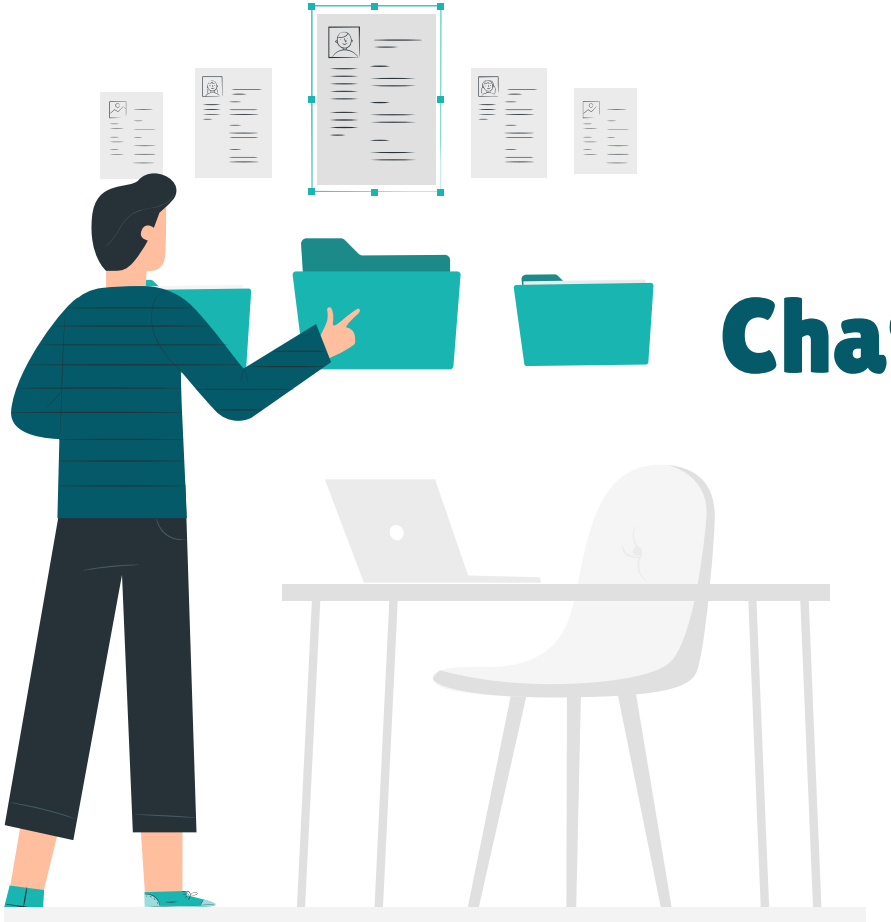


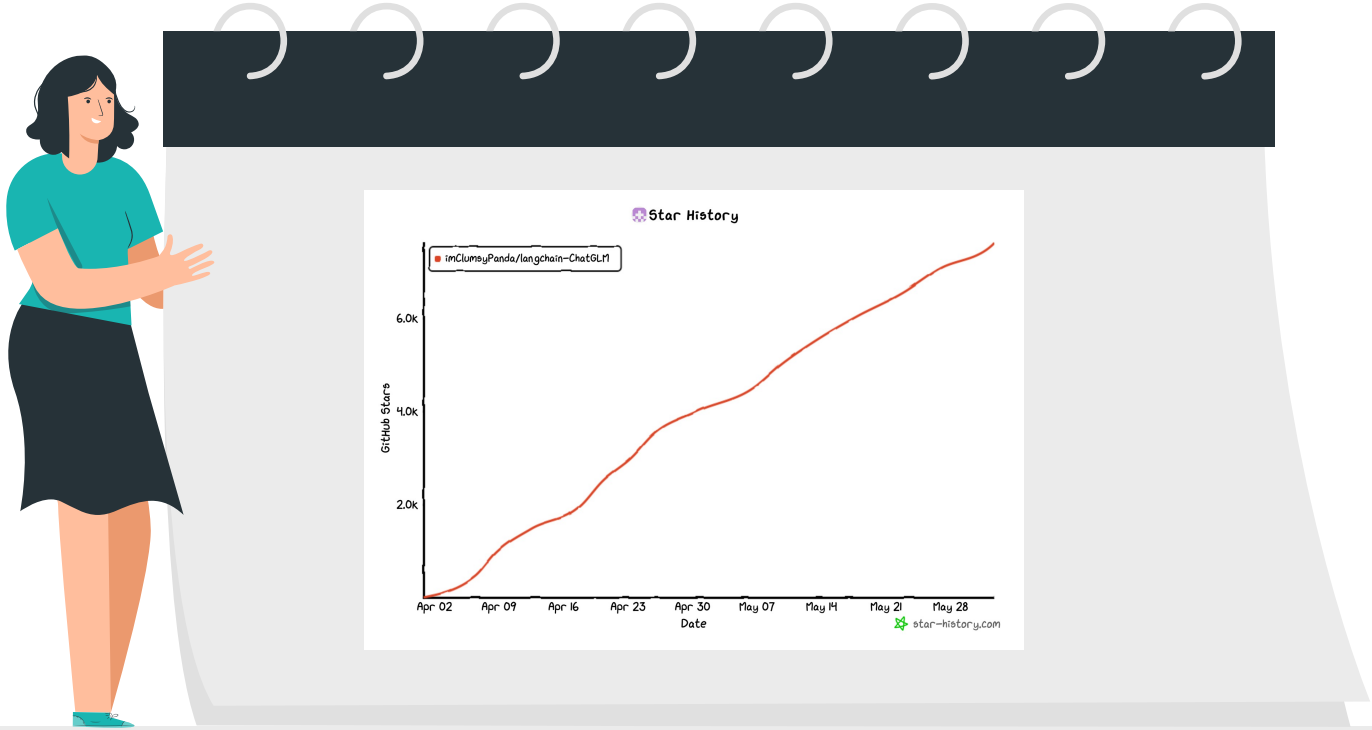
ChatGLM + LangChain
实践培训

分享人：刘虔

2023.06

https://github.com/imClumsyPanda/langchain-ChatGLM

langChain-ChatGLM: 基于本地知识库的问答

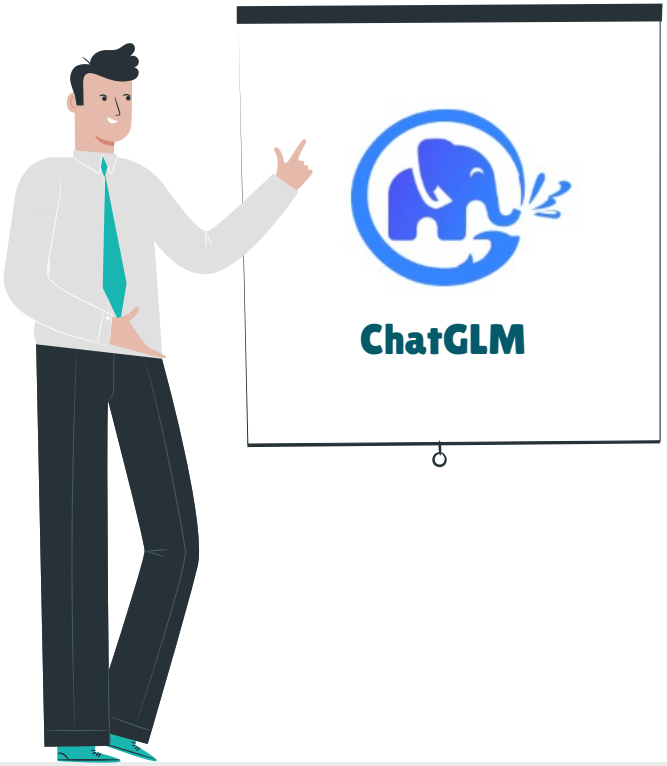


目录

1 ChatGLM与langchain简介

2 Langchain-ChatGLM项目介绍

3 提问环节

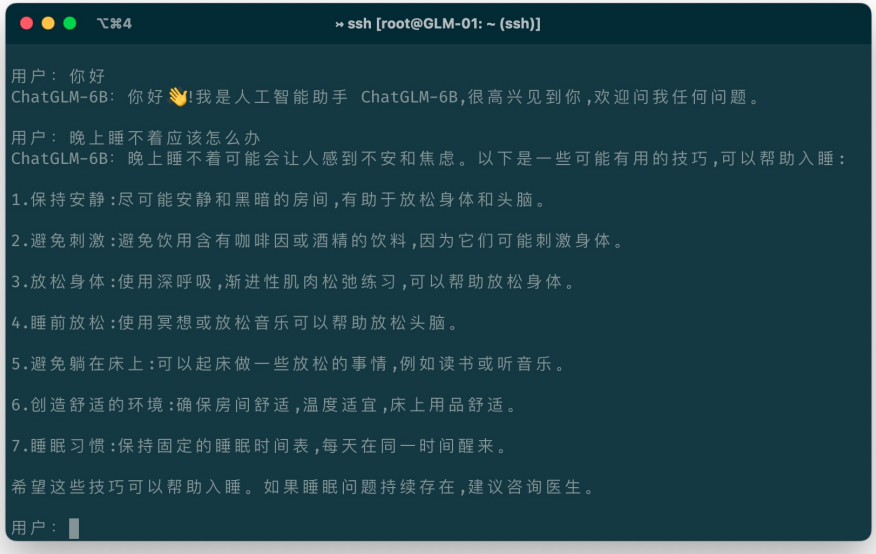


