大模型的探索与实践

Introduction to Large Language Models

§ 2.4 RAG

Retrieval Augmented Generation

滕佳烨
上海财经大学
www.tengjiaye.com

回顾 Recall

RL

- Agent, Environment
- State, Action, Reward
- Policy Iteration, Value Iteration
- No transition matrix → Sampling
- No policy function → DL

RLHF

- Actor Model, Reference Model, Reward Model, Critic Model

今天的任务：RAG

The Challenge: Knowledge Boundaries in GenAI

⚠ Knowledge Lag

LLMs rely on static pre-training data snapshots. They cannot access real-time information (e.g., current stock prices, breaking news), making them unsuitable for dynamic business environments without external aid.

Hallucinations

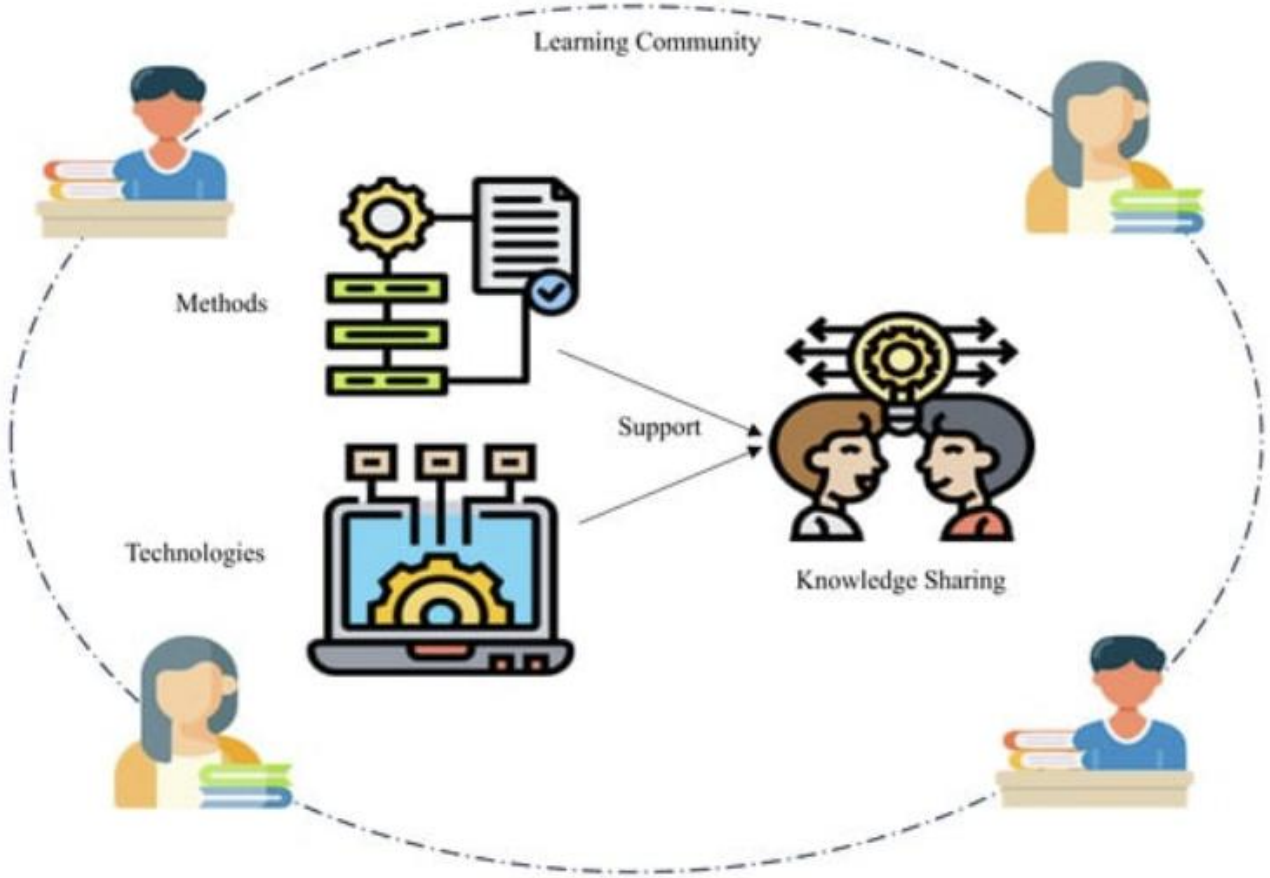
Generative models prioritize fluency over factuality. In high-stakes industries like finance or healthcare, confident but incorrect answers ("hallucinations") pose unacceptable risks to trust and compliance.

The RAG Solution

Bridging Parametric & Non-Parametric Memory

RAG connects the LLM's internal linguistic capabilities (parametric memory) with external, verifiable knowledge bases (non-parametric memory).

By enforcing citations and referencing external data, RAG acts as a **"Factual Guardrail,"** transforming output from creative writing to auditable reporting.



RAG

- 检索 Retrieval：当用户交互时，从外部知识库中检索与用户相关的内容
- 增强 Augmentation：将检索内容与用户输入结合，扩展上下文
- 生成 Generation：基于增强后内容生成最终回答

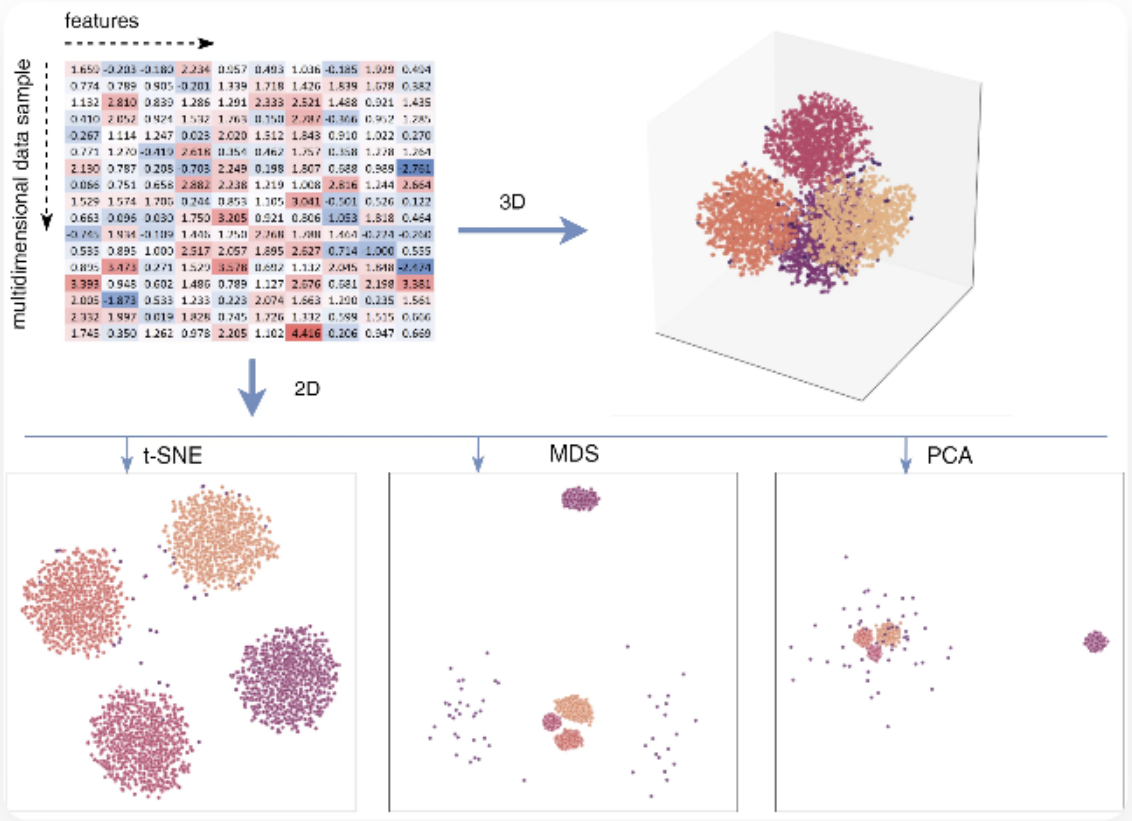
* 能够快速部署 (c.f. finetuning)

Retrieval Mechanics: From Sparse to Dense

- **Sparse Retrieval (BM25):** Keywords based. Exact matches. Good for specific acronyms but misses semantic meaning.
- **Dense Retrieval (Embeddings):** Vector based. Captures semantic intent even without keyword overlap.
- **Hybrid Search:** Best of both worlds. Combines BM25 precision with Vector semantic breadth.

