

InterLM线上课程训练营

#AI

<https://studio.intern-ai.org.cn/console/dashboard>

baidu <https://aistudio.baidu.com/overview>

11班-492-40017566

<https://aicarrier.feishu.cn/wiki/Vv4swUFMni5DiMkcasUczUp9nid>

作业笔记提交地址: (<https://aicarrier.feishu.cn/wiki/TqjawZsoqiiRXEkRpcScmKAAAn8d?table=tblNCUy9PeGmgd9l&view=vewQagjCL1>)

作业: 课程及 **InternLM2** 技术报告笔记

InternLM2 技术报告链接: [link](#)

觉得不错欢迎 Star

<https://github.com/InternLM/InternLM/>

2.2.3 笔记&作业打卡



- 按照时间安排完成笔记和作业。
 - **笔记**：课程学习的知识整理总结和心得体会。
 - **作业**：在算力平台上实践操作的过程整理总结和心得体会，**必须有关键步骤完成截图，否则视为未完成。**
- 笔记和作业可以放在github、CSDN等自己的笔记平台，再把链接粘贴到 **学员信息&作业登记** 表格中对应列。
- 注意笔记和作业线上链接中的内容需要**公开对所有人可见**（注意开启权限），方便助教评审，并注明教程来源链接。



- **优秀作业反馈&存档**：特别优秀的作业会被助教推荐给浦语小助手（告知学员昵称、作业链接），可能会在 InternLM 新媒体平台宣传推广。
- 如果希望被推荐或者不希望被公开，可以在打卡中显眼处加以文字说明。

第一课3/27

- 增强了数学内生能力+代码解释器（超过gpt4）
- 数据分析能力
- 全链条开源
- 高质量语料：书生-万卷1.0，CC [opendatalab.org.cn]
- 训练加速：8卡~千卡训练
- 微调：增量续训、有监督微调、微调框架**XTuner**（消费卡8G显存调7B模型）
- 评测：OpenCompass 轮换策略

思考&总结：InternLM似乎具有更好的中文能力和调用工具能力，值得进一步探索

第二课4/2

https://github.com/InternLM/Tutorial/blob/camp2/helloworld/hello_world.md

1、部署1.8B模型

- 开发机: 10%GPU, cuda11.7
- terminal环境配置

```
studio-conda -o internlm-base -t demo
# 与 studio-conda 等效的配置方案
# conda create -n demo python==3.10 -y
# conda activate demo
# conda install pytorch==2.0.1 torchvision==0.15.2
torchaudio==2.0.2 pytorch-cuda=11.7 -c pytorch -c nvidia

## 进入conda环境
conda activate demo

## 安装依赖包
pip install huggingface-hub==0.17.3 transformers==4.34
psutil==5.9.8 accelerate==0.24.1 streamlit==1.32.2
matplotlib==3.8.3 modelscope==1.9.5 sentencepiece==0.1.99

=====
### 2 下载 `InternLM2-Chat-1.8B` 模型
=====
mkdir -p /root/demo
touch /root/demo/cli_demo.py
touch /root/demo/download_mini.py
cd /root/demo

## 打开 `/root/demo/download_mini.py` 文件，复制代码

import os
from modelscope.hub.snapshot_download import snapshot_download

# 创建保存模型目录
os.system("mkdir /root/models")

# save_dir是模型保存到本地的目录
save_dir="/root/models"
snapshot_download("Shanghai_AI_Laboratory/internlm2-chat-1_8b",
                  cache_dir=save_dir,
                  revision='v1.1.0')

=====
## 执行命令，下载模型参数文件：
```

```
python /root/demo/download_mini.py
```

```
#####  
### 3 运行 cli_demo
```

```
#####  
## 打开 `/root/demo/cli_demo.py` 文件, 复制
```

```
import torch  
from transformers import AutoTokenizer, AutoModelForCausalLM
```

```
model_name_or_path =  
"/root/models/Shanghai_AI_Laboratory/internlm2-chat-1_8b"
```

```
tokenizer = AutoTokenizer.from_pretrained(model_name_or_path,  
trust_remote_code=True, device_map='cuda:0')
```

```
model =  
AutoModelForCausalLM.from_pretrained(model_name_or_path,  
trust_remote_code=True, torch_dtype=torch.bfloat16,  
device_map='cuda:0')  
model = model.eval()
```

```
system_prompt = """You are an AI assistant whose name is  
InternLM (书生·浦语).
```

```
- InternLM (书生·浦语) is a conversational language model that is  
developed by Shanghai AI Laboratory (上海人工智能实验室). It is  
designed to be helpful, honest, and harmless.
```

```
- InternLM (书生·浦语) can understand and communicate fluently in  
the language chosen by the user such as English and 中文.
```

```
"""
```

```
messages = [(system_prompt, '')]
```

```
print("=====Welcome to InternLM chatbot, type 'exit' to  
exit.=====")
```

```
while True:
```

```
    input_text = input("\nUser >>> ")
```

```
    input_text = input_text.replace(' ', '')
```

```
    if input_text == "exit":
```

```
        break
```

```
length = 0
for response, _ in model.stream_chat(tokenizer, input_text,
messages):
    if response is not None:
        print(response[length:], flush=True, end="")
        length = len(response)
```

