# 大模型及 Dify 平台介绍

## 一、大模型概述

### 1.1定义

大模型，是指利用海量数据，通过先进的算法和技术，训练得到的具有强大预测和决策能力的模型。具体而言，AI 大模型是 “大数据 + 大算力 + 强算法” 结合的产物，是一种能够利用大数据和神经网络来模拟人类思维和创造力的人工智能算法。它运用海量的数据和深度学习技术来理解、生成和预测新内容，通常拥有数百亿乃至数万亿个参数，具备在不同领域和任务中展现智能的能力 。例如，在自然语言处理方面，大模型可以理解人类语言的语义、语法，生成流畅自然的文本；在图像识别领域，能够精准识别图像中的物体、场景等 。

### 1.2特点

1. **巨大的规模**：大模型包含数十亿个参数，模型大小可达数百 GB 甚至更大。庞大的模型规模赋予大模型强大的表达能力和学习能力，使其能够学习到数据中复杂的模式和特征 。
2. **涌现能力**：当模型的训练数据突破一定规模，会突然展现出之前小模型所没有的、意料之外的复杂能力和特性，能够综合分析和解决更深层次问题，呈现出类似人类的思维和智能，这是大模型区别于小模型的显著特点之一 。
3. **更好的性能和泛化能力**：大模型通常具有更强大的学习能力和泛化能力，能够在各种任务上表现出色，无论是自然语言处理中的文本分类、机器翻译，还是图像识别中的目标检测、图像分割，亦或是语音识别等任务，都能取得较好的效果 。
4. **多任务学习**：大模型通常会同时学习多种不同的任务，比如在自然语言处理领域，能够同时进行机器翻译、文本摘要、问答系统等任务的学习，从而使模型掌握更广泛和通用的语言理解能力 。
5. **大数据训练**：大模型需要海量的数据来训练，数据集通常在 TB 以上甚至达到 PB 级别。只有大量的数据输入，才能充分发挥大模型参数规模的优势，学习到丰富多样的知识和模式 。
6. **强大的计算资源**：训练大模型通常需要数百甚至上千个 GPU，以及大量的时间，一般训练周期在几周到几个月不等。这对计算设备的性能和计算资源的投入要求极高 。
7. **迁移学习和预训练**：大模型可以先在大规模数据上进行预训练，学习到通用的知识和特征，然后在特定任务上进行微调，从而快速适应新任务，提高模型在新任务上的性能 。
8. **自监督学习**：大模型能够通过自监督学习在大规模未标记数据上进行训练，减少对标记数据的依赖，提高模型训练的效率和效能 。
9. **领域知识融合**：大模型可以从多个领域的数据中学习知识，并将这些知识应用到不同领域，促进跨领域的创新和应用 。
10. **自动化和效率**：大模型可以自动化许多复杂的任务，如自动编程、自动翻译、自动摘要等，大大提高工作效率，减少人力成本 。

### 1.3分类

1. **按输入数据类型分类**

* **语言大模型（NLP）**：在自然语言处理领域，这类大模型在大规模语料库上进行训练，学习自然语言的语法、语义和语境规则，用于处理文本数据和理解自然语言，常见应用有文本生成、机器翻译、问答系统等 。
* **视觉大模型（CV）**：应用于计算机视觉领域，通过在大规模图像数据上训练，实现图像分类、目标检测、图像分割、姿态估计、人脸识别等视觉任务 。
* **多模态大模型**：能够处理多种不同类型数据，如文本、图像、音频等多模态数据，综合理解和分析多模态信息，例如 DingoDB 多模向量数据库、DALL - E、悟空画画、midjourney 等 。

1. **按应用领域分类**

* **通用大模型 L0**：可在多个领域和任务通用，利用大算力在海量开放数据上训练，形成强大的泛化能力，无需或只需少量微调即可完成多场景任务，如同 AI 完成了 “通识教育” 。
* **行业大模型 L1**：针对特定行业或领域，使用行业相关数据进行预训练或微调，以提高在该领域的性能和准确度，使 AI 成为 “行业专家” 。
* **垂直大模型 L2**：针对特定任务或场景，使用任务相关数据进行预训练或微调，提升在该任务上的性能和效果 。

## 二、大模型的应用领域

### 2.1自然语言处理领域

1. **对话系统**：如智能客服，能够理解用户问题并提供准确回答，提高客户服务效率和质量。例如，金融机构的客服通过大模型与客户流畅对话，解答账户查询、业务办理、产品咨询等问题 。
2. **自动翻译**：实现不同语言之间的快速准确翻译，促进跨语言交流。像一些翻译软件借助大模型可以支持多种语言的互译，满足人们日常交流和商务沟通需求 。
3. **语音识别**：将语音转换为文本，广泛应用于语音助手、智能家电控制等场景。比如，用户通过语音指令控制智能音箱播放音乐、查询信息等 。
4. **文本生成**：包括新闻写作、剧本创作、诗歌生成、文案撰写等。例如，一些媒体利用大模型生成新闻稿件，快速报道事件；广告公司借助大模型创作广告文案 。
5. **语义分析**：分析文本的语义、情感倾向等，帮助企业了解客户反馈、市场舆情等。例如，通过分析社交媒体上的文本，判断用户对某产品的满意度和情感态度 。