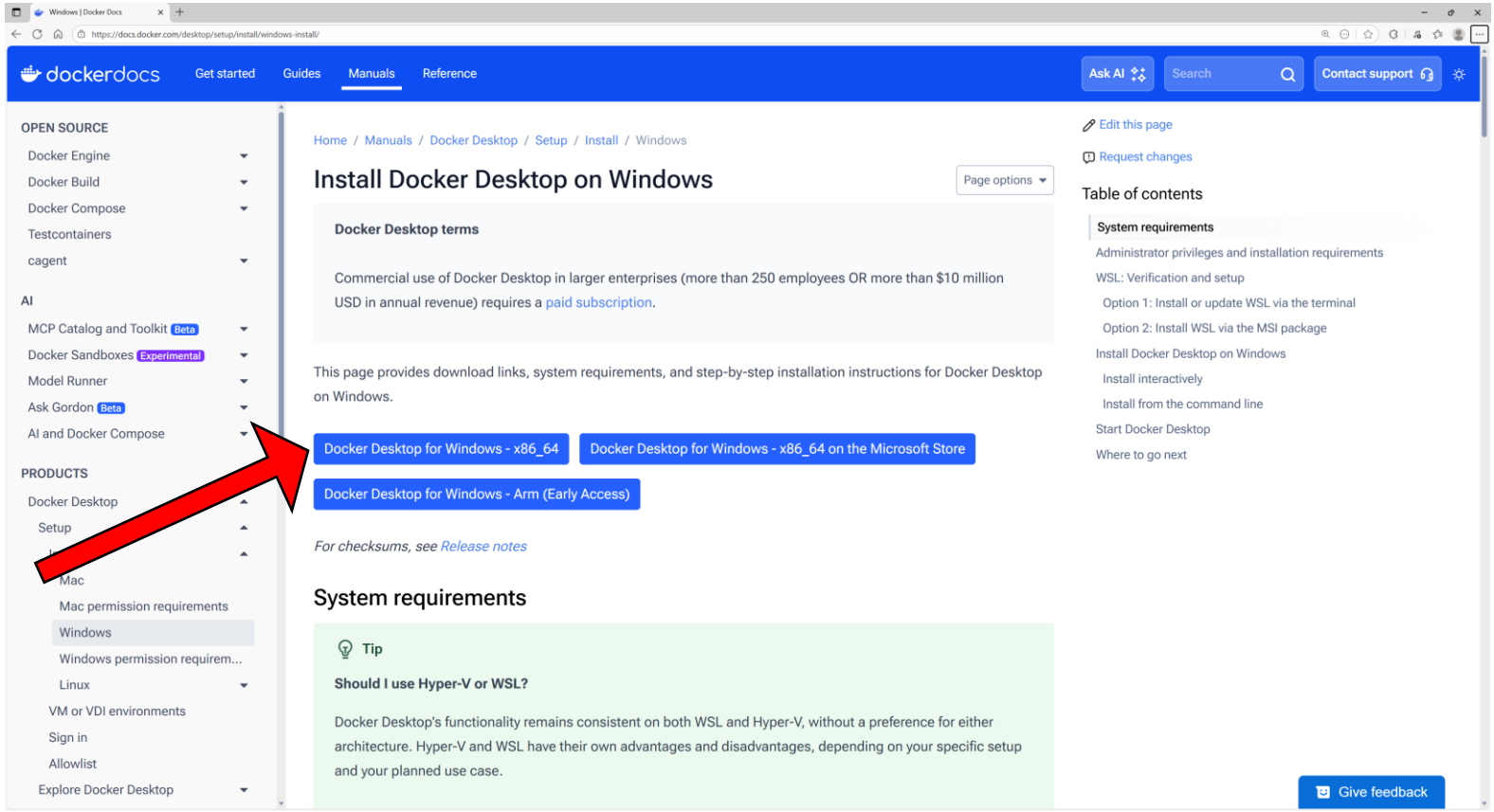
$$\text{Similarity}\left(A, B\right) = \frac{A \cdot B}{||A||||B||}$$