=====

输入命令，执行 Demo 程序：

```
conda activate demo
python /root/demo/cli_demo.py
```

等待模型加载完成，键入内容示例：

User >>> 请创作一个 300 字、有关小猫咪的小故事

在一个美丽的花园里，住着一只名叫小咪的小猫咪。小咪非常喜欢在花园里玩耍，尤其是当它看到那些美丽的花朵时，总是忍不住地停下来，用它的小爪子轻轻地抚摸花瓣，仿佛在和它们进行一场心灵的交流。

有一天，小咪发现花园里有一只小兔子在玩耍。它好奇地看着小兔子，然后悄悄地走近，想看看小兔子到底在玩什么游戏。小兔子看到小咪来了，它停下了玩耍，转过头来看着小咪。

小咪开始轻轻地舔小兔子的毛发，就像是在和它说话一样。小兔子被小咪的举动所感动，它开始和小咪玩耍起来，一边唱歌，一边跳着舞。小咪和小兔子成为了好朋友，它们一起在花园里玩耍，享受着美好的时光。

小咪发现，只要和动物交朋友，就可以享受到它们所带来的快乐和温暖。它决定继续和花园里的其他动物交朋友，让它的小世界变得更加美好和充满生机。

3 实战：部署 八戒-Chat-1.8B 模型

- 介绍：<https://www.modelscope.cn/models/JimmyMa99/BaJie-Chat-mini/summary>

```
=====
### 2 **配置基础环境**
=====
```

```
conda activate demo
```

```
## 使用 `git` 命令来获得仓库内的 Demo 文件
```

```
cd /root/
```

```
git clone https://gitee.com/InternLM/Tutorial -b camp2
```

```
# git clone https://github.com/InternLM/Tutorial -b camp2
```

```
cd /root/Tutorial
```

```
=====
### 3 **下载运行 Chat-八戒 Demo**
=====
```

```
## 在 `Web IDE` 中执行 `bajie_download.py`
```

```
python /root/Tutorial/helloworld/bajie_download.py
```

```
## 下载完成后，输入运行命令
```

```
streamlit run /root/Tutorial/helloworld/bajie_chat.py --
server.address 127.0.0.1 --server.port 6006
```

```
## 程序运行的同时，对端口环境配置本地 `PowerShell` (**Mac 用户打开终端即可**)
```

```
## 打开 PowerShell 后，先查询端口，再根据端口键入命令（例如图中端口示例为 38374）
```

```
# 从本地使用 ssh 连接 studio 端口
```

```
# 将下方端口号 38374 替换成自己的端口号
```

```
ssh -CNg -L 6006:127.0.0.1:6006 root@ssh.intern-ai.org.cn -p
40773
```

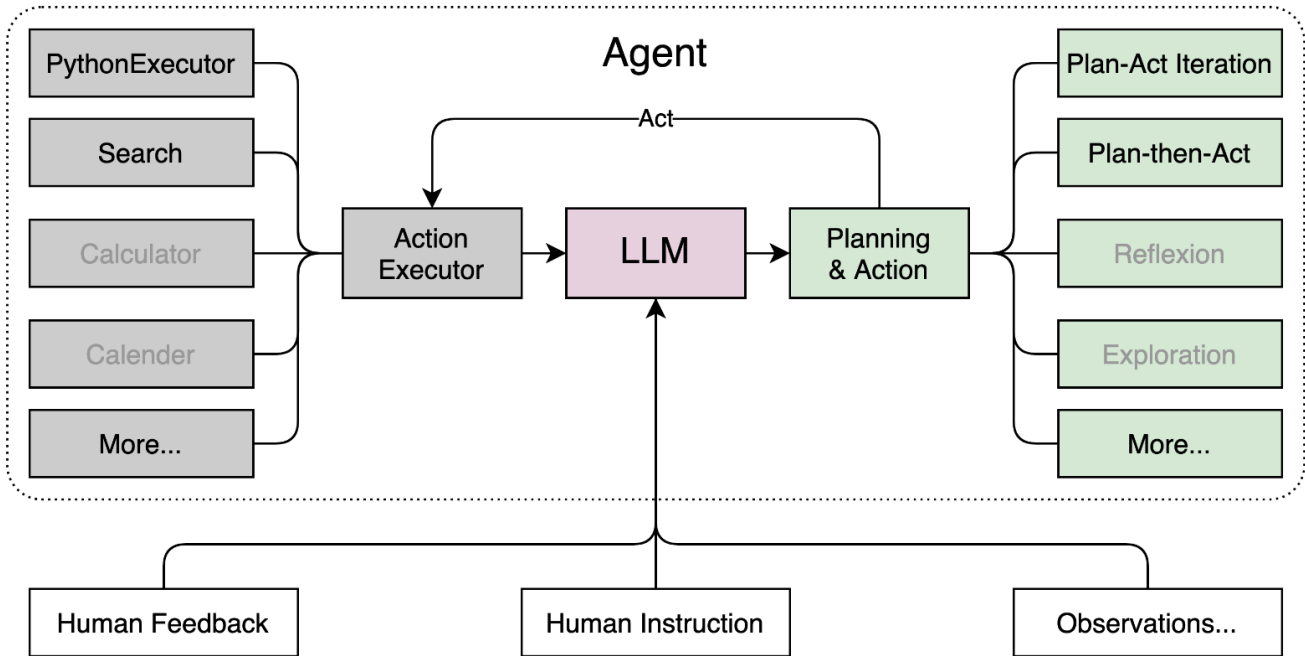
```
## 打开(http://127.0.0.1:6006/) 后，等待加载完成即可进行对话
```

```
》》》你觉得人间怎么样
```

