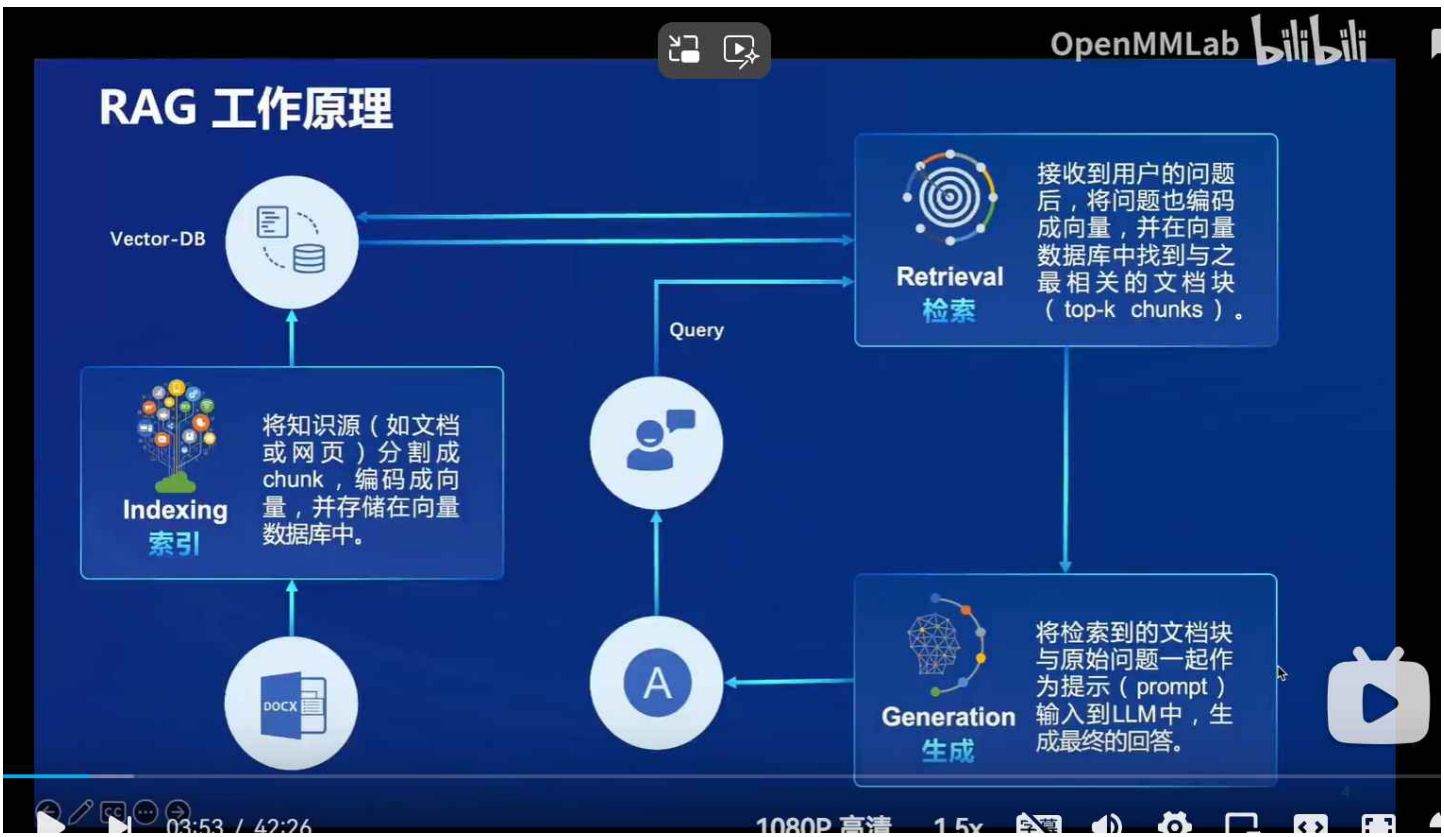
书生二期第三课笔记


HuixiangDou
上海人工智能实验室
Shanghai Artificial Intelligence Laboratory
茴香豆：搭建你的RAG智能助理
北 辰 | 书生·浦语社区贡献者