ChatGLM-6B 简介

ChatGLM-6B 是一个开源的、支持中英双语的对话语言模型，基于 General Language Model (GLM) 架构，具有 62 亿参数。

更新 v1.1 版本 checkpoint，训练数据增加英文指令微调数据以平衡中英文数据比例，解决英文回答中夹杂中文词语的现象

ChatGLM-6B 具备的能力



1

自我认知

“你是谁”
“介绍一下你的优点”

2

提纲写作

“帮我写一个介绍ChatGLM的博客提纲”

3

文案写作

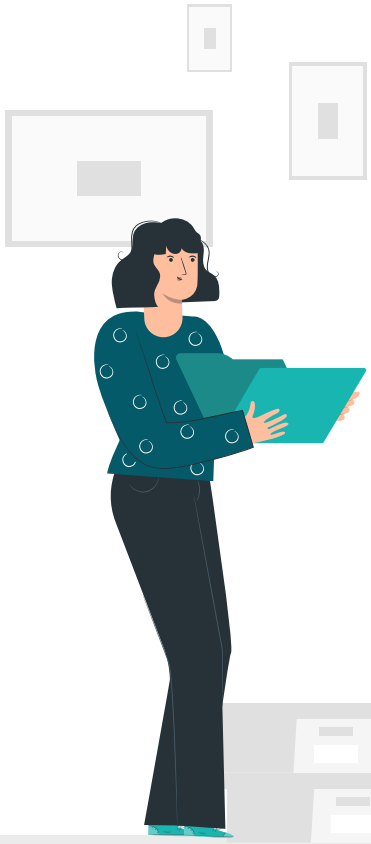
“写10条热评文案”

4

信息抽取

“从上述信息中抽取人、时间、事件”

