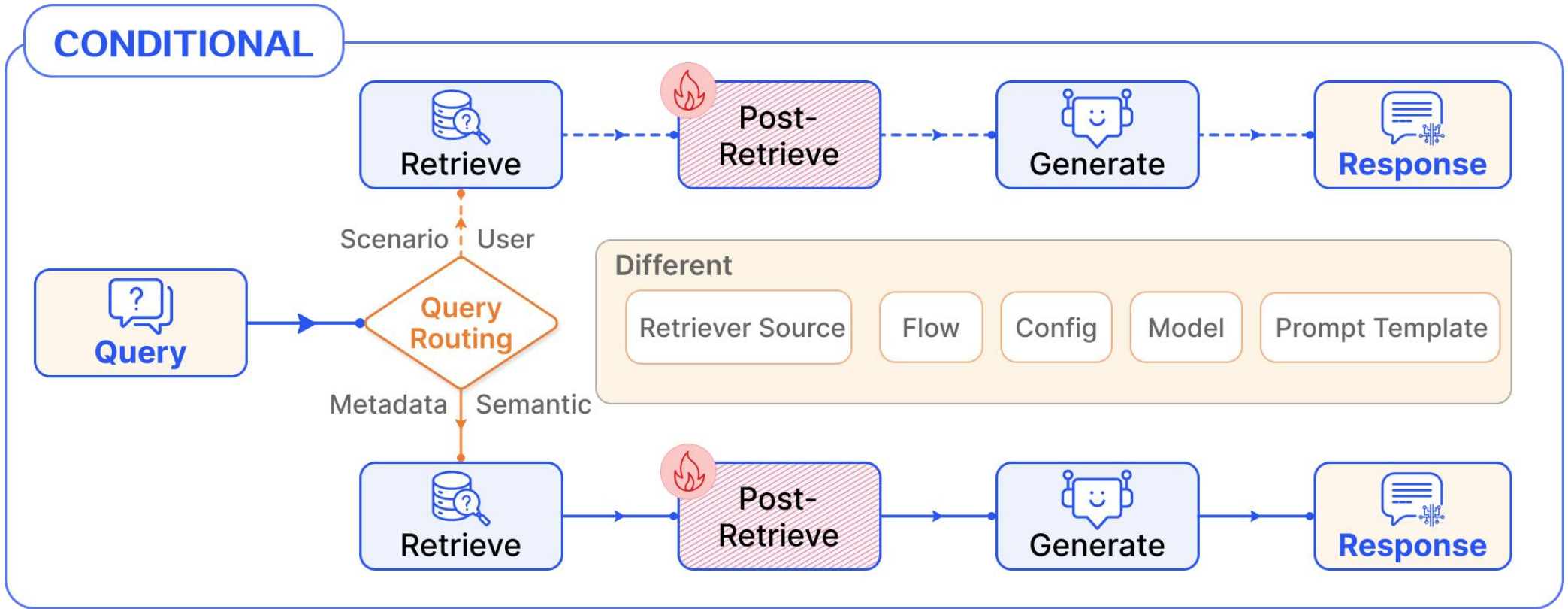


Conditional 条件 RAG Flow

Conditional Flow Pattern

根据不同的条件选择不同的RAG路线。通常由一个Routing模块进行路由。

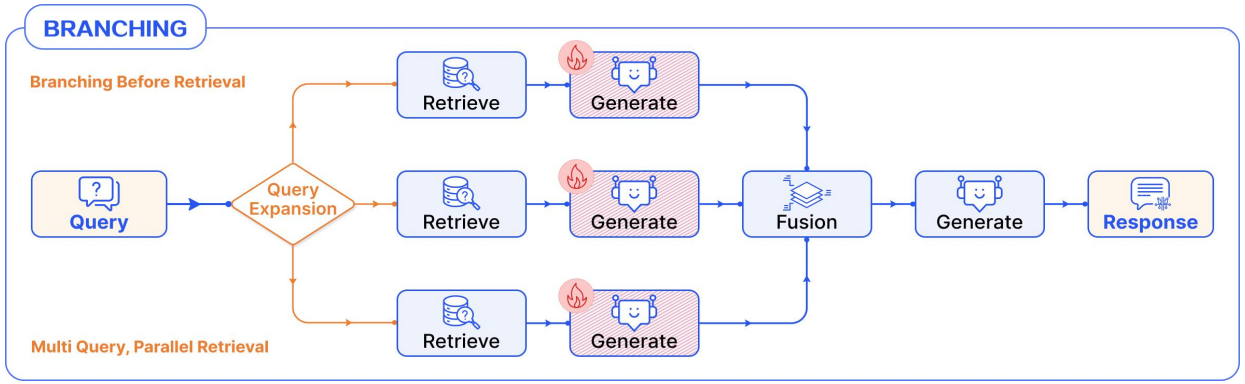
- 判断依据包括：语义、任务类型、元数据、用户权限。
- 不同路由分支在检索源、检索流程、配置信息、模型选择和Prompt Template上进行差异化。
- 严肃场景下的任务（例如 医疗、法律等）与娱乐问题，对大模型幻觉的容忍度是不同的。



Branching 分支结构 RAG Flow

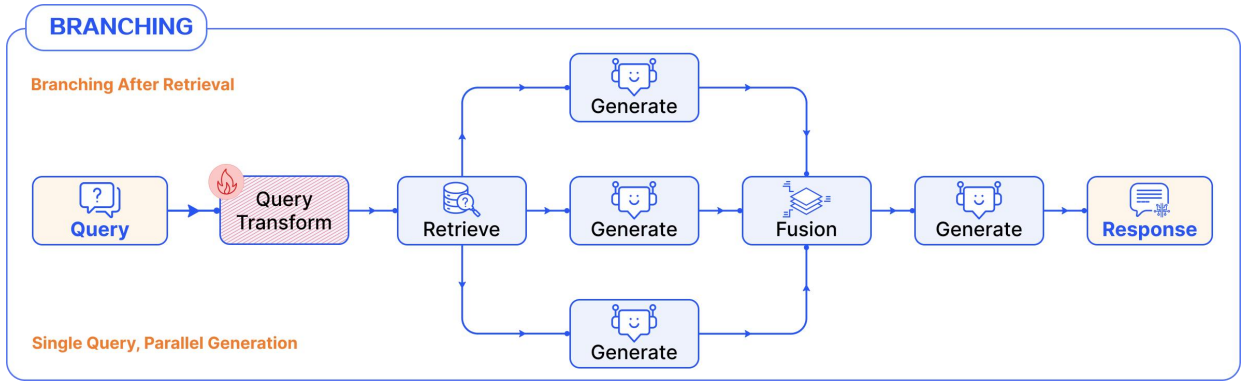
检索前分支

- 对原始Query进行扩展，得到多个Sub-Query。
- 分别对Sub-Query进行检索、生成
- 聚合所有得到的答案



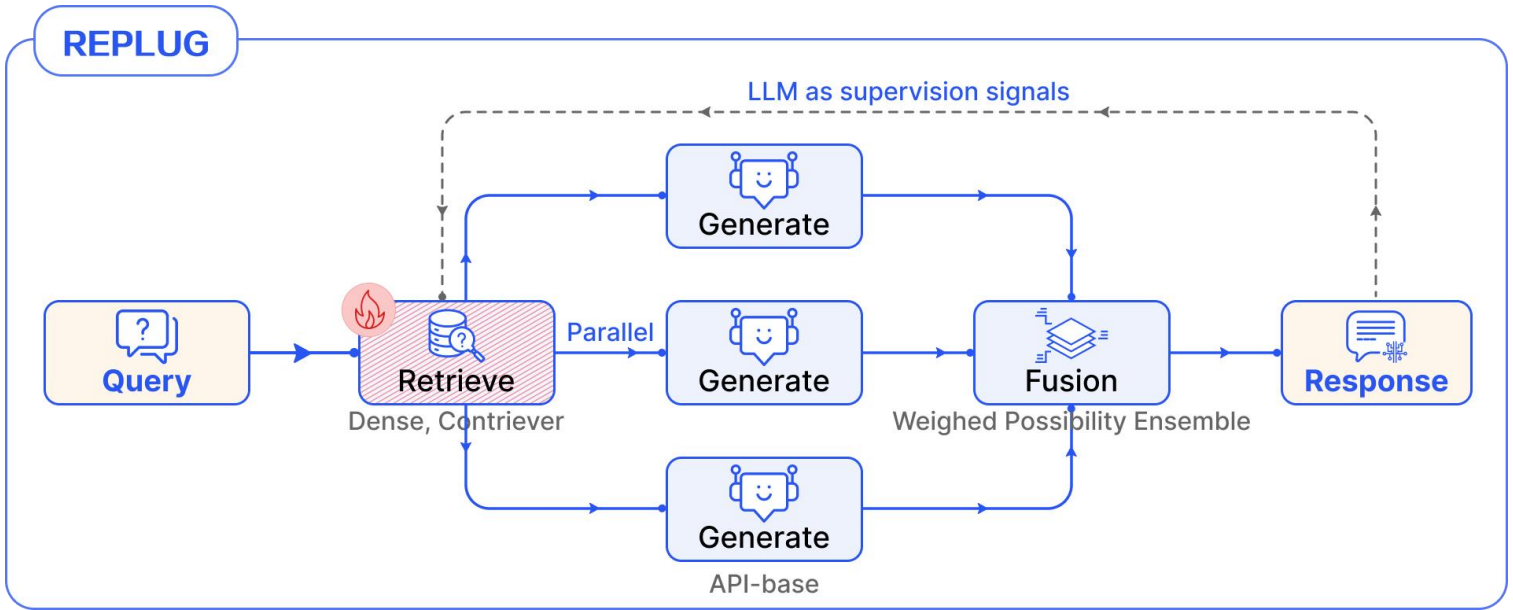
检索后分支

- 保持原来的Query，检索到多个文档块后，
- 并行使用原始Query和每一个文档块进行生成
- 对多个生成结果进行聚合



REPLUG

- 典型的检索后分支的分结构
- 根据每一个分支预测token的概率
- 通过Weighted possibility Ensemble 将不同的分支聚合。
- 并通过最后生成结果作为反馈微调检索器Contriever。



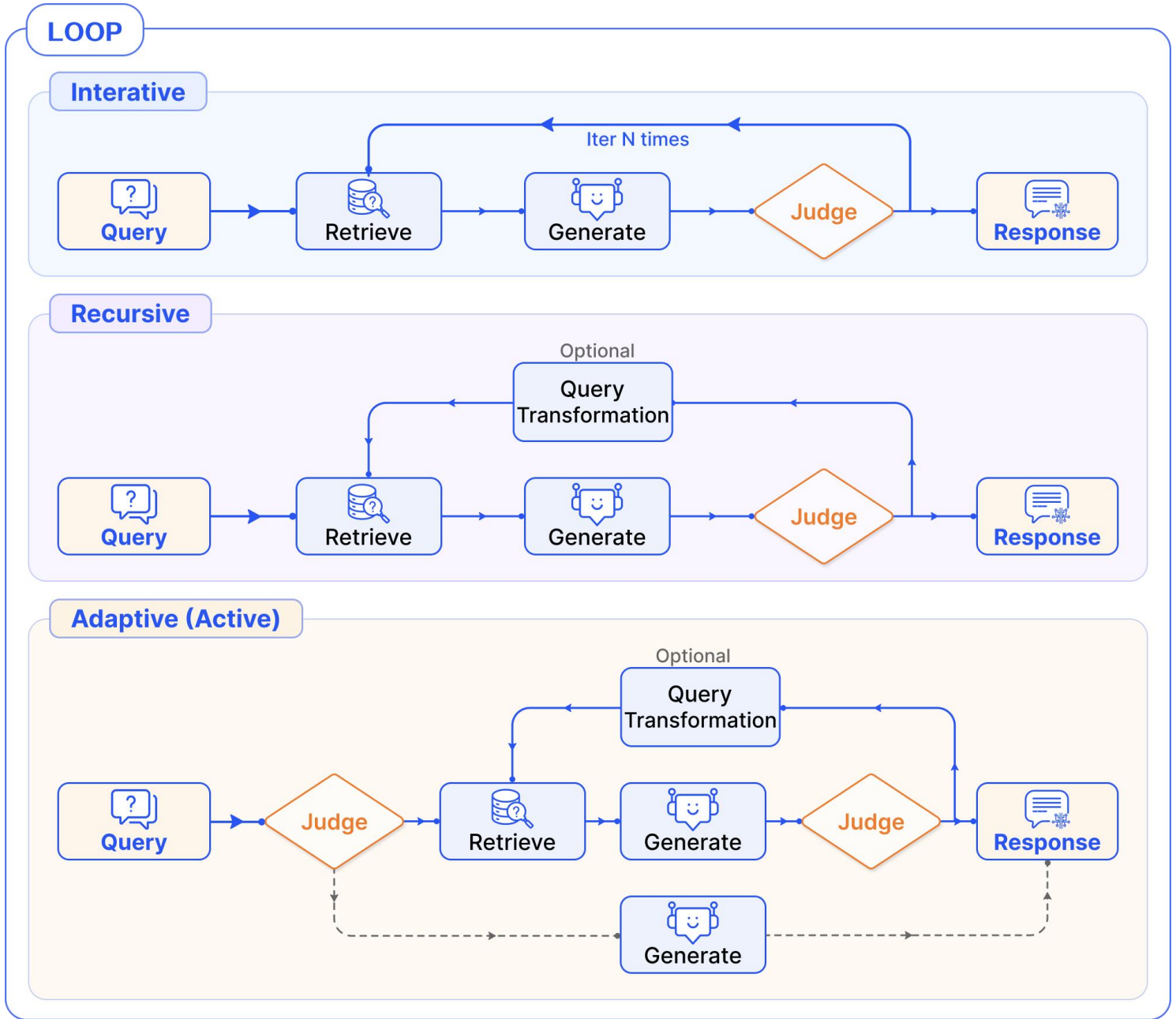
Loop 环状 RAG Flow

Loop Flow Pattern

- 面对复杂问题，一次的检索生成效果并不理想，通过多次检索增强
- 具有循环结构的RAG Flow，是Modular RAG的重要特点。
- 检索和推理步骤相互影响的。通常包括Judge模块，用于控制流程。

具体又可以分成:

- 迭代检索
- 递归检索
- 自适应（主动）检索

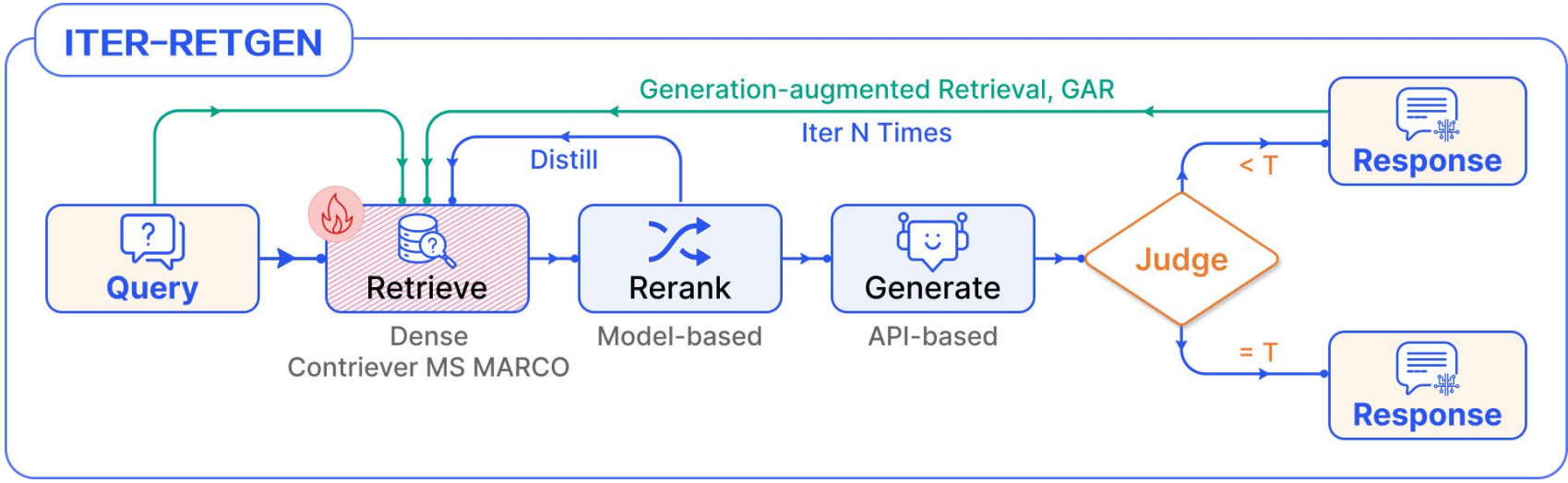


Iterative / Recursive Retrieval 迭代/递归检索

迭代检索

ITER-RETGEN

- 每次迭代，利用前一次迭代的模型输出帮助检索更相关的知识。
- 循序的终止通过预设的迭代次数来判断。

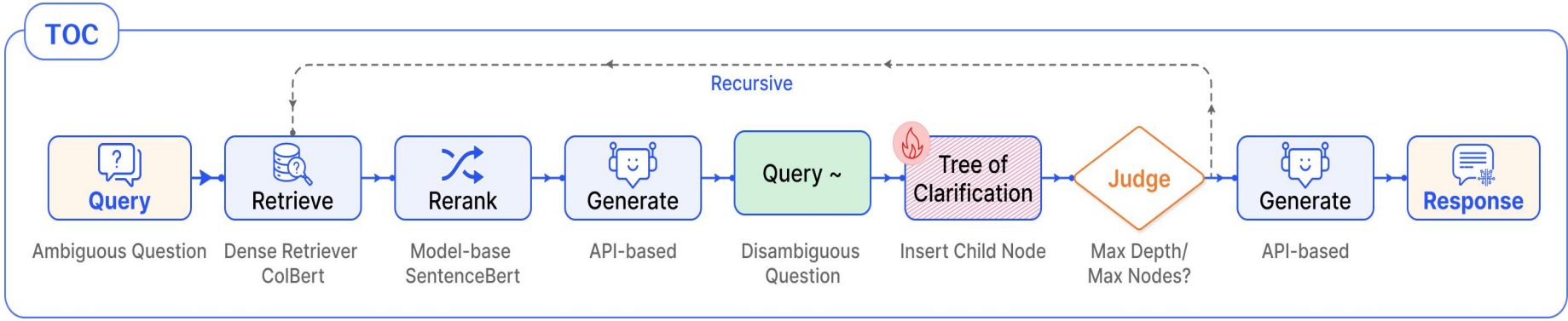


递归检索

不同于迭代检索，递归检索的特点是有明显依赖上一步并不断深入的检索。通常搭配Query Transformation，每次检索时依赖于新改写后的Query。

ToC

- 迭代根据查询重排序段落，并生成新的澄清问题。
- 插入子节点到澄清树中。
- 根据新的问题继续检索
- 树的探索在达到了最大有效节点或深度时结束。
- 收集树中所有节点生成长上下文回答。



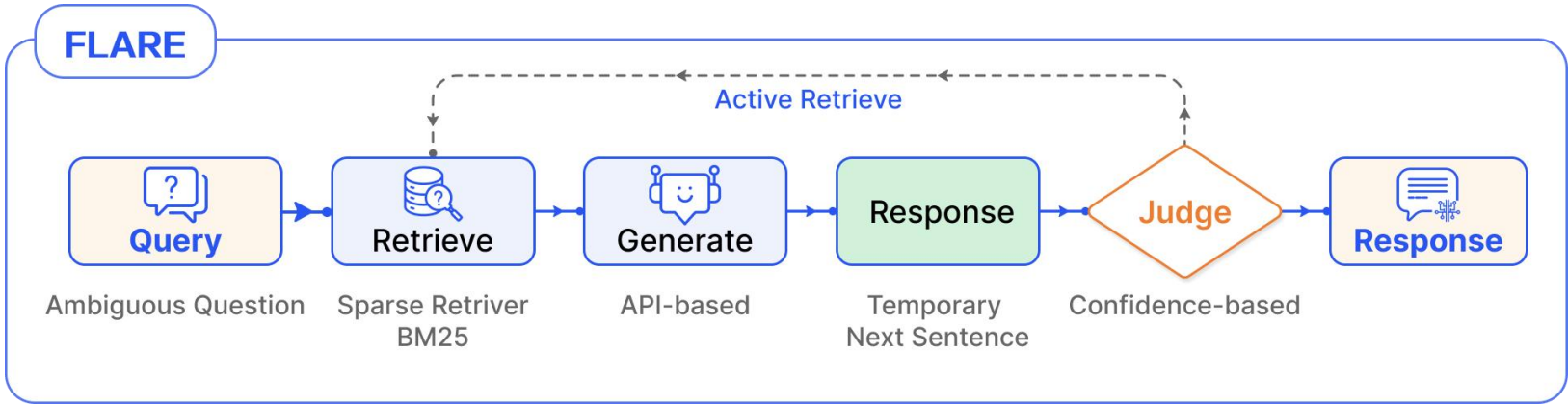
Adaptive (Active) Retrieval 自适应检索

Prompt-based

通过Prompt Engineering的方式让LLM对流程实现自主控制。

FLARE

- 应该仅在缺乏所需知识时进行检索，以避免被动检索增强中出现不必要或不适当的检索。
- 迭代地生成下一个临时句子，并检查是否包含低概率标记。
- 如果是，系统将检索相关文档并重新生成句子。

