

Langchain-ChatGLM：基于本地知识库问答

文章目录

ChatGLM与Langchain简介	
ChatGLM-6B简介	
ChatGLM-6B是什么	
ChatGLM-6B具备的能力	
ChatGLM-6B具备的应用	
Langchain简介	
Langchain是什么	
Langchain的核心模块	
Langchain的应用场景	
ChatGLM与Langchain项目介绍	
知识库问答实现步骤	
ChatGLM与Langchain项目特点	
项目主体结构	
项目效果优化方向	
项目后续开发计划	
ChatGLM与Langchain项目实战过程	
实战（一）	
实战（二）	

ChatGLM与Langchain简介

ChatGLM-6B简介

ChatGLM-6B是什么

ChatGLM-6B地址：<https://github.com/THUDM/ChatGLM-6B>

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。结合模型量化技术，用户可以在消费级的显卡上进行本地部署（INT4 量化级别下最低只需 6GB 显存）。

ChatGLM-6B 使用了和 ChatGPT 相似的技术，针对中文问答和对话进行了优化。经过约 1T 标识符的中英双语训练，辅以监督 **微调** 、反馈自助、人类反馈强化学习等技术的加持，62 亿参数的 ChatGLM-6B 已经能生成相当符合人类偏好的回答。

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

- 自我认知（可以对自己进行介绍，优点缺点等）
- 提纲写作（比如：帮我写一个介绍ChatGLM的博客提纲）
- 文案写作（根据一段话来生成一段文案）
- 信息抽取（抽取一段文本的人物，时间，地点等实体信息）
- 角色扮演（指定ChatGLM为一个角色，进行对话）

ChatGLM-6B具备的应用

大语言模型通常基于通识知识进行训练的，而在面向某些领域的具体场景时，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：常见的场景如下：

- 垂直领域知识的特定任务（金融领域，法律领域）
- 基于垂直领域知识库的问答

模型微调与提示词工程的区别：

模型微调：针对预训练好的语言模型，在特定任务的数据集上进行进一步的微调训练，需要有标记好的特定任务的数据。

提示工程：核心是设计自然语言提示或指定，引导模型完成特定任务，适合需要明确输出的任务。

Langchain简介

Langchain是什么

LangChain 是一个用于开发由语言模型驱动的应用程序的框架。他主要拥有 3个能力：

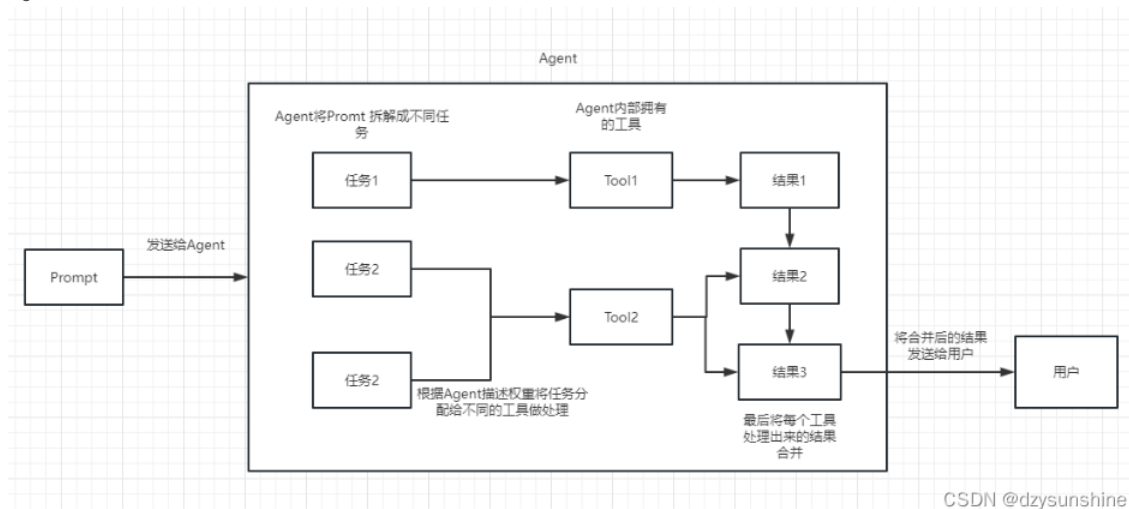
- 可以调用LLM模型
- 可以将 LLM 模型与外部数据源进行连接
- 允许与 LLM 模型进行交互