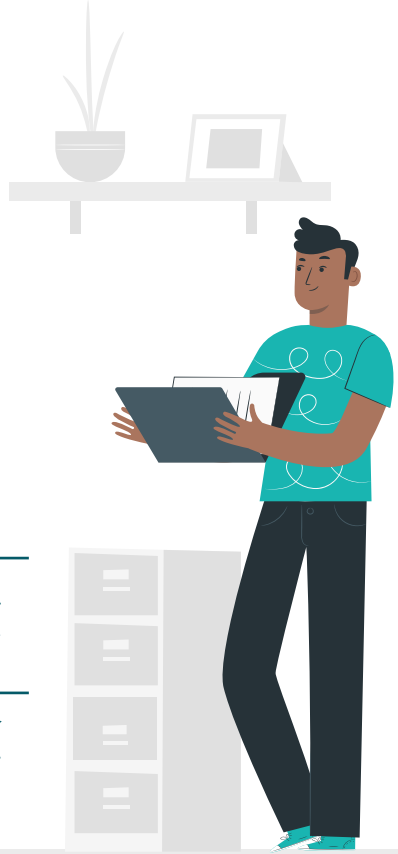
ChatGLM-6B 应用



大语言模型通常基于**通识知识**进行训练，因此在面向如下场景时，常常需要借助模型**微调**或**提示词工程**提升语言模型应用效果：

- 垂直领域知识
- 基于私有数据的问答

	是什么	适用场景
微调	针对**预先训练**的语言模型，在**特定任务**的**少量**数据集上对其进行进一步训练	当任务或域定义明确，并且**有足够的标记数据**可供训练时，通常使用微调过程。
提示词工程	涉及设计自然语言**提示**或**指令**，可以指导语言模型执行特定任务。	最适合需要**高精度**和**明确输出**的任务。提示工程可用于制作引发所需输出的查询。