Cosine Similarity Formula



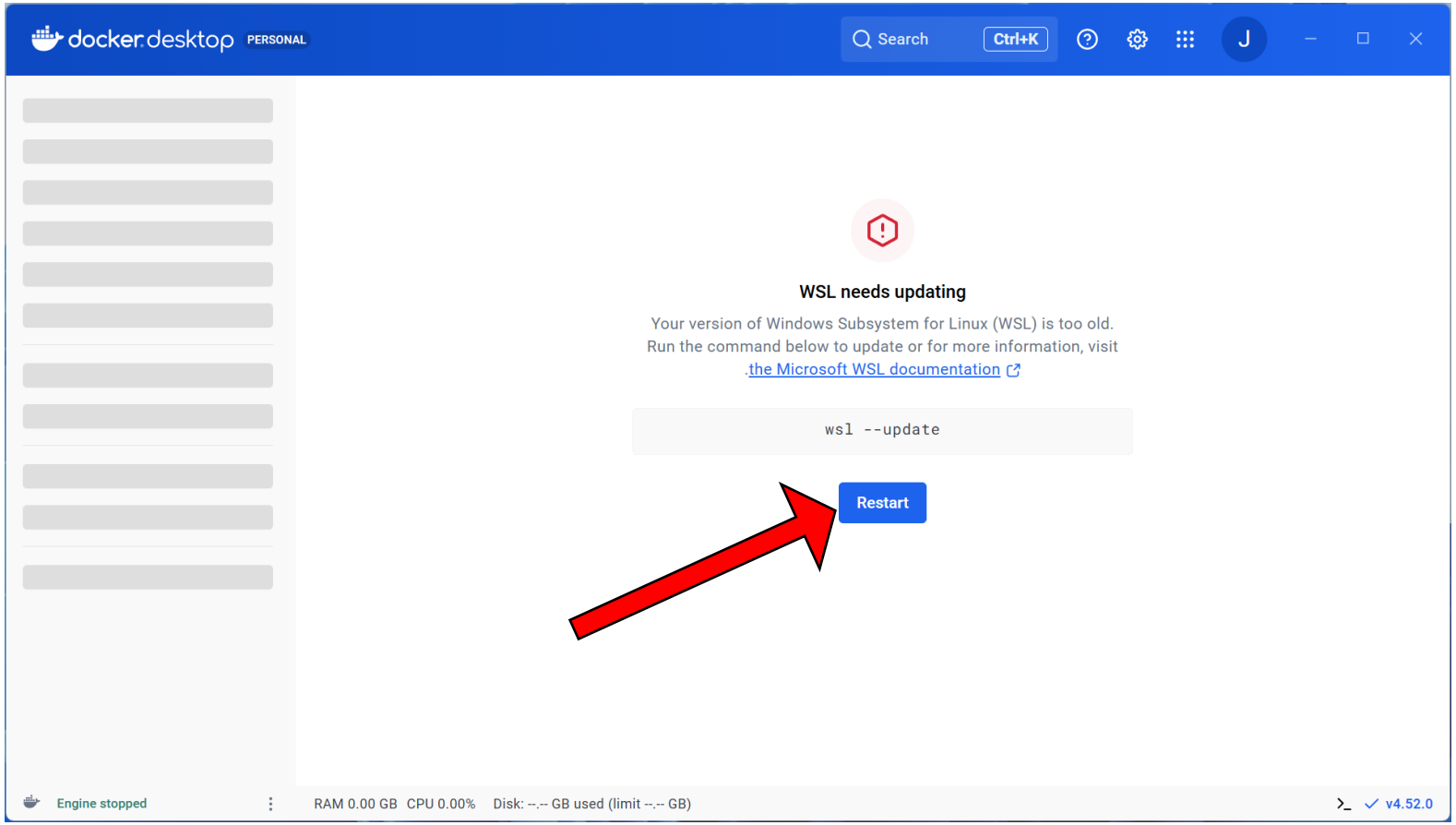
RAG with ragflow

1. 下载（Windows） Docker （一个封装好的环境）



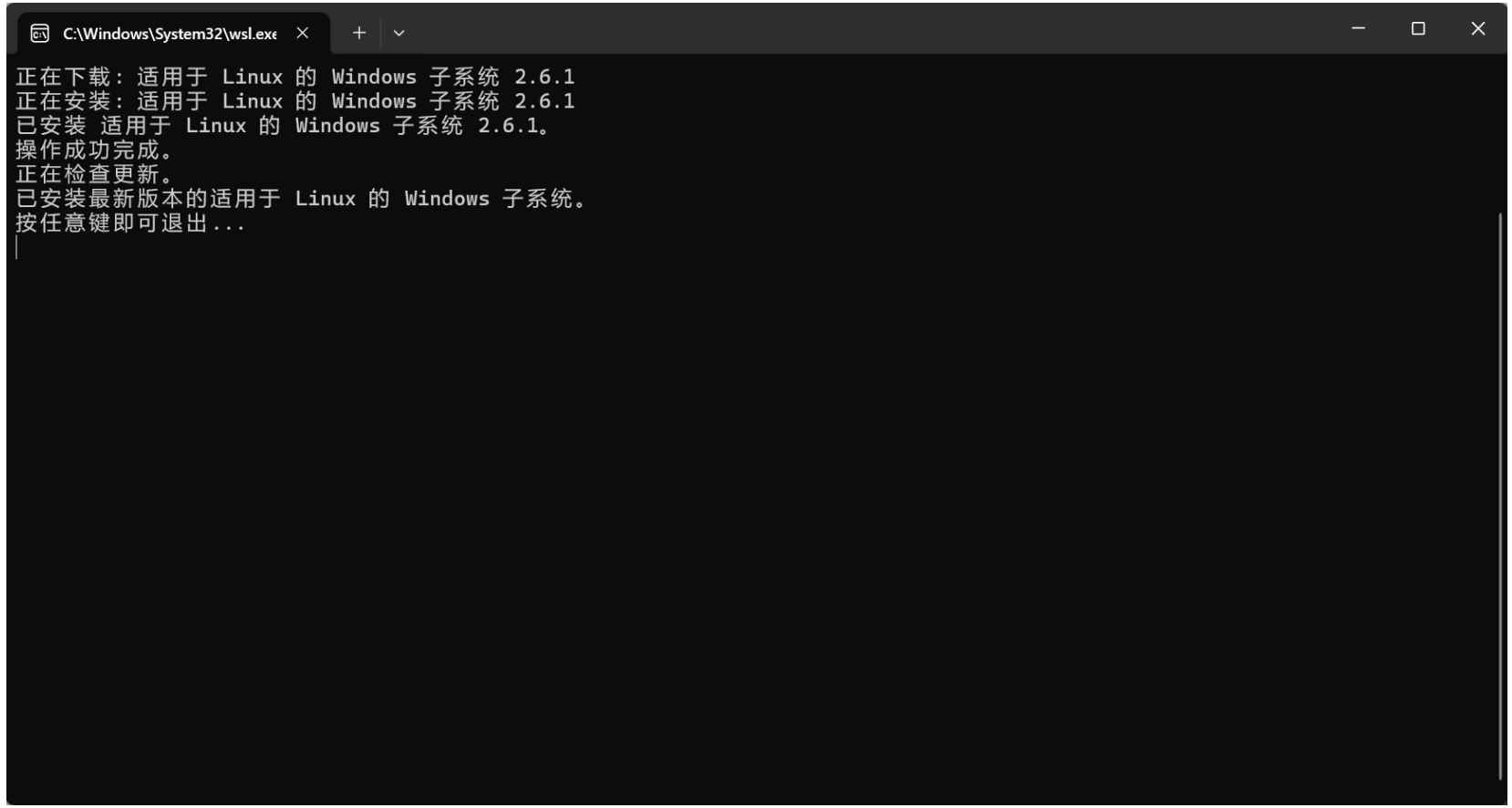
RAG with ragflow

1. 下载Windows Docker



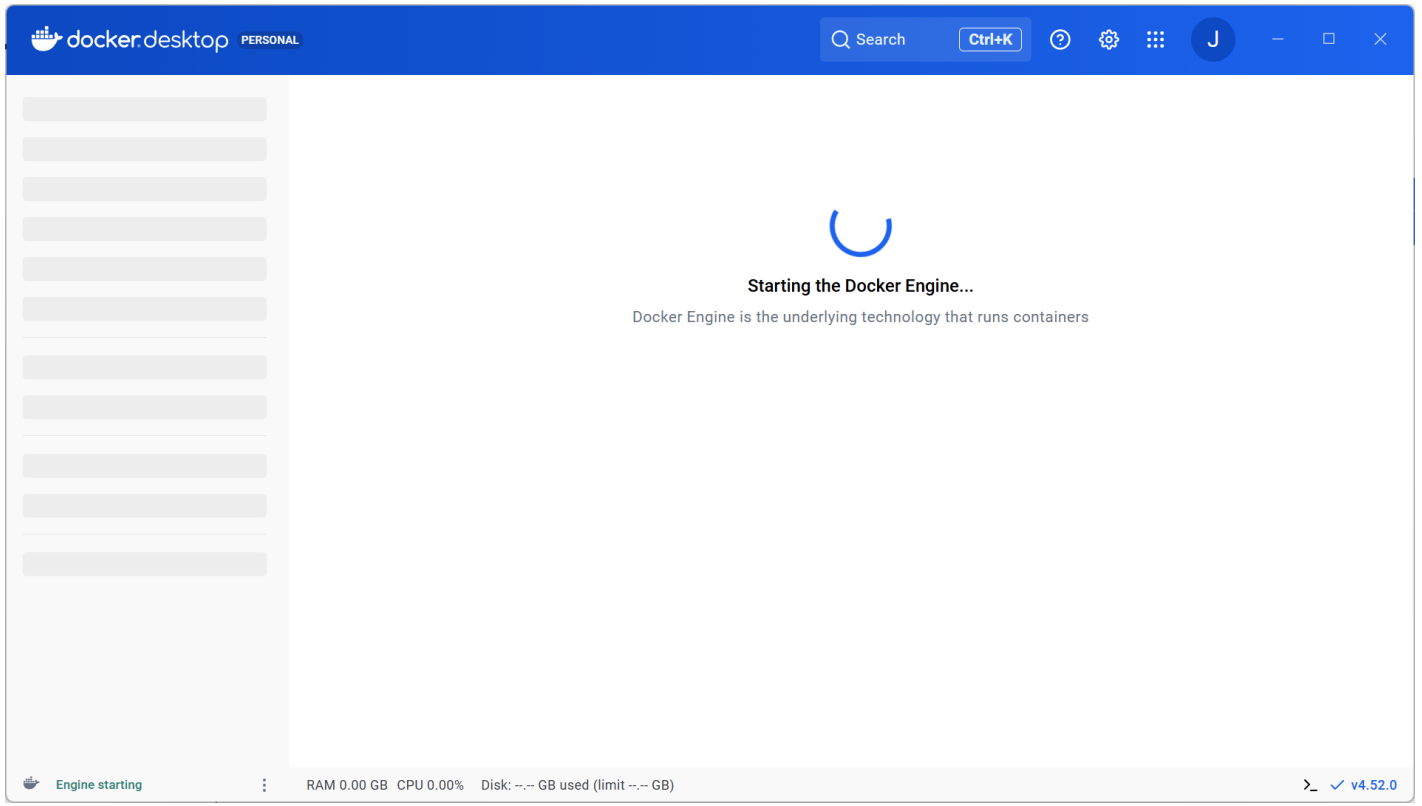
RAG with ragflow

1. 下载Windows Docker



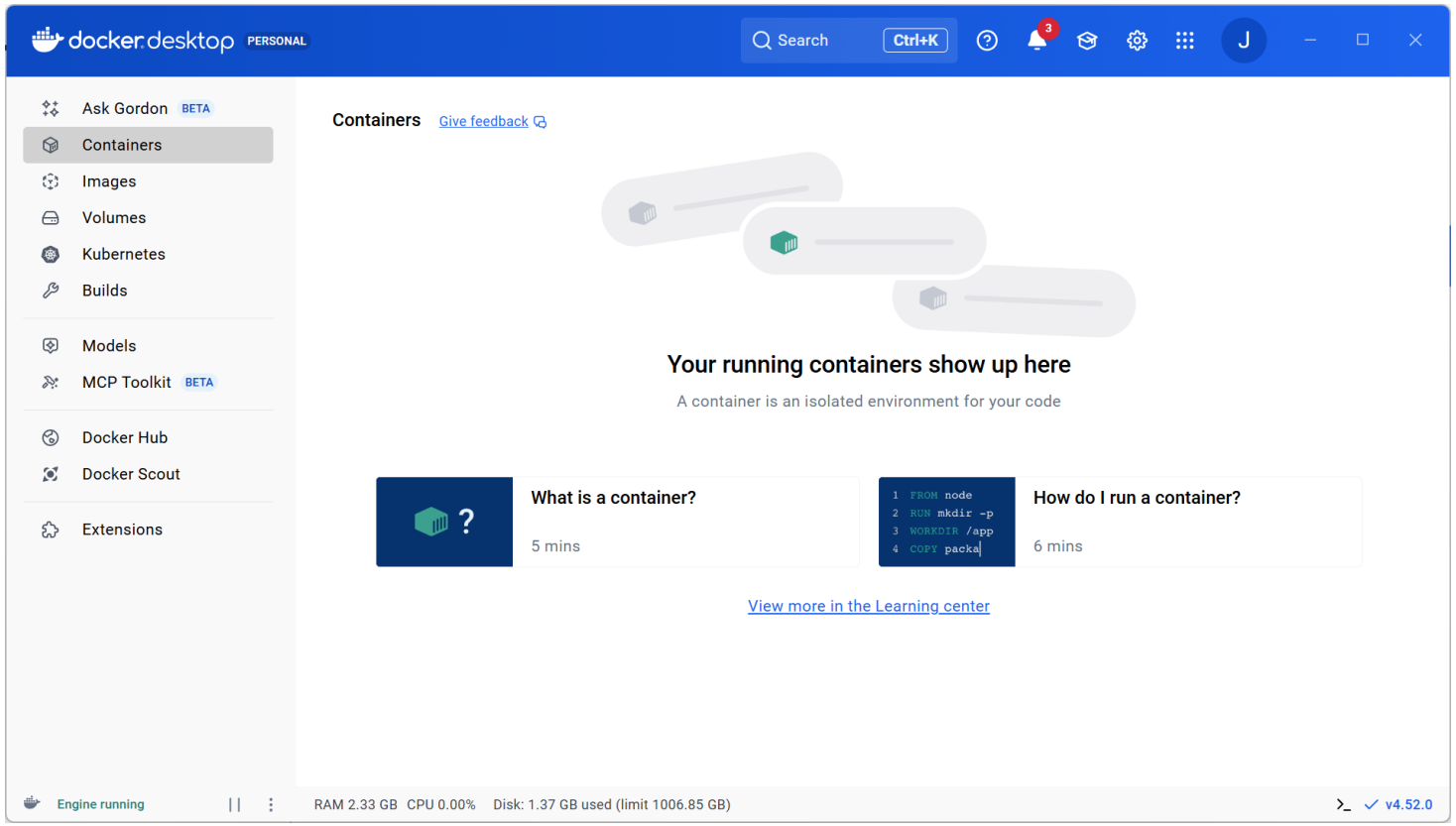
RAG with ragflow

1. 下载Windows Docker



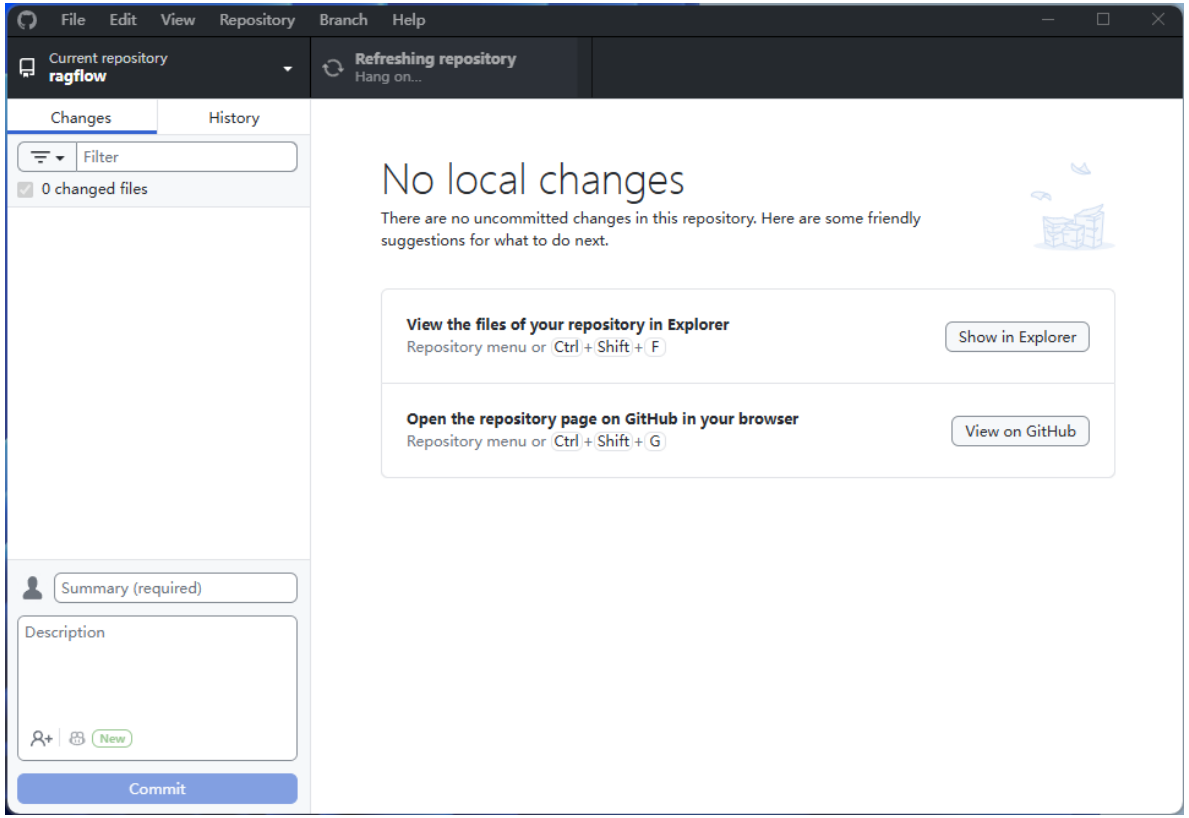
RAG with ragflow

1. 下载Windows Docker



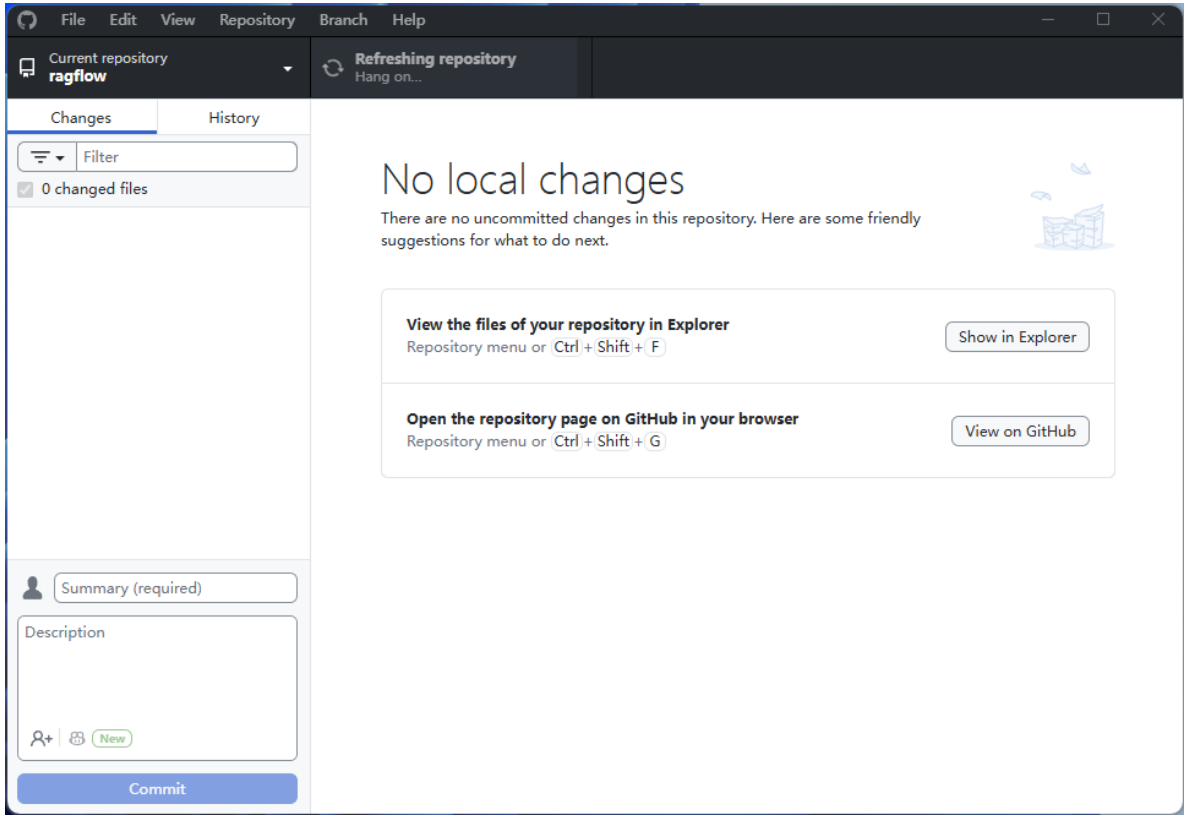
RAG with ragflow

2. Clone repo to your local computer



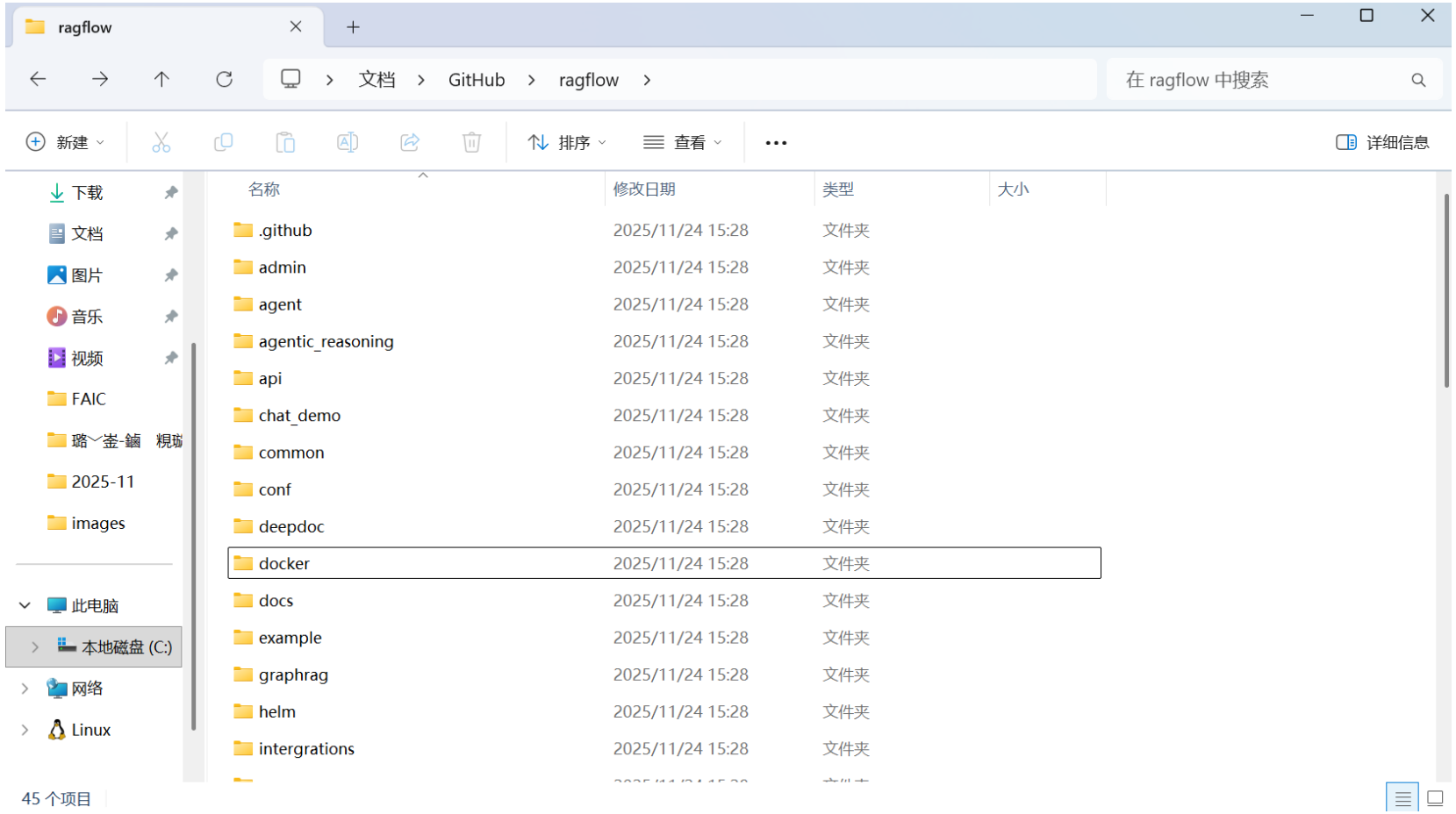
RAG with ragflow

2. Clone repo to your local computer



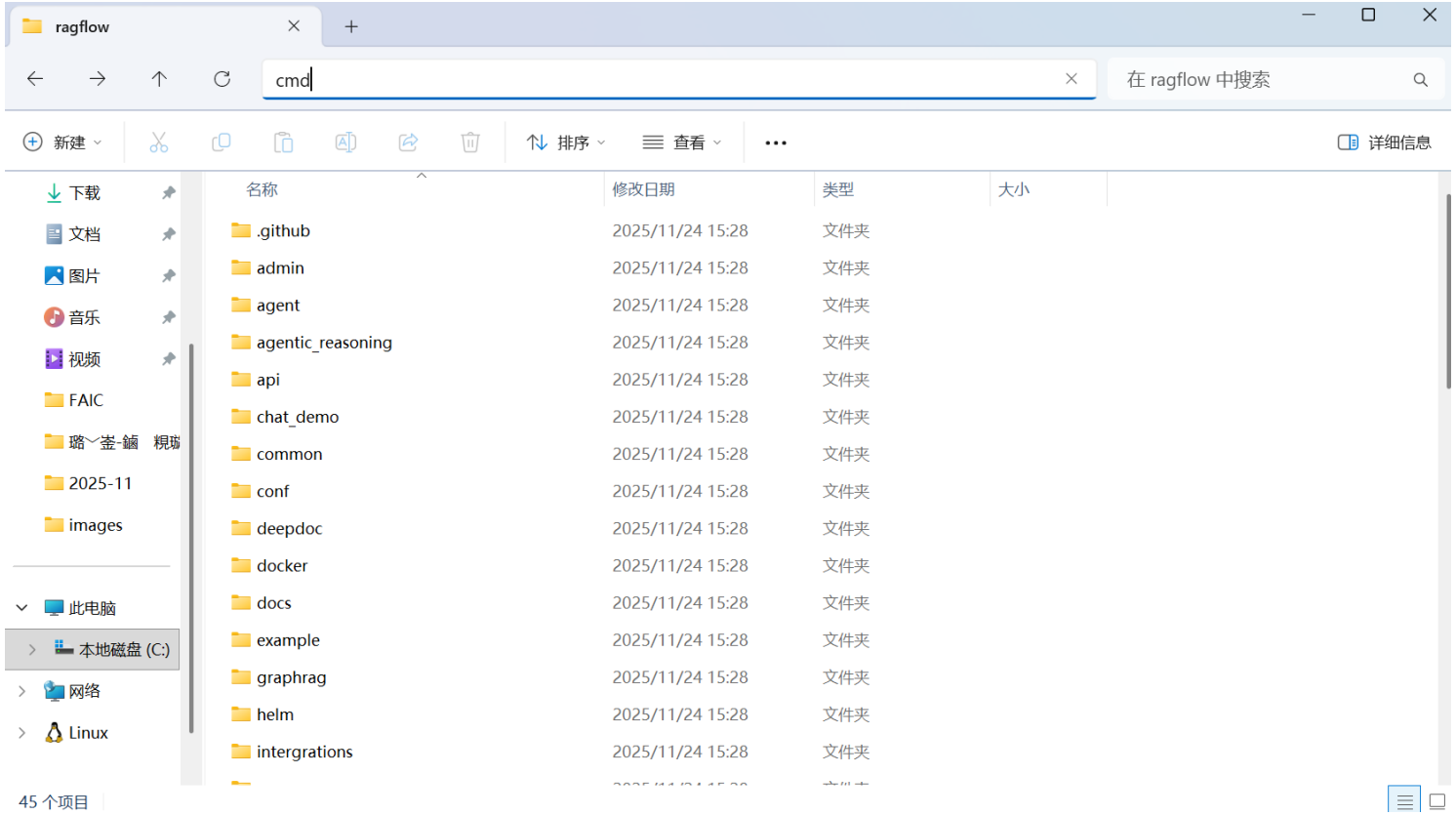
RAG with ragflow

2. Clone repo to your local computer



RAG with ragflow

3. Link ragflow and docker



RAG with ragflow

3. Link ragflow and docker

```
C:\Users\Jiaye\Documents\GitHub\ragflow\docker>docker compose -f docker-compose.yml up -d
time="2025-11-24T16:01:49+08:00" level=warning msg="The \"HOME\" variable is not set. Defaulting to a blank string."