目录
RAG：
RAG 是什么、原理、
RAG vs. Fine-tune、架构、向量数据库、评估和测试
HuixiangDou
茴香豆介绍、茴香豆的特点、
架构、构建步骤
实践演示：
茴香豆 Web 版演示
Intern Studio 部署茴香豆知识助手


RAG 技术概述
定义
RAG（Retrieval Augmented Generation）是一种结合了检索（Retrieval）和生成（Generation）的技术，旨在通过利用外部知识库来增强大型语言模型（LLMs）的性能。它通过检索与用户输入相关的信息片段，并结合这些信息来生成更准确、更丰富的回答。
知识
解决LLMs在处理知识密集型任务时可能遇到的挑战。提供更准确的回答、降低成本、实现外部记忆。
VS
生成幻觉（hallucination）
过时知识
缺乏透明和可追溯的推理过程
应用
问答系统
文本生成
信息检索
图片描述


OpenMMLab
bilibili
RAG 工作原理
Vector-DB
Retrieval
检索
接收到用户的问题后，将问题也编码成向量，并在向量数据库中找到与之最相关的文档块（top-k chunks）。
Query
Indexing
索引
将知识源（如文档或网页）分割成chunk，编码成向量，并存储在向量数据库中。
DOCX
A
Generation
生成
将检索到的文档块与原始问题一起作为提示（prompt）输入到LLM中，生成最终的回答。
03:53 / 42:26
1080P 高清
1.5x

向量数据库（Vector-DB）

数据存储

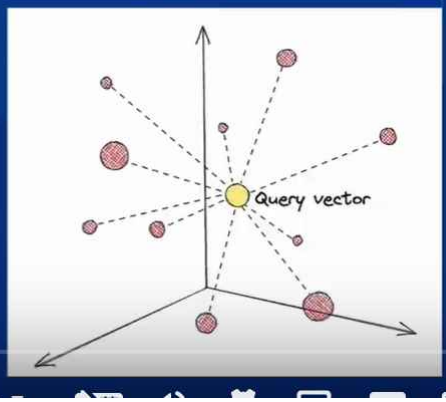
将文本及其他数据通过其他预训练的模型转换为固定长度的向量表示，这些向量能够捕捉文本的语义信息。

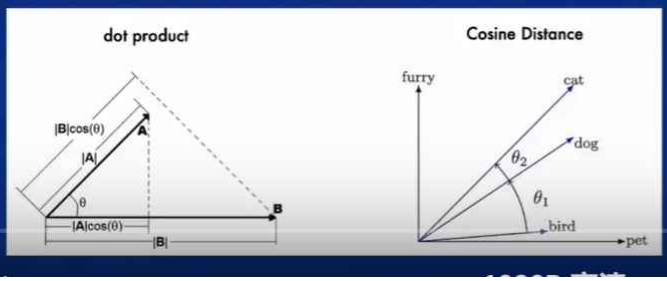
相似性检索

根据用户的查询向量，使用向量数据库快速找出最相关的向量的过程。通常通过计算余弦相似度或其他相似性度量来完成。
检索结果根据相似度得分进行排序，最相关的文档将被用于后续的文本生成。