嘿嘿，人间啊，俺老猪觉得人间可真是热闹非凡的地方！有吃有喝，还有美女，简直是俺老猪的天堂啊！不过，俺老猪可不喜欢那些妖精和妖怪，总是想要吃掉人间的美食，真是让人头疼。而且人间的规矩可多了，有些规矩俺老猪是不懂的，比如那些什么礼仪礼貌，俺老猪就觉得有点太繁琐了。不过，人间的风景也挺好的，有山有水，还有各种各样的花，俺老猪就喜欢看着这些美景，心情也变得愉快起来。总之，人间嘛，有好吃的、好玩的，就是有些规矩和繁琐的地方让俺老猪不太适应，不过，还是愿意留在人

间享受这些美食和美景的！

4 实战：使用 **Lagent** 运行 **InternLM2-Chat-7B** 模型（开启 30% A100）



Lagent 的特性总结如下：

- 流式输出：提供 `stream_chat` 接口作流式输出，本地就能演示酷炫的流式 Demo。
- 接口统一，设计全面升级，提升拓展性，包括：
 - **Model**：不论是 OpenAI API, Transformers 还是推理加速框架 LMDeploy 一网打尽，模型切换可以游刃有余；
 - **Action**：简单的继承和装饰，即可打造自己个人的工具集，不论 InternLM 还是 GPT 均可适配；
 - **Agent**：与 Model 的输入接口保持一致，模型到智能体的蜕变只需一步，便捷各种 agent 的探索实现；
- 文档全面升级，API 文档全覆盖。
- 关闭开发机，【升降】配置
- 附录：小技巧


```
=====
### 2 **配置基础环境**
=====
```

```
conda activate demo
```

```
cd /root/demo
```

```
## 使用 git 命令下载 Lagent 相关的代码库
```

```
git clone https://gitee.com/internlm/lagent.git
```

```
# git clone https://github.com/internlm/lagent.git
```

```
cd /root/demo/lagent
```

```
git checkout 581d9fb8987a5d9b72bb9ebd37a95efd47d479ac
```

```
pip install -e . # 源码安装
```

```
=====
### 3 **使用 `Lagent` 运行 `InternLM2-Chat-7B` 模型为内核的智能体**
=====
```

```
cd /root/demo/lagent
```

```
## 构造软链接快捷访问方式:
```

```
ln -s /root/share/new_models/Shanghai_AI_Laboratory/internlm2-  
chat-7b /root/models/internlm2-chat-7b
```

```
## 打开 `lagent` 路径下 `examples/internlm2_agent_web_demo_hf.py`  
文件, 并修改对应位置 (71行左右) 代码
```

```
value='/root/models/internlm2-chat-7b'
```

```
## 输入运行命令 - **点开 6006 链接后, 大约需要 5 分钟完成模型加载**
```

```
streamlit run
```

```
/root/demo/lagent/examples/internlm2_agent_web_demo_hf.py --  
server.address 127.0.0.1 --server.port 6006
```

```
## 待程序运行的同时, 对本地端口环境配置本地 `PowerShell`。使用快捷键组合  
`Windows + R` (Windows 即开始菜单键) 打开指令界面, 并输入命令, 按下回车  
键。(Mac 用户打开终端即可) 打开 PowerShell 后, 先查询端口, 再根据端口键  
入命令
```

```
# 从本地使用 ssh 连接 studio 端口
```

```
# 将下方端口号 38374 替换成自己的端口号
```

```
ssh -CNg -L 6006:127.0.0.1:6006 root@ssh.intern-ai.org.cn -p 40986
```

打开 [http://127.0.0.1:6006]后，（会有较长的加载时间）勾选数据分析，其他的选项不要选择，进行计算方面的 Demo 对话，即完成本章节实战
请解方程 $2 * X * X = 1360$ 之中 X 的结果

=====
数据分析提示词

你现在已经能够在一个有状态的 Jupyter 笔记本环境中运行 Python 代码。当你向 python 发送含有 Python 代码的消息时，它将在该环境中执行。这个工具适用于多种场景，如数据分析或处理（包括数据操作、统计分析、图表绘制），复杂的计算问题（解决数学和物理难题），编程示例（理解编程概念或特性），文本处理和分析（比如文本解析和自然语言处理），机器学习和数据科学（用于展示模型训练和数据可视化），以及文件操作和数据导入（处理CSV、JSON等格式的文件）。