### 2.2计算机视觉领域

1. **图像识别**：识别图像中的物体、场景、人物等，应用于安防监控、交通违章识别、医疗影像诊断等。例如，安防系统通过图像识别技术识别闯入的陌生人；医疗领域通过图像识别辅助医生检测疾病 。
2. **图像生成**：根据给定的文本描述或其他条件生成图像，在设计、娱乐等行业有广泛应用。如设计师利用图像生成大模型快速生成设计草图，游戏公司生成游戏场景和角色图像 。
3. **图像增强**：提高图像的质量，如清晰度、对比度等，可用于老照片修复、监控视频优化等。例如，将模糊的监控图像进行增强处理，以便更清晰地识别目标 。
4. **人脸识别**：用于身份验证、门禁系统、考勤管理等。在机场、火车站等场所，通过人脸识别快速验证旅客身份 。

### 2.3其他领域

1. **医疗领域**

* **智能诊断辅助**：深度学习海量医疗影像数据，辅助医生精确识别病症特征，如肺部结节探测、骨折识别等，提高诊断准确性和效率，尤其在早期疾病筛查中发挥重要作用 。
* **药物研发优化**：分析药物分子结构与活性关系，预测药物疗效和副作用，加速新药研发进程，缩短研发周期，降低成本 。
* **个性化医疗方案制定**：结合患者基因数据、病史、生活习惯等，量身定制治疗方案，包括药物选择、剂量确定、治疗流程规划等，提高治疗效果，减少不良反应 。
* **医疗智能机器人手术协助**：操控手术机器人在复杂手术中精准操作，如脑部微创手术、心脏搭桥手术等，降低手术风险，减少患者创伤，提升手术成功率和康复效果 。
* **健康管理与疾病预测**：持续分析个人健康数据，预测疾病发生风险，为用户提供个性化健康建议，实现疾病预防和健康促进 。

1. **金融领域**

* **智能投资顾问服务**：基于市场趋势分析、宏观经济数据解读以及企业财报研究，为投资者定制个性化投资组合，实时追踪市场动态并优化投资策略，实现财富稳健增长 。
* **风险评估与预警系统**：精确评估贷款申请人信用风险、企业财务风险以及市场系统性风险，超前识别潜在风险，触发预警机制，帮助金融机构防范损失 。
* **金融市场走势预测**：分析金融市场各类数据，预测股票、外汇、期货等市场价格走势和波动情况，为投资者制定交易策略提供参考 。
* **智能客服提升服务效率**：在金融机构客服领域，运用自然语言处理技术解答客户问题，收集反馈，优化产品与服务，提升客户满意度和忠诚度 。
* **反欺诈智能监测与拦截**：实时识别金融交易中的欺诈行为，如信用卡盗刷、洗钱、保险诈骗等，保护金融机构和客户资金安全 。

1. **教育领域**

* **智能辅导与答疑平台**：作为虚拟辅导教师，针对学生在多学科学习中遇到的难题提供解答和学习指导，培养解题思维，提升学习成绩 。
* **个性化学习路径规划引擎**：依据学生学习表现、习惯和兴趣，定制专属学习计划与路径，智能推荐学习资源，提高学习效率 。
* **智能作业批改与学情分析**：自动批改作业和试卷，给出详细评语和改进建议，分析学生知识薄弱点和学习进步趋势，助力教师调整教学策略 。

## 三、Dify 平台介绍

### 3.1 Dify 平台概述

* Dify（Define + Modify）是一个面向未来的开源 LLM 应用开发平台，融合后端即服务（Backend as Service）与 LLMOps 理念，为开发者和企业提供生产级的生成式 AI 应用构建能力。自 2023 年创立以来，已服务全球超过 200 万开发者，GitHub 星标数突破 60,000，成为 LLM 工具链领域的标杆产品 。

### 3.2 Dify 平台的技术架构

1. **支持多种模型**：可与数百个开源与商业模型实现无缝集成，包括 GPT、Llama、DeepSeek 等，兼容任意符合 OpenAI API 标准的模型，开发者能够根据自身需求灵活选择最适配的模型来构建 AI 应用 。
2. **独创蜂巢架构设计**：实现模型、插件、数据源的动态编排，能够根据不同的应用场景和任务需求，灵活组合和配置各种资源，提高应用开发的灵活性和效率 。
3. **内置企业级 RAG 引擎**：支持 PDF、PPT 等 20 + 文档格式的语义化处理，能够从多种格式的文档中提取有价值的信息，并进行语义理解和分析，为后续的应用开发提供丰富的数据支持 。
4. **可视化工作流设计器**：支持复杂 Agent 逻辑与多模态任务编排，通过直观的可视化界面，开发者无需编写大量代码，即可轻松设计和构建复杂的 AI 工作流程，实现多模态任务的协同处理 。
5. **提供 LLMOps 监控体系**：实现成本分析、效果评估与持续优化，帮助开发者实时了解 AI 应用的运行情况，对模型调用成本、应用效果等进行监控和分析，并根据分析结果进行优化和调整 。