向量表示的优化

包括使用更高级的文本编码技术，如句子嵌入或段落嵌入，以及对数据库进行优化以支持大规模向量搜索。

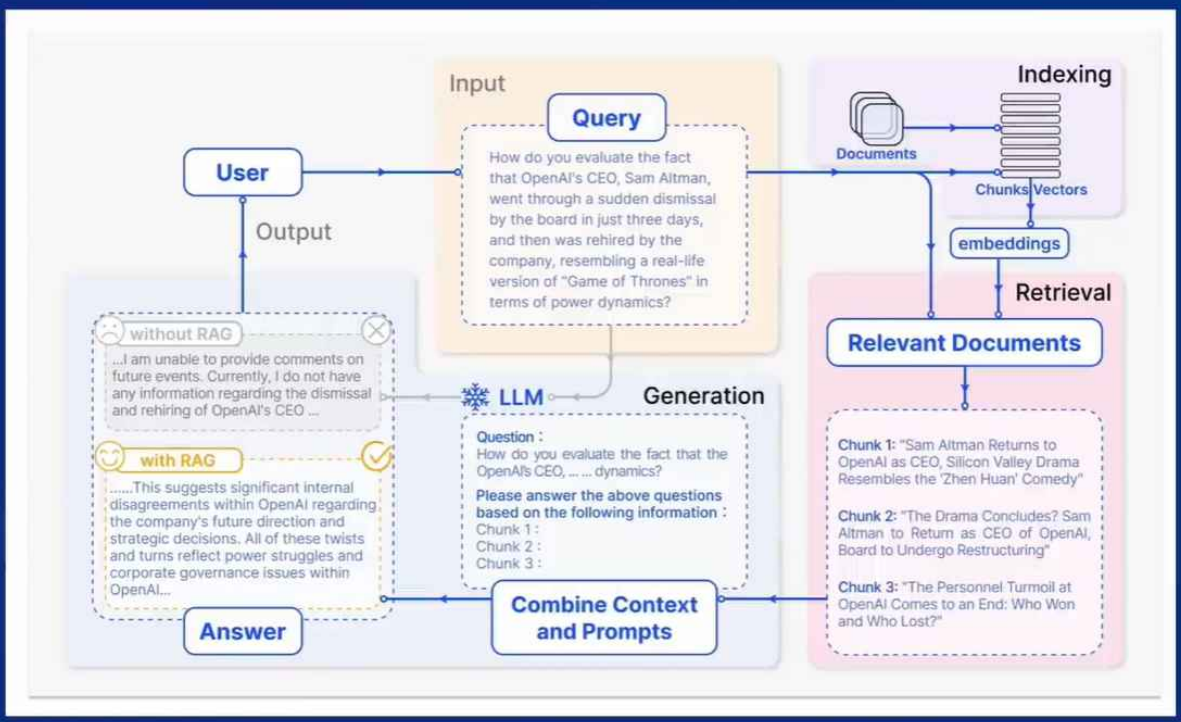




Images from : https://github.com/chenzomi12/AISystem/blob/main/06Foundation/05Dataset/04VectorDB/02VectorDB.pdf

针对向量的优化，也影响了RAG的效果。

RAG 流程示例



Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., Wang, H., 2024. Retrieval-Augmented Generation for Large Language Models: A Survey.

RAG 发展进程

RAG的概念最早是由Meta（Facebook）的Lewis等人在2020《Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks》中提出的。

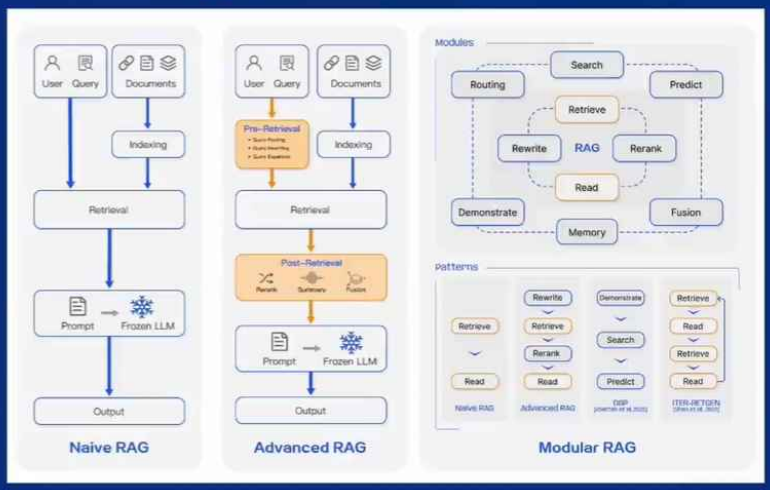


Image from: Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., Wang, H., 2024. Retrieval-Augmented Generation for Large Language Models: A Survey.