time="2025-11-24T16:01:49+08:00" level=warning msg="The \"HOME\" variable is not set. Defaulting to a blank string."
time="2025-11-24T16:01:49+08:00" level=warning msg="The \"HOME\" variable is not set. Defaulting to a blank string."
time="2025-11-24T16:01:49+08:00" level=warning msg="The \"HOME\" variable is not set. Defaulting to a blank string."
time="2025-11-24T16:01:49+08:00" level=warning msg="The \"HOME\" variable is not set. Defaulting to a blank string."
time="2025-11-24T16:01:49+08:00" level=warning msg="The \"HOME\" variable is not set. Defaulting to a blank string."
time="2025-11-24T16:01:49+08:00" level=warning msg="The \"HOME\" variable is not set. Defaulting to a blank string."
time="2025-11-24T16:01:49+08:00" level=warning msg="The \"HOME\" variable is not set. Defaulting to a blank string."
time="2025-11-24T16:01:49+08:00" level=warning msg="The \"HOME\" variable is not set. Defaulting to a blank string."
time="2025-11-24T16:01:49+08:00" level=warning msg="The \"HOME\" variable is not set. Defaulting to a blank string."
[+] Running 72/72
 ✔ragflow-cpu Pulled                                          183.0s
 ✔minio Pulled                                                 13.5s
 ✔redis Pulled                                                 15.5s
 ✔es01 Pulled                                                 127.5s
 ✔mysql Pulled                                                 29.8s

[+] Running 10/10
 ✔Network docker_ragflow          Created                       0.0s
 ✔Volume docker_mysql_data        Created                       0.0s
 ✔Volume docker_minio_data        Created                       0.0s
 ✔Volume docker_redis_data        Created                       0.0s
 ✔Volume docker_esdata01          Created                       0.0s
 ✔Container docker-es01-1         Started                       1.1s
 ✔Container docker-mysql-1        Healthy                      21.6s
 ✔Container docker-redis-1        Started                       1.1s
 ✔Container docker-minio-1        Started                       1.1s
 ✔Container docker-ragflow-cpu-1  Started                      21.5s

C:\Users\Jiaye\Documents\GitHub\ragflow\docker>
```