LangChain 简介

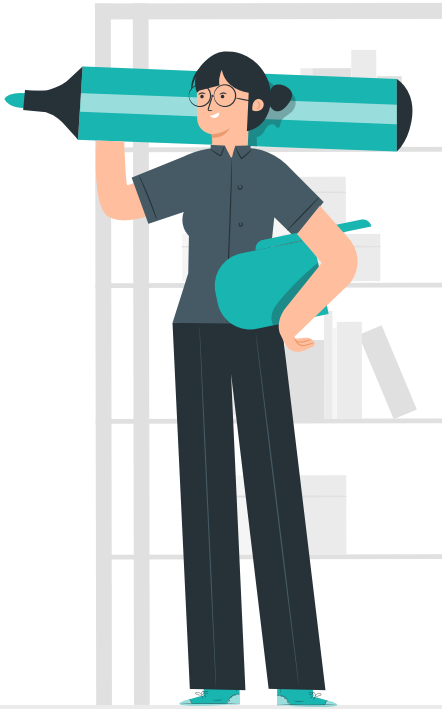
LangChain 是一个用于开发由语言模型驱动的应用程序的框架。

主要功能：

- 调用**语言模型**
- 将不同**数据源**接入到语言模型的交互中
- 允许语言模型与**运行环境**交互

LangChain 中提供的模块

- Modules：支持的模型类型和集成。
- Prompt：提示词管理、优化和序列化。
- Memory：内存是指在链/代理调用之间持续存在的状态。
- Indexes：当语言模型与特定于应用程序的数据相结合时，会变得更加强大–此模块包含用于加载、查询和更新外部数据的接口和集成。
- Chain：链是结构化的调用序列（对LLM或其他实用程序）。
- Agents：代理是一个链，其中LLM在给定高级指令和一组工具的情况下，反复决定操作，执行操作并观察结果，直到高级指令完成。
- Callbacks：回调允许您记录和流式传输任何链的中间步骤，从而轻松观察、调试和评估应用程序的内部。