RAG 常见优化方法

嵌入优化
Embedding Optimization

- ✓ 结合稀疏和密集检索
- ✓ 多任务

索引优化
Indexing Optimization

- ✓ 细粒度分割（Chunk）
- ✓ 元数据

查询优化
Query Optimization

- ✓ 查询扩展、转换
- ✓ 多查询

上下文管理
Context Curation

- ✓ 重排（rerank）
- ✓ 上下文选择/压缩

RAG 常见优化方法

嵌入优化
Embedding Optimization

- ✓ 结合稀疏和密集检索
- ✓ 多任务

索引优化
Indexing Optimization

- ✓ 细粒度分割（Chunk）
- ✓ 元数据

查询优化
Query Optimization

- ✓ 查询扩展、转换
- ✓ 多查询

上下文管理
Context Curation

- ✓ 重排（rerank）
- ✓ 上下文选择/压缩

迭代检索
Iterative Retrieval

- ✓ 根据初始查询和迄今为止生成的文本进行重复搜索

递归检索
Recursive Retrieval

- ✓ 迭代细化搜索查询
- ✓ 链式推理（Chain-of-Thought）指导检索过程

自适应检索
Adaptive Retrieval

- ✓ Flare，Self-RAG
- ✓ 使用LLMs主动决定检索的最佳时机和内容

RAG 常见优化方法

嵌入优化
Embedding Optimization

- ✓ 结合稀疏和密集检索
- ✓ 多任务

索引优化
Indexing Optimization

- ✓ 细粒度分割（Chunk）
- ✓ 元数据

查询优化
Query Optimization

- ✓ 查询扩展、转换
- ✓ 多查询

上下文管理
Context Curation

- ✓ 重排（rerank）
- ✓ 上下文选择/压缩

迭代检索
Iterative Retrieval

- ✓ 根据初始查询和迄今为止生成的文本进行重复搜索