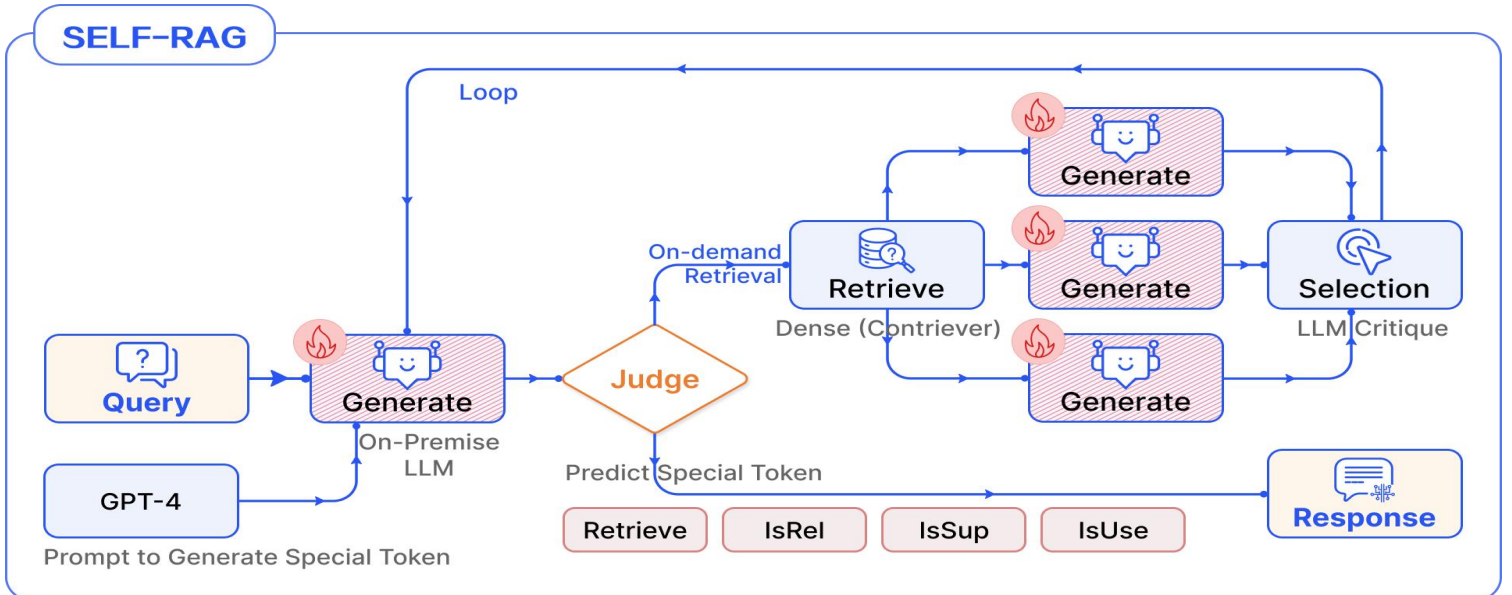


Tuning-based

对LLM进行微调使其生成特殊的Token，以此来触发和控制检索与生成。

SELF-RAG

- 给定输入提示和前一个生成结果，预测特殊标记“Retrieve”
- 如果需要检索，则模型生成:一个评论Token来评估检索到的文章的相关性，以及一个评估Token来评估响应段中的信息是否被该文章所支持。



Agentic RAG Flow 模式总结与讨论

Flow Pattern	特点		适用场景
Sequential	简单有效	复杂任务处理能力弱	难度不大的场景 适合作为**快速上手**
Conditional	多路由配置，适合多任务场景	每个路由分支仍是Sequential结构，复杂任务能力弱	数据**结构丰富**，同时包括文本、表格、图结构等
Branching	拆解复杂查询，分别检索生成后聚合，复杂任务处理能力提升	分支后需要聚合，各项开销提高	查询较复杂，**易于分解**成多个子问题，容忍一定的时延
Loop	多次RAG，具有较高的自主性和灵活性。复杂任务能力较高	可控性下降，时延和开销较大	任务**困难**，对**时延要求不高**，允许模型有一定灵活性的**非严肃任务**场景

02 RAG 新趋势

2-1. Agentic RAG 崛起 | 2-2. Graph与RAG 协同 | 2-3. 如何在RAG中注入推理能力

RAG+Graph 成为香饽饽

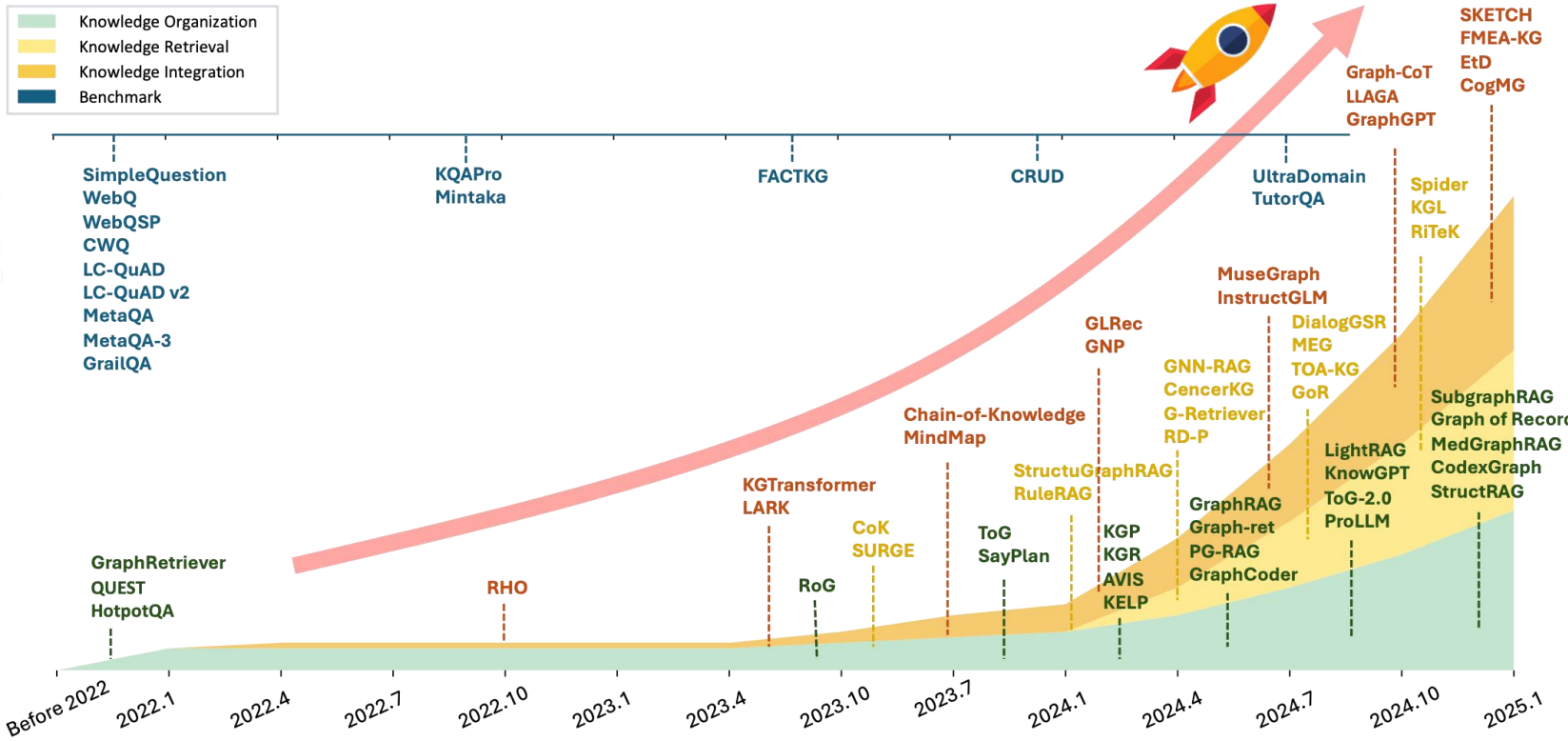
GraphRAG成为前沿探索方向

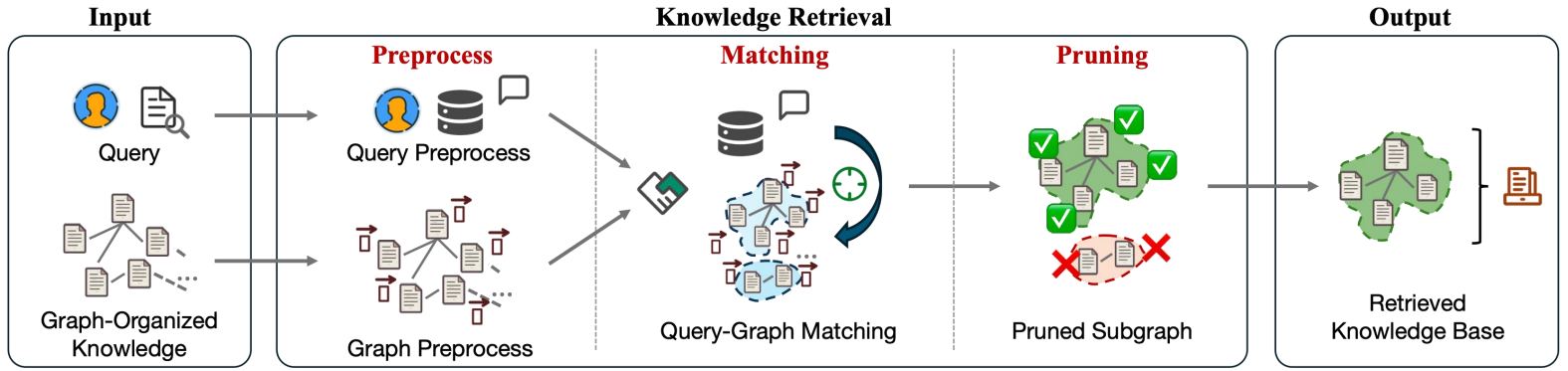
Graph结构的潜在优势

- 更强的复杂推理能力
- 更清晰的可解释可追溯性
- 更好的知识表达与关联性
- 更灵活的知识源集成能力

GraphRAG的核心议题

- 图推理能力增强
- 图结构化的知识表示
- 高效的图信息检索
- 利用图上知识的校验





A Survey of Graph Retrieval-Augmented Generation for Customized Large Language Models, 2025.

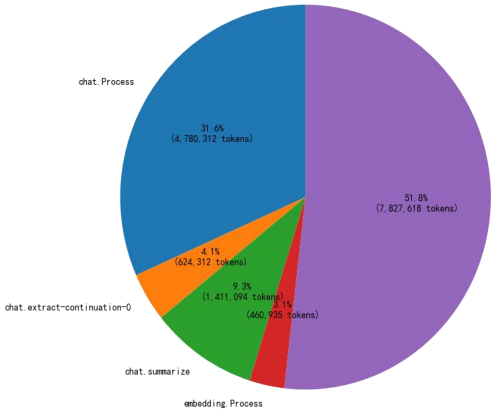
更轻量高效的 GraphRAG

Microsoft GraphRAG 较高的构建成本

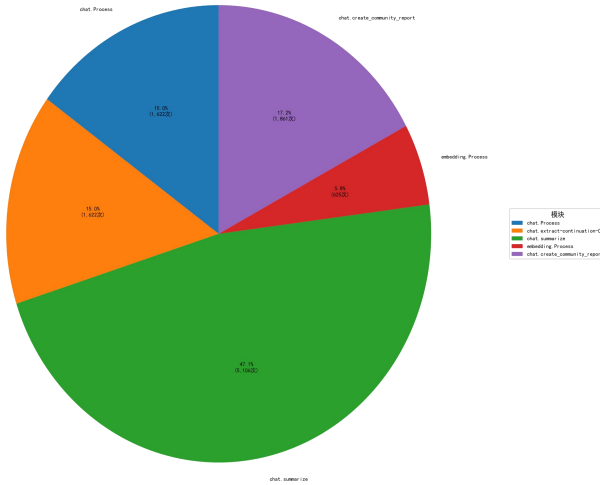
- 原始语料: 0.8M （约300篇新闻）
- 索引消耗: 16 M token（2000%+ 原始语料）
- 其中: 抽取（35%）摘要（10%）社区摘要（50%）
- Http 请求: 10,840+

如何构建更轻量级、更高效的GraphRAG？

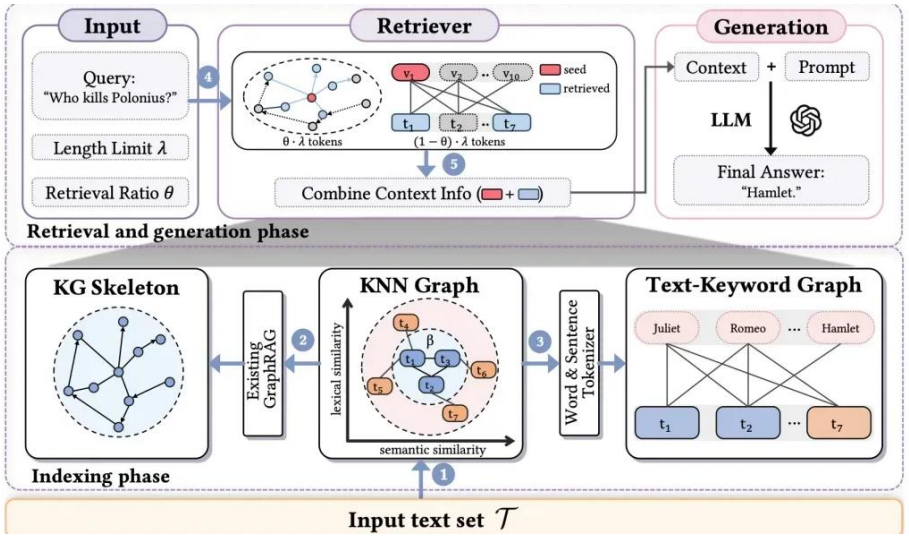
各个模块Token消耗占比



各个模块Http请求占比

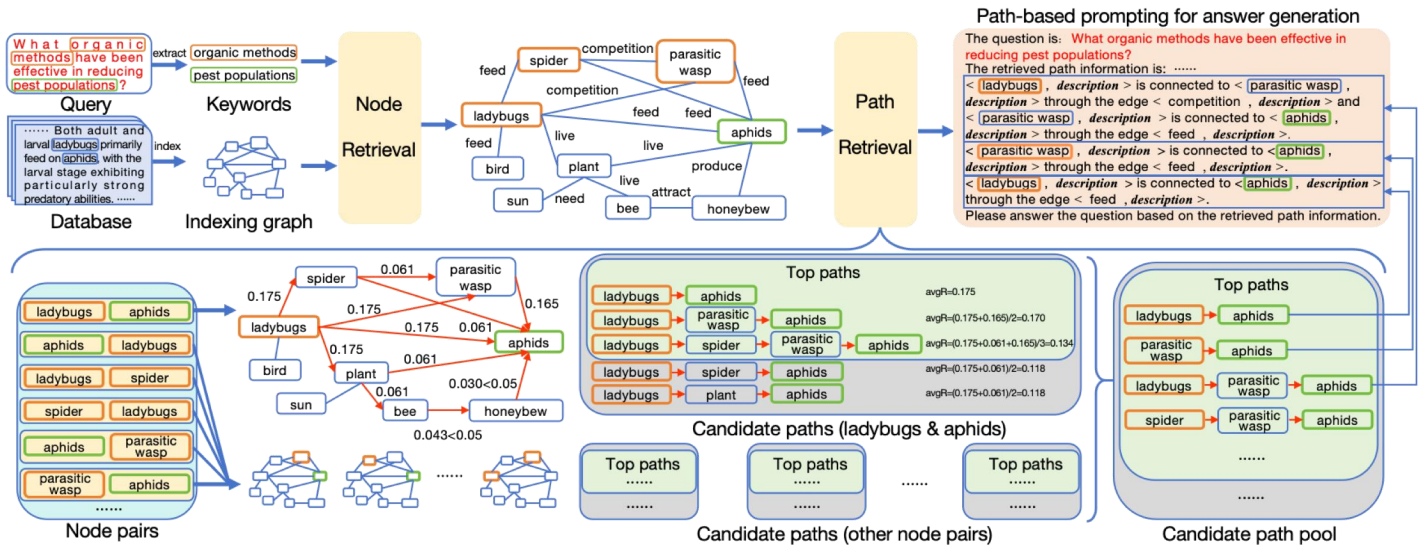


KET: 只构建核心图



KET-RAG: A Cost-Efficient Multi-Granular Indexing Framework for Graph-RAG, 2025.

PathRAG: 基于路径的剪枝

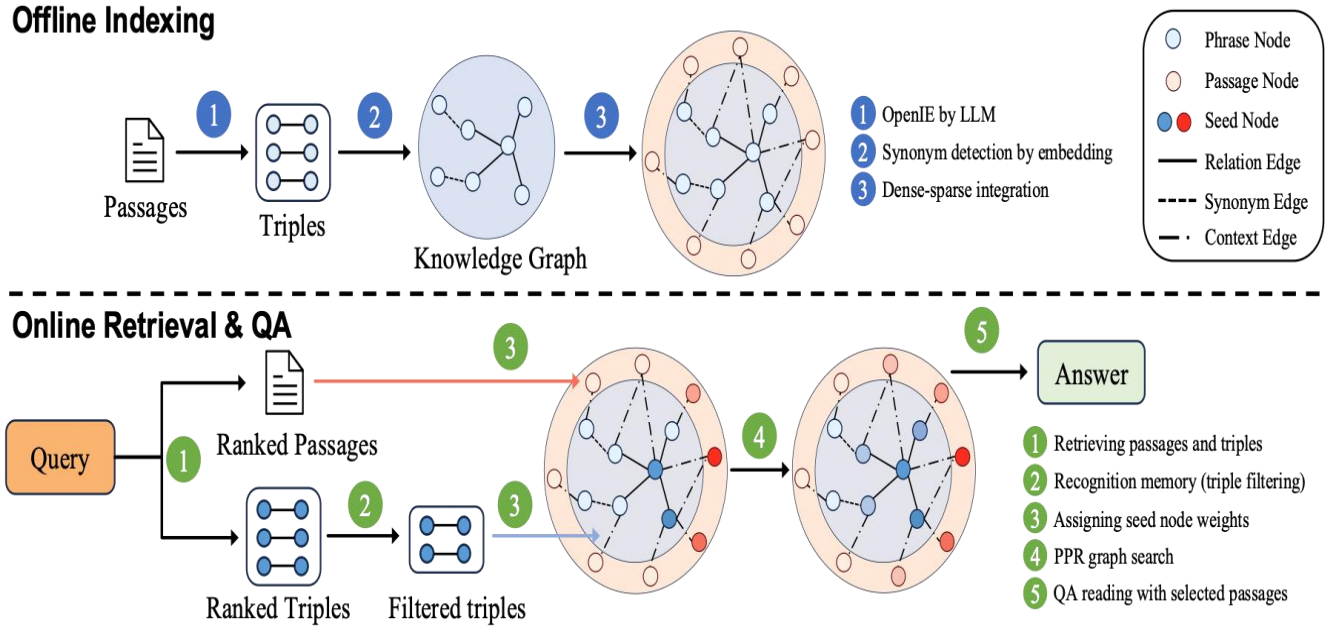


PathRAG: Pruning Graph-based Retrieval Augmented Generation with Relational Paths, 2025.