### 3.3 Dify 平台的核心优势

1. **生产就绪性**：通过 ISO 27001 认证的基础设施，具备强大的稳定性和安全性，能够支持千万级日请求处理，满足企业级应用的高并发需求 。
2. **开放性设计**：可无缝对接 AWS Bedrock、阿里云 PAI 等云服务，同时支持私有化部署，企业可以根据自身的安全需求、数据管理政策以及预算等因素，选择合适的部署方式 。
3. **开发者友好**：提供声明式 YAML 配置标准，大大降低了 AI 工程化的门槛，即使是对 AI 技术不太熟悉的开发者，也能够快速上手，进行 AI 应用的开发 。
4. **企业级特性**：具备 RBAC 权限管理、审计日志、数据隔离等合规功能，保障企业数据的安全性和合规性，满足企业在数据管理和安全方面的严格要求 。
5. **成本控制**：通过智能路由算法，实现模型调用成本降低 30% - 50%，帮助企业在使用大模型进行应用开发时，有效控制成本，提高经济效益 。

### 3.4 Dify 平台的产品功能矩阵

1. **AI 应用工厂**：开发者通过低代码界面，3 分钟即可创建客服机器人、智能助手等场景化应用，快速将 AI 技术应用到实际业务场景中，满足企业对特定场景应用的快速开发需求 。
2. **企业知识中枢**：帮助企业构建私有化 AI 大脑，支持 50 + 语言的知识检索与推理，能够整合企业内部的各类知识资源，为企业员工提供智能的知识查询和推理服务，提升企业的知识管理和应用能力 。
3. **AI Gateway**：统一管理模型 API，实现流量控制与安全审计，对模型 API 的调用进行集中管理和监控，保障 API 调用的安全性和稳定性，同时优化流量分配，提高资源利用效率 。
4. **Workflow Studio**：可视化编排包含 API 调用、数据库查询的复杂业务流，通过直观的可视化界面，开发者可以轻松设计和编排复杂的业务流程，实现不同系统和服务之间的协同工作 。

**在线体验**

速度比较慢。不推荐

# 四. 本地部署

**4.1 Docker安装**

安装Docker环境

bash <(curl -sSl https://cdn.jsdelivr.net/gh/SuperManito/LunuxMirrors@main/DockerInstallation.sh)

安装Docker Compose

curl -L "https://github.com/docker/compose/releases/latest/download/docker-compose-$(uname -

s)-$(uname -m)" -o /usr/local/bin/docker-compose && chmod +x /usr/local/bin/docker-compose

执行查看Docker-compose版本

docker-compose --version



说明安装成功了

docker-compse拉取镜像很慢

sudo tee /etc/docker/daemon.json <<-'EOF'

{

"registry-mirrors": [

"https://do.nark.eu.org",

"https://dc.j8.work",

"https://docker.m.daocloud.io",

"https://dockerproxy.com",

"https://docker.mirrors.ustc.edu.cn",

"https://docker.nju.edu.cn"

]

