一文搞懂大模型RAG应用（附实践案例）

zhuanlan.zhihu.com/p/668082024

写在前面

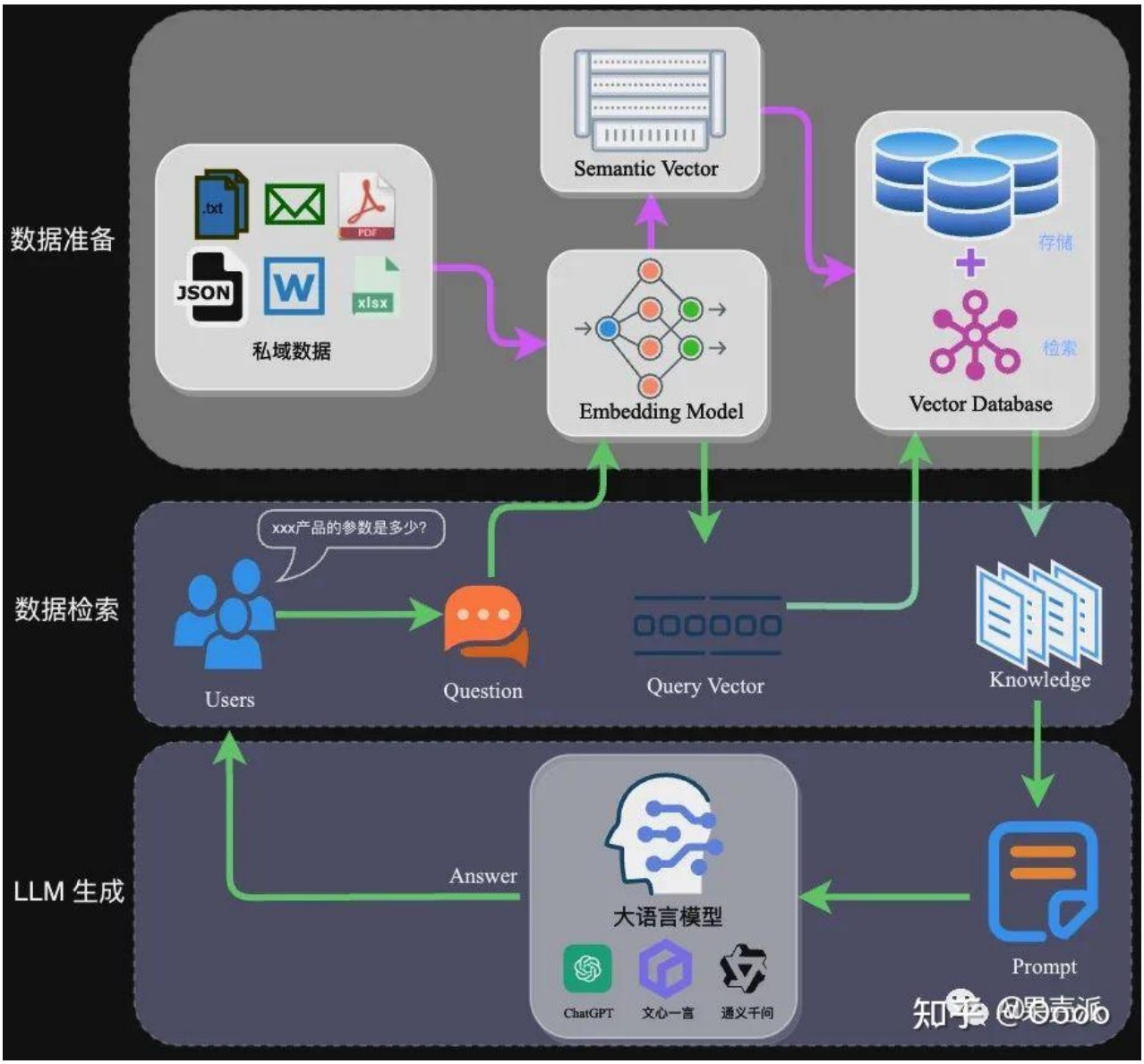
什么是RAG?