个性化 GraphRAG

HippoRAG2: 模拟人类长期记忆

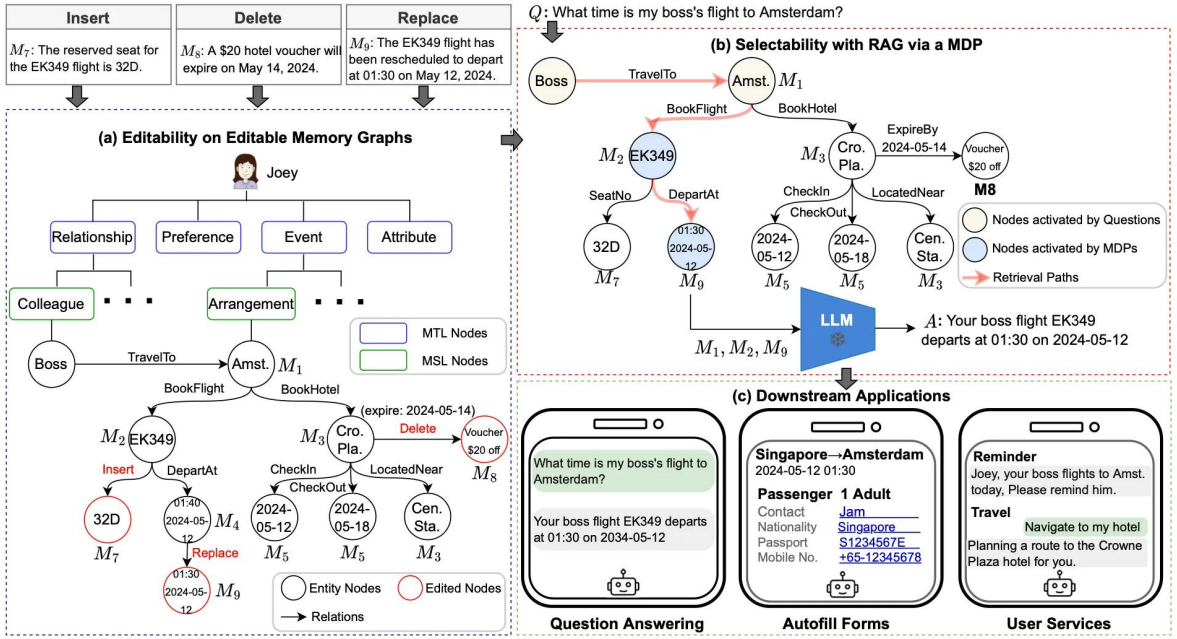
- 个性化的PageRank：进行多跳推理，提升复杂问题的关联性检索能力。
- 密集-稀疏编码结合：引入短语节点和段落节点，模拟人类大脑的密集-稀疏编码机制，更好地整合概念和上下文。



From RAG to Memory: Non-Parametric Continual Learning for Large Language Models, 2025.

EMG-RAG: 个人的记忆构建与利用

- 可编辑记忆图：多层数据结构，记忆类型层、记忆子类层和记忆图层，支持记忆的插入、删除和替换操作，能够动态管理个人记忆。
- 强化学习优化：学习在图上选择相关记忆，动态调整记忆选择策略，以生成更准确的答案，优化记忆选择过程



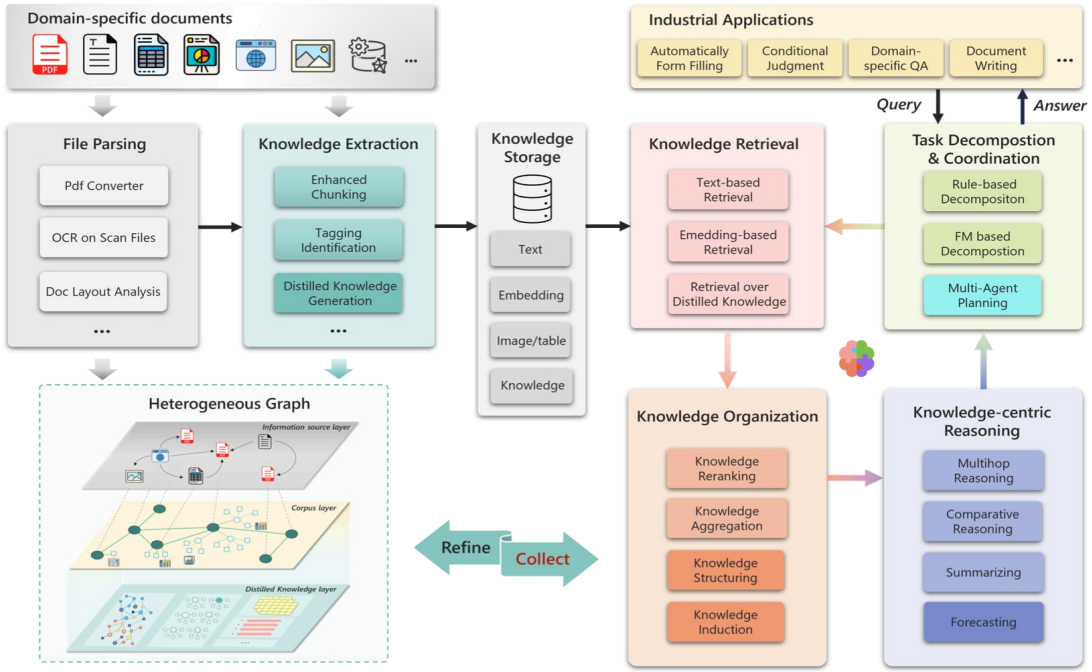
Crafting Personalized Agents through Retrieval-Augmented Generation on Editable Memory Graphs, 2024.

GraphRAG 企业级应用场景

PIKE-RAG: 更精细化的领域知识运用

专业知识增强：
- 知识原子化：将文档分解为细粒度原子知识单元
- 多层异构知识图谱：信息资源、语料和提炼知识的三层图结构。

动态任务分解：
- 知识感知任务分解：结合可用知识动态生成子查询，迭代优化推理路径，解决多跳推理问题。
- 多智能体规划：模拟多视角推理。协作处理创造性问题，

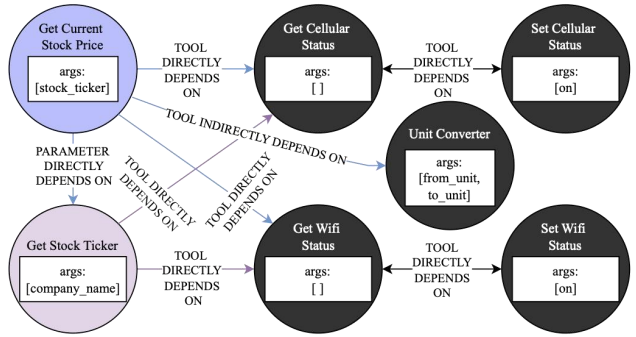


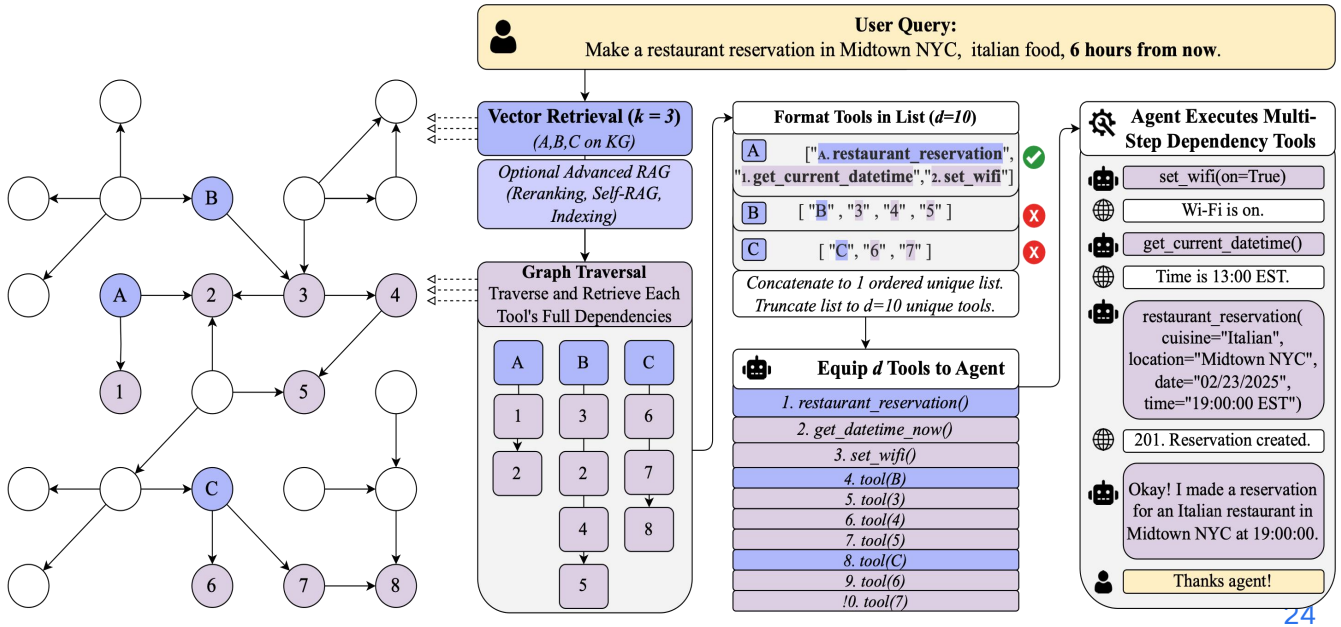
PIKE-RAG: sPecIalized KnowledgE and Rationale Augmented Generation, 2025.

Graph RAG-Tool Fusion: 工具使用和管理

工具图谱：
向量检索+图遍历，捕获工具及其嵌套依赖关系。

多依赖关系建模：
工具间的四种关系（直接依赖、间接依赖、参数直接依赖、参数间接依赖），可扩展性高。





Graph RAG-Tool Fusion, 2025.

02 RAG 新趋势

2-1. Agentic RAG 崛起 | 2-2. Graph与RAG 协同 | 2-3. 如何在RAG中注入推理能力

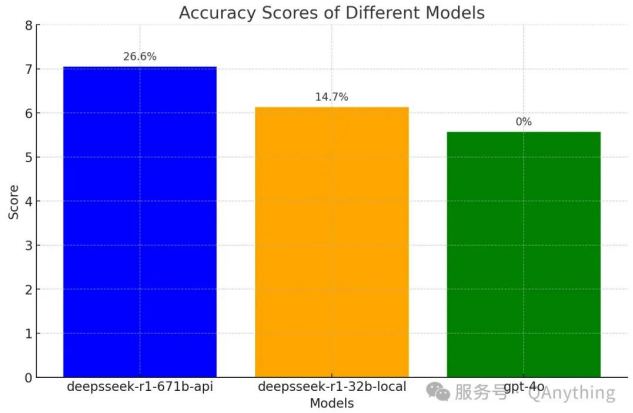
RAG+Reasoning 是否成就新的CP？

OpenAI O1 和DeepSeek-R1等慢思考模型展示出了强大的复杂任务处理能力，对RAG的影响和启发是什么？

推理模型（DeepSeek-R1）在RAG场景上的表现如何？

QAnything测试

- R1模型无需复杂prompt工程，直接明确需求即可，否则可能适得其反。
- 能假设用户真实意图，有上下文理解和推理能力强，关键在于检索环节。



R1 as Embedding

- R1 作为检索模型表现不好，不如专门的Embedding模型
- 擅长推理重点是顺序思考和逻辑连接。不擅长将文档映射到语义空间中。
- 有时会遵循导致主题相关但实际上无关的结果的推理路径

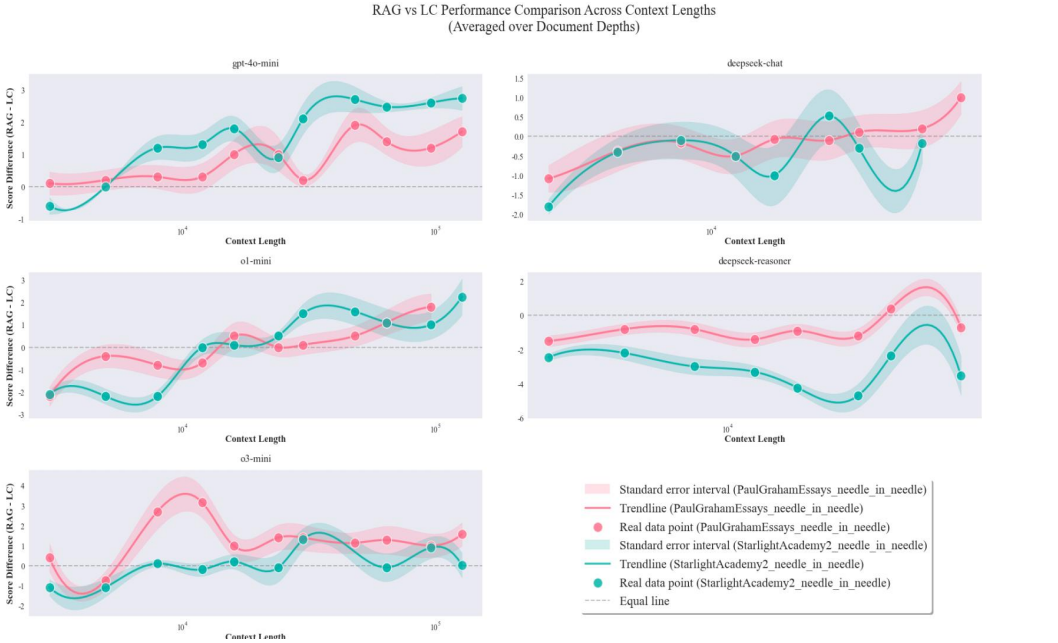
Break a lease | Small Claim Court

Query: `I want to break my lease. My landlord doesn't allow me to do that.`

Results by Qwen	Results by DeepSeek-R1
Leasing Agent saying i have to stay another month because of 30 days of notice?	100% At Fault for Car Accident, now Insurance Company has my case moving to "litigation department"
Moving into new apartments and one of my room mates can't be on the lease because she's on another lease with her ex boyfriend for a couple more months.	Landlord telling tenants we must pay for her chimney to be swept, must use a sweep of her choice
Landlord Asking to Sign New Lease	Ex Girlfriend stole my car
AZ Landlord requiring us to vacate for house showings.	1/5 roommates did not pay rent
[MO] both of our names are on the lease - what's the best course of action if I want to kick my boyfriend out?	I got into a car crash, and after I was kind of assaulted.

U-NIAH测试

在大海捞针（Needle-In-A-Haystack, NIAH）的复杂RAG场景中（更长的上下文、更多的干扰项、更分散的关键信息）R1在RAG下的表现并不如直接使用R1，更容易受到干扰。



[1] QAnything引擎升级技术通告：DeepSeek-R1适配实践与效果验证.QAnything 官方公众号

[2] U-NIAH: Unified RAG and LLM Evaluation for Long Context Needle-In-A-Haystack, 2025.

[3] Using DeepSeek R1 for RAG: Do's and Don'ts. https://blog.skypilot.co/deepseek-rag/

如何将推理能力注入RAG

三个核心问题：1）谁来进行推理？ 2）推理什么？ 3）以什么方式进行推理？

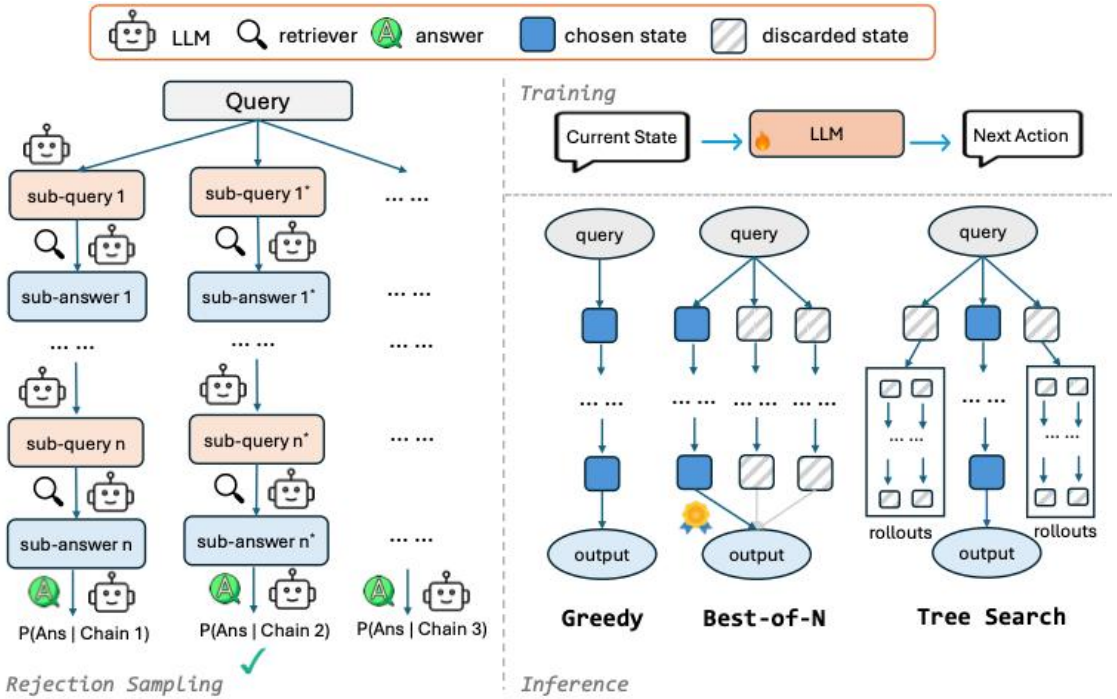
推理主体	推理对象	推理方法	优化方法
大语言模型	查询	思维链	监督微调
推理大模型	检索内容	一般树搜索	传统强化学习
策略网络	动作/算子	蒙特卡洛树搜索	结果奖励
专有大模型	实体/图	动作预测	过程奖励
符号系统		图遍历	

推理主体——大语言模型/推理大模型

CoRAG: LLM 检索链规划

动态检索过程

- 逐步检索信息并进行调整；
- 根据需要重新制定查询；
- 自主决定需要多少次检索步骤



Chain-of-Retrieval Augmented Generation, 2025.

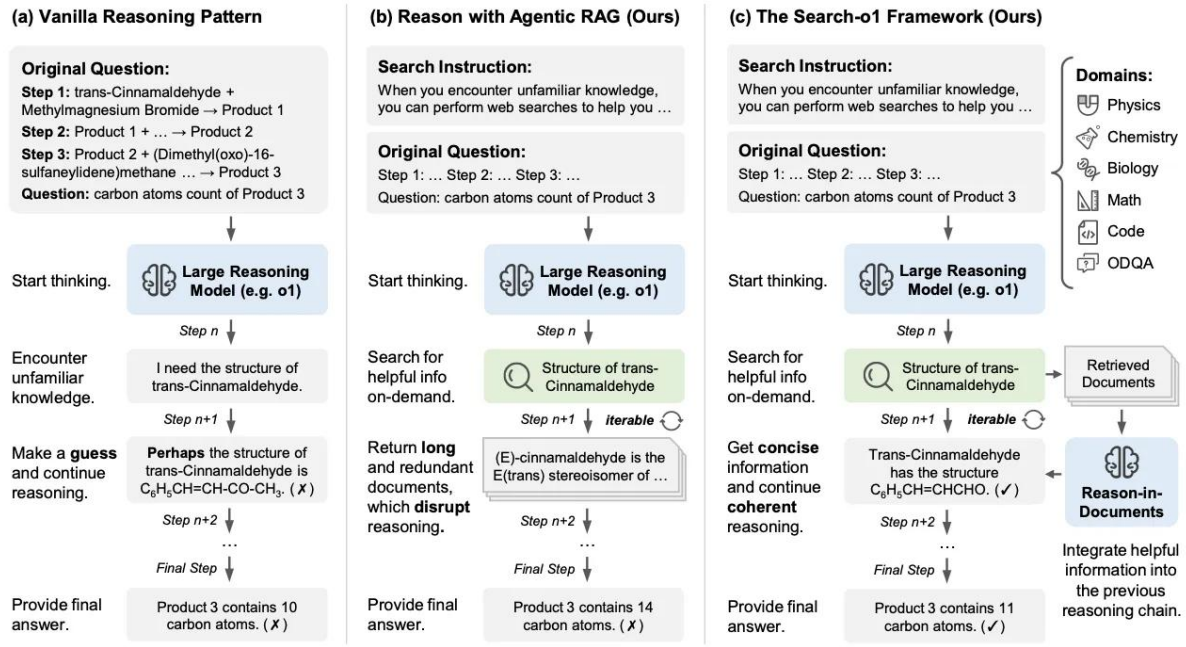
Search O1: LRM 检索链规划与精炼

- 动态代理检索机制

LRM自主触发外部知识检索，动态补充知识缺口，支持多次迭代检索。

- Reason-in-Documents 知识精炼模块

独立处理检索到的冗长文档，提取关键信息并整合到推理链中。避免直接注入原始文档导致的噪声和逻辑断裂。

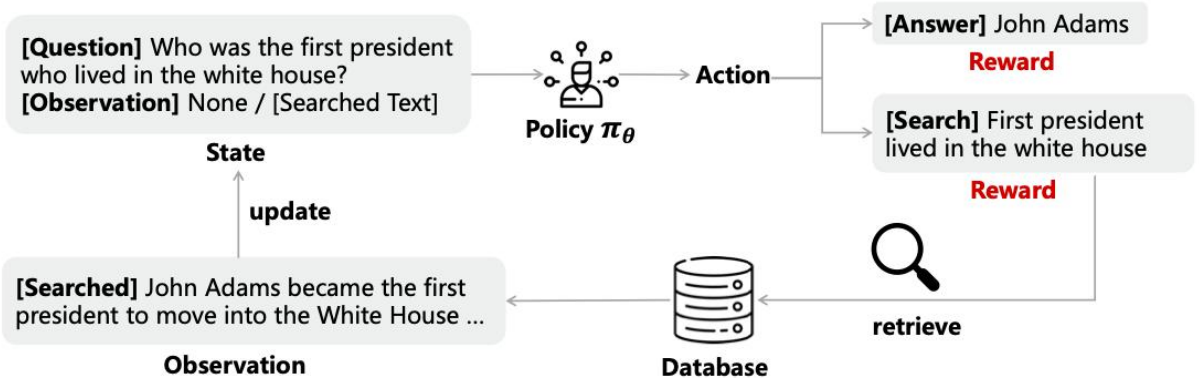


Search-o1: Agentic Search-Enhanced Large Reasoning Models, 2025.