}

EOF

执行上面的代码

sudo systemctl daemon-reload # 重新加载 systemd 的配置文件

systemctl restart docker # 重启docker

然后去GitHub上拉取dify的代码。解压后进入到docker目录中

docker-compose up -d

执行即可

**4.2 DockerDeskTop**

在Windows环境下我们可以通过DockerDesktop 来安装。直接去官网下载对应的版本即可。同样的我们需要拉

git clone <https://github.com/langgenius/dify.git>

进入 Dify 源代码的 Docker 目录

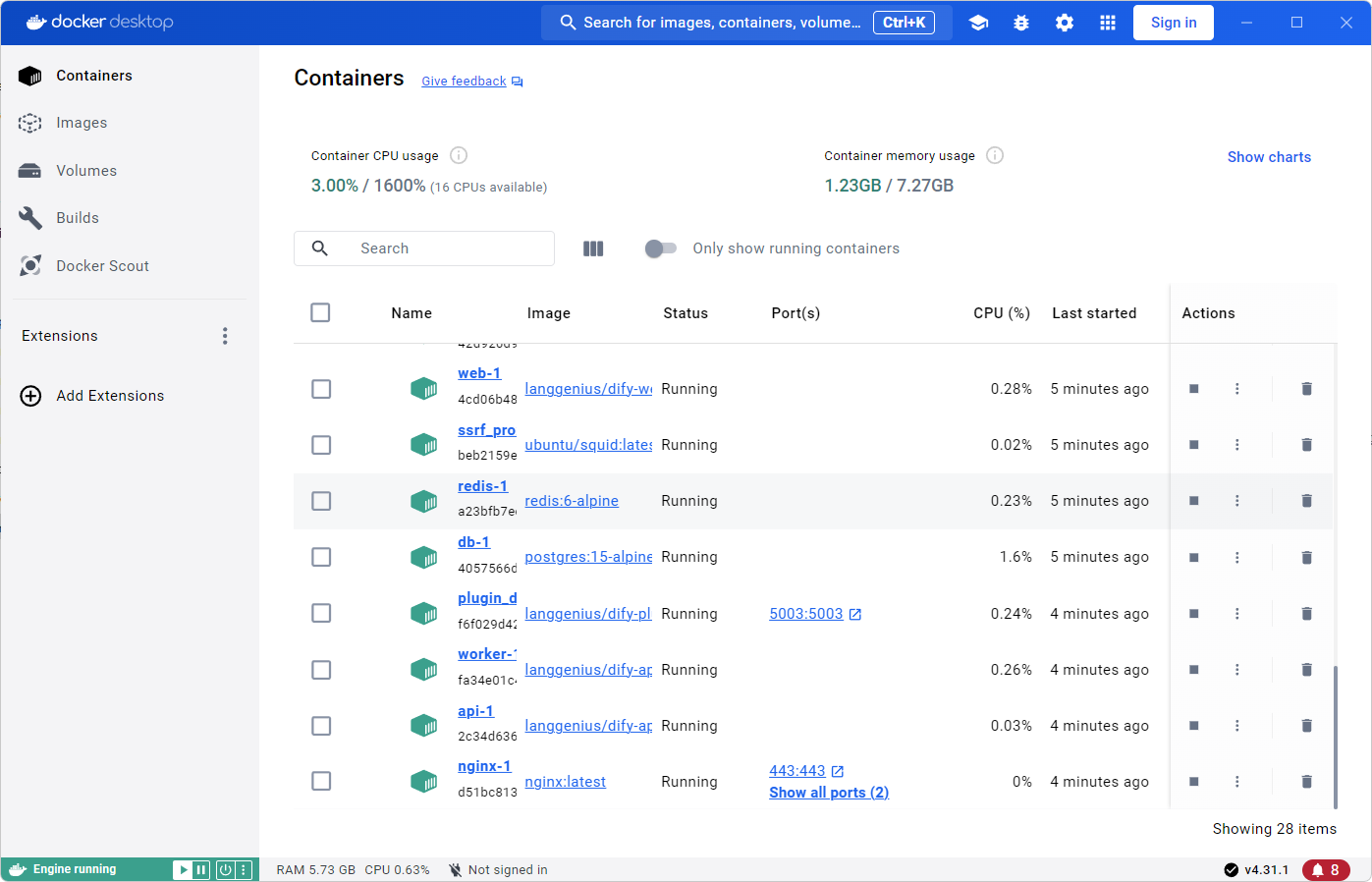
cd dify/docker

复制环境配置文件

cp .env.example .env

同样的执行这个代码

docker-compose up