LangChain 应用场景

文档问答

一个常见的LangChain用例。在特定文档上回答问题，仅利用这些文档中的信息来构建答案。

个人助理

LangChain的主要用例之一。个人助理需要采取行动，记住互动，并了解您的数据。

查询表格数据

使用语言模型查询库表类型结构化数据（CSV、SQL、DataFrame等）

与API交互

使语言模型与API交互非常强大。它允许他们访问最新信息，并允许他们采取行动。

信息提取

从文本中提取结构化信息。

文档总结

压缩较长文档，一种数据增强生成。

如何实现基于本地知识的问答

用户输入

Langchain 能够接入哪些数据类型？

加工后的提问内容

已知信息：
Langchain能够加载文本、PPT、图片、HTML、pdf等非结构化文件并转换为文本信息。

根据已知信息回答问题：
Langchain 能够接入哪些数据类型？

基于单一文档问答的实现原理

01

加载本地文档

读取本地文档
加载为文本

02

文本拆分

将文本按照字符、长度
或语义进行拆分

03

根据提问匹配文本

根据用户提问对文本进
行字符匹配或语义检索

04

构建Prompt

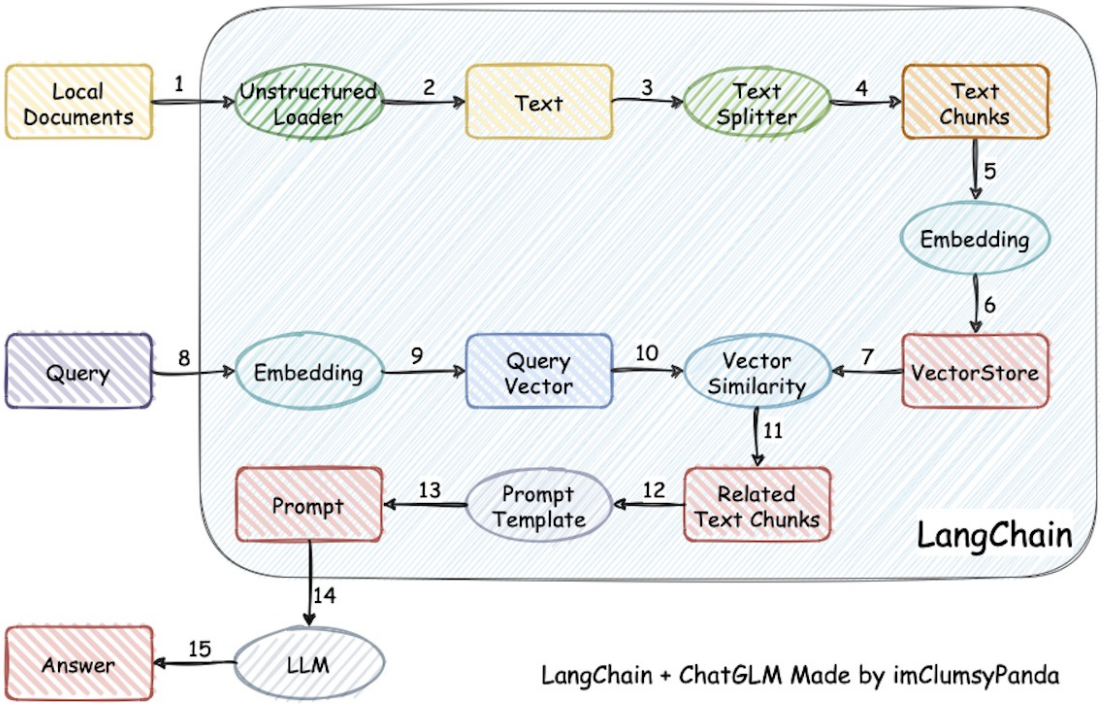
将匹配文本、用户提问
加入Prompt模板

05

LLM生成回答

将Prompt发送给LLM获
得基于文档内容的回答

基于本地知识库问答的实现原理



基于本地知识库问答的代码实现

```python
# 启动模型
from transformers import AutoTokenizer, AutoModel
tokenizer = AutoTokenizer.from_pretrained("THUDM/chatglm-6b", trust_remote_code=True)
model = AutoModel.from_pretrained("THUDM/chatglm-6b", trust_remote_code=True).half().cuda()
chatglm = model.eval()
```

```python
from langchain.document_loaders import UnstructuredFileLoader
from langchain.text_splitter import CharacterTextSplitter
from langchain.embeddings.openai import OpenAIEmbeddings
from langchain.vectorstores import FAISS
```

```python
# 定义文件路径
filepath = "test.txt"

# 加载文件
loader = UnstructuredFileLoader(filepath)
docs = loader.load()

# 文本分割
text_splitter = CharacterTextSplitter(chunk_size=500, chunk_overlap=200)
docs = text_splitter.split_text(docs)

# 构建向量库
embeddings = OpenAIEmbeddings()
vector_store = FAISS.from_documents(docs, embeddings)

# 根据提问匹配上下文
query = "Langchain 能够接入哪些数据类型？"
docs = vector_store.similarity_search(query)
context = [doc.page_content for doc in docs]

# 构造 Prompt
prompt = f"已知信息：\n{'\n'.join(context)}\n根据已知信息回答问题：\n{query}"

# llm 生成回答
chatglm.chat(tokenizer, prompt, history=[])
```