推理主体——外部模型/系统

SmartRAG: 策略网络推理

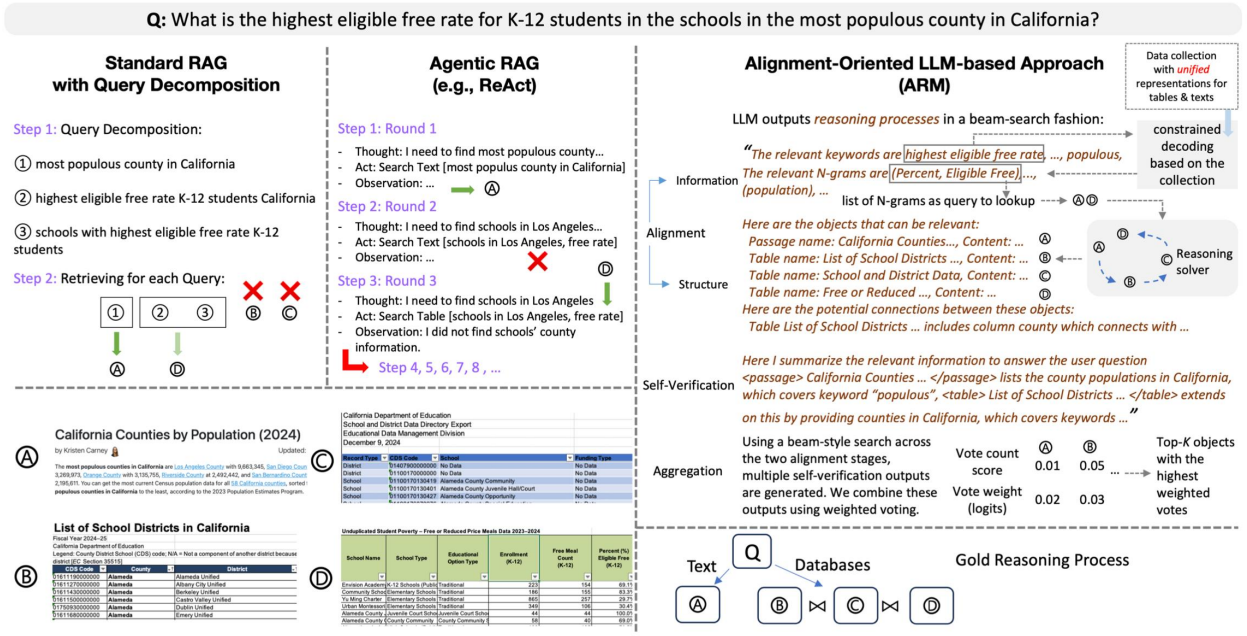
- 策略网络设计：决定何时检索、生成优化查询、基于检索结果生成答案，
- 联合优化框架：通过PPO联合优化 RAG 相关模块（检索决策、查询重写、答案生成）端到端训练
- 环境反馈驱动学习：奖励函数结合准确性（精确匹配和 F1 分数）和检索成本（惩罚过度检索），鼓励系统在最少检索下正确回答。



SmartRAG: Jointly Learn RAG-Related Tasks From the Environment Feedback, 2025.

ARM: 外部求解器推理

- 对齐导向检索：将检索视为一个生成过程，来找到所有必要的数据，实现一次检索所有。
- 结构对齐模块：引入外部求解器（Mixed-Integer Programming, MIP），动态选择并连接数据对象（如表格、文本）
- 自验证与聚合：LLM 验证所选对象的完整性，通过束搜索和投票机制综合多候选结果。



Can we Retrieve Everything All at Once? ARM: An Alignment-Oriented LLM-based Retrieval Method, 2025.

如何将推理能力注入RAG

三个核心问题：1）谁来进行推理？ 2）推理什么？ 3）以什么方式进行推理？

推理主体

- 大语言模型
- 推理大模型
- 策略网络
- 专有大模型
- 符号系统

推理对象

- 查询
- 检索内容
- 动作/算子
- 实体/图

推理方法

- 思维链
- 一般树搜索
- 蒙特卡洛树搜索
- 动作预测
- 图遍历

优化方法

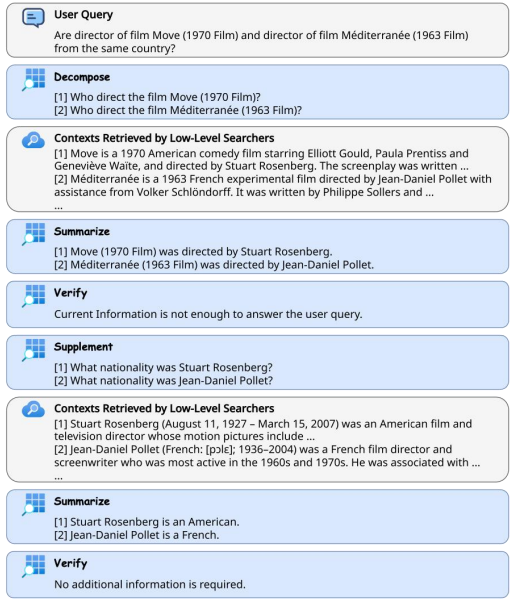
- 监督微调
- 传统强化学习
- 结果奖励
- 过程奖励

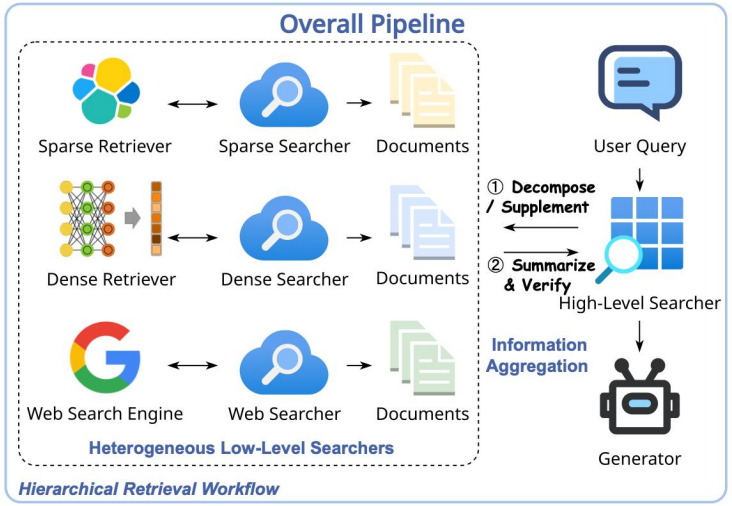
推理对象——查询/检索内容

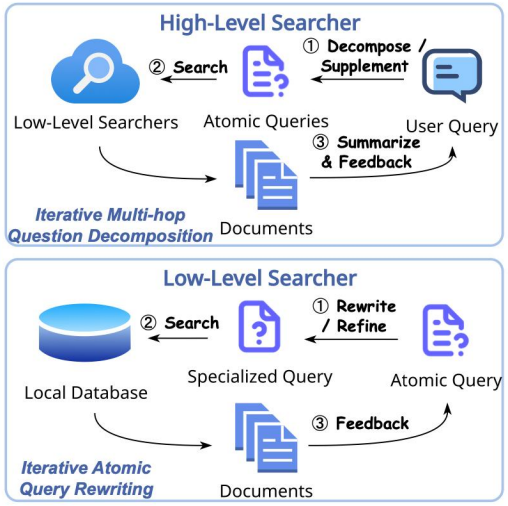
LevelRAG: 对查询的分解与扩展

分层架构设计

- 高层搜索器：负责多跳逻辑规划，将复杂查询分解为原子查询
- 低层搜索器：包含稀疏、密集检索器，针对不同检索器优化查询





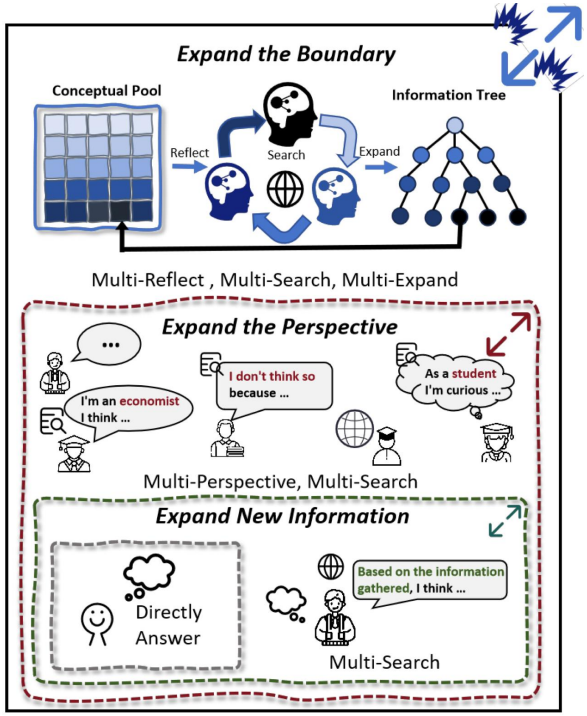


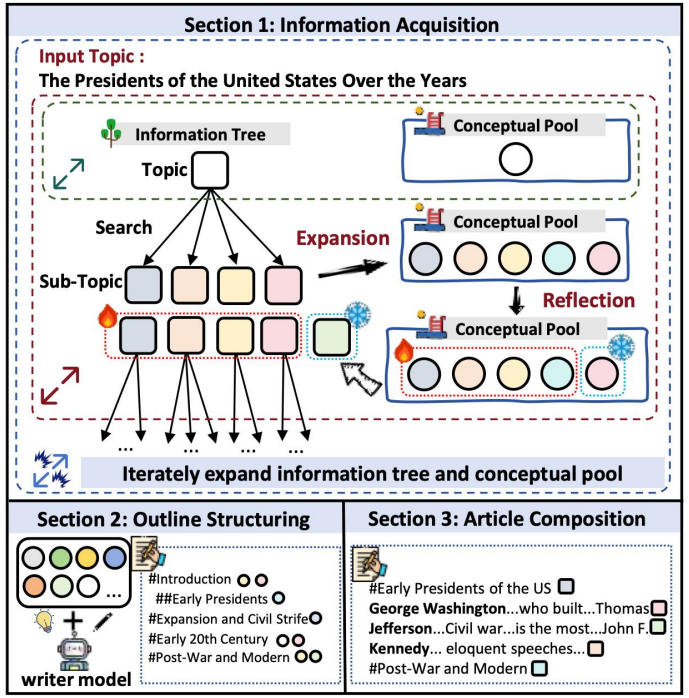
OmniThink: 对检索内容的拓展与精炼

模拟人类慢思考过程：通过迭代扩展与反思机制，动态整合外部知识并优化内部认知框架，突破知识边界限制。

动态知识扩展与结构化：

- 信息树：基于主题构建层次化知识结构，通过子节点扩展多维度信息检索。
- 概念池：持续提炼检索信息的核心洞见，形成可迭代更新的认知边界，指导后续扩展方向。





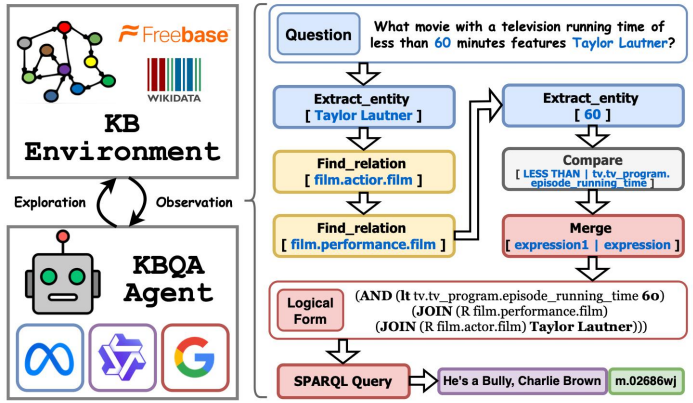
LevelRAG: Enhancing Retrieval-Augmented Generation with Multi-hop Logic Planning over Rewriting Augmented Searchers, 2025.

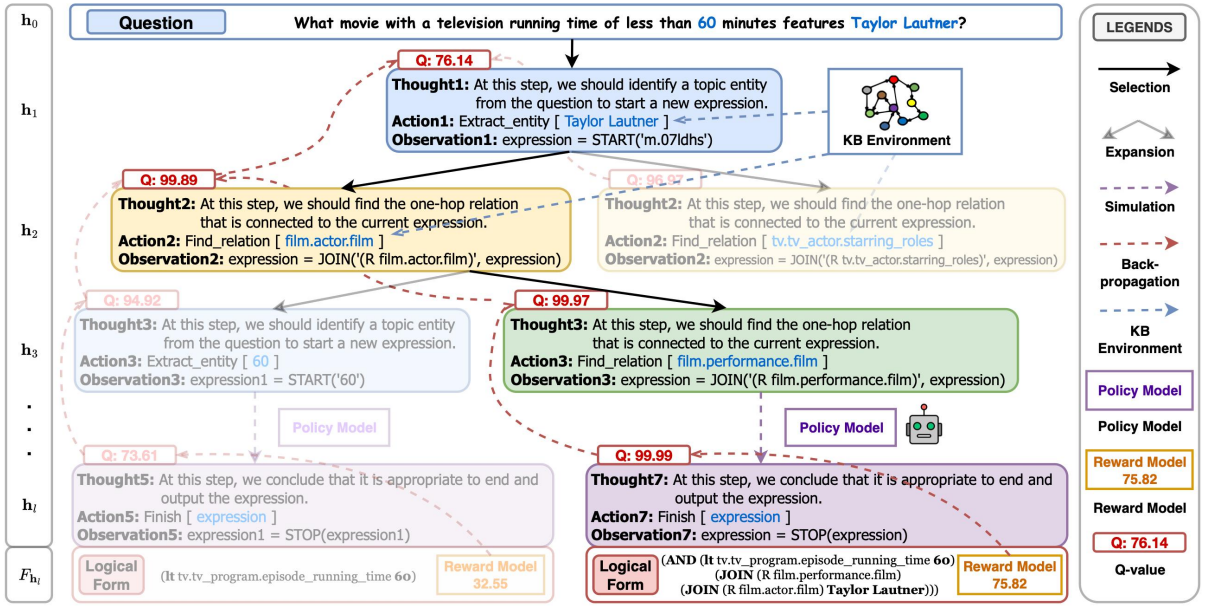
OmniThink: Expanding Knowledge Boundaries in Machine Writing through Thinking, 2025.

推理对象——算子/实体

KBQA-O1: KG上的算子规划

- ReAct-based 过程：通过逐步生成逻辑形式并与 KB 环境交互，提升对 KB 结构的感知能力。
- KB—Based推理：预测下一步动作，动作是KG上可执行算子（例如查询、聚合、抽取），生成候选路径





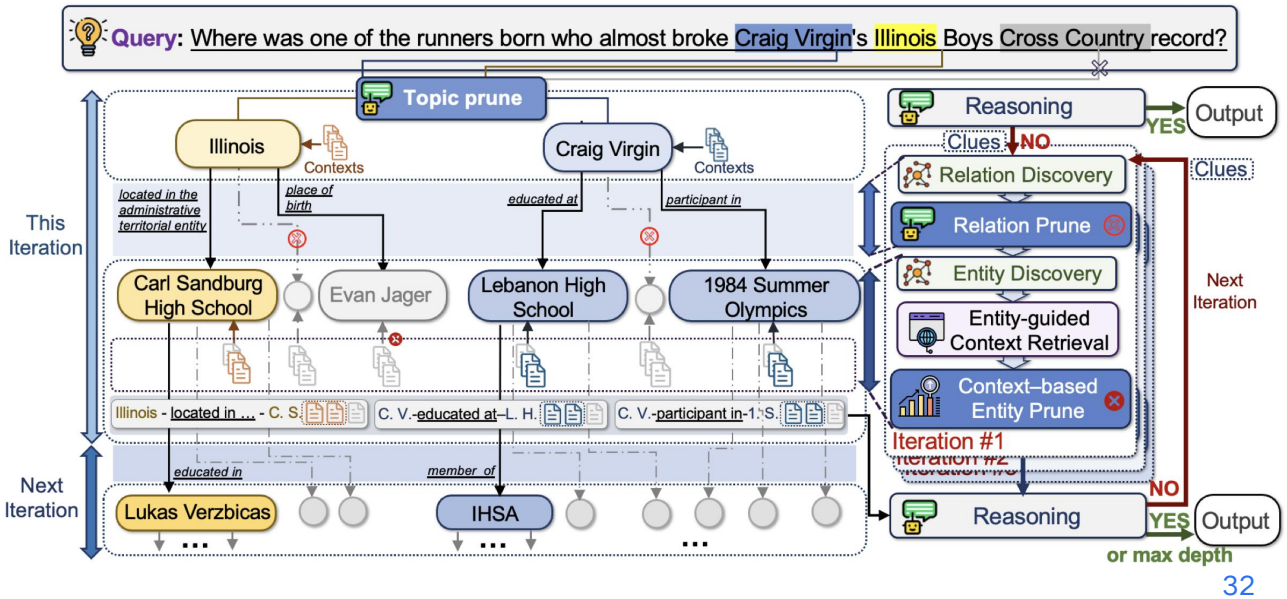
ToG 2.0: KG上的知识（实体/关系）推理

紧密耦合知识图谱与文本检索：

通过KG连接文档实体，利用文档内容作为实体上下文，解决中知识碎片化和结构关系缺失的问题。

迭代式混合知识探索

- 图检索：基于 KG 的关系发现与剪枝，扩展候选实体路径。
- 上下文检索：筛选高相关实体，反向优化图检索方向。
- 交替迭代直至收集足够线索，模拟人类逐步推理过程。



KBQA-o1: Agentic Knowledge Base Question Answering with Monte Carlo Tree Search, 2025. *Think-on-Graph 2.0: Deep and Faithful Large Language Model Reasoning with Knowledge-guided Retrieval Augmented Generation, 2024.*