5 实战：实践部署 浦语·灵笔2 模型（开启 50% A100）

5.1 初步介绍 XComposer2 相关知识

浦语·灵笔2 是基于 书生·浦语2 大语言模型研发的突破性的图文多模态大模型，具有非凡的图文写作和图像理解能力，在多种应用场景表现出色，总结起来其具有：

- 自由指令输入的图文写作能力：浦语·灵笔2 可以理解自由形式的图文指令输入，包括大纲、文章细节要求、参考图片等，为用户打造图文并貌的专属文章。生成的文章文采斐然，图文相得益彰，提供沉浸式的阅读体验。
- 准确的图文问题解答能力：浦语·灵笔2 具有海量图文知识，可以准确的回复各种图文问答难题，在识别、感知、细节描述、视觉推理等能力上表现惊人。
- 杰出的综合能力：浦语·灵笔2-7B 基于 书生·浦语2-7B 模型，在13项多模态评测中大 幅领先同量级多模态模型，在其中6项评测中超过 GPT-4V 和 Gemini Pro。

```
conda activate demo
```

```
# 补充环境包
```

```
pip install timm==0.4.12 sentencepiece==0.1.99
```

```
markdown2==2.4.10 lxmlwriter==3.1.2 gradio==4.13.0
```

```
modelscope==1.9.5
```

```
=====
```

下载 ****InternLM-XComposer 仓库**** 相关的代码资源

```
=====
```

```
cd /root/demo
```

```
git clone https://gitee.com/internlm/InternLM-XComposer.git
```

```
# git clone https://github.com/internlm/InternLM-XComposer.git
```

```
cd /root/demo/InternLM-XComposer
```

```
git checkout f31220eddc2cf6246ee2ddf8e375a40457ff626
```

```
=====
```

构造软链接快捷访问方式

```
=====
```

```
ln -s /root/share/new_models/Shanghai_AI_Laboratory/internlm-xcomposer2-7b /root/models/internlm-xcomposer2-7b
```

```
ln -s /root/share/new_models/Shanghai_AI_Laboratory/internlm-xcomposer2-vl-7b /root/models/internlm-xcomposer2-vl-7b
```

```
=====
```

3 ****图文写作实战****

```
=====
```

```
# 继续输入指令，用于启动 `InternLM-XComposer`
```

```
python /root/demo/InternLM-
```

```
XComposer/examples/gradio_demo_composition.py \
```

```
--code_path /root/models/internlm-xcomposer2-7b \
```

```
--private \
```

```
--num_gpus 1 \
```

```
--port 6006
```

```
# 待程序运行的同时，参考章节 3.3 部分对端口环境配置本地 `PowerShell`
```

```
# 从本地使用 ssh 连接 studio 端口
```

```
# 将下方端口号 38374 替换成自己的端口号
```

```
ssh -CNg -L 6006:127.0.0.1:6006 root@ssh.intern-ai.org.cn -p 40986
```

根据以下标题：“中国水墨画：流动的诗意与东方美学”，创作长文章，字数不少于800字。请结合以下文本素材：

“水墨画是由水和墨调配成不同深浅的墨色所画出的画，是绘画的一种形式，更多时候，水墨画被视为中国传统绘画，也就是国画的代表。也称国画，中国画。墨水画是中国传统画之一。墨水是国画的起源，以笔墨运用的技法基础画成墨水画。线条中锋笔，侧锋笔，顺锋和逆锋，点染，擦，破墨，拨墨的技法。墨于水的变化分为五色。画成作品，题款，盖章。就是完整的水墨画作品。

基本的水墨画，仅有水与墨，黑与白色，但进阶的水墨画，也有工笔花鸟画，色彩缤纷。后者有时也称为彩墨画。在中国画中，以中国画特有的材料之一，墨为主要原料加以清水的多少引为浓墨、淡墨、干墨、湿墨、焦墨等，画出不同浓淡（黑、白、灰）层次。别有一番韵味称为“墨韵”。而形成水墨为主的一种绘画形式。”

4 **图片理解实战**

```
# 关闭并重新启动一个新的 `terminal`，继续输入指令，启动 `InternLM-
XComposer2-vl`
conda activate demo
```

```
cd /root/demo/InternLM-XComposer
python /root/demo/InternLM-
XComposer/examples/gradio_demo_chat.py \
--code_path /root/models/internlm-xcomposer2-vl-7b \
--private \
--num_gpus 1 \
--port 6006
```

请分析一下图中内容



Chatbot

图像中，松鼠正飞跃森林中的一个树桩。它以独特的方式抓着树枝，使自己能够腾空。雪花从天而降，形成了一个美丽而寒冷的雪景。松鼠位于图像的正中央，背景中是雪花纷飞的森林。



请分析图片

这张图片展示了伦敦泰晤士河上的著名景点——伦敦眼。从对岸的角度看，伦敦眼的高大轮子屹立在河畔，成为伦敦市中心的高地标志。泰晤士河上，一艘快艇疾驰而过，船体留下一道涟漪。河边的建筑物倒映在河水中，与河对岸的景色相呼应。一片雪花纷飞的景象出现在远处，为整个场景增添了一番冬日的气息。

第三课4/7

第三节课来啦！由书生·浦语社区贡献者【北辰】带来【茴香豆：搭建你的 RAG 智能助理】讲解

- 1、RAG 基础介绍
- 2、茴香豆产品简介
- 3、使用茴香豆搭建RAG知识库实战

视频地址：<https://www.bilibili.com/video/BV1QA4m1F7t4/>

文档地址：

<https://github.com/InternLM/Tutorial/blob/camp2/huixiangdou/readme.md>

作业地址：

<https://github.com/InternLM/Tutorial/blob/camp2/huixiangdou/homework.md>

1. 在茴香豆 Web 版中创建自己领域的知识问答助手

参考视频零编程玩转大模型，学习茴香豆部署群聊助手

完成不少于 400 字的笔记 + 线上茴香豆助手对话截图(不少于5轮)