递归检索
Recursive Retrieval

- ✓ 迭代细化搜索查询
- ✓ 链式推理（Chain-of-Thought）指导检索过程

自适应检索
Adaptive Retrieval

- ✓ Flare，Self-RAG
- ✓ 使用LLMs主动决定检索的最佳时机和内容

LLM微调
LLM Fine-tuning

- ✓ 检索微调
- ✓ 生成微调
- ✓ 双重微调

RAG vs. 微调（Fine-tuning）

RAG

- 非参数记忆，利用外部知识库提供实时更新的信息。
- 能够处理知识密集型任务，提供准确的事实性回答。
- 通过检索增强，可以生成更多样化的内容。

适用场景

适用于需要结合最新信息和实时数据的任务：开放域问答、实时新闻摘要等。

优势：动态知识更新，处理长尾知识问题。

局限：依赖于外部知识库的质量和覆盖范围。依赖大模型能力。

Fine-tuning

- 参数记忆，通过在特定任务数据上训练，模型可以更好地适应该任务。
- 通常需要大量标注数据来进行有效微调。
- 微调后的模型可能过拟合，导致泛化能力下降。

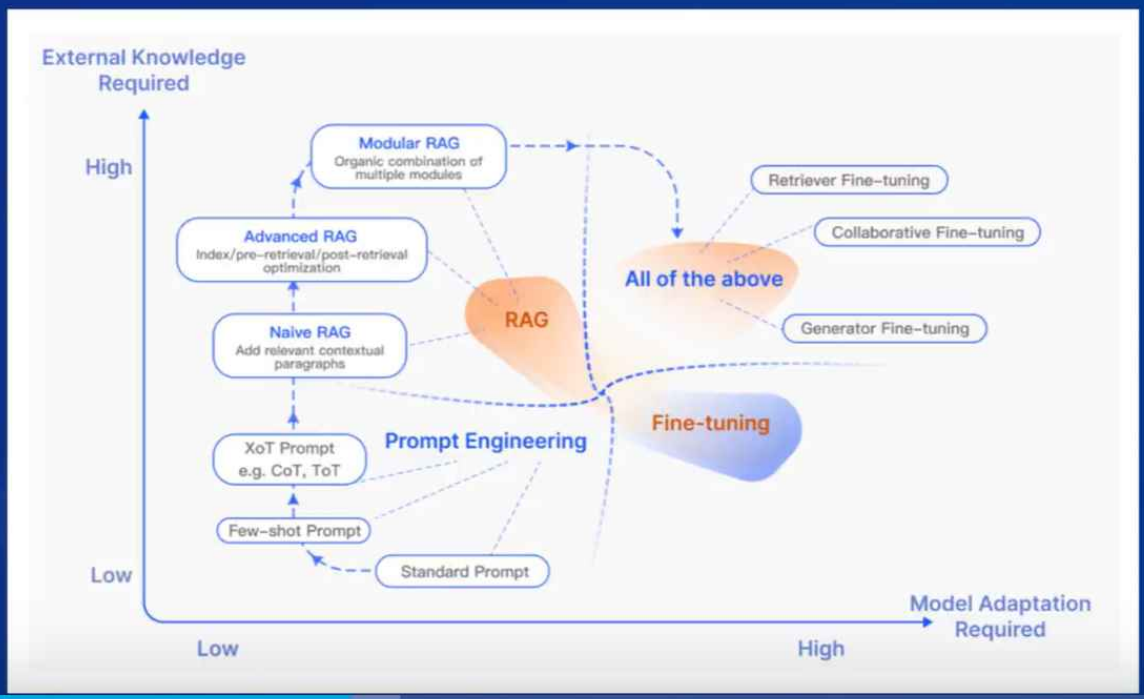
适用场景

适用于数据可用且需要模型高度专业化的任务，如特定领域的文本分类、情感分析、文本生成等。

优势：模型性能针对特定任务优化。

局限：需要大量的标注数据，且对新任务的适应性较差。

LLM 模型优化方法比较



Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., Wang, H., 2024. Retrieval-Augmented Generation for Large Language Models: A Survey.

评估框架和基准测试

经典评估指标：

- 准确率（Accuracy）
- 召回率（Recall）
- F1分数（F1 Score）
- BLEU分数（用于机器翻译和文本生成）
- ROUGE分数（用于文本生成的评估）

