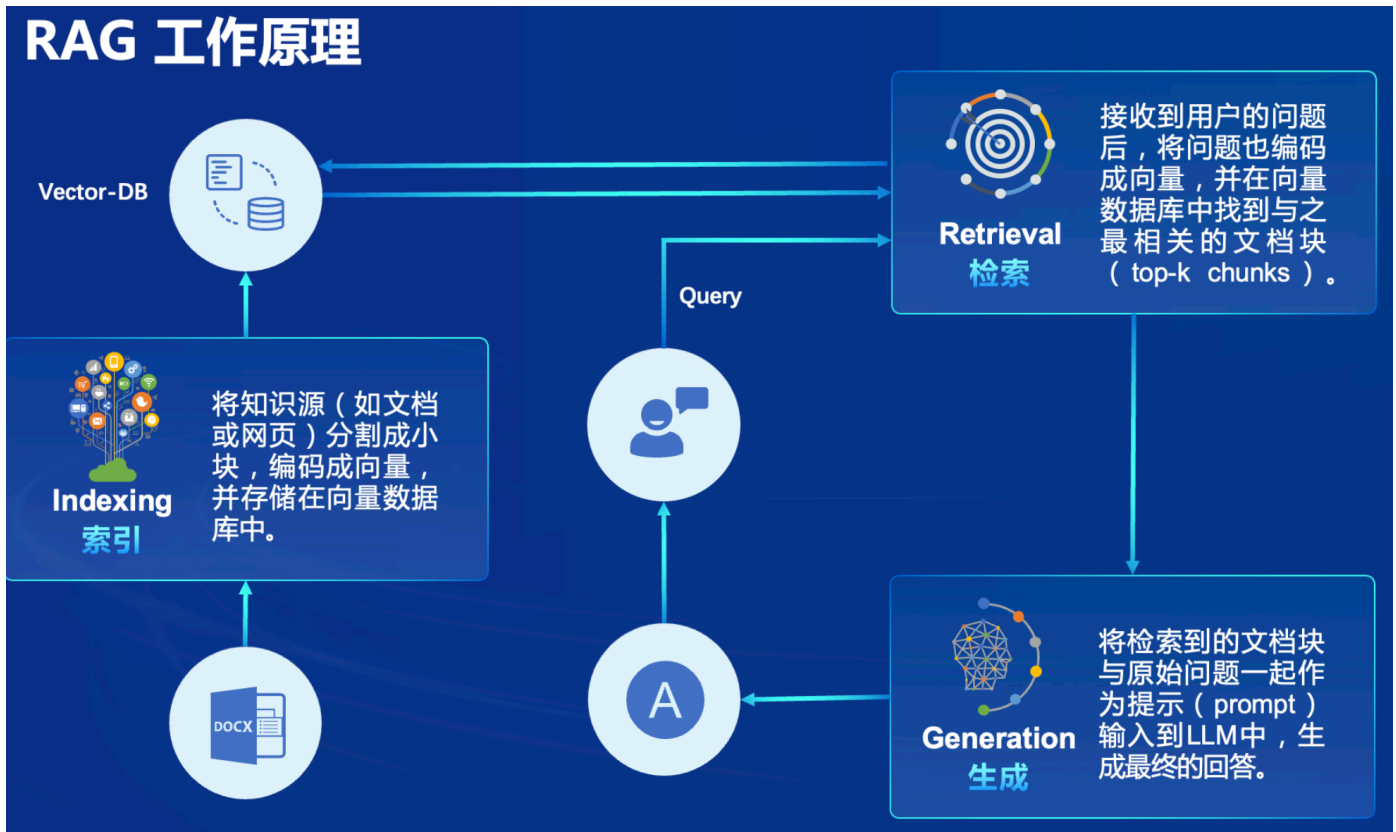


第三节笔记

RAG优势

解决 LLMs 在处理知识密集型任务时可能遇到的挑战, 如幻觉、知识过时和缺乏透明、可追溯的推理过程等。提供更准确的回答、降低推理成本、实现外部记忆。

RAG流程图



- 流程
 1. 输入文本转化为向量
 2. 在向量数据库中匹配相似文本
 3. 作为prompt在大模型中寻找答案

RAG部署与实践

1. 环境配置

主要库的功能分析：

protobuf==4.25.3 #负责与server端通信

accelerate==0.28.0 #分布式部署

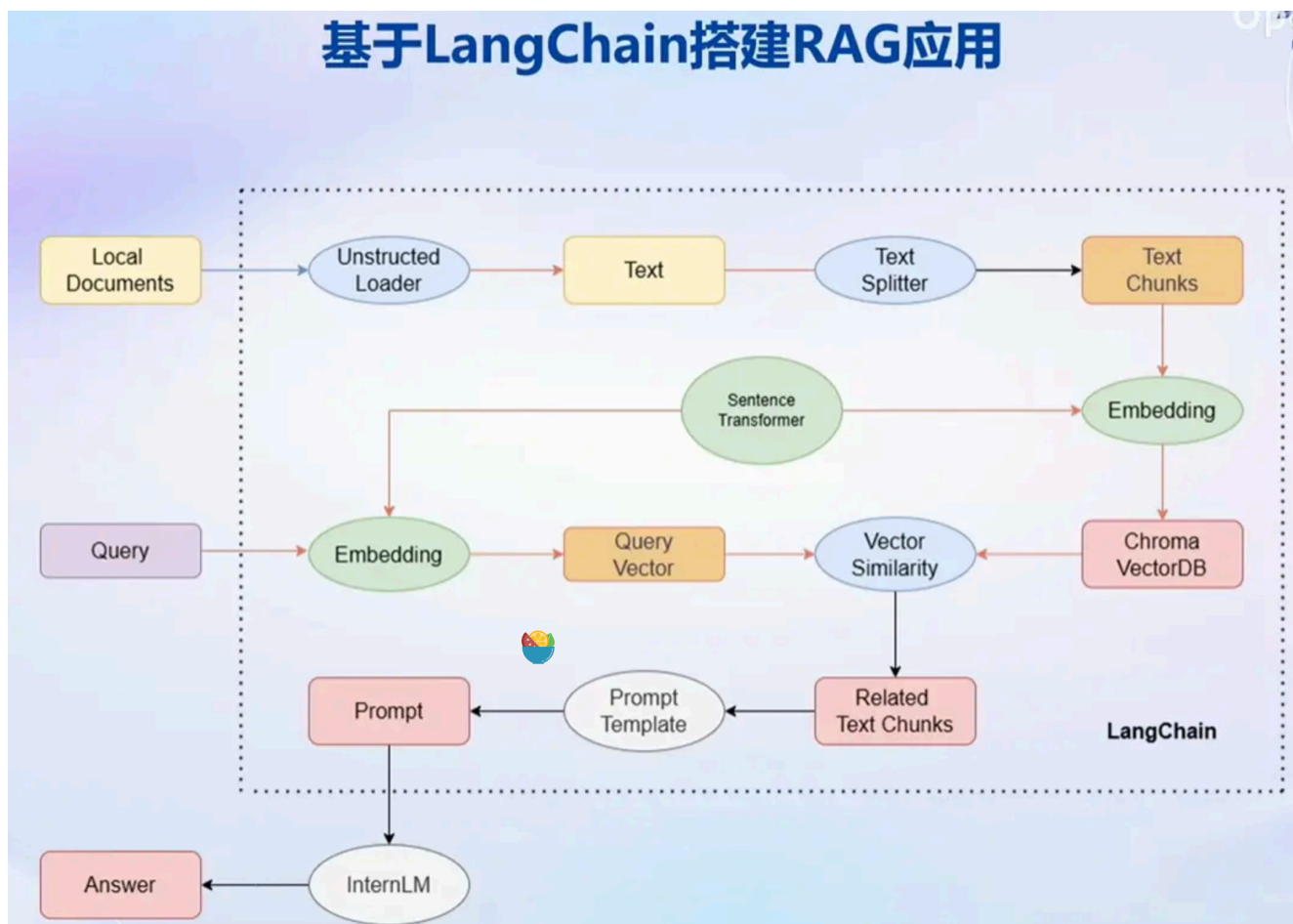
bcembedding==0.1.3 # BCE embedding和reranker模型部署

beautifulsoup4==4.8.2 #网页内容解析

faiss-gpu==1.7.2 #向量数据库

langchain==0.1.14 # RAG框架搭建

langchain可以实现如下图中的功能



loguru==0.7.2 # 日志打印

pandas==2.2.1 # 构建知识库

pymupdf==1.24.1 python-docx==1.1.0 pytoml==0.1.21 readability-lxml==0.8.1 # 文档读取相关

2. 下载茴香豆的script文件及文档

3. 修改配置文件

主要修改了三个模型的存储路径

4. 构建知识库

提取知识库特征，创建向量数据库。数据库向量化的过程应用到了 **LangChain** 的相关模块，默认嵌入和重排序模型调用的网易 **BCE 双语模型**，查看发现引用了这些文档作为知识库

📁 / ... / workdir / preprocess /

Name

📄 repodir_huixiangdou_huixiangdou-inside.md

📄 repodir_huixiangdou_requirements-lark-group.txt

📄 repodir_huixiangdou_README_zh.md