(可选) 参考 代码 在自己的服务器部署茴香豆 Web 版

2. 在 InternLM Studio 上部署茴香豆技术助手

根据教程文档搭建 茴香豆技术助手，针对问题"茴香豆怎么部署到微信群？"进行提问

完成不少于 400 字的笔记 + 截图

github

<https://github.com/InternLM/HuixiangDou>

RAG的3种方式

RAG 发展进程

RAG的概念最早是由Meta (Facebook) 的Lewis等人在2020《Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks》中提出的。

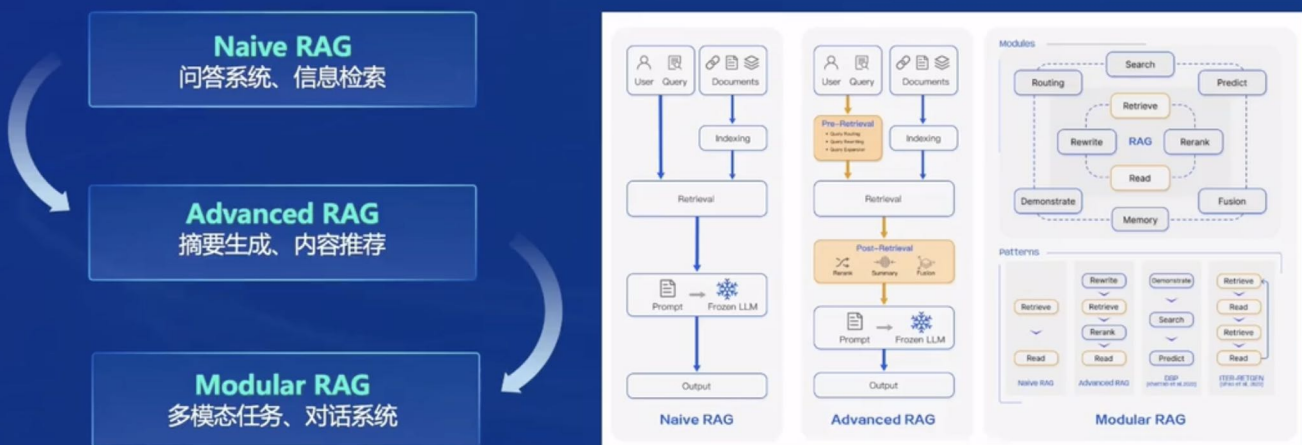


Image from: Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., Wang, H., 2024. Retrieval-Augmented Generation for Large Language Models: A Survey.

RAG 常见优化方法

嵌入优化 Embedding Optimization

- ✓ 结合稀疏和密集检索
- ✓ 多任务

索引优化 Indexing Optimization

- ✓ 细粒度分割 (Chunk)
- ✓ 元数据

查询优化 Query Optimization

- ✓ 查询扩展、转换
- ✓ 多查询

上下文管理 Context Curation

- ✓ 重排 (rerank)
- ✓ 上下文选择/压缩

迭代检索 Iterative Retrieval

- ✓ 根据初始查询和迄今为止生成的文本进行重复搜索

递归检索 Recursive Retrieval

- ✓ 迭代细化搜索查询
- ✓ 链式推理 (Chain-of-Thought) 指导检索过程

自适应检索 Adaptive Retrieval

- ✓ Flare, Self-RAG
- ✓ 使用LLMs主动决定检索的最佳时机和内容

LLM微调 LLM Fine-tuning

- ✓ 检索微调
- ✓ 生成微调
- ✓ 双重微调

RAG VS. 微调ft

RAG vs. 微调 (Fine-tuning)