如何将推理能力注入RAG

三个核心问题：1）谁来进行推理？ 2）推理什么？ 3）以什么方式进行推理？

推理主体

- 大语言模型
- 推理大模型
- 策略网络
- 专有大模型
- 符号系统

推理对象

- 查询
- 检索内容
- 动作/算子
- 实体/图

推理方法

- 思维链
- 一般树搜索
- 蒙特卡洛树搜索
- 动作预测
- 图遍历

优化方法

- 监督微调
- 传统强化学习
- 结果奖励
- 过程奖励

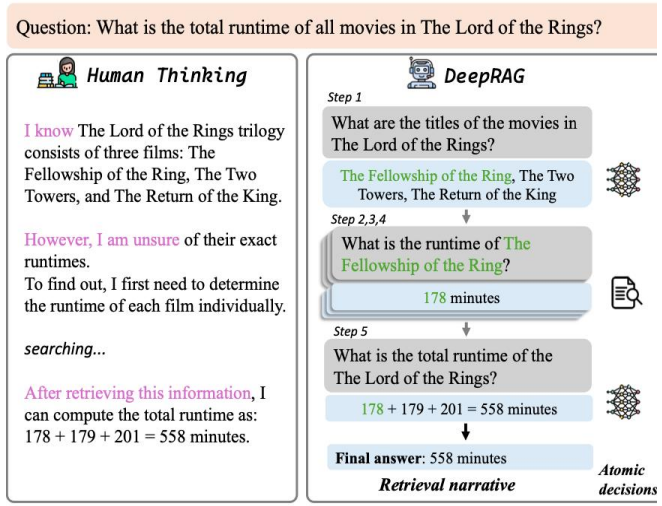
推理方式——树搜索

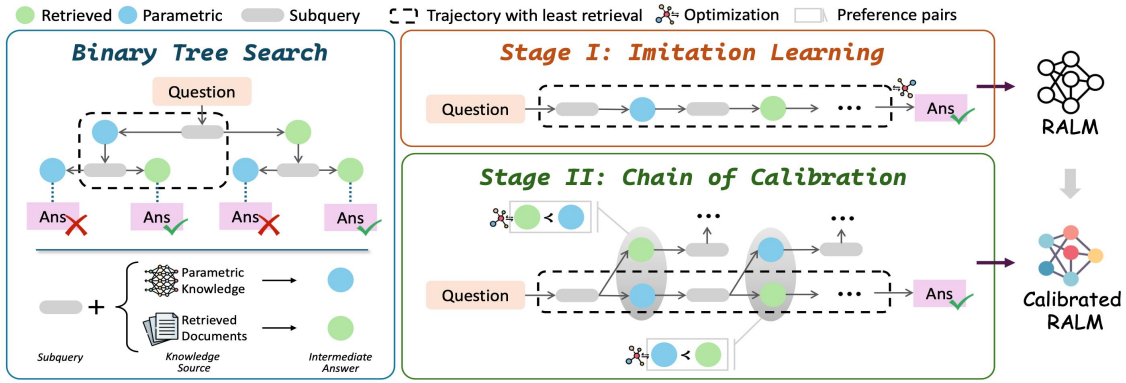
DeepRAG: 二叉树搜索

动态检索决策：将RAG建模为马尔可夫决策过程（MDP），通过状态转移动态决定是否检索外部知识或依赖模型内部知识。

模仿学习：让模型学会以最小化检索成本（如次数）将复杂问题分解为子查询序列。

校准链（Chain of Calibration）：通过偏好数据优化原子决策，提升模型对自身知识边界的认知。





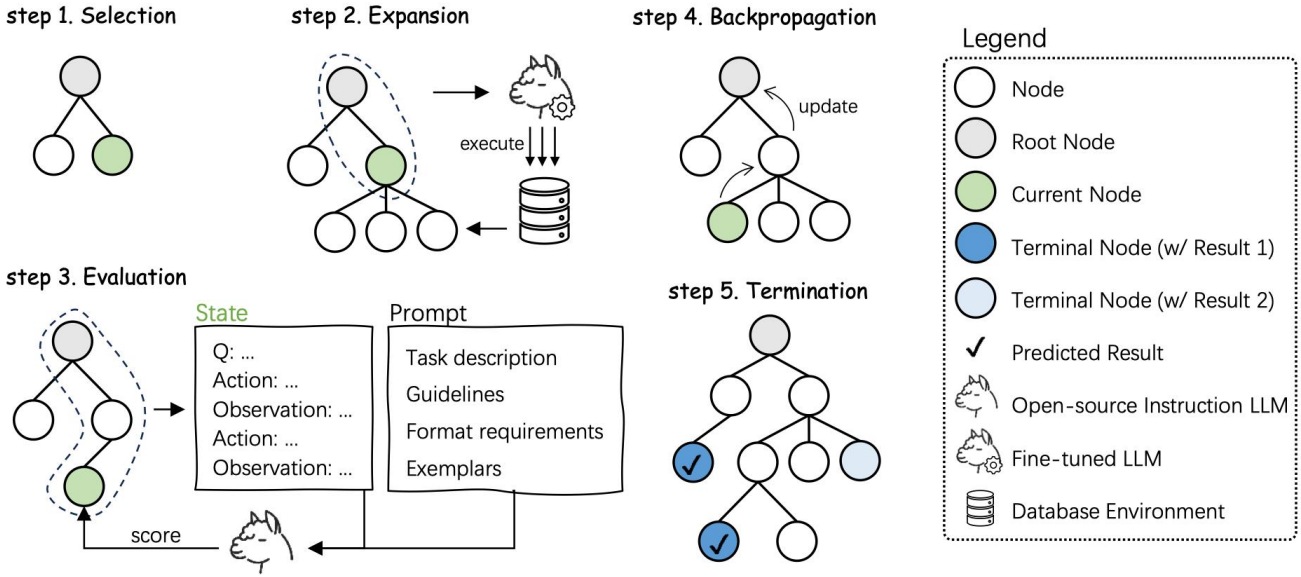
DeepRAG: Thinking to Retrieval Step by Step for Large Language Models, 2025.

MCTS-KBQA: 蒙特卡洛树搜索

MCTS 与 KBQA 结合：借助 MCTS 的选择、扩展、评估、模拟和反向传播步骤，对解决方案空间进行有效探索，提升推理效率。

分步奖励机制：依据 KBQA 任务特点，设计规则提示文本，评估中间状态是否成功识别元素和解决子问题。

中间推理标注数据：运用远程监督技术，为现有的问题 - SPARQL 数据集标注中间推理过程，助力低资源场景下模型性能提升。



MCTS-KBQA: Monte Carlo Tree Search for Knowledge Base Question Answering, 2025.

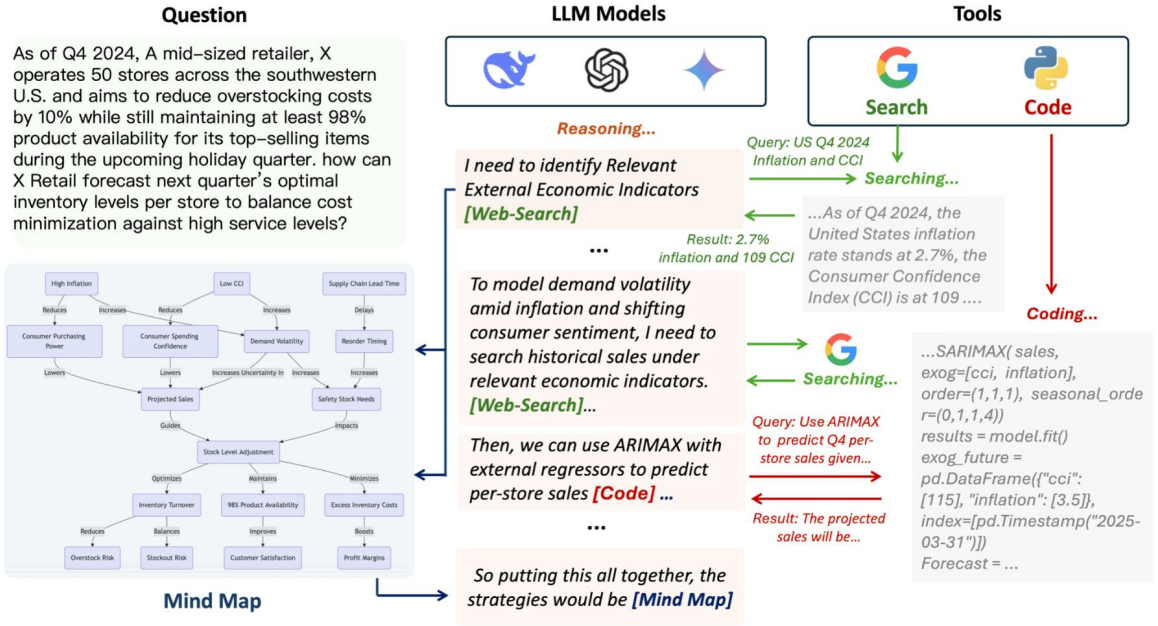
推理方式——动作预测

Agentic Reasoning: LLM 特殊Token预测

动态任务委托：通过Special Token（如[Web-Search]/[Code]）实时触发代理，按需检索信息、执行计算或构建知识图。

结构化知识管理：Mind Map 代理构建逻辑关系图，提升复杂问题的演绎推理能力，避免传统 LLM 的逻辑断裂。

任务专业化分工：使用专用 LLM（如代码生成、知识图构建）处理辅助任务，减少主模型干扰，提升效率。



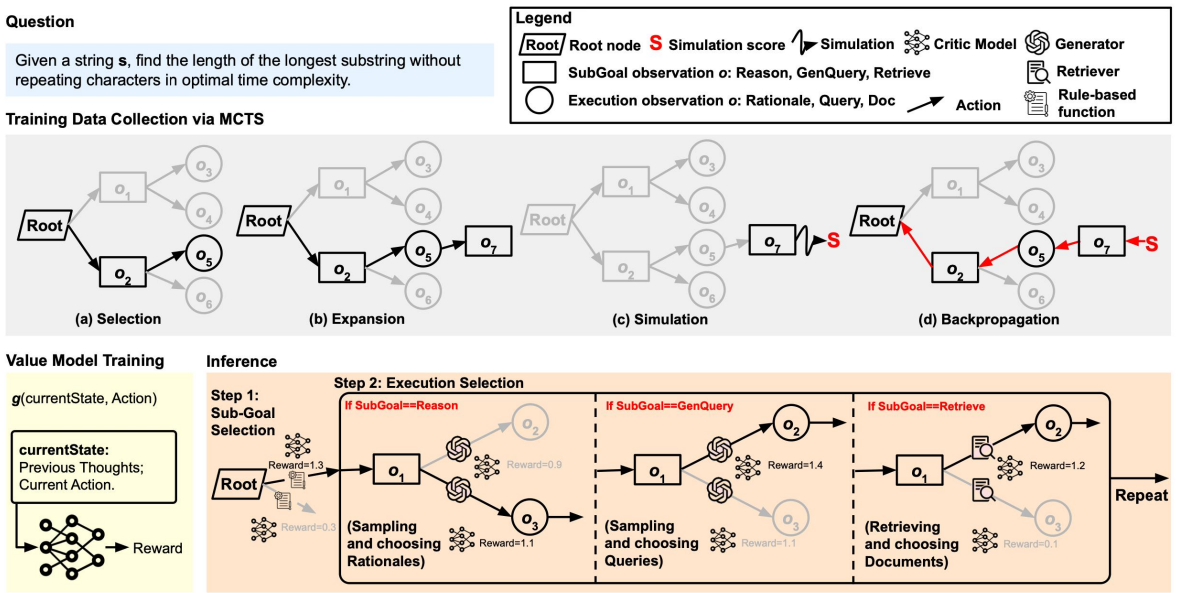
Agentic Reasoning: Reasoning LLMs with Tools for the Deep Research, 2025.

CR-Planner: 专有模型行为预测

评论家指导规划：引入子目标评论家和执行评论家，分别指导推理、检索的子目标选择和执行步骤，动态优化问题解决路径。

MCTS数据生成：通过 MCTS 模拟不同动作序列，收集带有长期奖励标签的训练数据，解决评论家模型训练数据稀缺问题。

检索增强与动态规划结合：在推理过程中动态决定何时检索、如何生成查询，并通过评论家筛选相关文档，减少无关知识干扰。



Can We Further Elicit Reasoning in LLMs? Critic-Guided Planning with Retrieval-Augmentation for Solving Challenging Tasks, 2024.

如何将推理能力注入RAG

三个核心问题：1）谁来进行推理？ 2）推理什么？ 3）以什么方式进行推理？

推理主体

- 大语言模型
- 推理大模型
- 策略网络
- 专有大模型
- 符号系统

推理对象

- 查询
- 检索内容
- 动作/算子
- 实体/图

推理方法

- 思维链
- 一般树搜索
- 蒙特卡洛树搜索
- 动作预测
- 图遍历

优化方法

- 监督微调
- 传统强化学习
- 结果奖励
- 过程奖励

推理优化——有监督微调/强化学习

ReARTeR: 过程监督优化

- 可信过程奖励框架：结合过程奖励模型（PRM）和过程解释模型（PEM），在测试时通过 PRM 提供准确标量分数，PEM 生成自然语言解释，指导推理步骤优化。
- 双阶段优化：通过后训练（MCTS 引导的迭代偏好优化）和测试时扩展（动态检索与推理）协同提升多步推理能力。

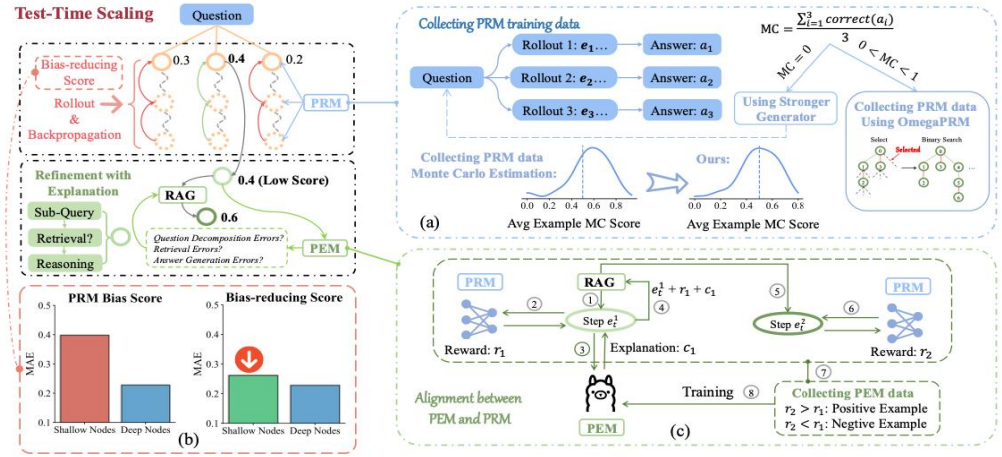


Figure 2: Test-Time Scaling of ReARTeR, which includes collecting unbiased PRM training data for PRM, reducing early-step bias in PRM, and alignment between PEM and PRM.

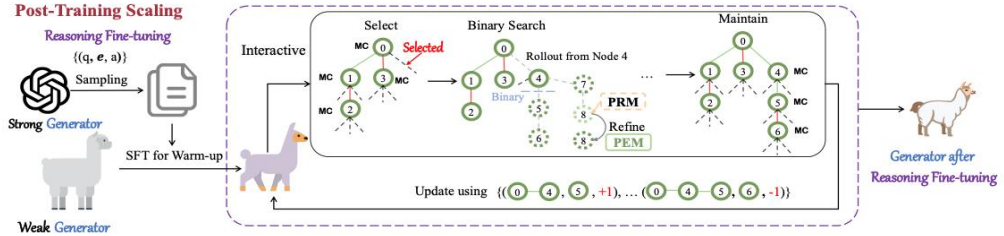


Figure 3: Post-Training Scaling of ReARTeR, which includes Warm-Up and Step-Level Offline Reinforcement Stages.

ReARTeR: Retrieval-Augmented Reasoning with Trustworthy Process Rewarding, 2025.

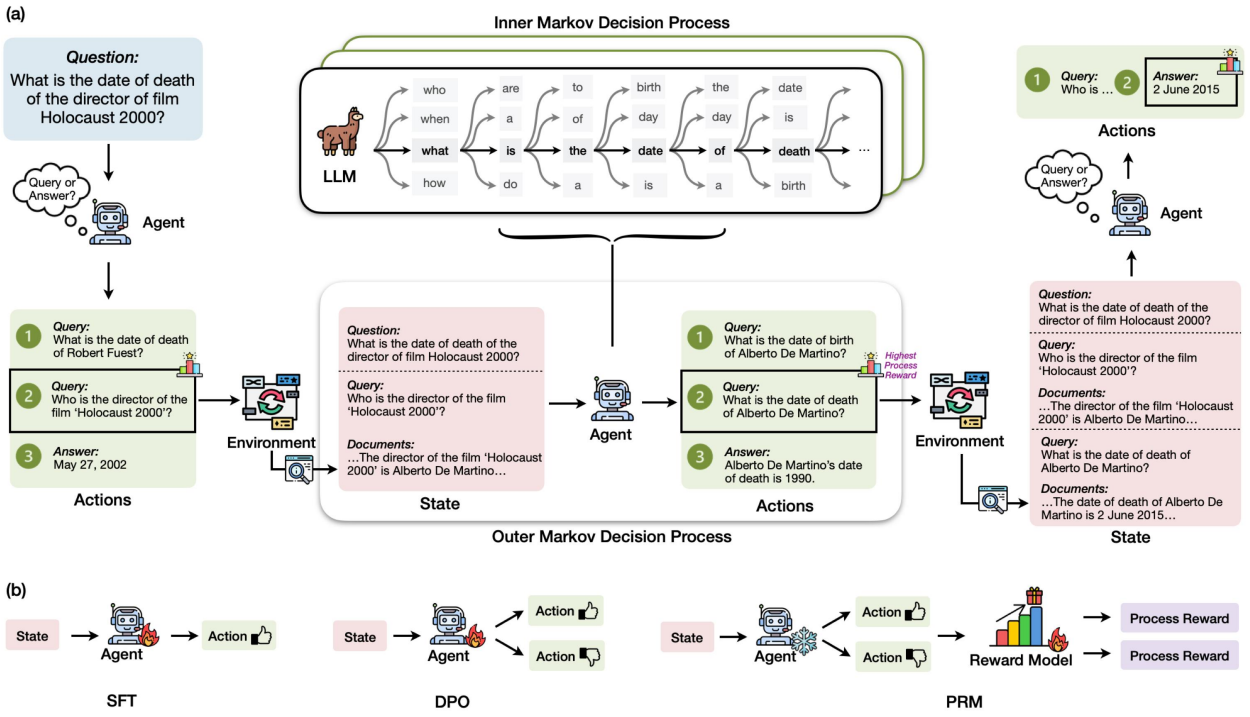
RAG-Gym: 多阶段过程优化

嵌套 MDP 建模：

- 外层 MDP 管理动作生成（查询 / 答案）
- 内层 MDP 控制 LLM 的 token 生成，支持多代理架构。

过程监督技术：

SFT：监督微调优化动作生成策略。
DPO：对比学习优化偏好选择。
PRM：训练独立奖励模型评估中间动作质量。



RAG-Gym: Optimizing Reasoning and Search Agents with Process Supervision, 2025.