然后在地址栏中输入 http://localhost/install 就可以访问了

我们先设置管理员的相关信息。设置后再登录



**4.3 Ollama**

https://ollama.com/

我们已经把Dify在本地部署了。然后我们可以通过Ollama在本地部署对应的大模型，比如 deepseek-r1:1.5b 这

种小模型

Ollama 是一个让你能在本地运行大语言模型的工具，为用户在本地环境使用和交互大语言模型提供了便利，具有以下

特点：

1）多模型支持：Ollama 支持多种大语言模型，比如 Llama 2、Mistral 等。这意味着用户可以根据自己的需求和场

景，选择不同的模型来完成各种任务，如文本生成、问答系统、对话交互等。

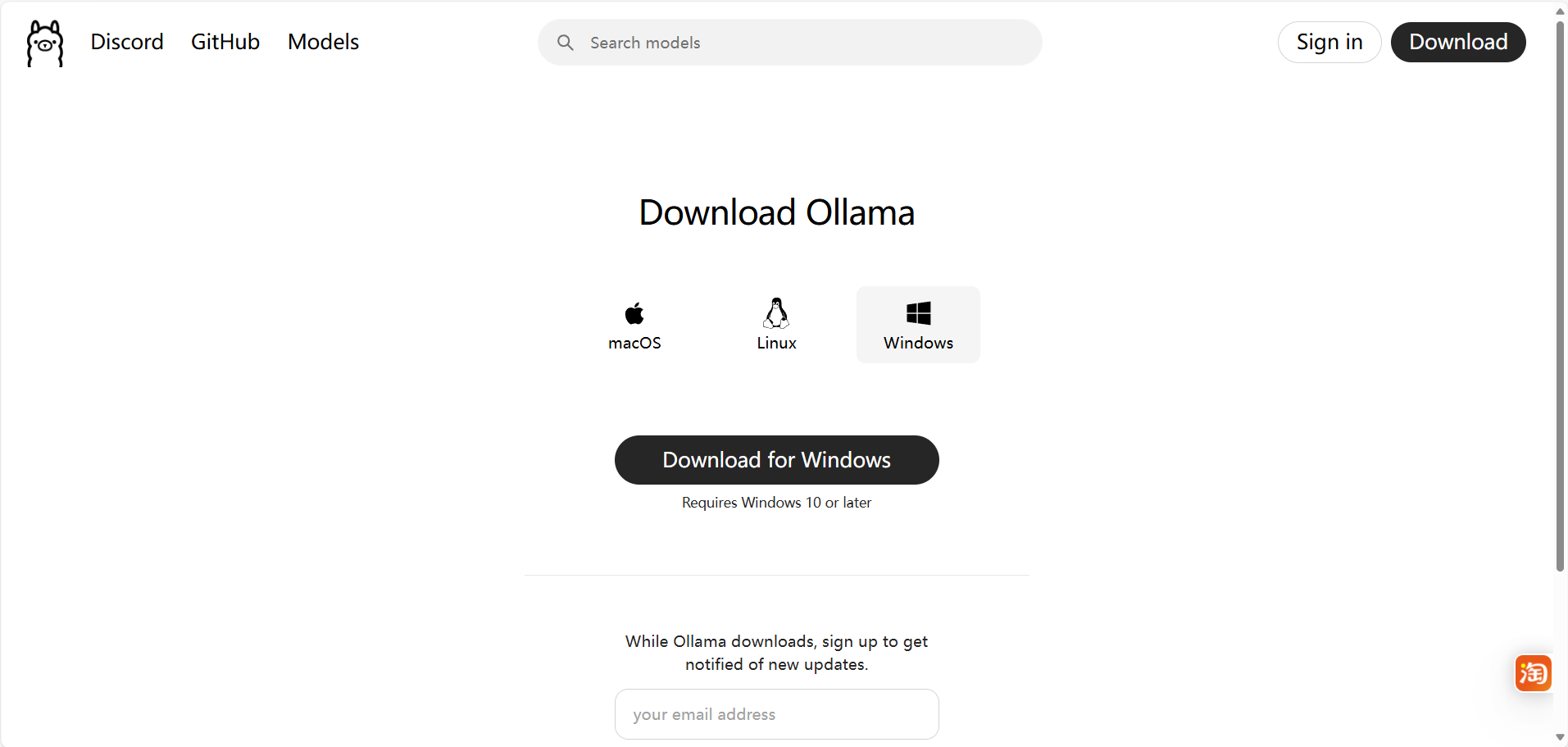
2）易于安装和使用：它的安装过程相对简单，在 macOS、Linux 和 Windows 等主流操作系统上都能方便地部署。