检索增强生成（Retrieval Augmented Generation），简称 RAG，已经成为当前最火热的LLM应用方案。经历今年年初那一波大模型潮，想必大家对大模型的能力有了一定的了解，但是当我们将大模型应用于实际业务场景时会发现，通用的基础大模型基本无法满足我们的实际业务需求，主要有以下几方面原因：

而RAG是解决上述问题的一套有效方案。

RAG架构

RAG的架构如图中所示，简单来讲，RAG就是通过检索获取相关的知识并将其融入Prompt，让大模型能够参考相应的知识从而给出合理回答。因此，可以将RAG的核心理解为“检索+生成”，前者主要是利用向量数据库的高效存储和检索能力，召回目标知识；后者则是利用大模型和Prompt工程，将召回的知识合理利用，生成目标答案。



RAG架构

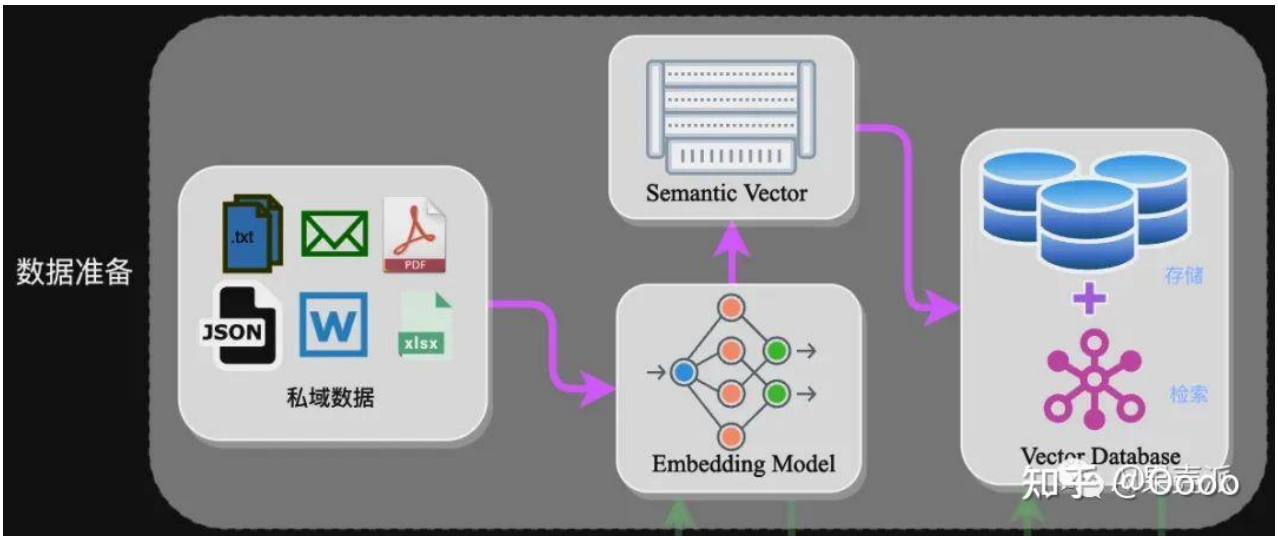
完整的RAG应用流程主要包含两个阶段：

- 数据准备阶段：数据提取——>文本分割——>向量化（embedding）——>数据入库
- 应用阶段：用户提问——>数据检索（召回）——>注入Prompt——>LLM生成答案

下面我们详细介绍一下各环节的技术细节和注意事项：

数据准备阶段：

数据准备一般是一个离线的过程，主要是将私域数据向量化后构建索引并存入数据库的过程。主要包括：数据提取、文本分割、向量化、数据入库等环节。



数据准备

- **数据提取**
  - 数据加载：包括多格式数据加载、不同数据源获取等，根据数据自身情况，将数据处理为同一个范式。
  - 数据处理：包括数据过滤、压缩、格式化等。
  - 元数据获取：提取数据中关键信息，例如文件名、Title、时间等 。
- **文本分割**：
  文本分割主要考虑两个因素：1）embedding模型的Tokens限制情况；2）语义完整性对整体的检索效果的影响。一些常见的文本分割方式如下：
  - 句分割：以”句”的粒度进行切分，保留一个句子的完整语义。常见切分符包括：句号、感叹号、问号、换行符等。
  - 固定长度分割：根据embedding模型的token长度限制，将文本分割为固定长度（例如256/512个tokens），这种切分方式会损失很多语义信息，一般通过在头尾增加一定冗余量来缓解。
- **向量化（embedding）**：

