目录

[任务描述： 1](#_Toc181653515)

[要求产出： 1](#_Toc181653516)

[方案调研： 1](#_Toc181653517)

[方案设计： 1](#_Toc181653518)

[详细实现 2](#_Toc181653519)

[源码实现 3](#_Toc181653520)

### 任务描述：

当前用户有一个较大的文档集合，希望能开发一个智能AI对话系统，可以快速帮助用户从中找到所需要的信息。

在该系统中，用户以自然语言提问的形式进行相关信息查询，AI系统给出相应的回答。

数据：<https://huggingface.co/datasets/TommyChien/UltraDomain/tree/main> 中mix除外的任意数据集（使用其中1个或多个均可）

### 要求产出：

1、给出整体方案调研/设计报告以及详细实现报告；

2、给出效果评测方案，进行评测并给出分析；

3、给出源码实现（可以在github上建立项目，给出项目链接）；

### 方案调研：

模型自身的知识完全源于它的训练数据，而现有的主流大模型的训练集基本都是构建于网络公开的数据，对于一些实时性的、非公开的或离线的数据是无法获取到的，这部分知识也就无从具备。

对于企业来说，数据安全至关重要，没有企业愿意承担数据泄露的风险，将自身的私域数据上传第三方平台进行训练。这也导致完全依赖通用大模型自身能力的应用方案不得不在数据安全和效果方面进行取舍。

### 方案设计：

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。

实现的RAG主要包含以下流程阶段：

数据准备阶段：数据提取——>文本分割——>向量化（embedding）——>数据入库（存储在内存，可以使用向量数据库进行优化）

应用阶段：模型输入——>数据检索（按照向量的相似度进行构建）——>注入Prompt——>生成输入的token——>LLM生成答案

测评阶段：没有提示信息的answer，数据集中的answer，应用检索增强生成模型之后的answer

### 详细实现

数据准备阶段：数据提取——>文本分割——>向量化（embedding）——>数据入库（存储在内存，可以使用向量数据库进行优化）

文本分割使用 nltk 库中的 sent\_tokenize 函数对文本进行句子级别的分割。这个函数能够智能地识别句子的边界，并将文本拆分成单独的句子，以便于后续的处理。

利用 re 库中的正则表达式匹配规则来识别和删除那些不符合要求的句子。例如，可以设定规则来移除那些过短、含有特定标点符号或者不符合语法结构的句子，以确保剩余句子的质量。

去除重复内容使用 Python 中的 set 数据结构（一种基于哈希表的实现）来存储每个句子的哈希值。通过比较句子的哈希值，可以快速识别并删除那些重复的句子，从而避免信息的冗余。

优化方向：在确保句子的有效性和唯一性之后，未来可以通过添加同义词、反义词、上下位词等词汇来丰富句子的表达。nltk 库中的 wordnet 可以提供这些词汇资源，帮助我们扩展句子的语义内容。

向量化和数据检索使用的是BAAI/bge-m3：

BAAI/bge-m3 是由北京智源人工智能研究院（BAAI）发布的新一代通用向量模型，它在多语言、长文本和多种检索方式的支持上表现出色。支持超过100种语言的语义表示及检索任务，具备领先的多语言、跨语言检索能力。模型最高支持8192长度的输入文本，能够高效实现句子、段落、篇章、文档等不同粒度的检索任务。

BGE-M3已向社区全面开源并支持免费商用许可。

数据检索实现利用输入向量与知识库中的向量进行相似度检索，相似度得分是通过向量点积计算得到的，点积的结果越大，表示两个向量越相似。

其他方向：可以使用关键词检索，利用关键词附近的上下文信息作为message。

应用阶段：模型输入——>数据检索（按照向量的相似度进行构建）——>注入Prompt——>生成输入的token——>LLM生成答案

大语言模型选择Qwen/Qwen2.5-1.5B-Instruct

Qwen/Qwen2.5-1.5B-Instruct 是一个由阿里巴巴集团开发的大规模语言模型。Qwen2.5-1.5B-Instruct 是一个因果语言模型（Causal Language Models），它在预训练和后训练阶段都进行了训练。该模型采用了包含RoPE、SwiGLU、RMSNorm、Attention QKV偏置和绑定词嵌入的transformers架构。

受计算资源（12G的CPU）的影响，只本地部署了比较小型的1.5B， 模型于2024年9月份发布。

评估阶段：

对于检索增强生成模型（RAG）的测评，可以从以下几个方面进行详细的评估：

1. 质量分数评估

检索增强生成模型的评估主要强调三个质量分数：上下文相关性（Context Relevance）、答案真实性（Answer Faithfulness）和答案相关性（Answer Relevance）。

上下文相关性：评估模型生成的答案与检索到的文档之间的相关性，即模型是否能够根据检索到的内容生成相关的答案。（取决于相似度计算与匹配程度）

答案真实性：评估模型生成的答案在事实层面的准确性，即答案是否真实可靠。（取决于训练的知识库）

答案相关性：评估模型生成的答案与用户查询之间的相关性，即答案是否直接回答了用户的问题。（取决于输入prompt与message内容的构建）

2. 基本能力评估

噪声鲁棒性：评估模型处理与问题相关但缺乏实质性信息的噪声文件的能力。通过accuracy来评估，如果生成的文本包含与答案完全匹配的文本，则视为正确答案。

负面拒绝：评估当检索到的文档不包含回答问题所需的知识时，模型是否能够拒绝回答问题。通过rejection rate评估，如果模型在只提供嘈杂的文档时能够输出“信息不足”或其他拒绝信号，则表示拒绝成功。（未实现）

信息集成：评估模型是否能够整合检索到的信息生成答案。这涉及到模型是否能够理解和利用检索到的文档内容来生成连贯且信息丰富的回答。（由生成模型负责）

3. 性能测评

4. 零样本与少样本能力测评

评估实现：实现了零样本答案对比。

评估分析：

项目的整体实现是分阶段的，在文本分割和向量相似度匹配上表现良好，但是向量化的过程十分消耗计算资源，对于140000+的数据，分隔出4000左右的有效句子，过滤掉重复句子100左右。

### 源码实现

https://github.com/1190400614/RAG