RAG

- 非参数记忆，利用外部知识库提供实时更新的信息。
- 能够处理知识密集型任务，提供准确的事实性回答。
- 通过检索增强，可以生成更多多样化的内容。

适用场景

适用于需要结合最新信息和实时数据的任务：开放域问答、实时新闻摘要等。

优势： 动态知识更新，处理长尾知识问题。

局限： 依赖于外部知识库的质量和覆盖范围。依赖大模型能力。

Fine-tuning


- 参数记忆，通过在特定任务数据上训练，模型可以更好地适应该任务。
- 通常需要大量标注数据来进行有效微调。
- 微调后的模型可能过拟合，导致泛化能力下降。

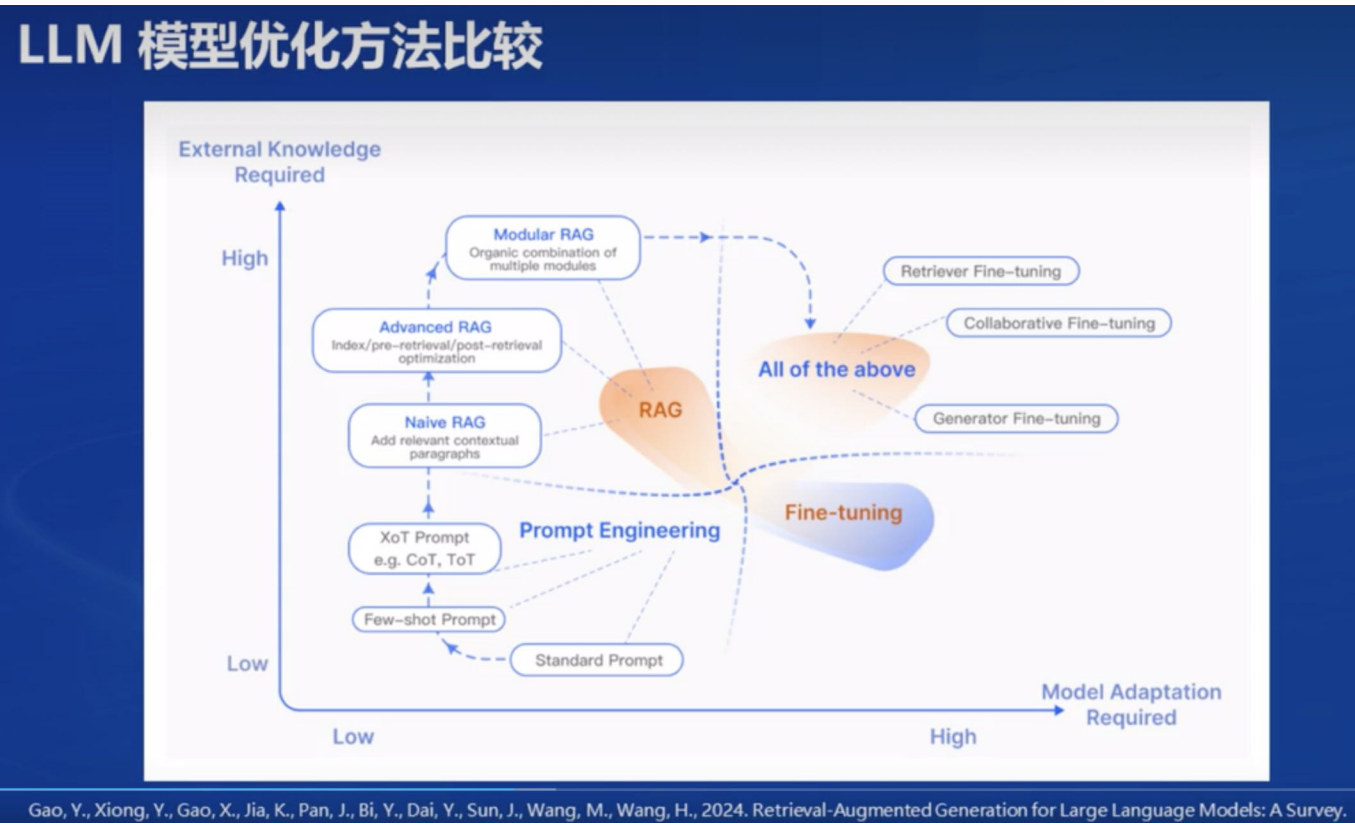
适用场景

适用于数据可用且需要模型高度专业化的任务，如特定领域的文本分类、情感分析、文本生成等。

优势： 模型性能针对特定任务优化。

局限： 需要大量的标注数据，且对新任务的适应性较差。

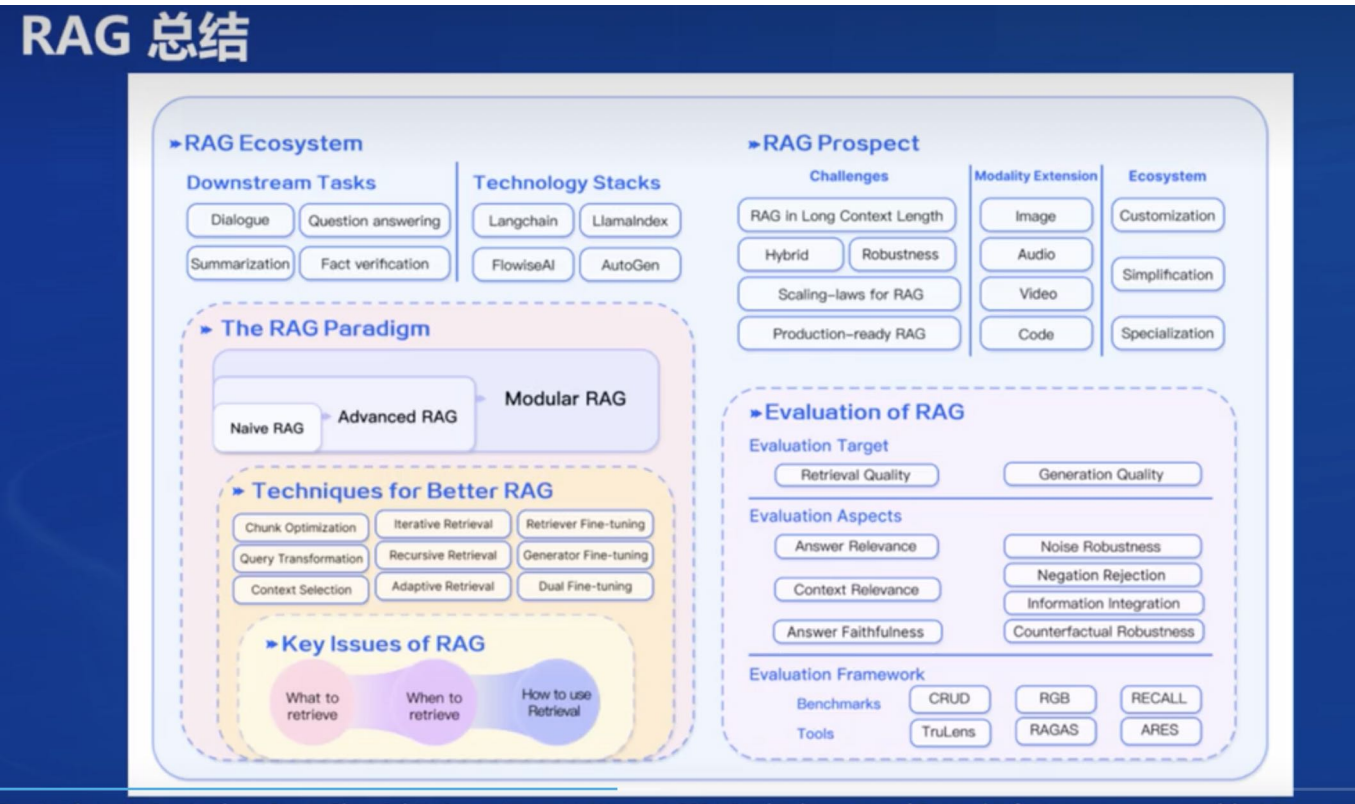




- RAG：高外部知识需求、低模型适配度要求
- 都高的话：三种技术结合

评估指标

评估框架和基准测试			
<div>经典评估指标：<ul style="list-style-type: none">○ 准确率 (Accuracy)○ 召回率 (Recall)○ F1分数 (F1 Score)○ BLEU分数 (用于机器翻译和文本生成)○ ROUGE分数 (用于文本生成的评估)</div> <div>RAG 评测框架：<ul style="list-style-type: none">○ 基准测试 - RGB、RECALL、CRUD○ 评测工具 - RAGAS、ARES、TruLens</div>			
Evaluation Framework	Evaluation Targets	Evaluation Aspects	Quantitative Metrics
RGB [†]	Retrieval Quality	Noise Robustness	Accuracy
	Generation Quality	Negative Rejection	EM
RECALL [‡]	Retrieval Quality	Information Integration	Accuracy
	Generation Quality	Counterfactual Robustness	Accuracy
RAGAS [‡]	Retrieval Quality	Context Relevance	*
	Generation Quality	Faithfulness	*
		Answer Relevance	Cosine Similarity