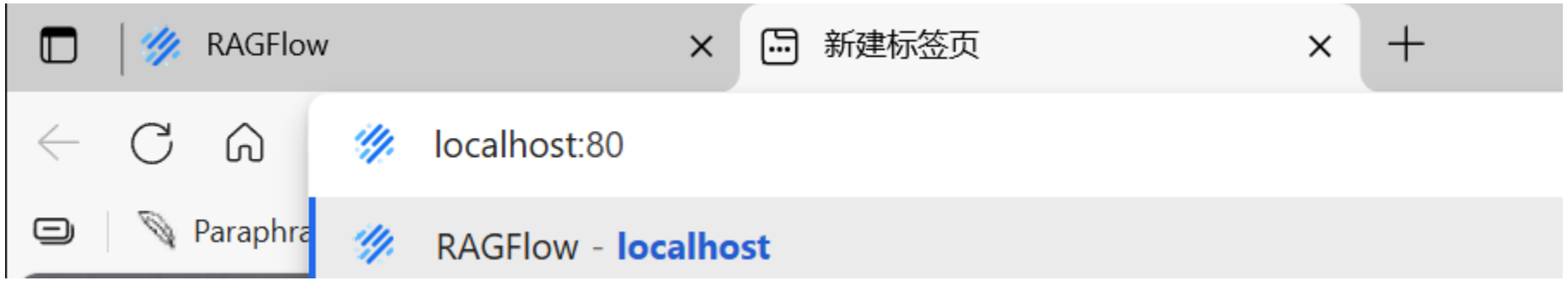
docker compose -f docker-compose.yml up -d