用户安装完成后，通过简洁的命令行界面就能与模型进行交互，降低了使用大语言模型的技术门槛。

3）本地运行：Ollama 允许模型在本地设备上运行，无需依赖网络连接来访问云端服务。这不仅提高了数据的安全性

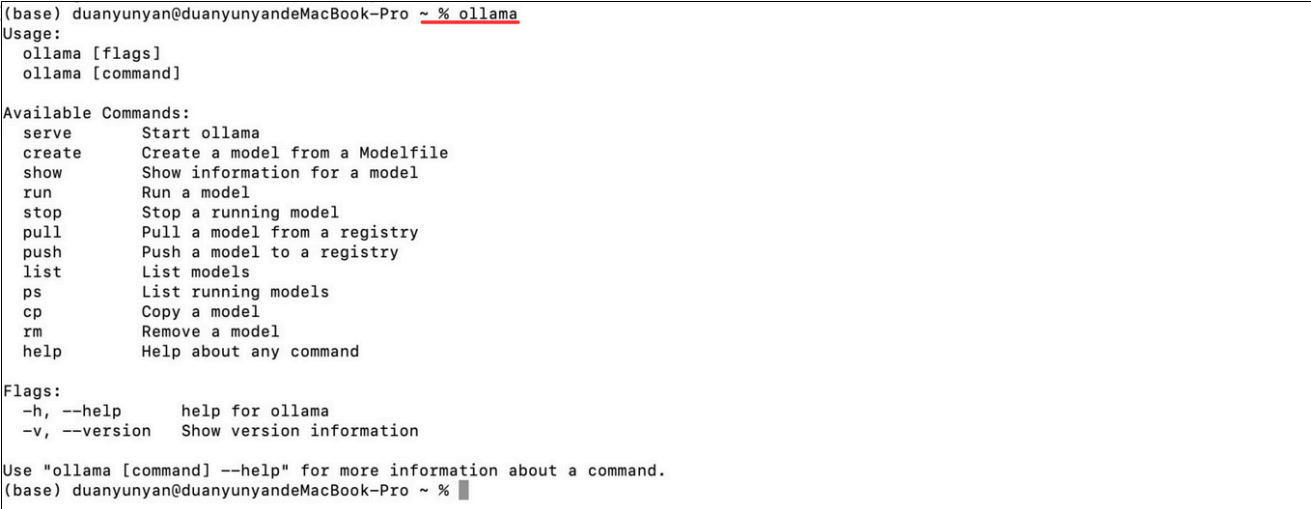
和隐私性，还能减少因网络问题导致的延迟，实现更快速的响应。

搜索Ollama进入官网https://ollama.com/download，选择安装安装包，点击安装即可



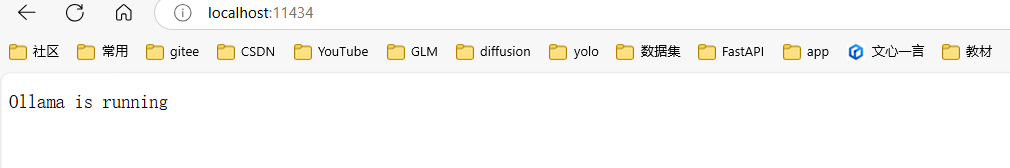
下载完成后直接双击安装即可

命令：ollama,出现下面内容，说明安装成功



启动Ollama服务

输入命令【ollama serve】，浏览器打开，显示running，说明启动成功



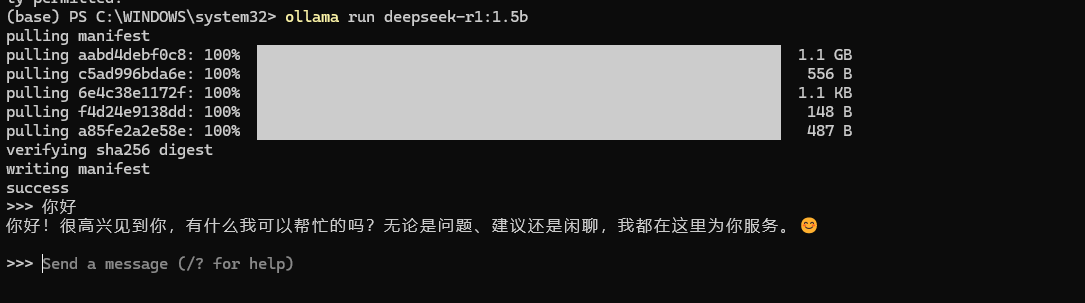
安装 deepseek-r1:1.5b模型

在https://ollama.com/library/deepseek-r1:1.5b 搜索deepseek-R1,跳转到下面的页面，复制这个命令，在终端执

行，下载模型

ollama run deepseek-r1:1.5b

cmd中执行这个命令



**4.4 Dify关联Ollama**

Dify 是通过Docker部署的，而Ollama 是运行在本地电脑的，得让Dify能访问Ollama 的服务。