向量化是一个将文本数据转化为向量矩阵的过程，该过程会直接影响到后续检索的效果。目前常见的embedding模型如表中所示，这些embedding模型基本能满足大部分需求，但对于特殊场景（例如涉及一些罕见专有词或字等）或者想进一步优化效果，则可以选择开源Embedding模型微调或直接训练适合自己场景的Embedding模型。

模型名称	描述	获取地址
ChatGPT-Embedding	ChatGPT-Embedding由OpenAI公司提供，以接口形式调用。	platform.openai.com/doc

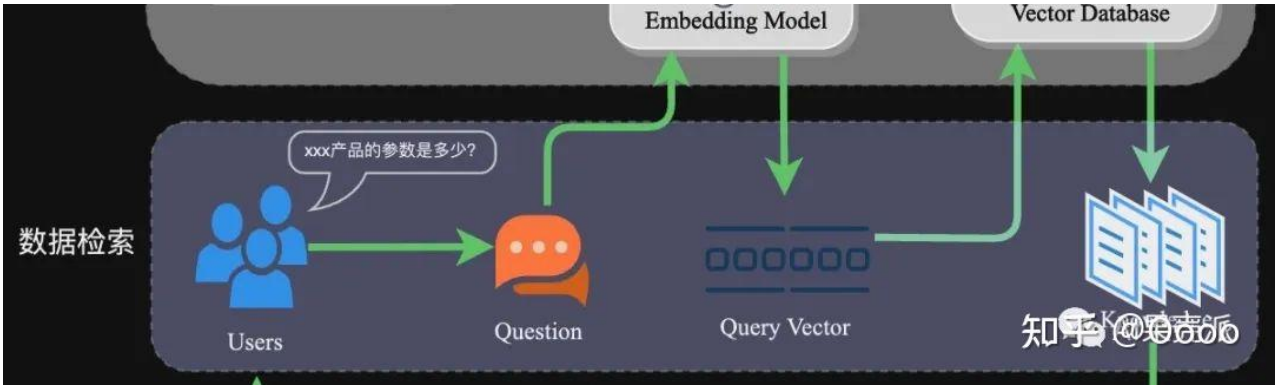
ERNIE-Embedding V1	ERNIE-Embedding V1由百度公司提供，依赖于文心大模型能力，以接口形式调用。	cloud.baidu.com/doc/WEN
M3E	M3E是一款功能强大的开源Embedding模型，包含m3e-small、m3e-base、m3e-large等多个版本，支持微调和本地部署。	huggingface.co/moka-ai/
BGE	BGE由北京智源人工智能研究院发布，同样是一款功能强大的开源Embedding模型，包含了支持中文和英文的多个版本，同样支持微调和本地部署。	huggingface.co/BAAI/bge

数据入库：

数据向量化后构建索引，并写入数据库的过程可以概述为数据入库过程，适用于RAG场景的数据库包括：FAISS、Chromadb、ES、milvus等。一般可以根据业务场景、硬件、性能需求等多因素综合考虑，选择合适的数据库。

应用阶段：

在应用阶段，我们根据用户的提问，通过高效的检索方法，召回与提问最相关的知识，并融入Prompt；大模型参考当前提问和相关知识，生成相应的答案。关键环节包括：数据检索、注入Prompt等。



数据检索

数据检索

常见的数据检索方法包括：相似性检索、全文检索等，根据检索效果，一般可以选择多种检索方式融合，提升召回率。

- 相似性检索：即计算查询向量与所有存储向量的相似性得分，返回得分高的记录。常见的相似性计算方法包括：余弦相似性、欧氏距离、曼哈顿距离等。
- 全文检索：全文检索是一种比较经典的检索方式，在数据存入时，通过关键词构建倒排索引；在检索时，通过关键词进行全文检索，找到对应的记录。

- **注入Prompt**

LLM生成