可能会出问题，若如此，可以使用国内镜像；需要进入到docker文件夹（cd docker）

RAG with ragflow

3. Link ragflow and docker

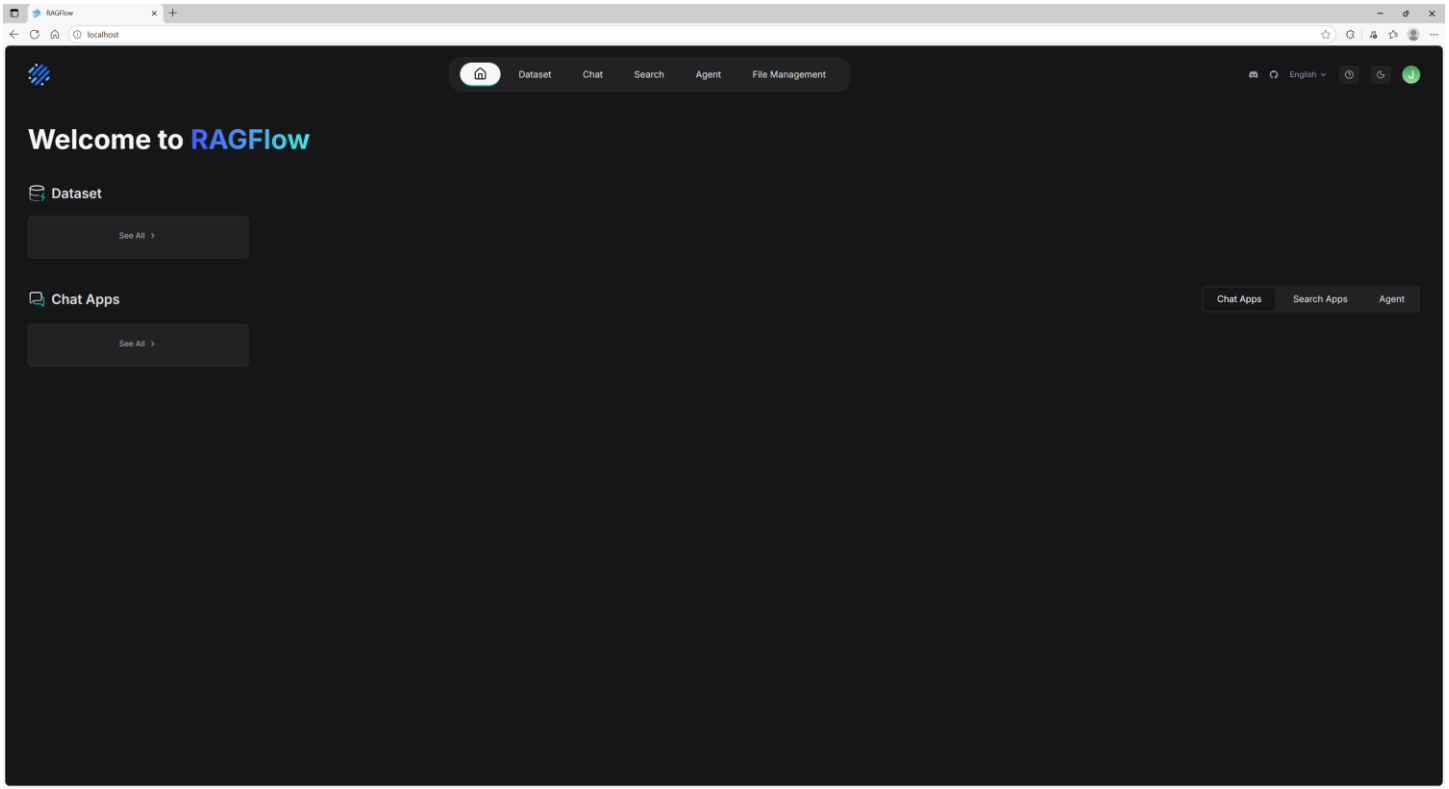
网页中输入　localhost:80　　进入虚拟机



RAG with ragflow

3. Link ragflow and docker

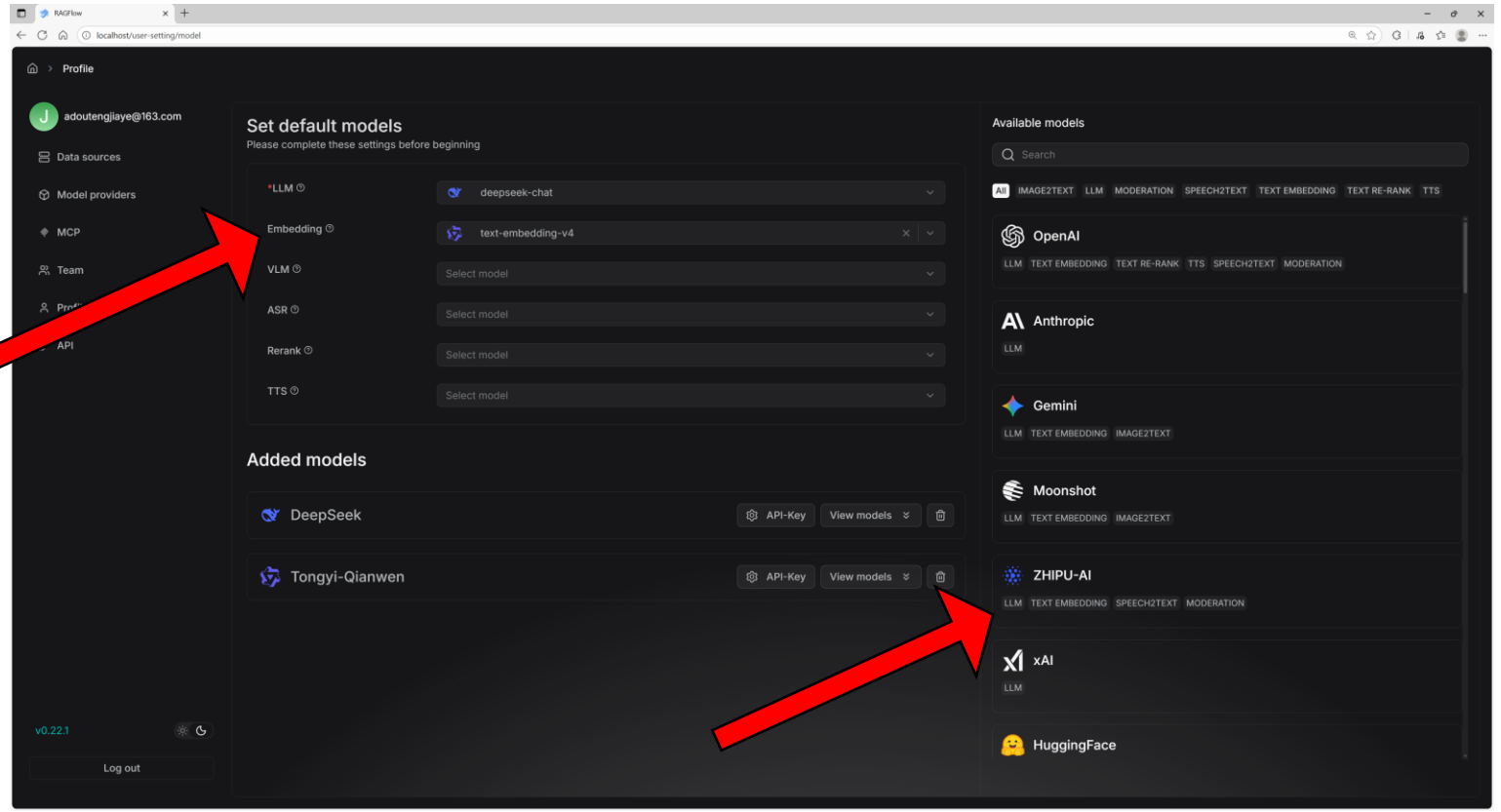
注册登录后出现界面：



RAG with ragflow

4. 使用 ragflow

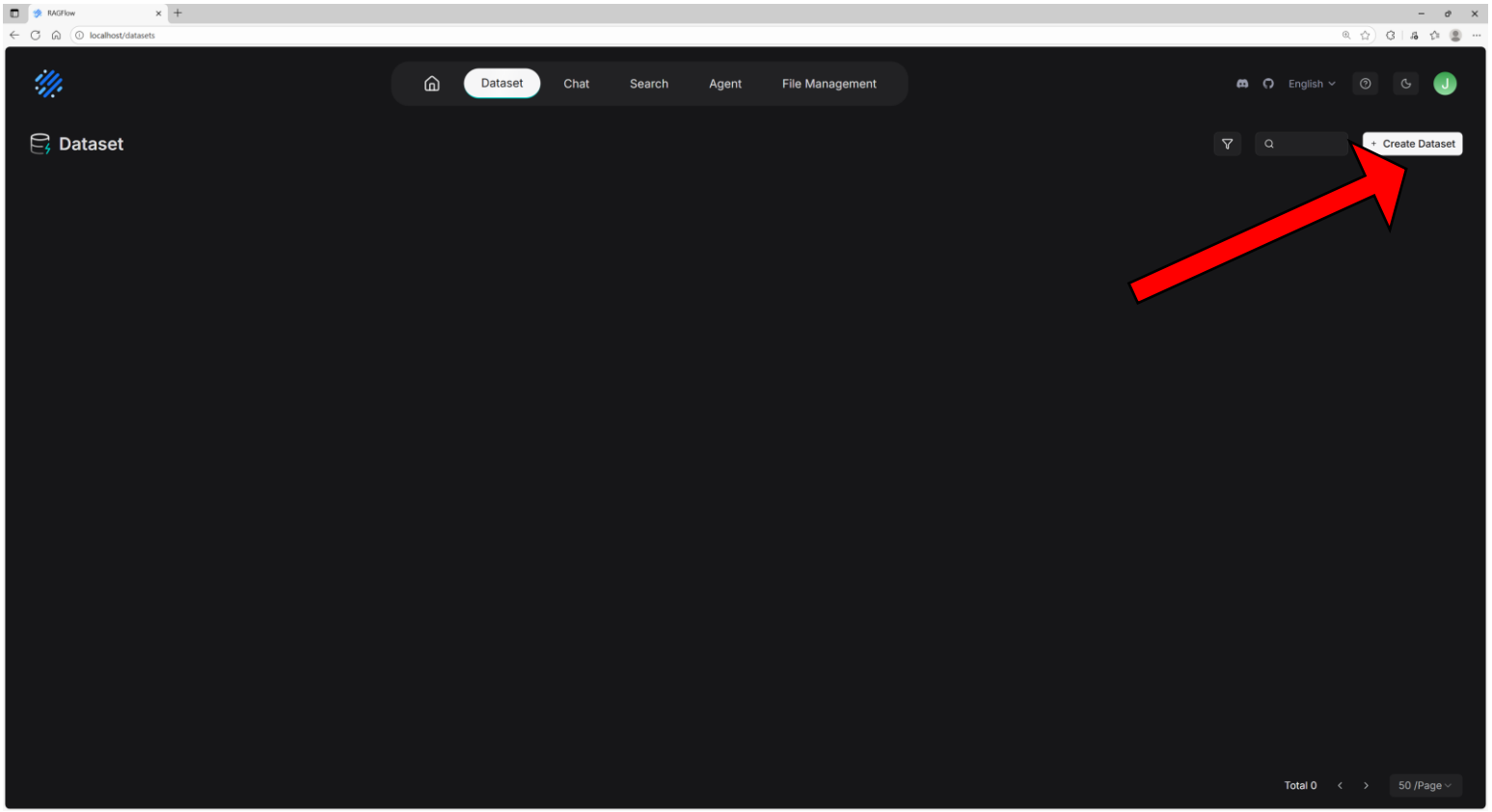
4.1 添加大模型基座 (和API keys)