Langchain的核心模块

Langchain的核心模块如下：

- Modules：支持的模型类型和集成，如：openai, huggingface等；
- Prompt：提示词管理、优化和序列化，支持各种自定义模板；
- Memory：内存管理（在链/代理调用之间持续存在的状态）；
- Indexes：索引管理，方便加载、查询和更新外部数据；
- Agents：代理，是一个链，可以决定和执行操作，并观察结果，直到指令完成；
- Callbacks：回调，允许记录和流式传输任何链的中间步骤，方便观察、调试和评估。

Agents代理执行过程如下：



Langchain的应用场景

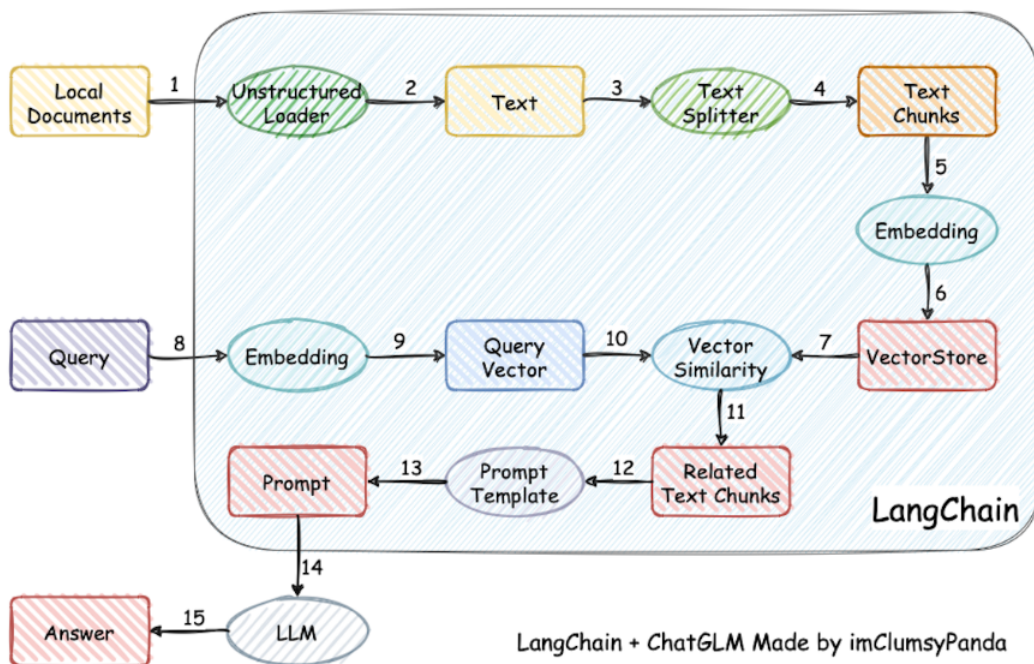
- 文档问答
- 个人助理
- 查询表格
- 与API交互
- 信息提取
- 文档总结