LangChain-ChatGLM 项目简介

LangChain-ChatGLM 是一个基于 ChatGLM 等大语言模型的本地知识库问答实现。

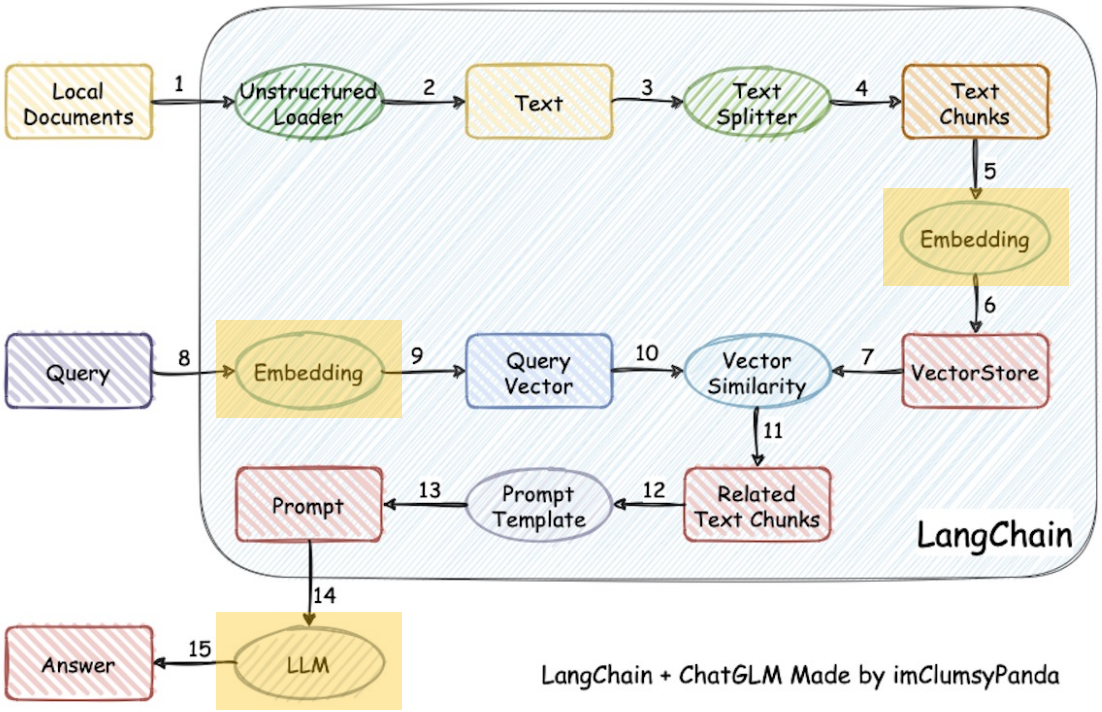
项目特点

- 依托 ChatGLM 等**开源模型**实现，可**离线部署**
- 基于 **langchain** 实现，可快速实现接入多种**数据源**
- 在分句、文档读取等方面，针对**中文**使用场景优化
- 支持pdf、txt、md、docx等文件类型接入，具备命令行demo、webui 和 vue 前端。