注意：必须得有embedding能力

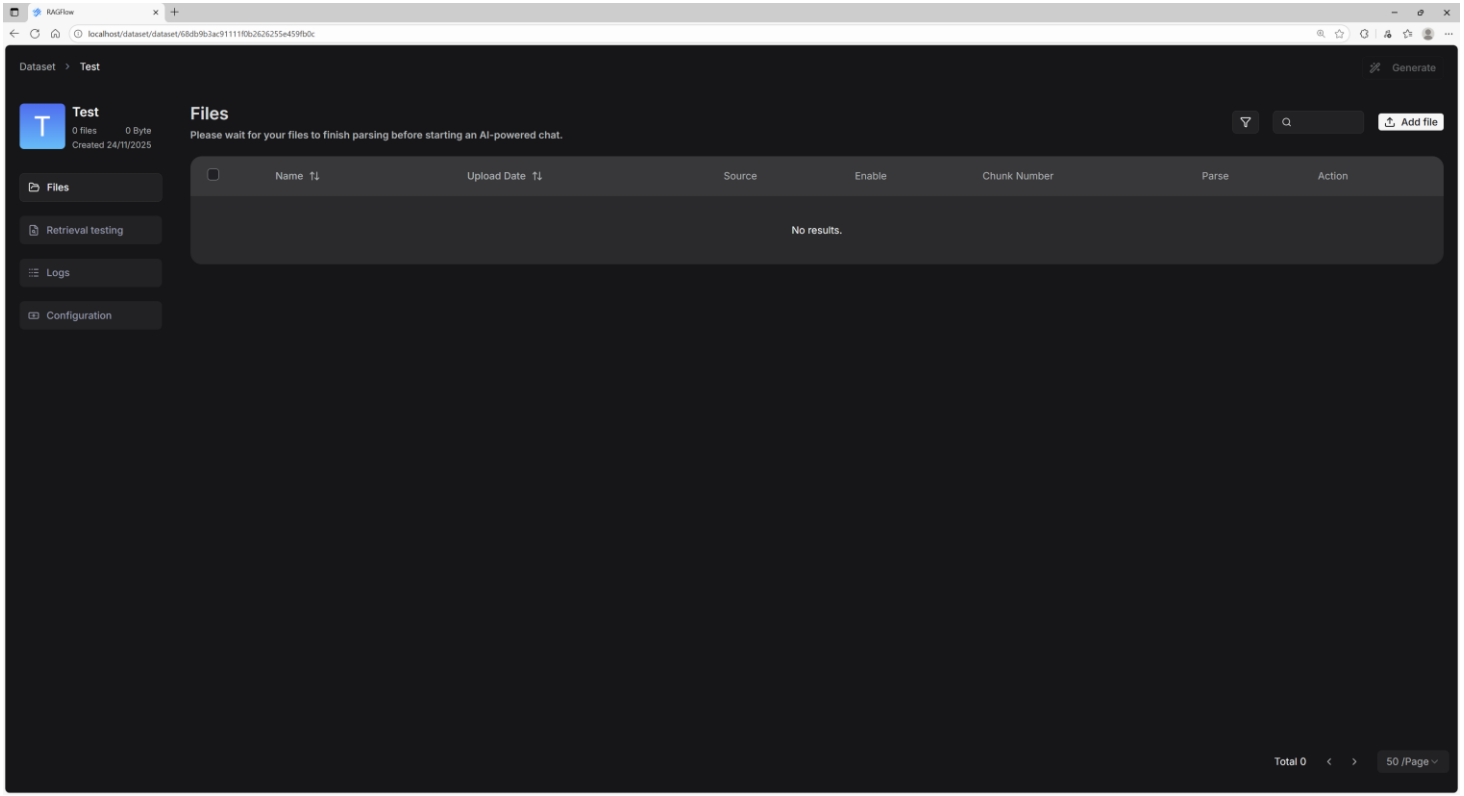
RAG with ragflow

4. 使用 ragflow
4.2 添加（私有）数据库



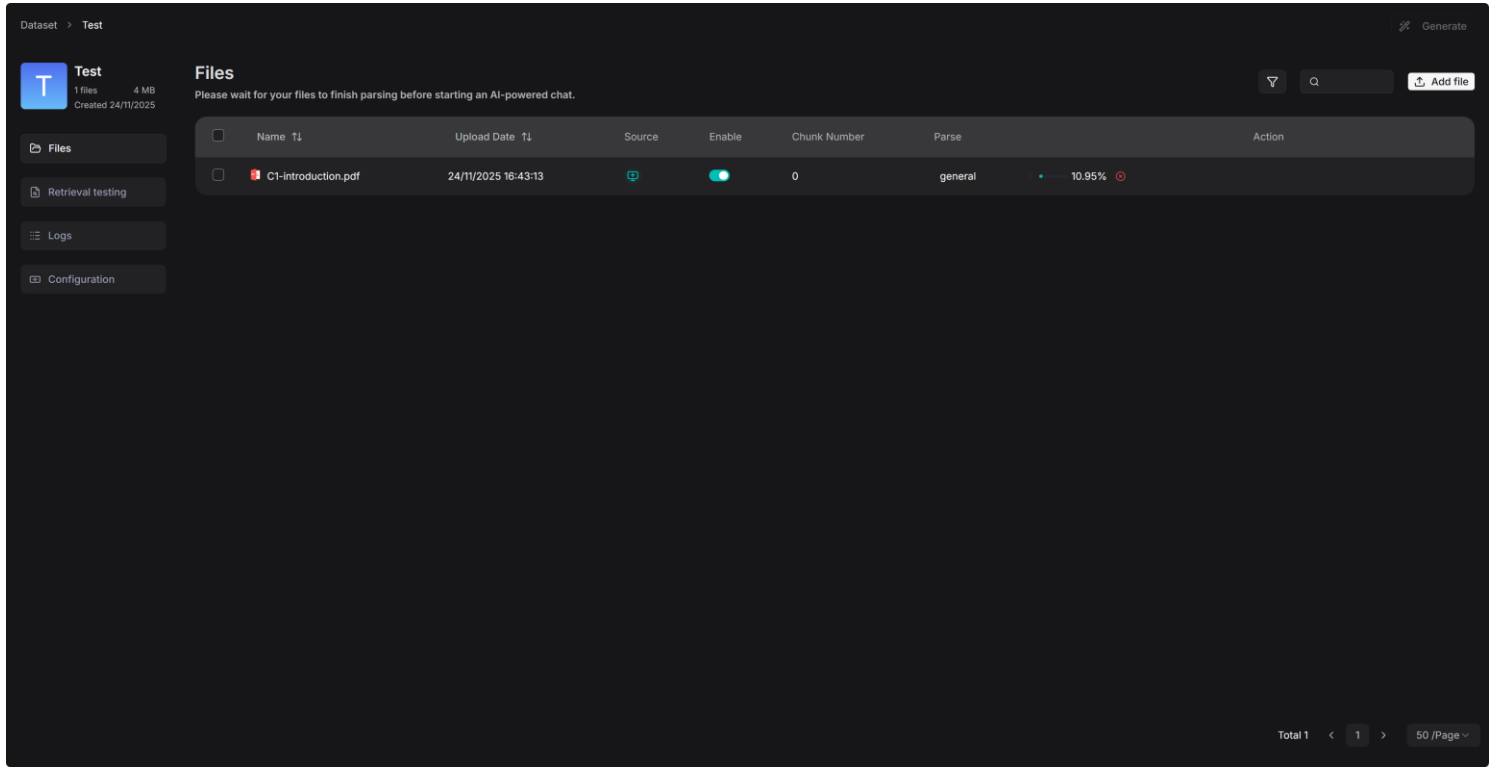
RAG with ragflow

4. 使用 ragflow
4.2 添加（私有）数据库



RAG with ragflow

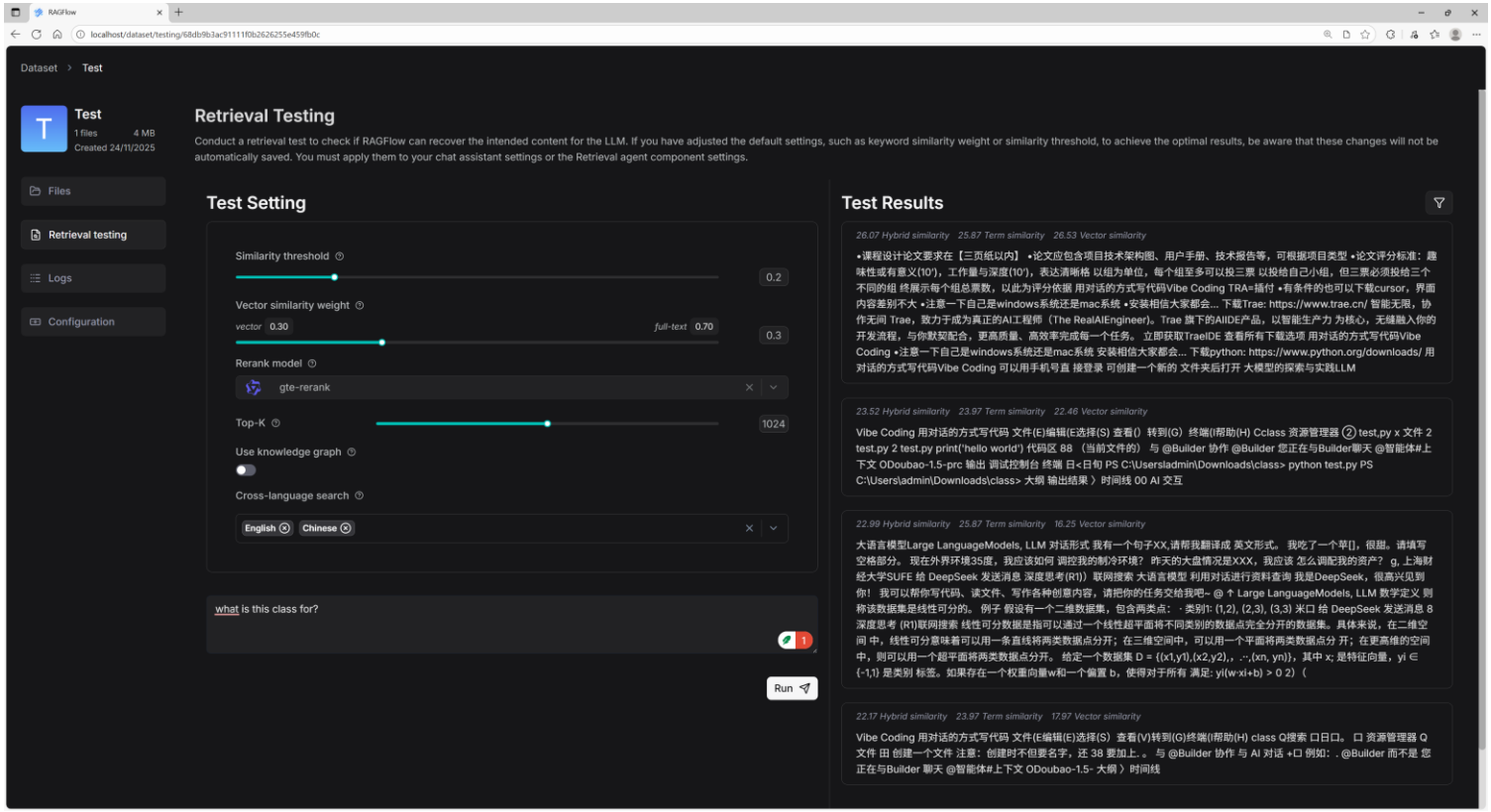
4. 使用 ragflow
4.2 添加（私有）数据库



需要解析后才能使用

RAG with ragflow

4. 使用 ragflow
4.2 添加（私有）数据库



最好进行调试

RAG with ragflow

4. 使用 ragflow
4.3 创建自己的个人助手！

总结 Take-away Messages

RAG
- Retrieval
- Augmentation
- Generation

RAG with ragflow

第九次作业：部署一个个人助手