ChatGLM与Langchain项目介绍

知识库问答实现步骤

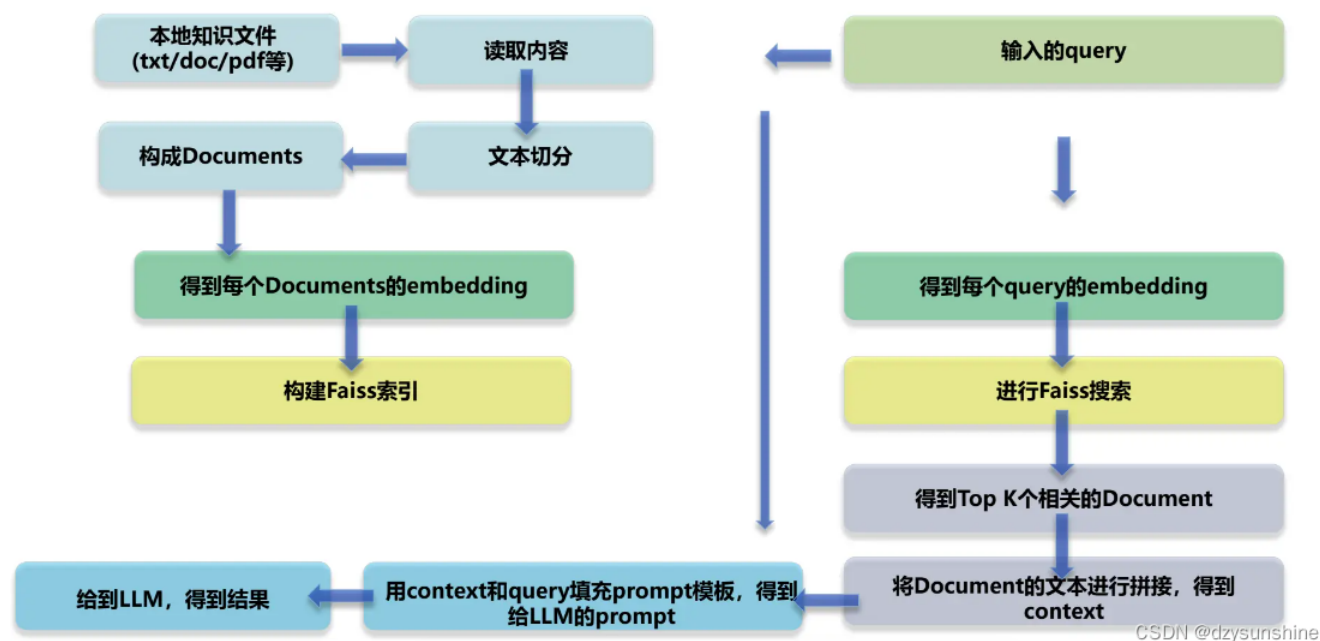
基于Langchain思想实现基于本地知识库的问答应用。实现过程如下：

- 1、加载文件
- 2、读取文本
- 3、文本分割
- 4、文本向量化
- 5、问句向量化
- 6、在文本向量中匹配出与问句向量最相似的top k个
- 7、匹配出的文本作为上下文和问题一起添加到prompt中
- 8、提交给LLM生成回答。



CSDN @dzysunshine

还有另一个版本（本质是一样的）



ChatGLM与Langchain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。

项目主体结构

- 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。
- textspliter: 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。

项目效果优化方向

- 模型微调：一个是对embedding模型的基于垂直领域的数据进行微调；一个是对LLM模型的基于垂直领域的数据进行微调；
- 文档加工：一种是使用更好的文档拆分的方式（如项目中已经集成的达摩院的语义识别的模型及进行拆分）；一种是改进填充的方式，判断中心句上下文的句子是否和中心句相关，仅添加相关度高的句子；另一种是文本分段后，对每段分别及进行总结，基于总结内容语义及进行匹配；
- 借助不同模型的能力：在 text2sql、text2cpyher 场景下需要产生代码时，可借助不同模型能力。

项目后续开发计划

- 扩充数据源：增加库表、图谱、网页等数据接入；
- 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；
- 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；
- 探索Agent应用：利用开源LLM探索Agent的实现与应用。

ChatGLM与Langchain项目实战过程

实战（一）

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

由于之前已经对ChatGLM进行过部署，所以考虑可以直接在原有环境中安装新的所需的包即可，同样也可以使用之前下载好的模型文件：ChatGLM部署

但看了下requirements.txt文件后还有不少需要安装的包，索性直接新建一个python3.8.13的环境（模型文件还是可以用的）

```
1 | conda create -n langchain python==3.8.13
```

拉取项目

```
1 | git clone https://github.com/imClumsyPanda/langchain-ChatGLM.git
```

进入目录

```
1 | cd langchain-ChatGLM
```

安装requirements.txt

```
1 | conda activate langchain
2 | pip install -r requirements.txt
```