📄 repodir_huixiangdou_requirements.txt

📄 repodir_huixiangdou_README.md

📄 repodir_huixiangdou_logs_work.txt

📄 repodir_huixiangdou_docs_add_wechat_accessibility_zh.md

📄 repodir_huixiangdou_docs_architecture_en.md

📄 repodir_huixiangdou_docs_add_wechat_group_zh.md

📄 repodir_huixiangdou_docs_add_lark_group_zh.md

📄 repodir_huixiangdou_docs_architecture_zh.md

📄 repodir_huixiangdou_.github_ISSUE_TEMPLATE_bug.md

📄 repodir_huixiangdou_.github_ISSUE_TEMPLATE_others.md

📄 repodir_huixiangdou_android_README.md

📄 repodir_huixiangdou_web_requirements.txt

📄 repodir_huixiangdou_web_README.md

📄 repodir_huixiangdou_web_proxy_traslate.txt

📄 repodir_huixiangdou_web_proxy_logs_work.txt

📄 repodir_huixiangdou_web_front-end_readme.md

📄 0bf33ad1.text

📄 d0a21c4e.text

📄 28eb6cfa.text

5. 构建good question和bad question

这一步应该是为更好地判断query和知识库中文档的相关性，good question也是来源与上一步建好的知识库而来的，如果query与good question比较相似，即判断可以用rag来检索增强

6. 运行并测试