在Dify项目-docker-找到.env文件，在末尾加上下面的配置：

# 启用自定义模型

CUSTOM\_MODEL\_ENABLED=true

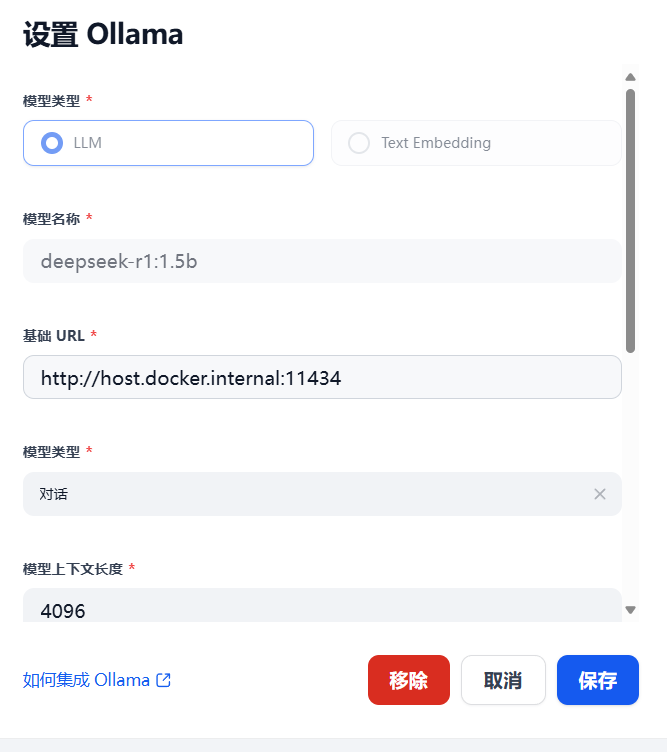
# 指定 Olama 的 API地址（根据部署环境调整IP）

OLLAMA\_API\_BASE\_URL=host.docker.internal:11434

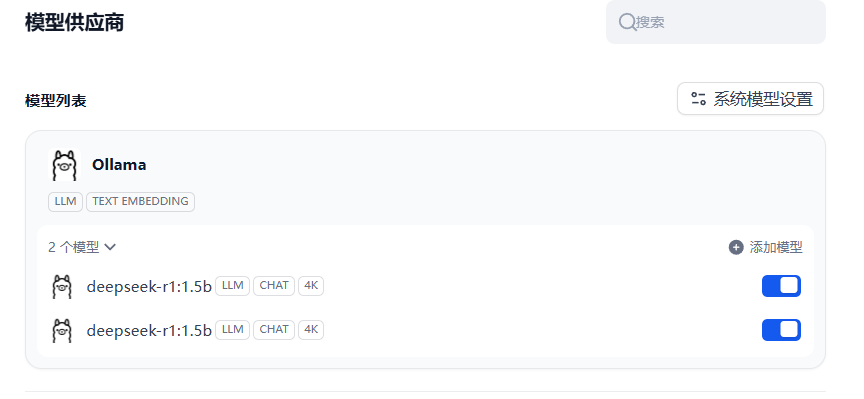
然后在模型中配置

在Dify的主界面 http://localhost/apps ，点击右上角用户名下的【设置】

在设置页面--Ollama--添加模型，如下：



添加成功后的



模型添加完成以后，刷新页面，进行系统模型设置。步骤：输入“http://localhost/install”进入Dify主页，用户名--设

置--模型供应商，点击右侧【系统模型设置】，如下



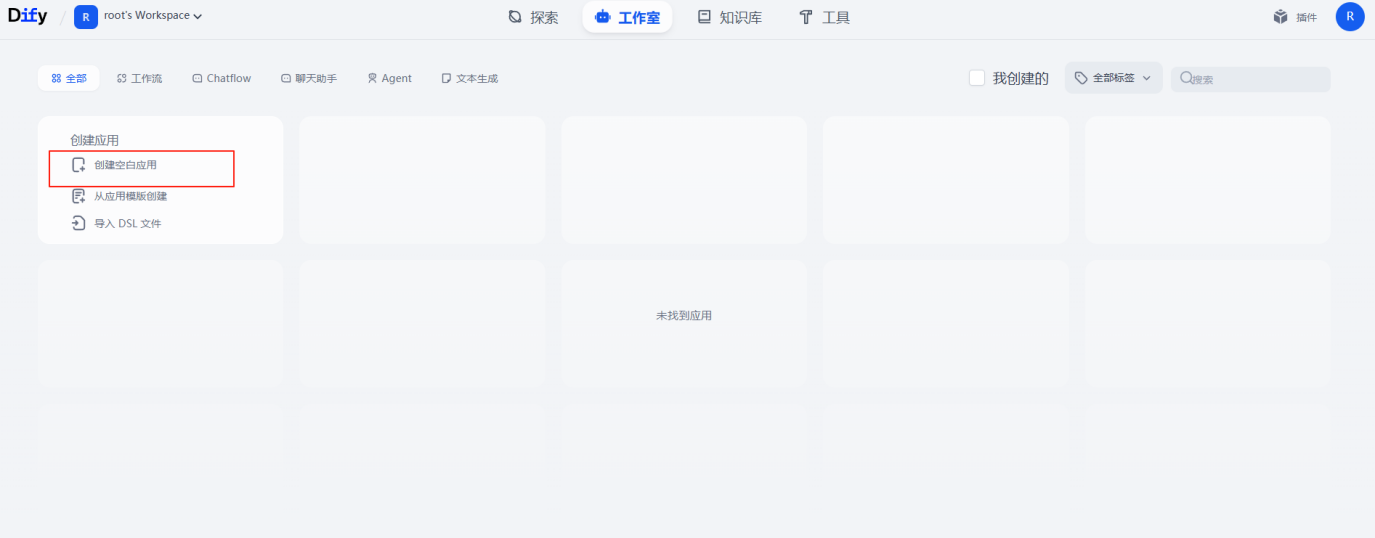
这样就关联成功了！！！

# 五、Dify应用讲解

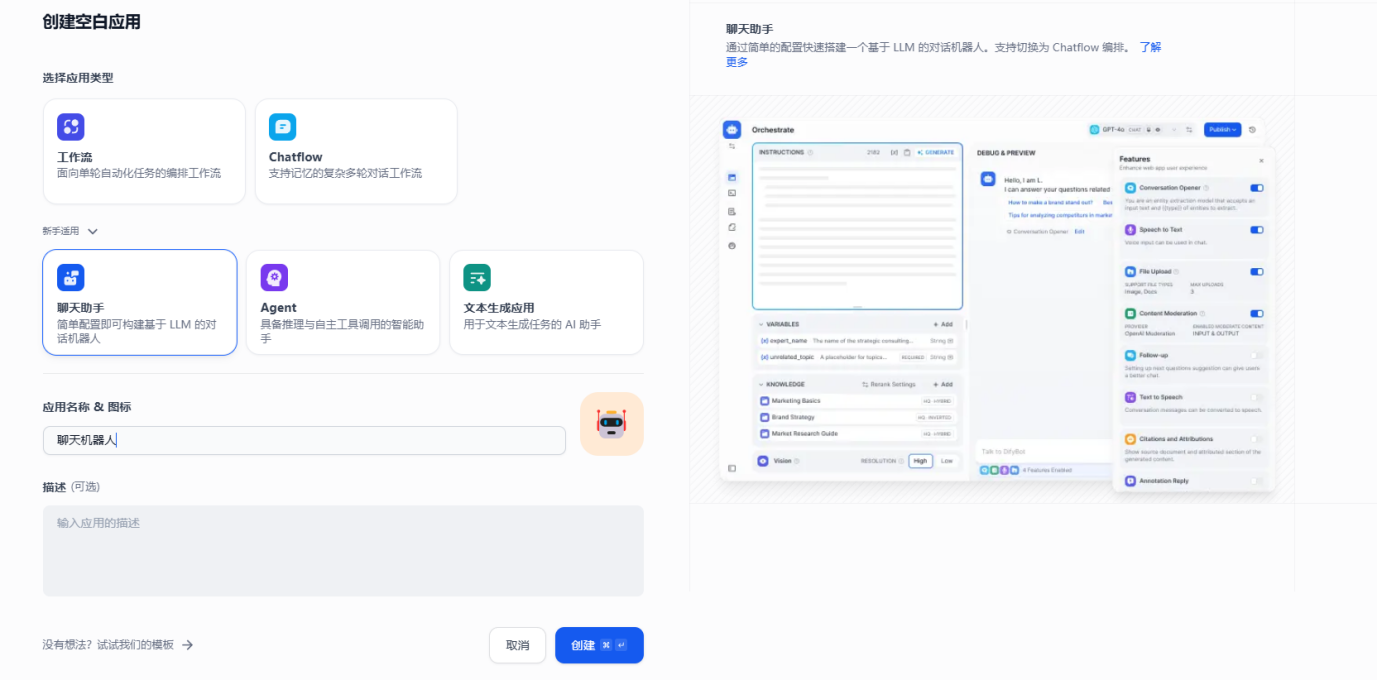
## 5.1创建对话应用

我们通过Dify来创建我们的第一个简单案例，智能聊天机器人