03 RAG 应用实践

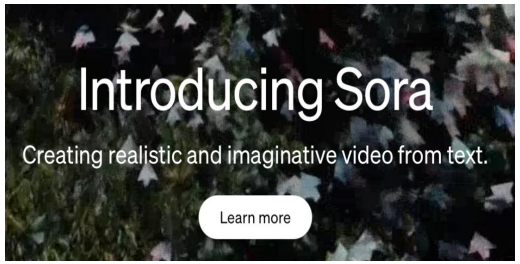
3-1. RAG 技术平台实践 | 3-2. RAG 领域应用

RAG 行业案例

从Modular RAG的视角看当前AI企业提供的RAG服务

OpenAI

当前AI时代的领导者

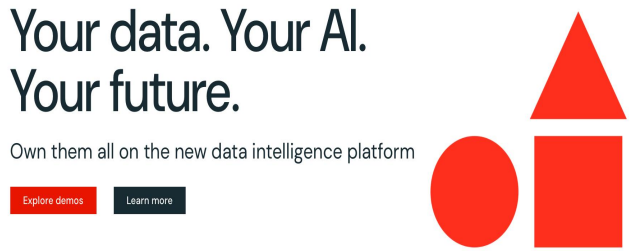


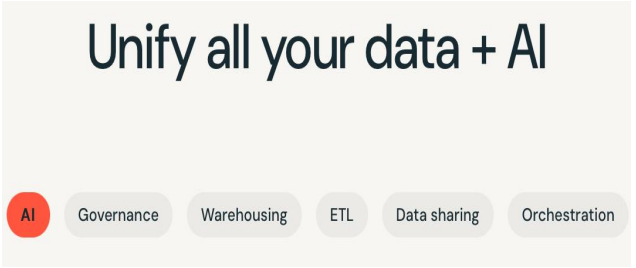
Cohere

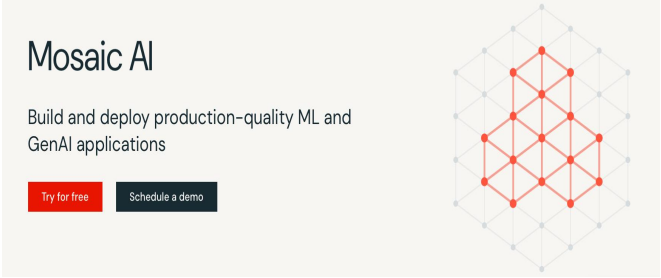
专注于RAG的AI创业公司

Databrick

正在向AI转型的大数据企业



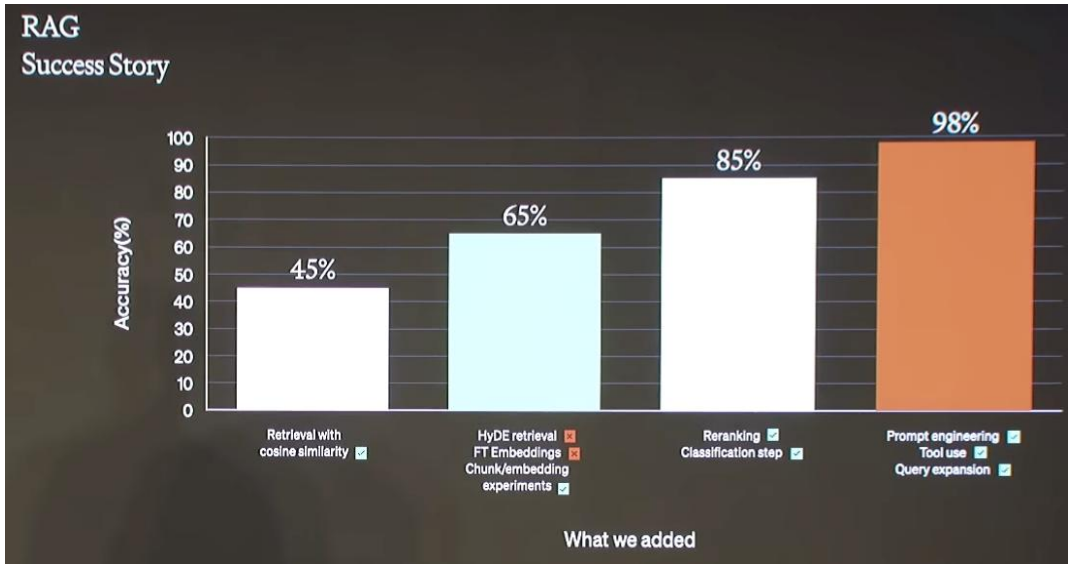


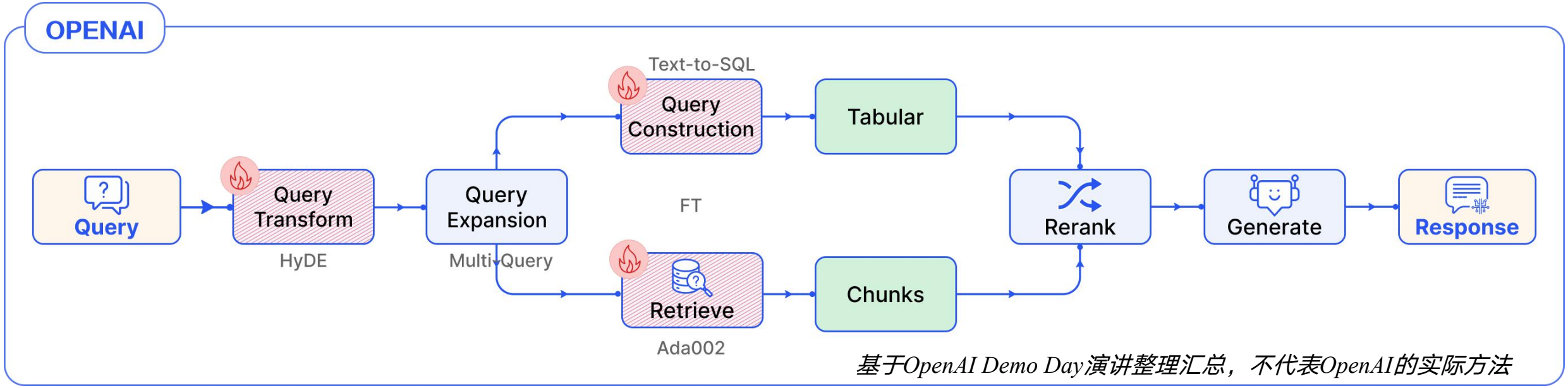


RAG 行业案例—— OpenAI

- OpenAI团队从**45%**的准确率开始
- 假设性文档嵌入（HyDE）和精调嵌入等，效果并不理想
- Chunks / Embedding 调整，达到**65%**准确率
- Reranking+对不同类别问题特别处理，提升到**85%**的准确率
- 提示工程、查询扩展和工具使用，最终达到了**98%**的准确率

团队强调了**模型精调和RAG结合**使用时的强大潜力，尤其是在没有使用复杂技术的情况下，接近行业领先水平。





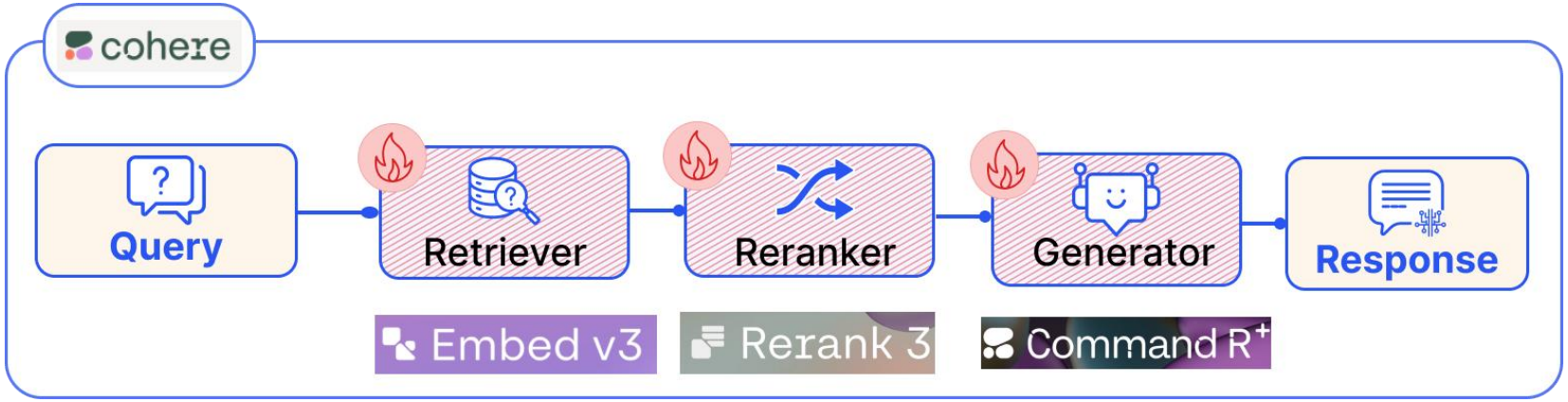
基于OpenAI Demo Day演讲整理汇总，不代表OpenAI的实际方法

RAG 行业案例—— Cohere

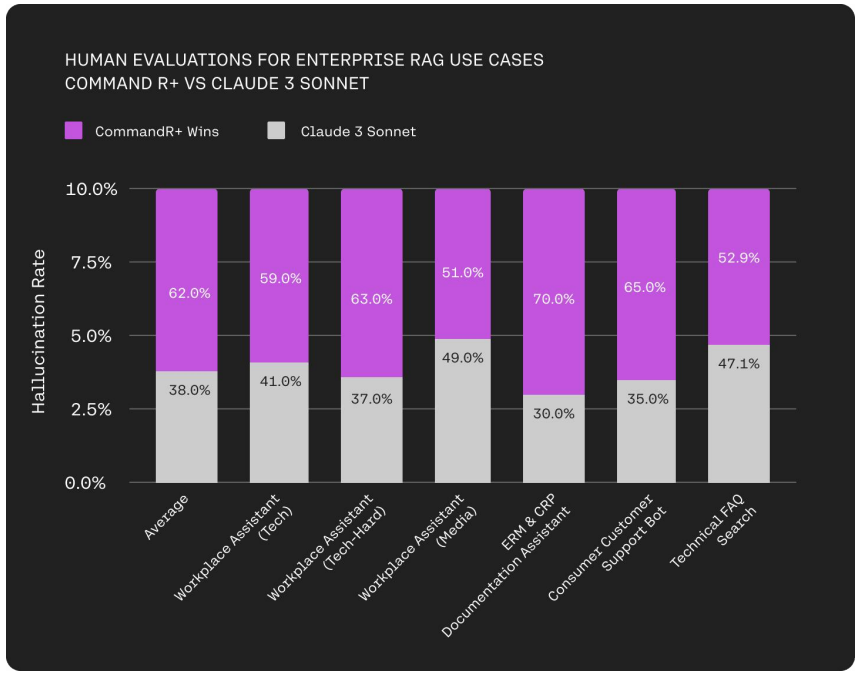
Cohere体现了简单有效的方法风格，不追求流程上的复杂，侧重各个功能模型的优化。

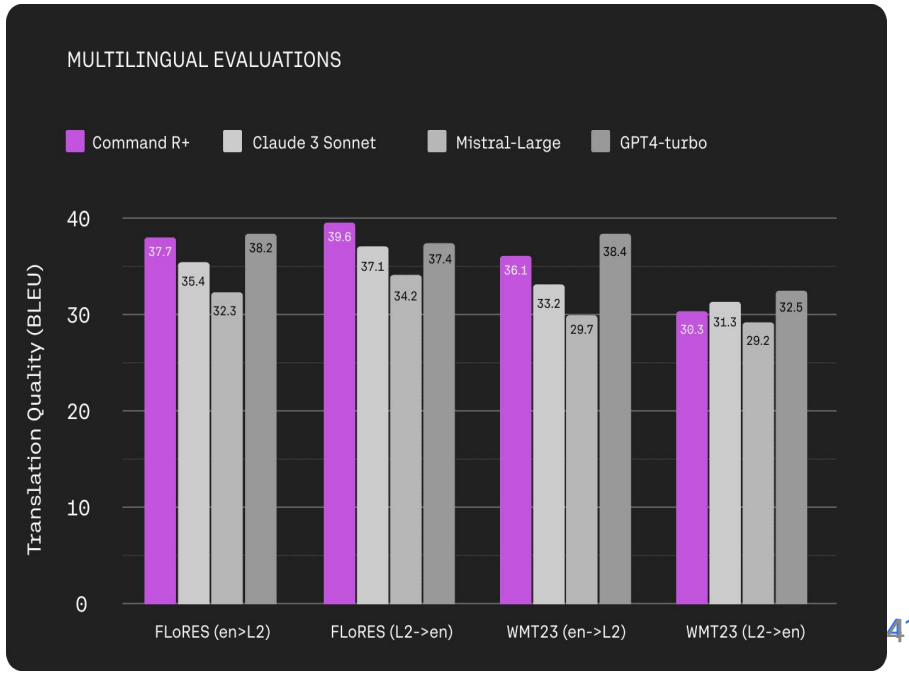
全流程自研模型

- Command R+
- Embed V3
- Rerank 3



- 当组合使用时，RAG任务上全面领先Claude 3 Sonnet
- 多语言表现领先于主流大模型。多语言能力一直是Cohere的优势之一

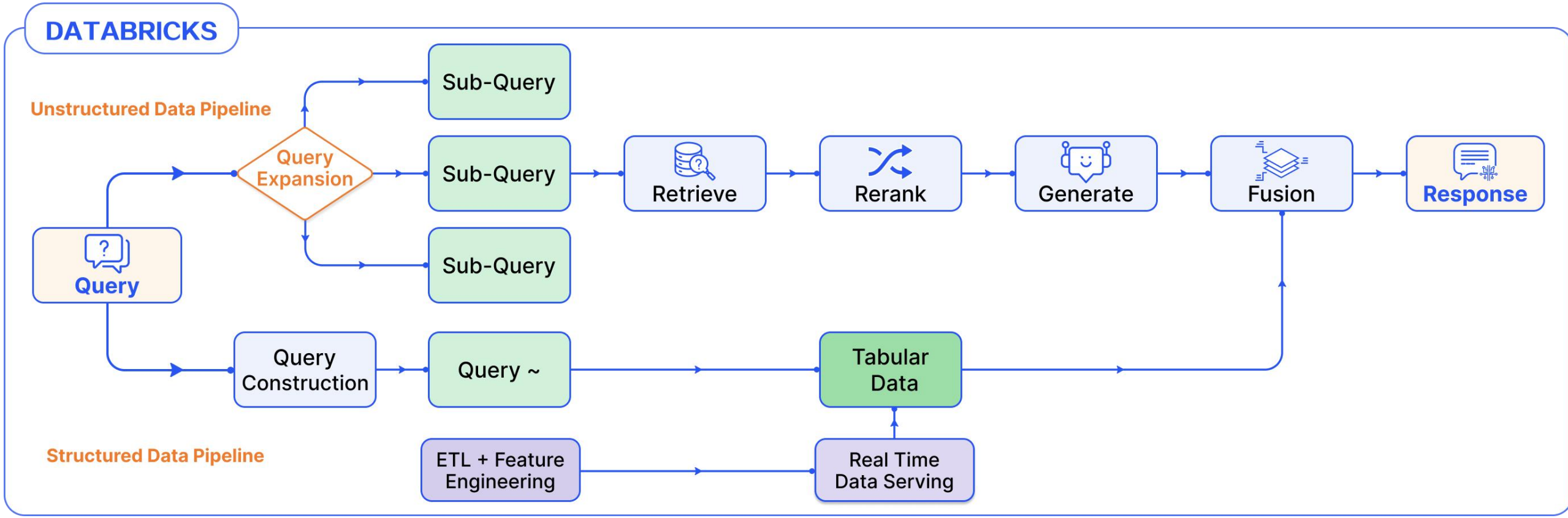




RAG 行业案例—— Databricks

Databricks作为大数据领域中领先的服务商，在RAG设计上依然保持了自己特点和优势。从准确度较高的数据存储中进行额外的检索，充分发挥自身在Real Time Data Serving 上的优势。

- （上分支）通过从事先处理好的文本向量索引里面获取问题相关信息，加上提示词工程，生成回答。
- （下分支）结合RAG与Structured Data Pipeline，是 Databricks 特征工程处理流程，也是Databricks RAG最大的特点。