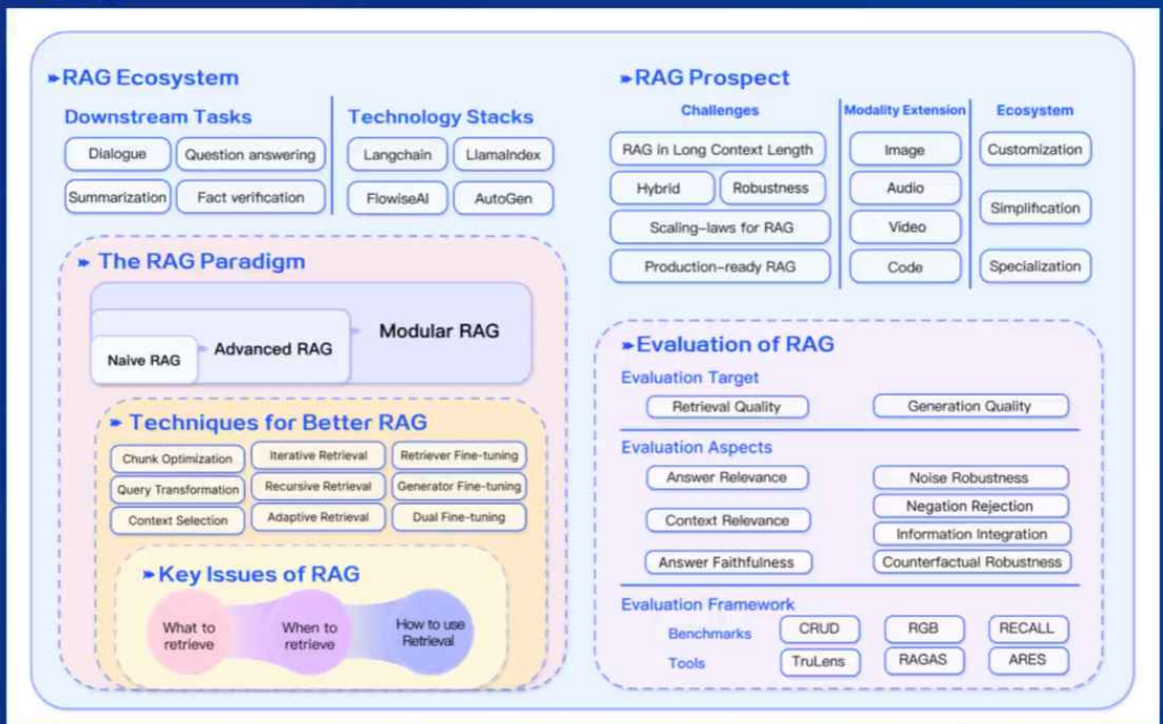
RAG 评测框架：

- 基准测试 - RGB、RECALL、CRUD
- 评测工具- RAGAS、ARES、TruLens

Evaluation Framework	Evaluation Targets	Evaluation Aspects	Quantitative Metrics
RGB†	Retrieval Quality Generation Quality	Noise Robustness Negative Rejection Information Integration Counterfactual Robustness	Accuracy EM Accuracy Accuracy
RECALL†	Generation Quality	Counterfactual Robustness	R-Rate (Reappearance Rate)
RAGAS‡	Retrieval Quality Generation Quality	Context Relevance Faithfulness Answer Relevance	* * Cosine Similarity
ARES‡	Retrieval Quality Generation Quality	Context Relevance Faithfulness Answer Relevance	Accuracy Accuracy Accuracy
TruLens‡	Retrieval Quality Generation Quality	Context Relevance Faithfulness Answer Relevance	* * *
CRUD†	Retrieval Quality Generation Quality	Creative Generation Knowledge-intensive QA Error Correction Summarization	BLEU ROUGE-L BertScore RAGQuestEval

Image from: Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., Wang, H., 2024. Retrieval-Augmented Generation for Large Language Models: A Survey.

RAG 总结



Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., Wang, H., 2024. Retrieval-Augmented Generation for Large Language Models: A Survey.

茴香豆介绍

茴香豆是一个基于LLMs的领域知识助手，由书生浦语团队开发的开源大模型应用。

- 专为即时通讯（IM）工具中的群聊场景优化的工作流，提供及时准确的技术支持和自动化问答服务。
- 通过应用检索增强生成（RAG）技术，茴香豆能够理解和高效准确的回应与特定知识领域相关的复杂查询。

应用场景

- 智能客服：技术支持、领域知识对话
- IM工具中创建用户群组，讨论、解答相关的问题。
- 随着用户数量的增加，答复内容高度重复，充斥大量无意义和闲聊，人工回复，成本高，影响工作效率。
- 茴香豆通过提供自动化的问答支持，帮助维护者减轻负担，同时确保用户问题得到有效解答。

场景难点

- 群聊中的信息量巨大，且内容多样，从技术讨论到闲聊应有尽有。
- 用户问题通常与个人紧密相关，需要准确的实时的专业知识解答。
- 传统的NLP解决方案无法准确解析用户意图，且往往无法提供满意的答案。
- 需要一个能够在群聊中准确识别与回答相关问题的智能助手，同时避免造成消息过载。