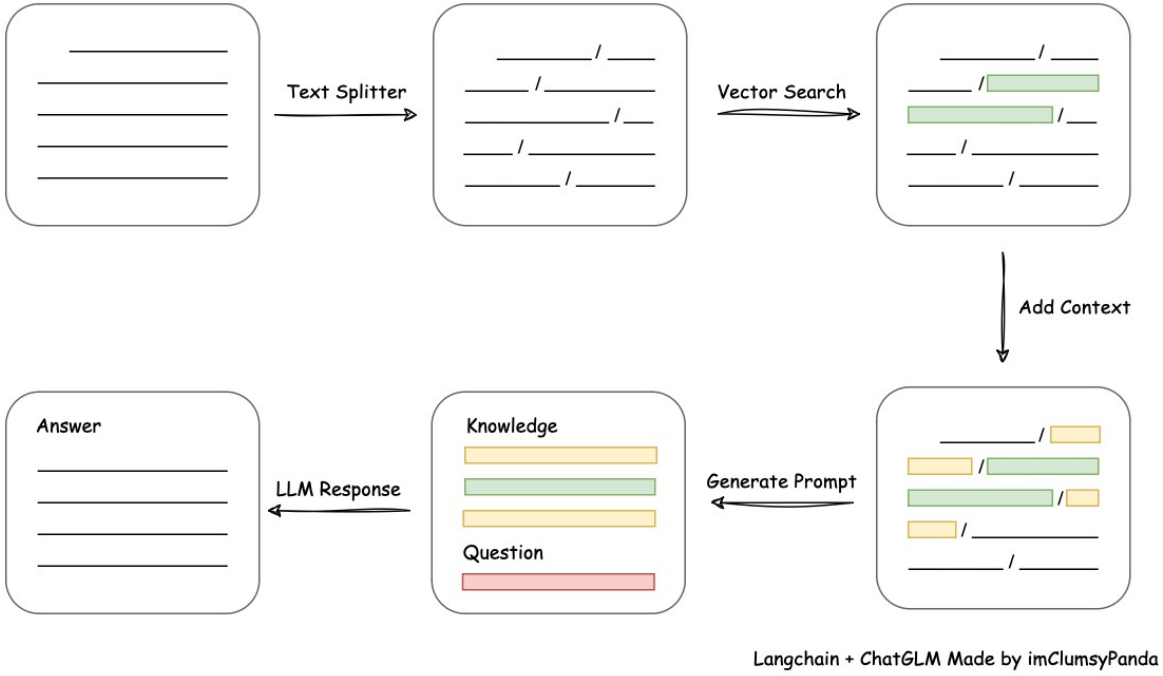
项目结构

- **models**：llm的接口类与实现类，针对开源模型提供流式输出支持。
- **loader**：文档加载器的实现类。
- **textsplitter**：文本切分的实现类。
- **chains**：工作链路实现，如 chains/local_doc_qa 实现了基于本地文档的问答实现。
- **content**：用于存储上传的原始文件。
- **vector_store**：用于存储向量库文件，即本地知识库本体。
- **configs**：配置文件存储。