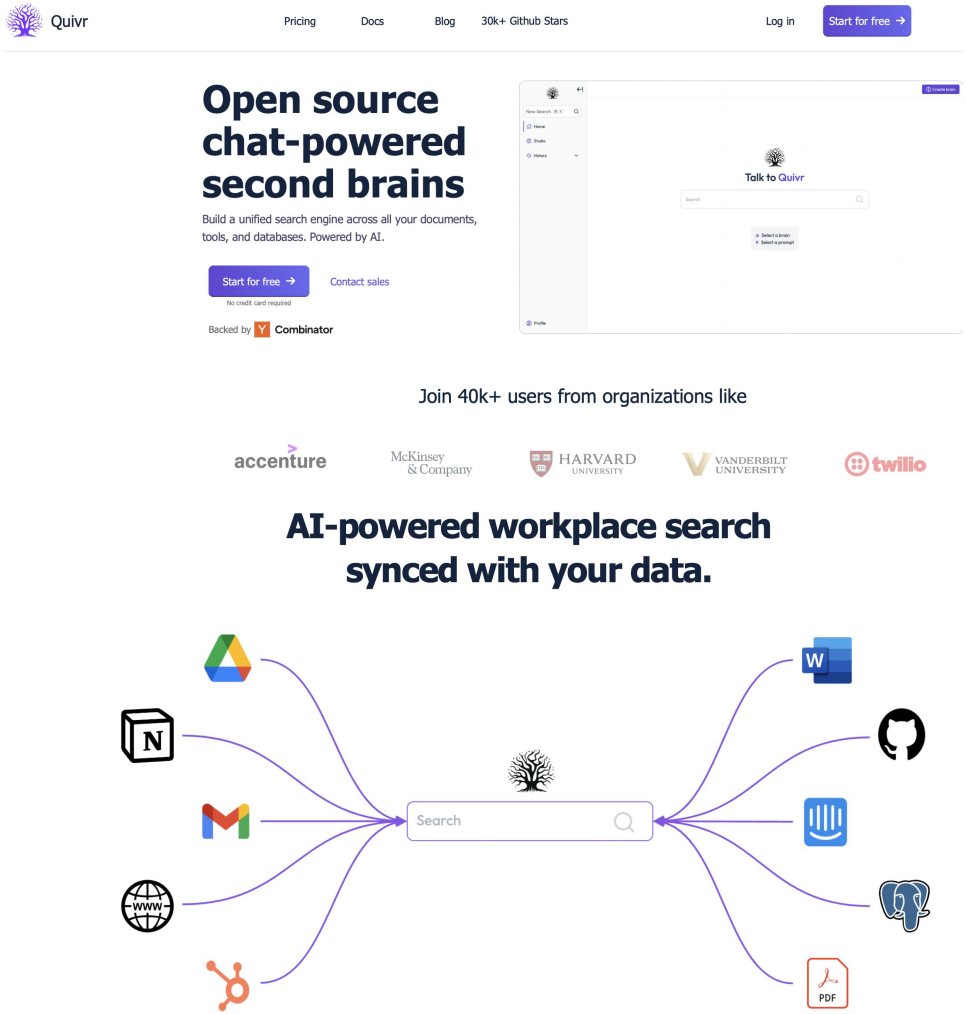


基于Databricks官网资料整理，不代表Databricks的实际方法

更多案例：个人知识助手

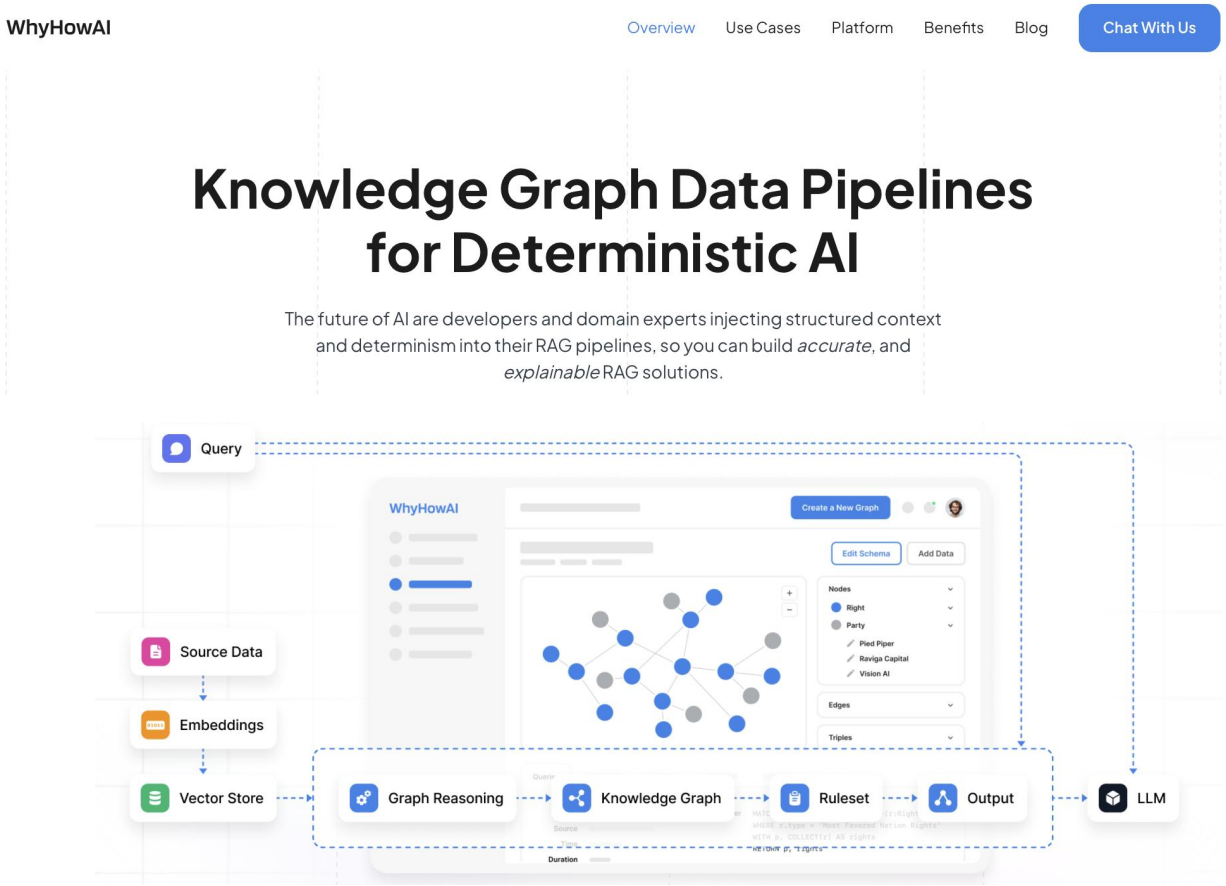
Quivr: Your Second Brain

整合多源文档语料



WhyHowAI

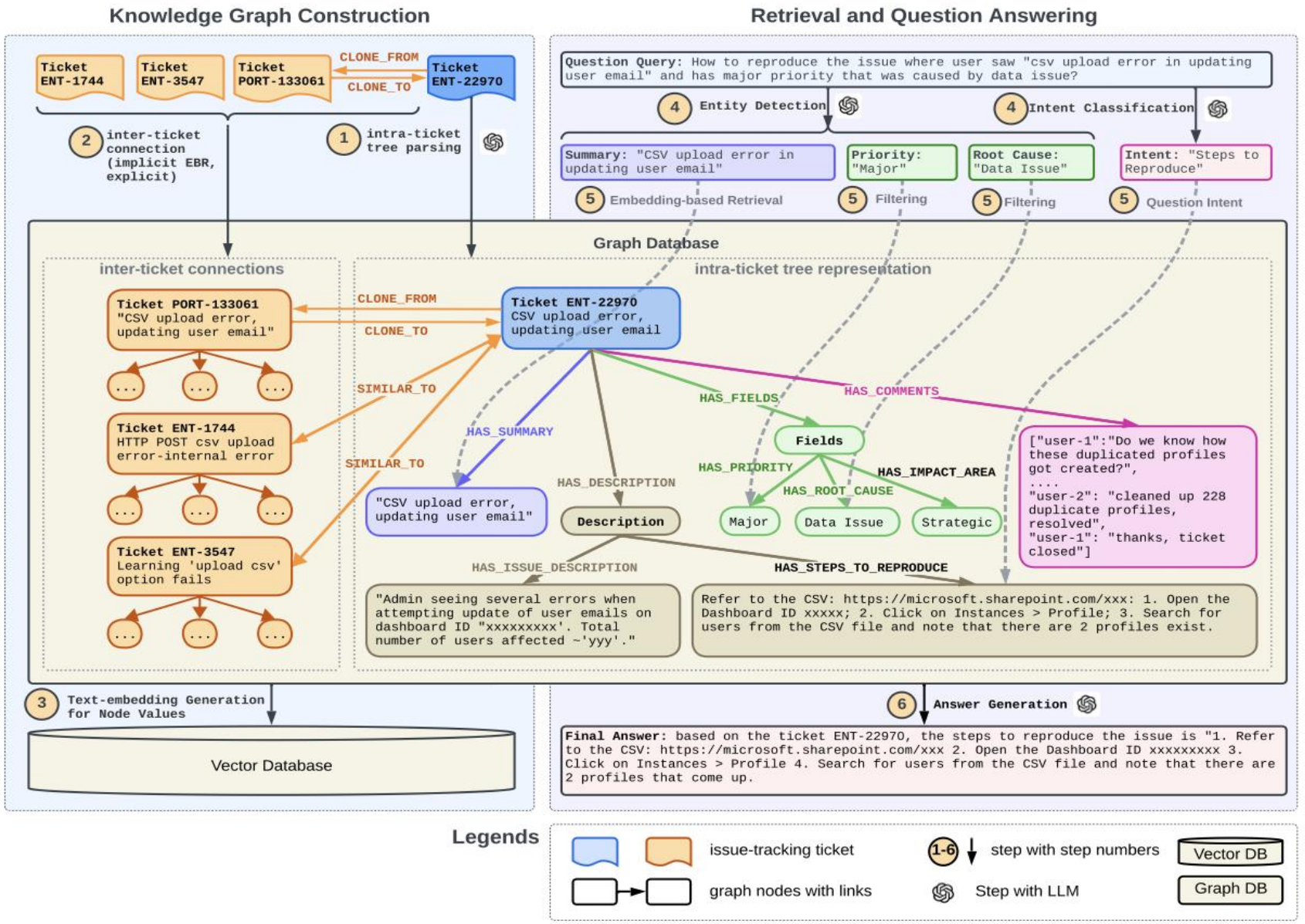
知识图谱组织多源知识



更多案例：LinkedIn 智能客服

KG+RAG
改善客户服务问答服务

- 从历史问题（JIRA工单）构建知识图谱，用于检索，并根据知识图谱中的相关子图生成答案。
- JIRA系统已知具有内置的层次结构和固有结构，使其更适合用于构建知识图谱。



Xu et.al., Retrieval-Augmented Generation with Knowledge Graphs for Customer Service Question Answering, 2024

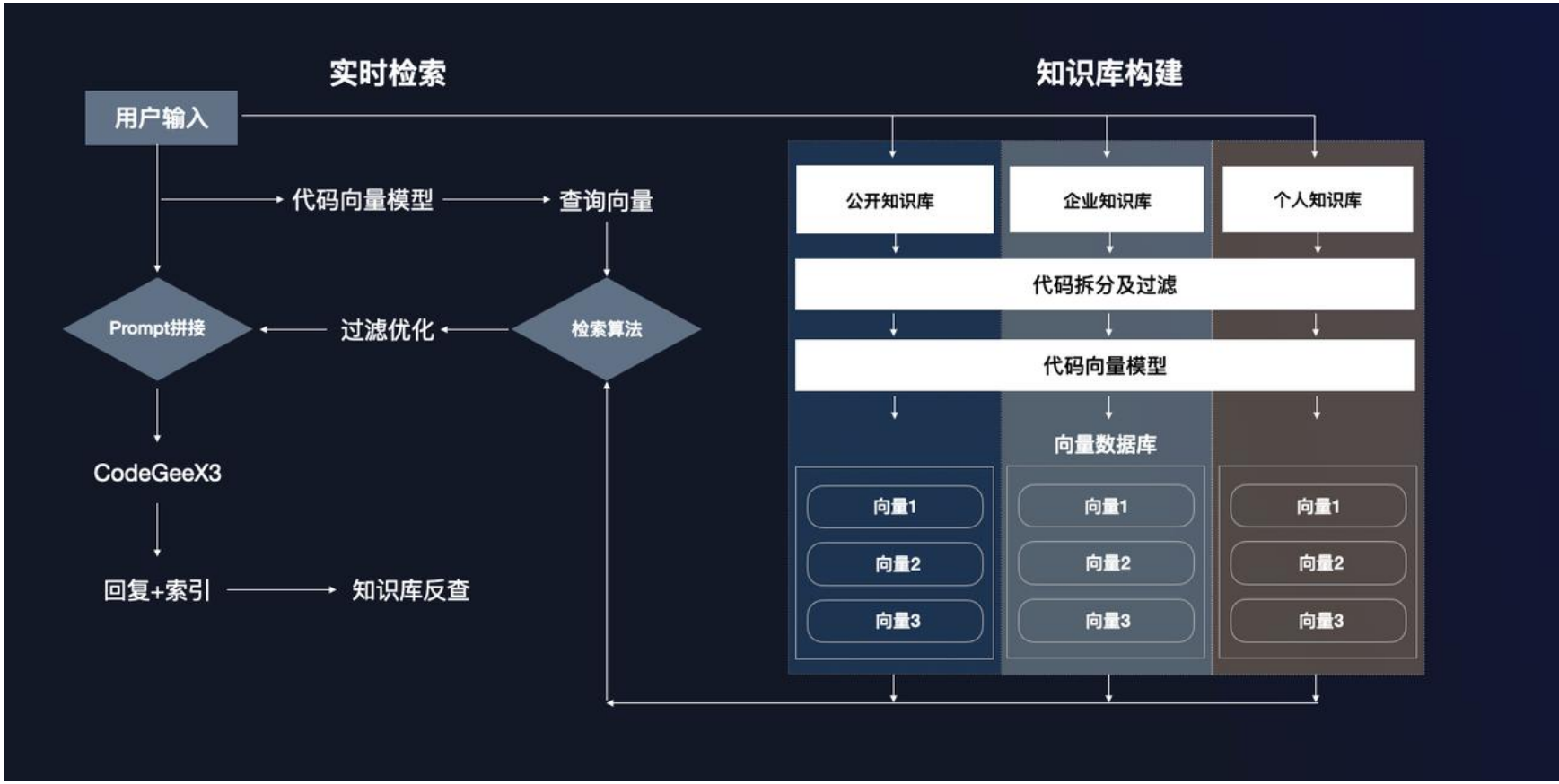
更多案例：软件工程

降低代码生成幻觉

CodeGeeX
A Multilingual Code Generation Model

- CodeGeeX 3代码模型引入RAG算法，构建流行公有仓库和私有仓库的代码向量数据库，缓解代码生成模型幻觉性问题。

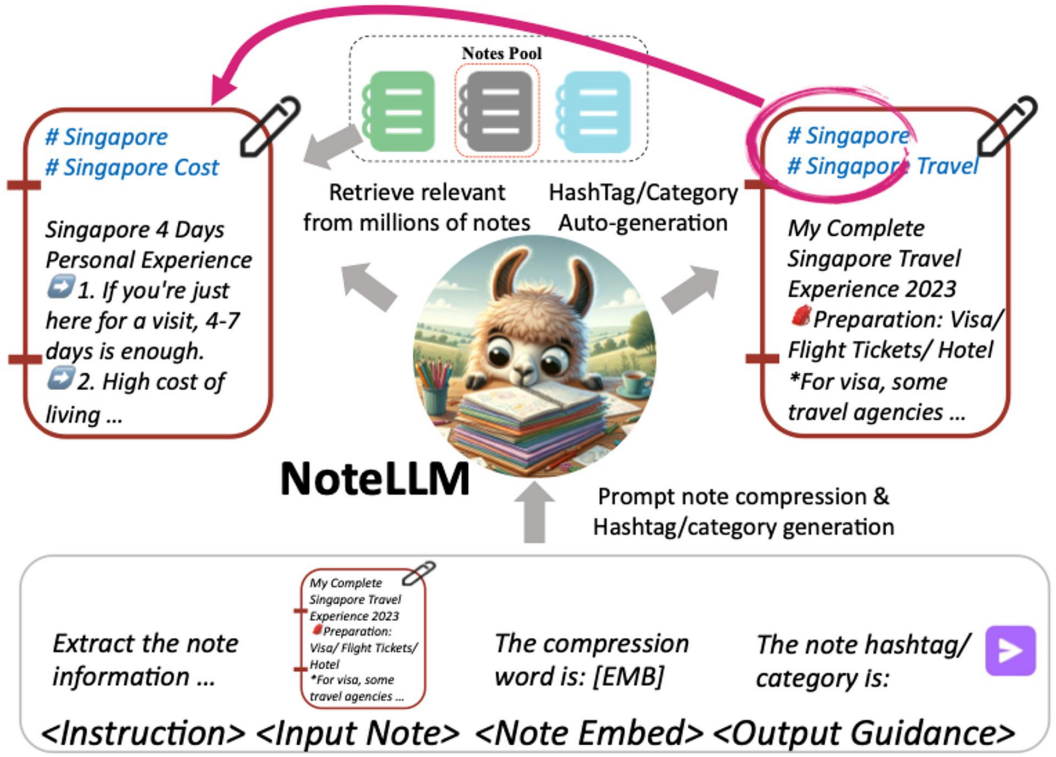
- 具体：避免生成错误的私有函数调用、让模型拥有最新的代码仓库知识、对私有代码仓库建立知识库等



更多案例：推荐系统

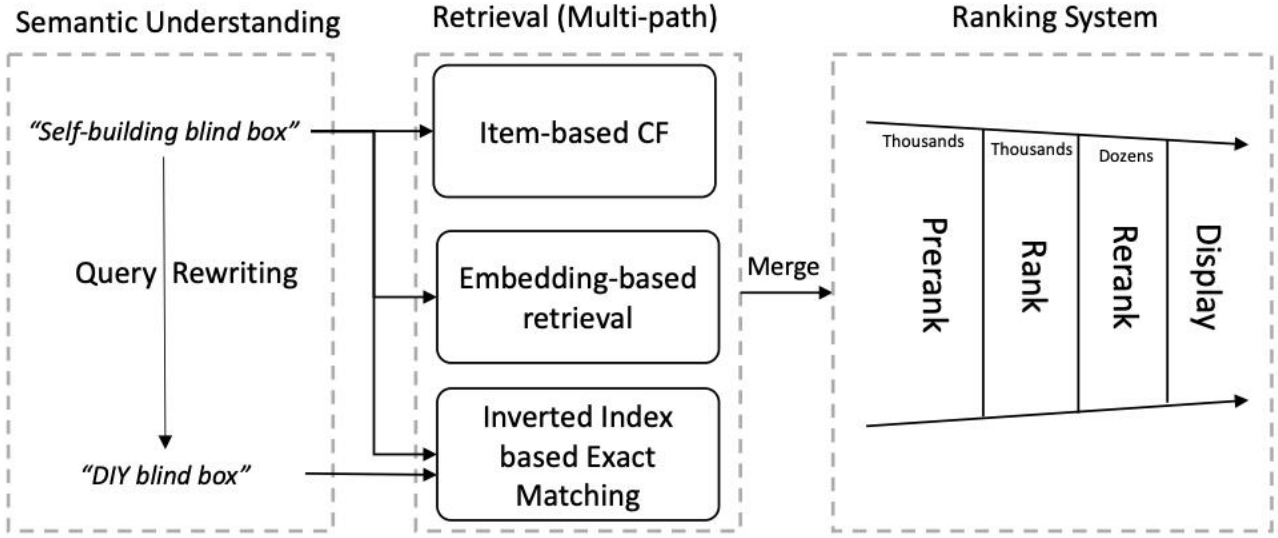
- 随着基于NL话界面的搜索和推荐系统以及分享社区的兴起，这些需求变得越来越重要，而传统RS支持不足
- 对推荐物品的文本内容进行编码检和索，适合Tex-Rich物品，例如笔记、攻略、指南

NoteLLM 小红书笔记推荐



BEQUE 淘宝搜索

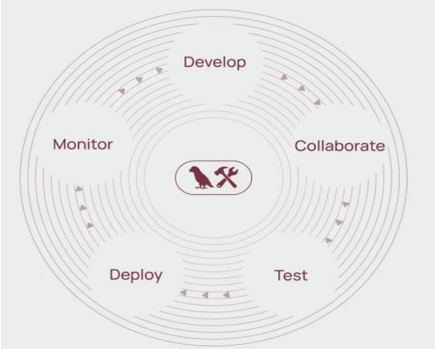
- 多指令监督微调：重写Query
- 离线反馈：评估重写和检索结果
- 目标对齐：根据离线对齐线上模型



Peng et al., Large Language Model based Long-tail Query Rewriting in Taobao Search, WWW, 2023
Zhang et al., NoteLLM: A Retrievable Large Language Model for Note Recommendation, WWW, 2024

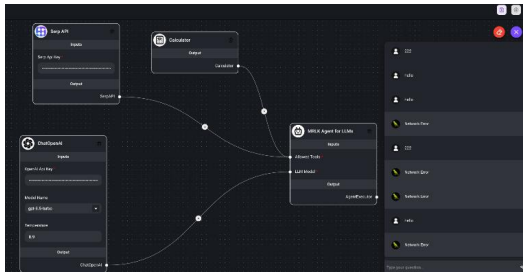
RAG 技术栈与工具

名称	优点	缺点
LangChain	模块化，功能全面	行为不一致并且隐藏细节 API复杂，灵活度低
LlamaIndex	专注知识检索	需组合使用，定制化程度低
FlowiseAI	上手简单，流程可视化	功能单一，不支持复杂场景
AutoGen	适配多智能体的场景	效率低，需要多轮对话

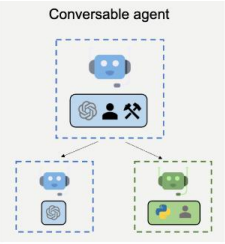


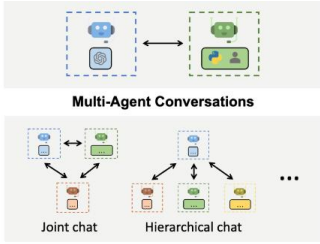
LangChain

LlamaIndex



FlowiseAI





AutoGen

- 用途**定制化**，满足蛮多样的需求
- 使用**简易化**，进一步降低上手门槛
- 功能**专业化**，逐渐面向生产环境

AnythingLLM

LangChain-Chatchat

RAGFlow

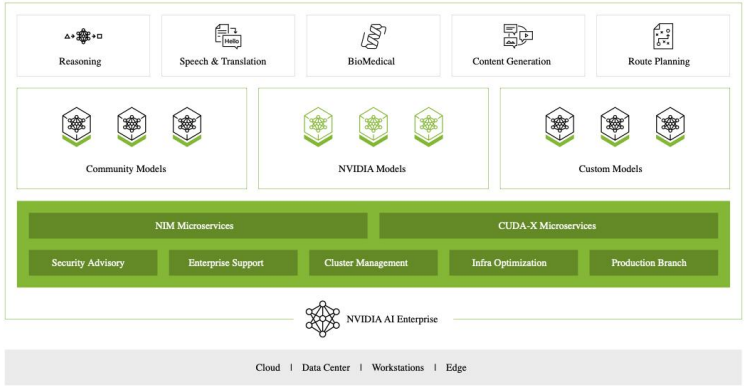
Anything

RAG 技术栈与工具

Databricks: GenAI

大数据

芯片

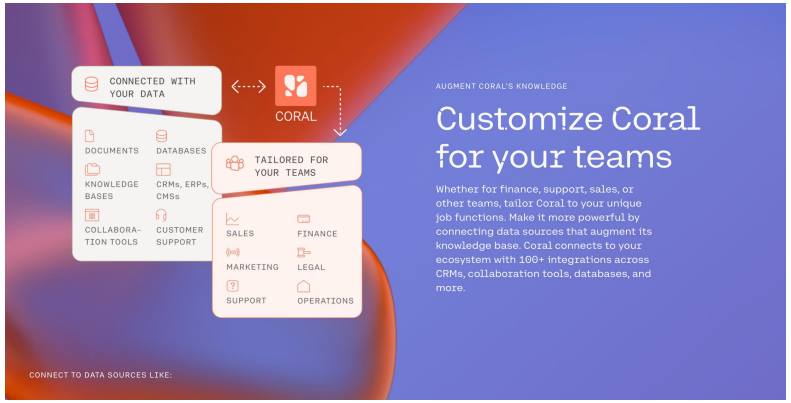


Nvidia: AI Enterprise

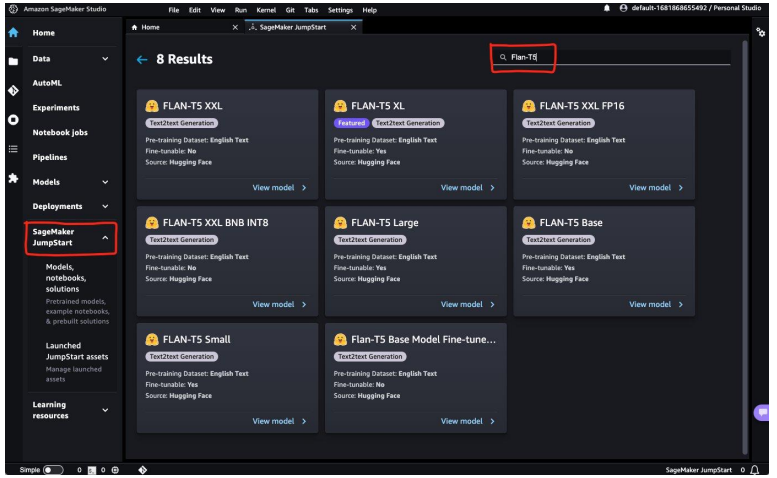
RAG

AI

云服务



Cohere: Coral



Amazon: Kendra

03 RAG 应用实践

3-1. RAG 技术平台实践 | 3-2. RAG 领域应用

垂域知识服务典型要求

私域文档&数据

建索引 + 检索 + 推理 + 生成

?