当前环境支持装langchain的最高版本是0.0.166，无法安装0.0.174，就先装下0.0.166试下。

修改配置文件路径：

```
1 | vi configs/model_config.py
```

将chatglm-6b的路径设置成自己的。

```
"chatglm-6b": {
    "name": "chatglm-6b",
    "pretrained_model_name": "/data/sim_chatgpt/chatglm-6b",
    "local_model_path": None,
    "provides": "ChatGLM"
```

修改要运行的代码文件：webui.py，

```
1 | vi webui.py
```

将最后launch函数中的share设置为True，inbrowser设置为True。

执行webui.py文件

```
1 | python webui.py
```

```
loading model config
llm device: cuda
embedding device: cuda
dir: /students/julyedu_522454/langchain-ChatGLM
flagging username: e5bd9e34fdbc470681a17f2990ddd04e

Loading /data/sim_chatgpt/chatglm-6b...
Loading checkpoint shards: 100% |██████████████████████████████████████| 8/8 [00:09<00:00, 1.19s/it]
Loaded the model in 13.13 seconds.
INFO 2023-06-06 17:41:54,001-ld: Load pretrained SentenceTransformer: GanymedeNil/text2vec-large-chinese
Downloading (...)cial_tokens_map.json: 100% |██████████████████████████████████████| 125/125 [00:00<00:00, 34.1kB/s]
Downloading (...)58aa3/tokenizer.json: 100% |██████████████████████████████████████| 439k/439k [00:00<00:00, 980kB/s]
Downloading (...)okenizer_config.json: 100% |██████████████████████████████████████| 514/514 [00:00<00:00, 222kB/s]
Downloading (...)026ff58aa3/vocab.txt: 100% |██████████████████████████████████████| 110k/110k [00:00<00:00, 393kB/s]
WARNING 2023-06-06 17:41:59,334-ld: No sentence-transformers model found with name /students/julyedu_522454/.cache/torch/sentence_transformers/GanymedeNil_text2vec-large-chinese. Creating a new one with MEAN pooling.
WARNING 2023-06-06 17:42:03,055-ld: The dtype of attention mask (torch.int64) is not bool
{'answer': '你好😊！我是人工智能助手 ChatGLM-6B，很高兴见到你，欢迎问我任何问题。'}
INFO 2023-06-06 17:42:07,843-ld: 模型已成功加载，可以开始对话，或从右侧选择模式后开始对话
Running on local URL: http://0.0.0.0:7860
```

CSDN @dzysunshine

可能是网络问题，无法创建一个公用链接。可以进行云服务器和本地端口的映射，参考：<https://www.cnblogs.com/monologuesmw/p/14465117.html>



对应输出:

选择知识库名称后，即可开始问答：当前知识库为空，如有需要可以在选择知识库名称后上传文件/文件夹至知识库。


暂不支持文件删除，该功能将在后续版本中推出。'], [None, '模型已成功加载，可以开始对话，或从右侧选择模式后开始对话'], [None, '已选择知识库ceshi, 当前知识库中未上传文件，请先上传文件后，再开始提问'], [None, '已添加


占用显存情况：大约15个G


NVIDIA-SMI 510.108.03 Driver Version: 510.108.03 CUDA Version: 11.6									
GPU Name		Persistence-M		Bus-Id		Disp.A		Volatile Uncorr. ECC	
Fan	Temp	Perf	Pwr:Usage/Cap		Memory-Usage		GPU-Util	Compute M.	MIG M.
0	Tesla	P100-PCIE...	On	00000000:00:0A.0	Off		0		
N/A	36C	P0	31W / 250W	14899MiB / 16384MiB			0%	Default	N/A
Processes:									
GPU	GI	CI	PID	Type	Process name		GPU Memory		
	ID	ID					Usage		
0	N/A	N/A	23243	C	python		14897MiB		
CSDN @dzysunshine									


实战（二）


项目地址：<https://github.com/thomas-yanxin/LangChain-ChatGLM-Webui>
HuggingFace社区在线体验：<https://huggingface.co/spaces/thomas-yanxin/LangChain-ChatLLM>