ARES [‡]	Retrieval Quality	Context Relevance	Accuracy
	Generation Quality	Faithfulness	Accuracy
TruLens [‡]	Retrieval Quality	Answer Relevance	Accuracy
	Generation Quality	Context Relevance	*
		Faithfulness	*
		Answer Relevance	*
CRUD [†]	Retrieval Quality	Creative Generation	BLEU
	Generation Quality	Knowledge-intensive QA	ROUGE-L
		Error Correction	BertScore
		Summarization	RAGQuestEval



茴香豆

- 豆哥：专为即时通讯工具群聊场景优化的工作流

茴香豆介绍

Hui

茴香豆是一个基于LLMs的领域知识助手，由书生浦语团队开发的开源大模型应用。

- 专为即时通讯（IM）工具中的群聊场景优化的工作流，提供及时准确的技术支持和自动化问答服务。
- 通过应用检索增强生成（RAG）技术，茴香豆能够理解和高效准确的回应与特定知识领域相关的复杂查询。

应用场景

- 智能客服：技术支持、领域知识对话
- IM工具中创建用户群组，讨论、解答相关的问题。
- 随着用户数量的增加，答复内容高度重复，充斥大量无意义和闲聊，人工回复，成本高，影响工作效率。
- 茴香豆通过提供自动化的问答支持，帮助维护者减轻负担，同时确保用户问题得到有效解答。

场景难点

- 群聊中的信息量巨大，且内容多样，从技术讨论到闲聊应有尽有。
- 用户问题通常与个人紧密相关，需要准确的专业知识解答。
- 传统的NLP解决方案无法准确解析用户意图，且往往无法提供满意的答案。
- 需要一个能够在群聊中准确识别与回答相关问题的智能助手，同时避免造成消息过载。

HuixiangDou

茴香豆

@所有人 web 版上线了，请试用。资源有限，可能要排队 qq

冲!

3/26/24 19:51

无权的八

3/26/24 20:24

aniv2401.08772 invited XiaoMin to the group chat

3/27/24 09:49

长大了的天维宝

mmsentation/mimseg/datasets/refuge.py 中，REFUGEDataset类的注释中写了“reduce_zero_label”是 fixed to True，但是代码中却将 reduce_zero_label 固定为 False，这是否为一个 bug?