专业性知识服务

专业问答
法律、政务、医疗、科学

写作助手
新闻稿、研报、分析

场景

知识精准
知识完备
逻辑严谨
时间敏感
数值敏感

要求

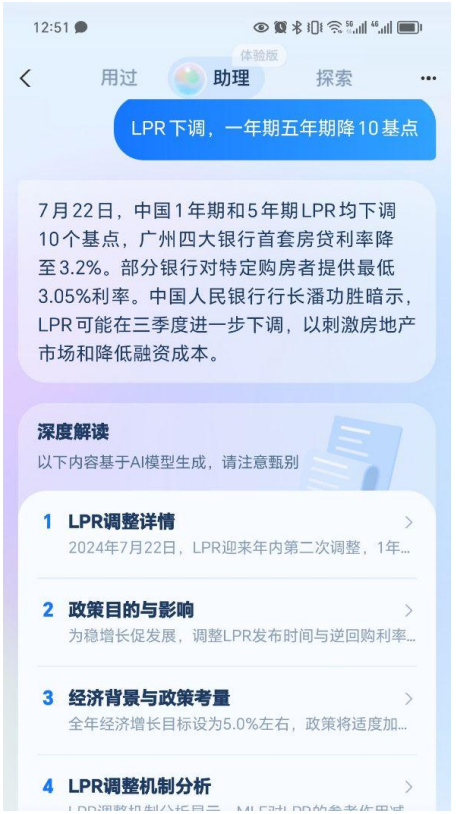
1.错误定性或错误逻辑　2 事实性错误或无依据　3. 时间、数值不敏感

4. 张冠李戴　5. 不能区分重要性　6. 语义不精准　7. 召回不完备

RAG 企业级应用场景

分析写作

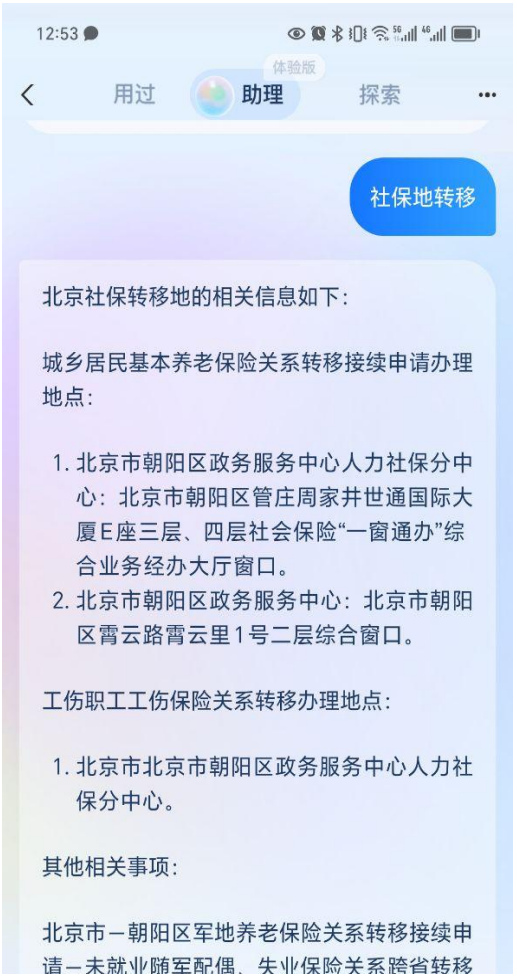
热点事件解读



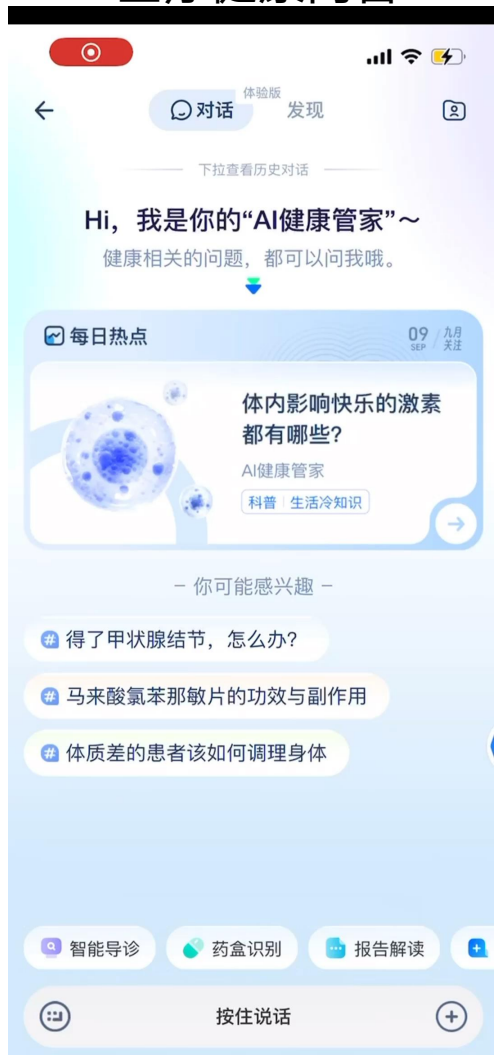
银行风险分析

知识问答

政务办事问答



医疗健康问答



RAG 企业级应用场景

保定市提取公积金需要哪些材料

购买自住住房提取住房公积金

不全，提取公积金的12种不同情况

生育险在哪里看

生育津贴支付

检索错误，参保人员参保信息查询

	有果 准确率	召回率
基本RAG	0.55	0.37
知识增强RAG	0.91	0.71

舟山市怎么查房产证

未找到相关信息

遗漏，不动产权属证明网上查询

社保月缴费多少

职工参保登记

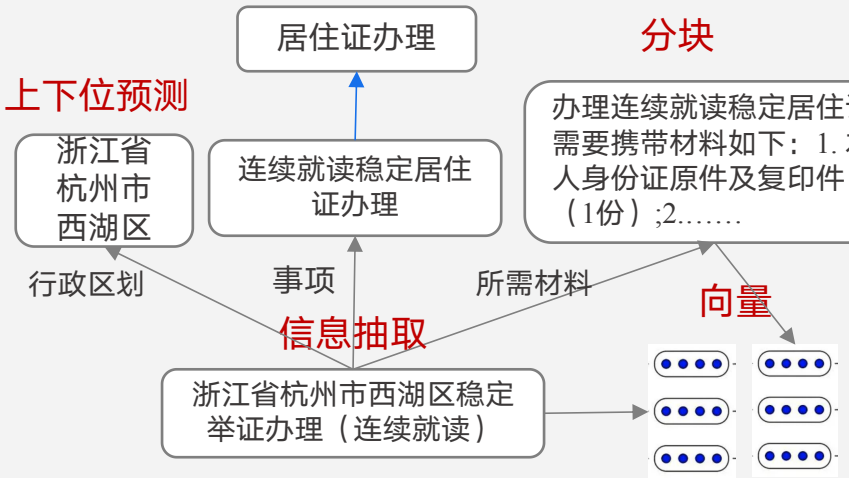
检索错误，没有事项

我买房申请了500万公积金贷款，计划20年还清，当前月利率是0.02，每个月应该还多少钱

已知公积金贷款月供计算公式为[贷款本金×月利率×(1+月利率)^还款月数]÷[(1+月利率)^还款月数-1]
算式：500万×0.02×（1+0.02）^(20×12)÷[(1+0.02)^(20×12)-1

RAG 企业级应用场景

离线索引



UIE+概念推理+向量化 「建立互索引」

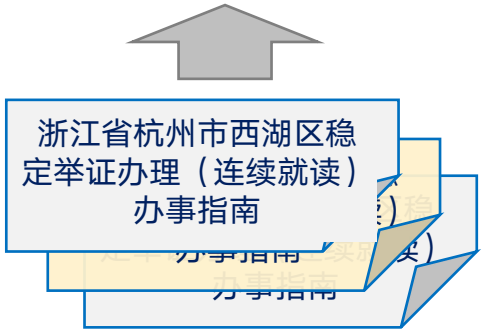
在线1: 查询改写

在读博士申请居住证是否需要学校开具在读证明

严谨规划：将问题拆解为多跳符号化的逻辑表式
逻辑形式 = F(查询 + 图模式)

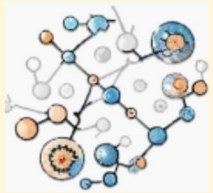
1. 查找博士办理居住证的办事指南
get_spo(s=s1:在读博士申请居住证|政务办事,p=p1:材料证明，o=o1:材料证明)
2. 判断材料中是否需要在读证明
verify(left_expr=o1, right_expr=在读证明，op=包含)

用户查询



政府官网办事指南

在线2: 检索



向量检索

连续就读

语义检索

在读博士

文本检索

S:在读博士申请居住证

P:材料证明

语义推理增强检索：查询和知识库分布差异

在线3: 生成

数学：Count/sum/sort/calculate/……

自然语言推断：has_property/isa/ entailment

Generator

答案生成

执行 「执行器+生成器」提升答案可信度

RAG 与推理结合的场景

兼顾了外部知识依赖和复杂推理的需求，使得LLM更加胜任复杂的真实场景。

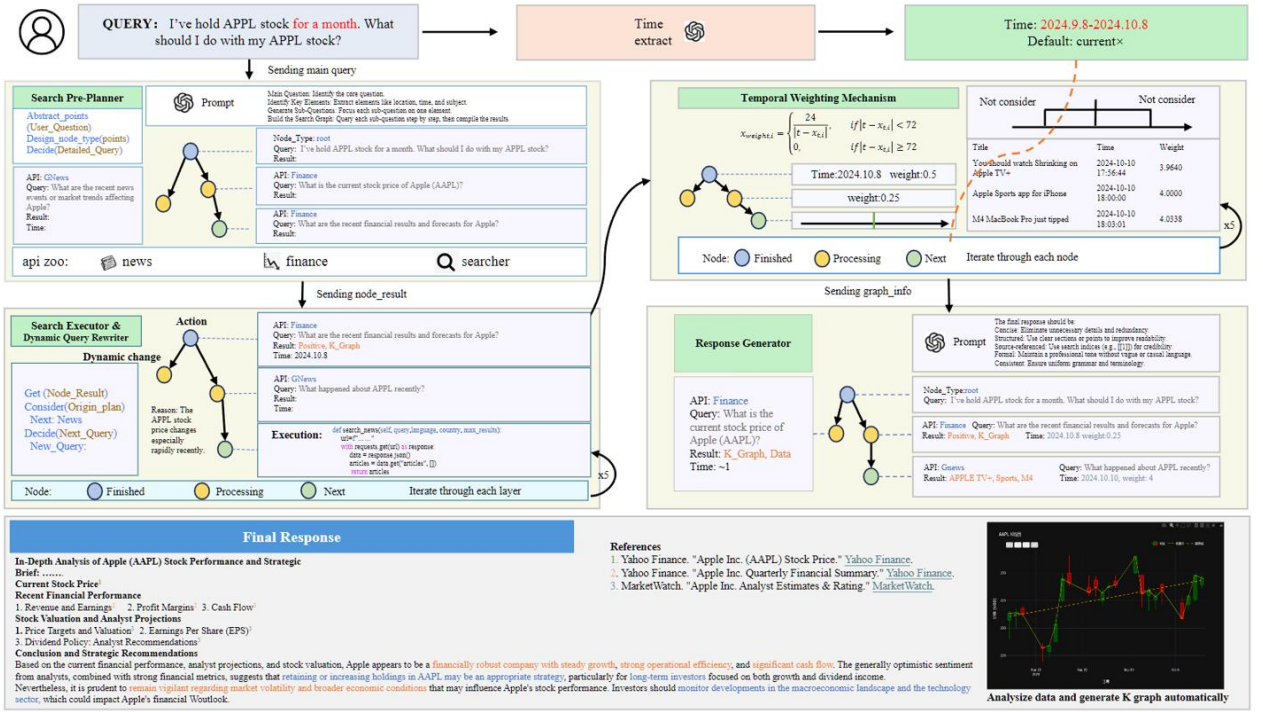
金融：FinSearch

- 金融：复杂推理任务+高频更新的外部专业知识

动态自适应搜索机制

- 分步查询规划：通过 LLM 将复杂金融问题分解为子查询，构建DAG，明确逻辑依赖关系。
- 实时查询优化：在搜索执行过程中，基于中间结果动态调整后续子查询，提升市场动态响应能力。

时间敏感加权机制: 引入 72 小时时间窗口的线性衰减函数，优先处理近期信息（如新闻、股价波动），强化时间相关性。



Method	Accuracy (%)					Time (s/answer)				
	GPT-4o	Llama3.1-405B	Claude3.5-Sonnet	Deepseek	Gemini-1.5-Flash	GPT-4o	Llama3.1-405B	Claude3.5-Sonnet	Deepseek	Gemini-1.5-Flash
Baseline	36.13 ±1.22	38.13 ±1.27	38.60 ±1.28	34.07 ±1.21	35.47 ±1.22	3.87 ±0.15	4.96 ±0.25	6.66 ±0.21	14.50 ±0.27	5.04 ±0.16
SearchAgent	46.33 ±1.30	43.80 ±1.24	41.87 ±1.29	44.27 ±1.26	42.33 ±1.28	**1.58** ±0.06	**1.61** ±0.07	**1.54** ±0.06	**1.32** ±0.04	**1.58** ±0.04
MindSearch	52.40 ±1.33	53.07 ±1.34	53.60 ±1.28	49.73 ±1.32	51.53 ±1.29	19.09 ±0.39	14.82 ±0.45	17.93 ±0.39	27.01 ±0.58	20.14 ±0.43
Perplexity Pro	60.27 ±1.26	61.47 ±1.28	56.67 ±1.24	–	–	6.12 ±0.26	3.94 ±0.15	5.85 ±0.22	–	–
Ours	**76.20** ±1.12	**75.53** ±1.08	**78.27** ±1.07	**72.33** ±1.15	**74.87** ±1.08	16.03 ±0.43	14.55 ±0.47	18.15 ±0.35	29.31 ±0.70	17.74 ±0.53

上图展示了不同智能体方法在金融基准FinSearchBench-24上的性能和计算效率的对比，其中本文的方法取得了**更高的准确率**与**更低的计算开销**

An Agent Framework for Real-Time Financial Information Searching with Large Language Models, 2024.

RAG 与推理结合的场景

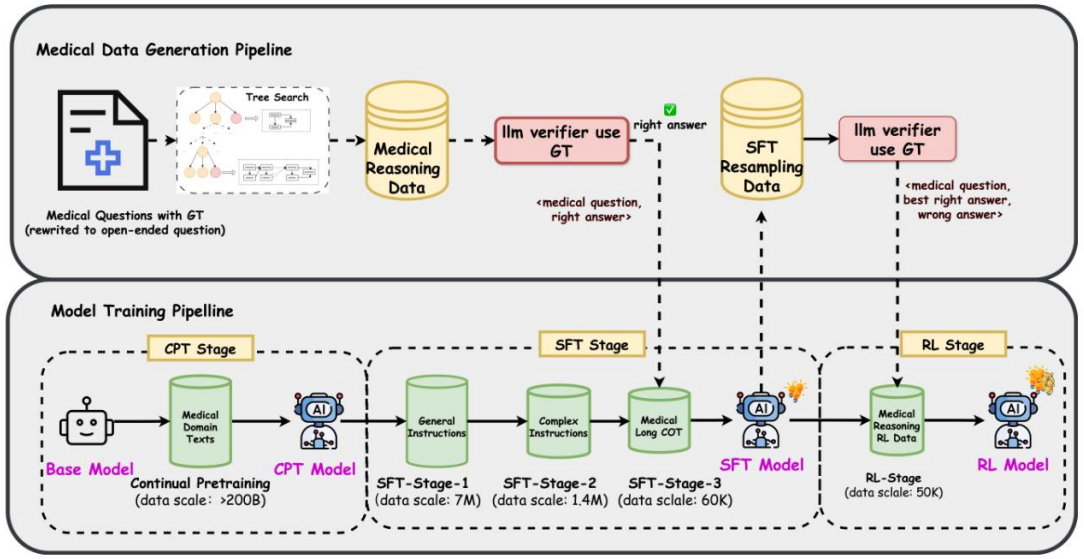
医疗: Citrus

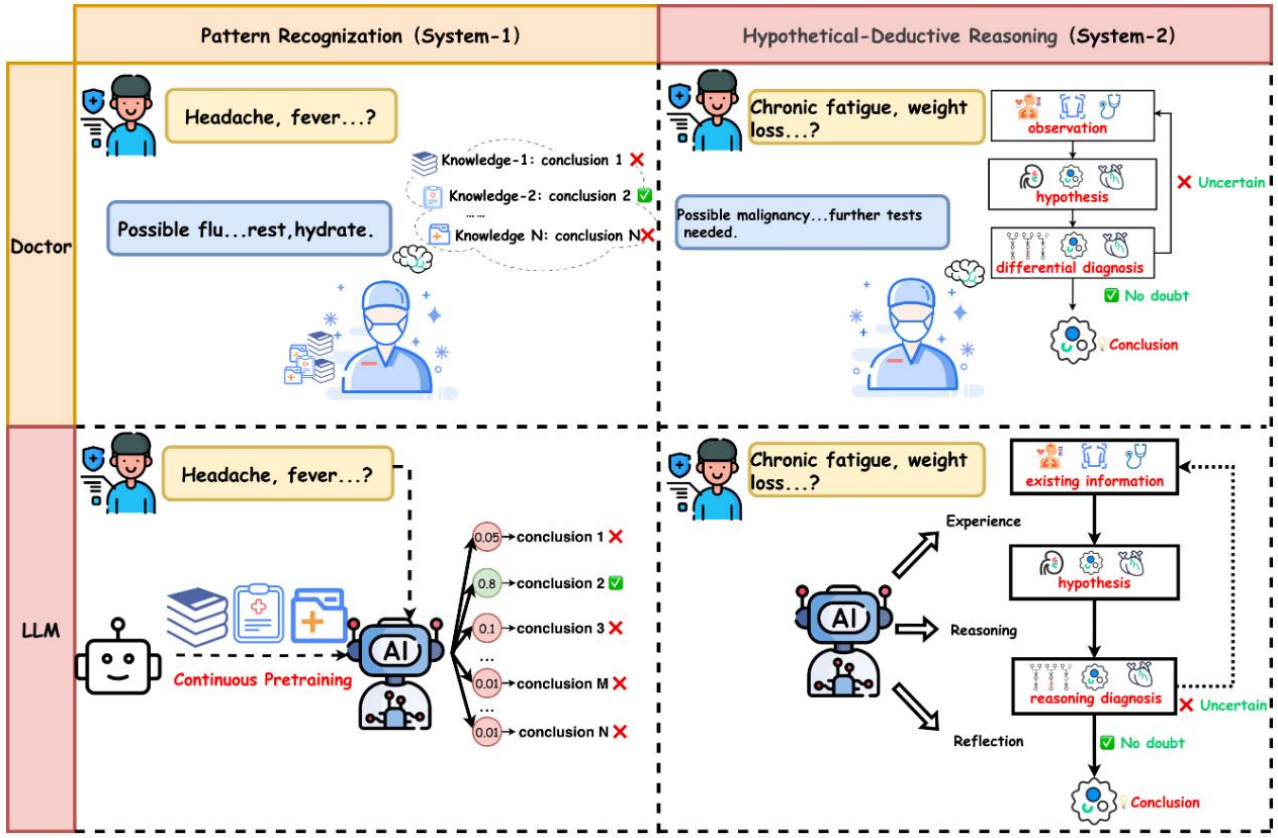
- 医疗：复杂推理任务+强领域知识+低幻觉容忍

模拟专家认知路径：提出结合 “假设 - 演绎推理” 和 “模式识别” 的双专家推理框架，模仿医生的临床决策过程。

双专家推理机制：

- 推理专家：生成假设 - 演绎推理链。
- 反思专家：基于真实答案验证推理逻辑，过滤无效步骤。



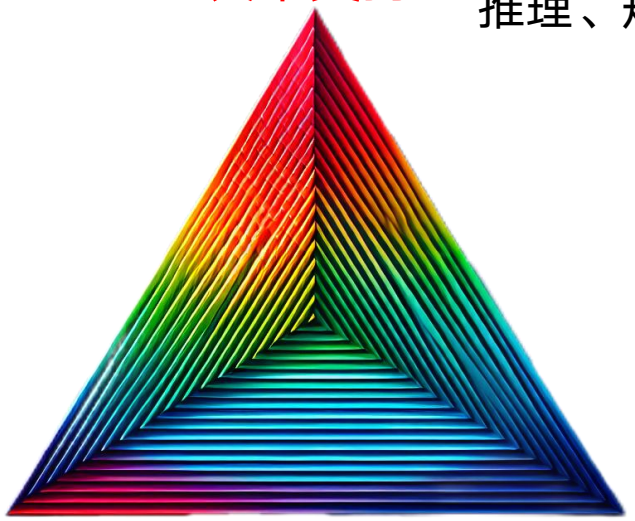


通过SFT和RL的训练，LLMs 表现出**与医学专家相似的临床决策路径**

Citrus: Leveraging Expert Cognitive Pathways in a Medical Language Model for Advanced Medical Decision Support, 2025.

04 总结与展望

新一代 RAG 呼之欲出



决策支持

Reasoning:
推理、规划、溯源、校验

知识库

结构化组织（Graph化）

业务流程

服务算子编排（Bussiness）

New RAG = Reasoning+ Graph +Business ？

开放问题：

1. 趋势：三者的组合是否能够构成下一代的RAG系统？

2. 技术：下一代RAG有什么挑战？

3. 场景：率先大规模落地的杀手级应用会是？

ขอบคุณ 谢谢

感谢 Thanks

Terima kasih ありがとう