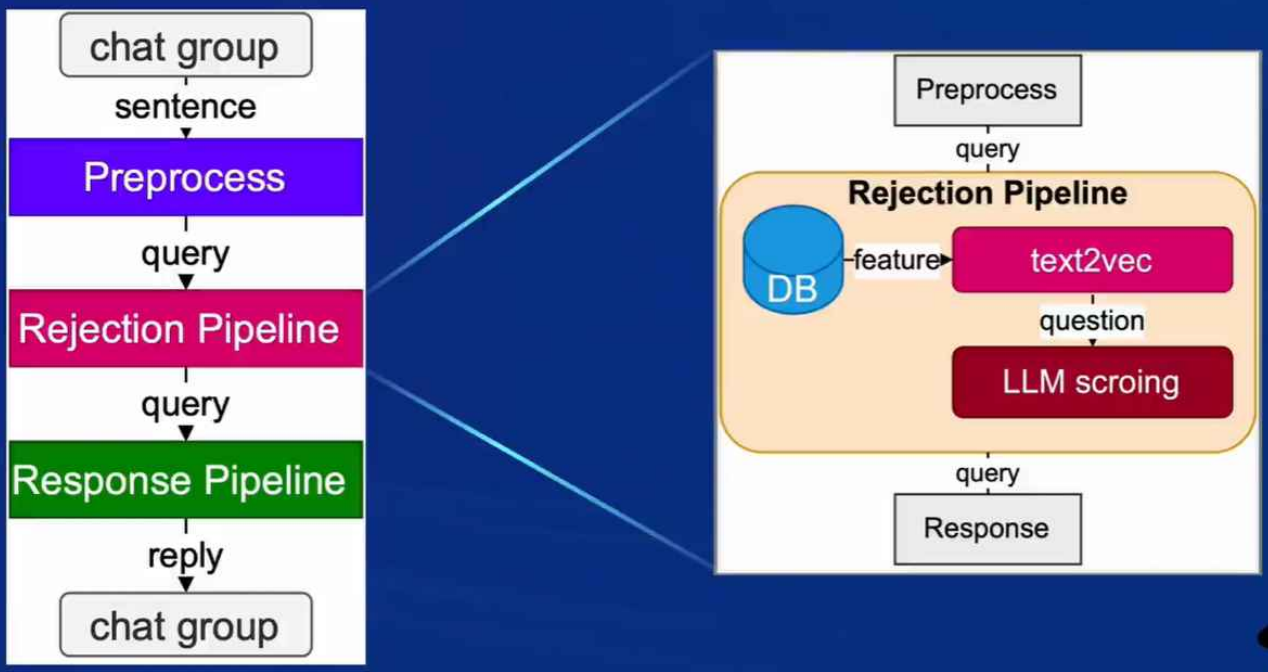
茴香豆构建

HuixiangDou

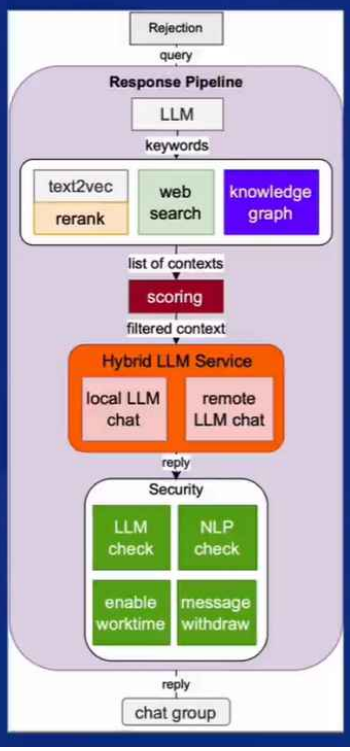
前端

LLM 后端

茴香豆的工作流



茴香豆完整工作流



多来源检索

- ✓ 向量数据库
- ✓ 网络搜索结果
- ✓ 知识图谱

混合大模型

- ✓ 本地LLM
- ✓ 远程LLM

多重评分拒答工作流

- ✓ 回答有效
- ✓ 避免信息泛滥

安全检查

- ✓ 多种手段
- ✓ 确保回答合规