 **Hugging Face**

 Models


 Datasets

 Spaces

 Docs




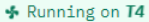
 Solutions


Pricing





Log In


Sign Up

 Spaces:  **thomas-yanxin / LangChain-ChatLLM**  like 84 

 App

 Files

 Community 4



LangChain-ChatLLM-Webui

本项目基于LangChain和大型语言模型系列模型,提供基于本地知识的自动问答应用.
目前项目提供基于[ChatGLM-6B](#)系列、Minimax的LLM和包括text2vec-base-chinese、ernie-3.0-zh系列以及由[Jina](#)提供的ViT-B-32::laion2b-s34b-b79k等多个Embedding模型,支持上传txt、docx、md等文本格式文件.
后续将提供更加多样化的LLM、Embedding和参数选项供用户尝试,欢迎关注[Github地址](#).
本项目已内置开发者自己的key, 用户无需输入自己的相关key.
当然, 更推荐您点击右上角的Duplicate this Space, 将项目Fork到自己的Space中, 保护个人隐私, 且避免排队!

模型选择

large language model

ChatGLM-6B-int4

Embedding model

text2vec-base

请上传知识库文件, 目前支持txt、docx、md格式

ChatLLM

CSDN @dzysunshine

另外也支持ModelScope魔搭社区、飞桨AIStudio社区等在线体验。

下载项目

```
1 | git clone https://github.com/thomas-yanxin/LangChain-ChatGLM-Webui.git
```

进入目录

```
1 | cd LangChain-ChatGLM-Webui
```

安装所需的包

```
1 | pip install -r requirements.txt
2 | pip install gradio==3.10
```

修改config.py

```
1 init_llm = "ChatGLM-6B"
2
3 llm_model_dict = {
4     "chatglm": {
5         "ChatGLM-6B": "/data/sim_chatgpt/chatglm-6b",
```

修改app.py文件，将launch函数中的share设置为True，inbrowser设置为True。

执行webui.py文件

```
1 | python webui.py
```

LangChain-ChatLLM-Webui

本项目基于LangChain和大型语言模型系列模型, 提供基于本地知识的自动问答应用.

目前项目提供基于ChatGLM-6B的LLM和包括GanymedeNil/text2vec-large-chinese、nghuyong/ernie-3.0-base-zh、nghuyong/ernie-3.0-nano-zh在内的多个Embedding模型. 支持上传 txt、docx、md、pdf等文本格式文件. 后续将提供更加多样化的LLM、Embedding和参数选项供用户尝试, 欢迎关注Github地址.

模型选择

large language model

ChatGLM-6B

Embedding model

text2vec-base

重新加载模型

模型参数配置

vector search top k

6

history len

3

temperature

0.01

top_p

0.9

请上传知识库文件

README.md

197.4 KB

Download

知识库文件向量化

ChatLLM

langchain都有哪些功能?

根据已知信息, Langchain 是一种利用本地知识库实现问答应用的框架, 它支持接入非结构化文档、结构化数据、图片 OCR 文字识别、搜索引擎、本地网页和知识图谱/图数据库等. 此外, Langchain 还提供了 Agent 实现的功能, 使得应用能够根据用户的问题自动调用相应的模型进行回答.

显存占用约13G。

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+									
NVIDIA-SMI		510.108.03		Driver Version: 510.108.03			CUDA Version: 11.6		
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+									
GPU	Name	Persistence-M		Bus-Id	Disp.A	Volatile Uncorr. ECC			
Fan	Temp	Perf	Pwr:Usage/Cap	Memory-Usage		GPU-Util	Compute M.	MIG M.	
=====									
0	Tesla P100-PCIE...	On		00000000:00:0A.0	Off	0			
N/A	37C	P0	36W / 250W	12901MiB / 16384MiB		0%	Default	N/A	
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+									
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+									
Processes:									
GPU	GI	CI	PID	Type	Process name	GPU Memory Usage			
	ID	ID							
=====									
0	N/A	N/A	21224	C	python	12899MiB			
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+									

CSDN @dzysunshine

参考

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

<https://liaokong.gitbook.io/llm-kai-fa-jiao-cheng/>

<https://github.com/thomas-yanxin/LangChain-ChatGLM-Webui>