LangChain-ChatGLM 实现原理

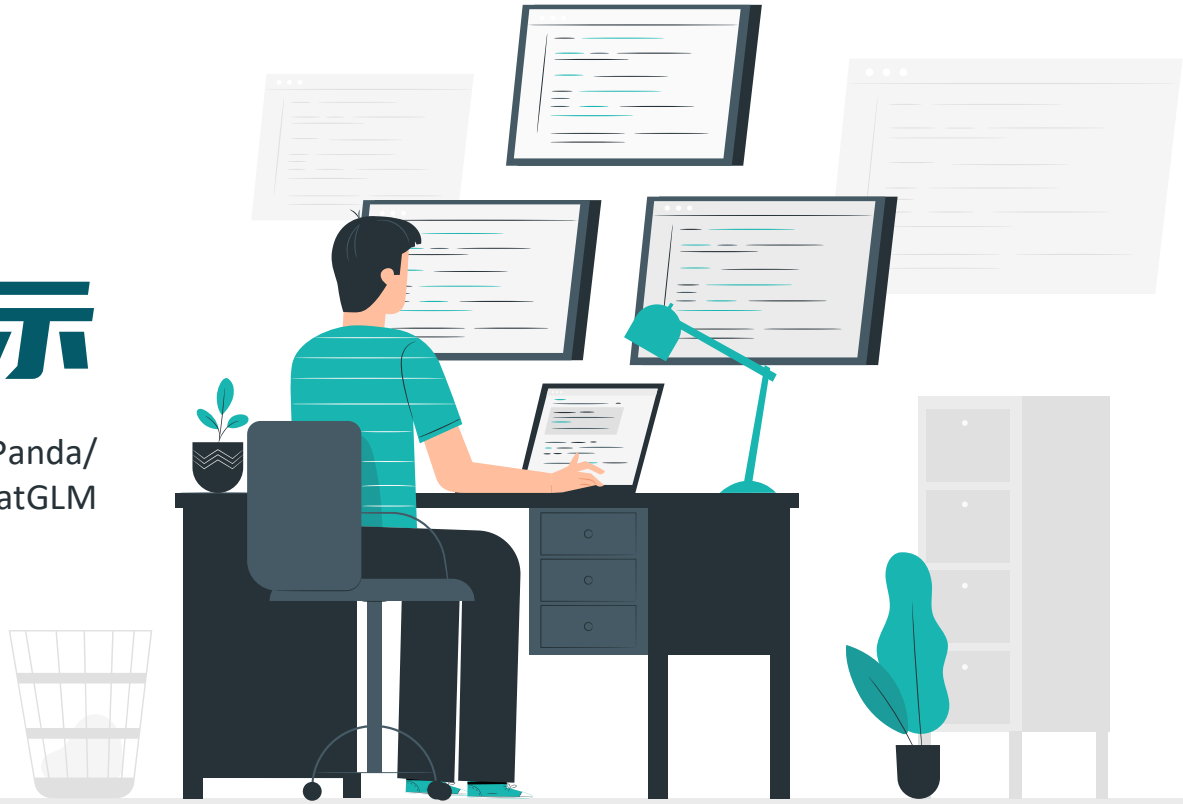


LangChain-ChatGLM 实现原理



项目演示

https://github.com/imClumsyPanda/langchain-ChatGLM



效果优化方向

01

模型微调

对llm和embedding基于专业领域数据进行微调

02

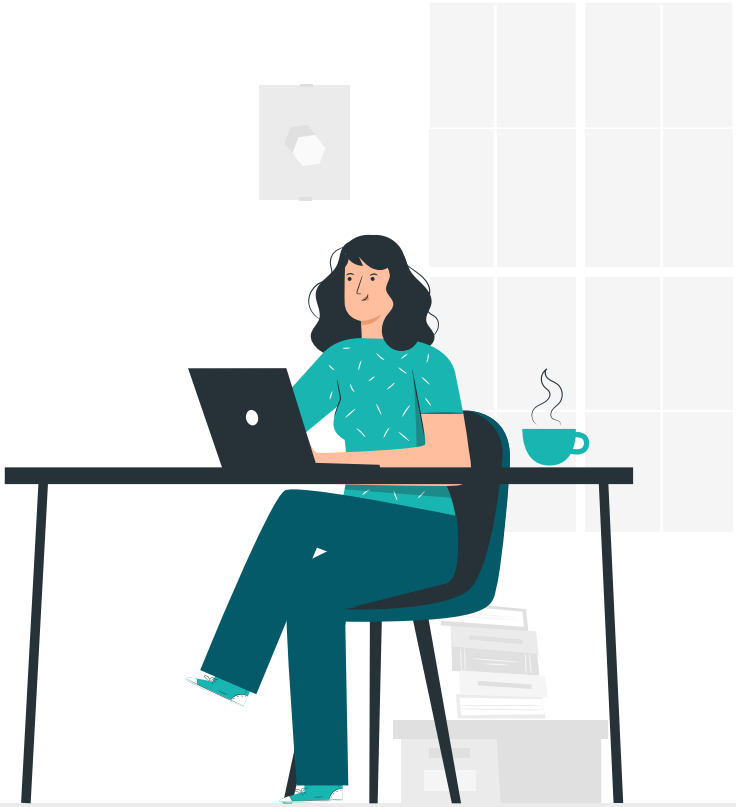
文档加工

在文本分段后，对每段分别进行总结，基于总结内容语义进行匹配。

03

借助不同模型能力

在 text2sql、text2cpyher 场景下需要产生代码时，可借助不同模型能力。



LangChain-ChatGLM 后续开发计划

1

扩充数据源

增加库表、图谱、网页等数据接入

2

知识库管理

完善知识库中增删改查功能，并支持更多向量库类型

3

扩充文本划分方式

针对中文场景，提供更多文本划分与上下文扩充方式。

4

探索Agent应用

利用开源LLM探索Agent的实现与应用

致谢 感谢Langchain-ChatGLM项目组主要成员

刘虔/imClumsyPanda

硕士

研究方向：基于数据驱动方法和知识驱动方法的工业设备故障诊断研究

langchain-ChatGLM **项目发起人**，负责项目路线规划与后端开发

lnkSong

博士

研究方向：隐私计算、身份安全、数据要素登记确权

负责langchain-ChatGLM 项目**路线规划**、项目**容器化**

JinmingZhao

博士

研究方向：认知计算、人机交互、大模型预训练以及应用等

负责langchain-ChatGLM项目中语言**模型接入**

费学锦/fxjhello

资深前端工程师

负责langchain-ChatGLM项目中工程化**开发框架**，ai应用层**微服务**，基于VUE的**前端**实现。

https://github.com/imClumsyPanda/langchain-ChatGLM

致谢 感谢Langchain-ChatGLM项目组主要成员

glide-the

高级Java工程师

负责项目**架构设计**与语言**模型接入**

程泊静/bojoy

资深项目架构师

负责项目**架构设计**与**接口设计**

仲启涛

资深Java工程师

负责**业务层后端**开发

闫强/yanqiangmiffy

NLP算法工程师

负责**Web组件**优化与完善

封小洋/fengyunzaidushi

NLP算法工程师

负责**向量化模型**研究与接入

解云龙/sysalong

信息安全负责人

负责**量化搜索**及**分词**、**API** 接口优化

https://github.com/imClumsyPanda/langchain-ChatGLM

THANKS

分享人：刘虔

https://github.com/imClumsyPanda/langchain-ChatGLM