茴香豆

mmsentation/mimseg/datasets/ref_

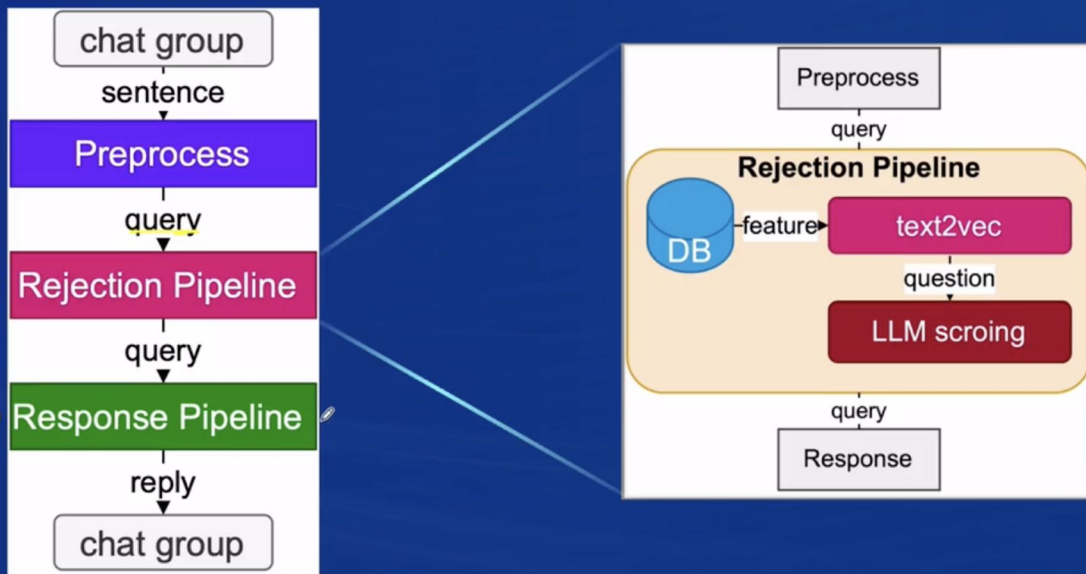
—

这可能是由于代码更新后文档没有及时更新导致的不一致。在实际代码中，reduce_zero_label 被设置为 False，因此应该以实际代码为准。

- 自动化问答、减轻消息过载
- 开源免费商用
- 专业知识快速获取

- 本地部署、保护数据隐私
- 兼容多种IM软件
- **拒答 workflow**：主要特色！看下代码是怎么实现的！

茴香豆的工作流



```

...
# Switch languages according to the scenario.
if self.language == 'zh':
    self.TOPIC_TEMPLATE = '告诉我这句话的主题，直接说主题不要解释：“{}”'
    self.SCORING_QUESTION_TEMPLTE = '“{}”\n请仔细阅读以上内容，判断句子
    是否是个有主题的疑问句，结果用 0~10 表示。直接提供得分不要解释。\n判断标
    准：有主语谓语宾语并且是疑问句得 10 分；缺少主谓宾扣分；陈述句直接得 0 分；
    不是疑问句直接得 0 分。直接提供得分不要解释。' # noqa E501
    self.SCORING_RELAVANCE_TEMPLATE = '问题：“{}”\n材料：“{}”\n请仔细阅
    读以上内容，判断问题和材料的关联度，用0~10表示。判断标准：非常相关得 10
    分；完全没关联得 0 分。直接提供得分不要解释。\n' # noqa E501
    self.KEYWORDS_TEMPLATE = '谷歌搜索是一个通用搜索引擎，可用于访问互联
    网、查询百科知识、了解时事新闻等。搜索参数类型 string， 内容是短语或关键
    字，以空格分隔。\n你现在是{}交流群里的技术助手，用户问“{}”，你打算通过谷歌
    搜索查询相关资料，请提供用于搜索的关键字或短语，不要解释直接给出关键字或短
    语。' # noqa E501
    self.SECURITY_TEMPLTE = '判断以下句子是否涉及政治、辱骂、色情、恐暴、
    宗教、网络暴力、种族歧视等违禁内容，结果用 0~10 表示，不要解释直接给出得
    分。判断标准：涉其中任一问题直接得 10 分；完全不涉及得 0 分。直接给得分不要
    解释：“{}”' # noqa E501
    self.PERPLESITY_TEMPLATE = '“question:{} answer:{}”\n阅读以上对
    话，answer 是否在表达自己不知道，回答越全面得分越少，用0~10表示，不要解释
    直接给出得分。\n判断标准：准确回答问题得 0 分；答案详尽得 1 分；知道部分答
    案但有不确定信息得 8 分；知道小部分答案但推荐求助其他人得 9 分；不知道任何
    答案直接推荐求助别人得 10 分。直接打分不要解释。' # noqa E501
    self.SUMMARIZE_TEMPLATE = '{} \n 仔细阅读以上内容，总结得简短有力点'
    # noqa E501
    # self.GENERATE_TEMPLATE = '材料：“{}”\n 问题：“{}” \n 请仔细阅读参
    考材料回答问题，材料可能和问题无关。如果材料和问题无关，尝试用你自己的理解来
    回答问题。如果无法确定答案，直接回答不知道。' # noqa E501
    self.GENERATE_TEMPLATE = '材料：“{}”\n 问题：“{}” \n 请仔细阅读参考
    材料回答问题。' # noqa E501
...

```