进入Dify 主界面，点击【创建空白应用】，如下图：



选择【聊天助手】，输入自定义应用名称和描述，点击【创建】



右上角选择合适的模型，进行相关的参数配置



输入有相关的回复了。此时说明Dify 与本地部署的DeepSeek大模型已经连通了。



上面的机器人有个不足之处就是无法回答模型训练后的内容和专业垂直领域的内容，这时我们可以借助本地知识库来

解决专业领域的问题。

## 5.2创建本地知识库

**5.2.1 向量模型**

Embedding模型是一种将数据转换为向量表示的技术，核心思想是通过学习数据的内在结构和语义信息，将其映射到一个低维

向量空间中，使得相似的数据点在向量空间中的位置相近，从而通过计算向量之间的相似度来衡量数据之间的相似性。

Embedding模型可以将单词、句子或图像等数据转换为低维向量，使得计算机能够更好地理解和处理这些数据。在NLP领域，

Embedding模型可以将单词、句子或文档转换为向量，用于文本分类、情感分析。机器翻译等任务。在计算机视觉中，

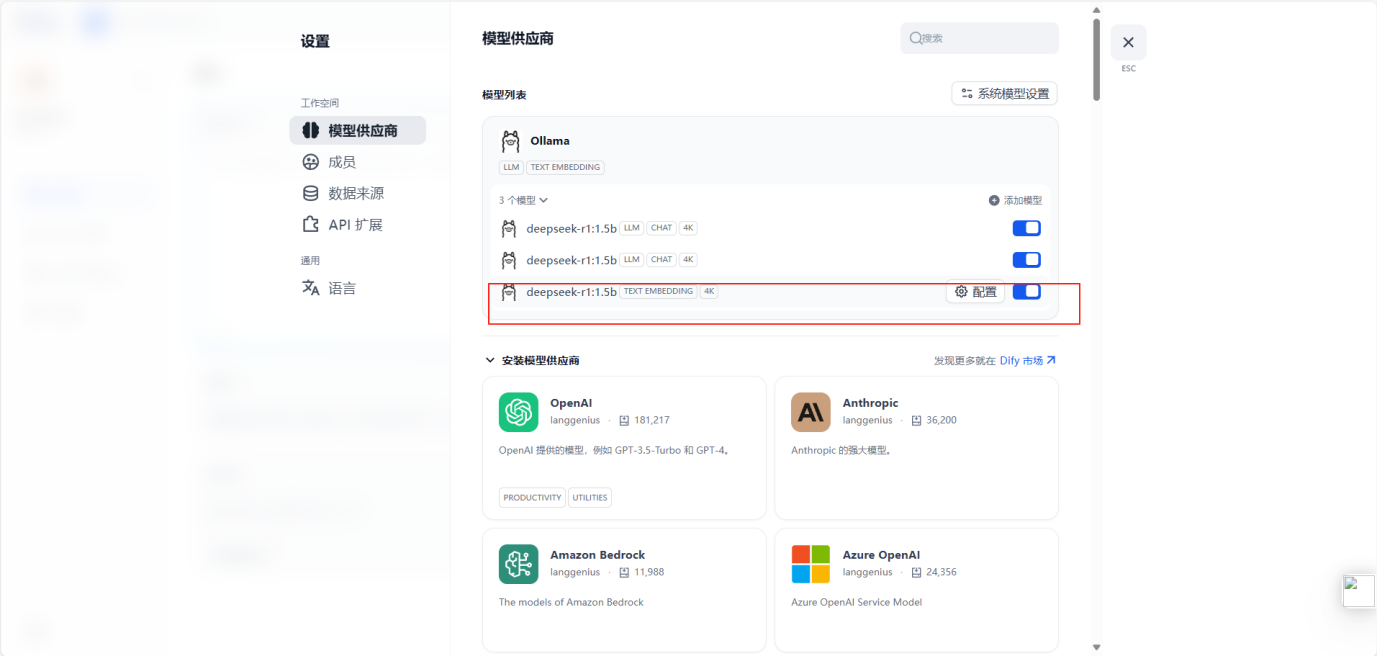
Embedding模型可以用于图像识别和检索等任务。

**5.2.2 添加Embedding模型**

点击右上角用户名--设置--模型供应商--右上角【添加模型】，填写相关配置信息如下

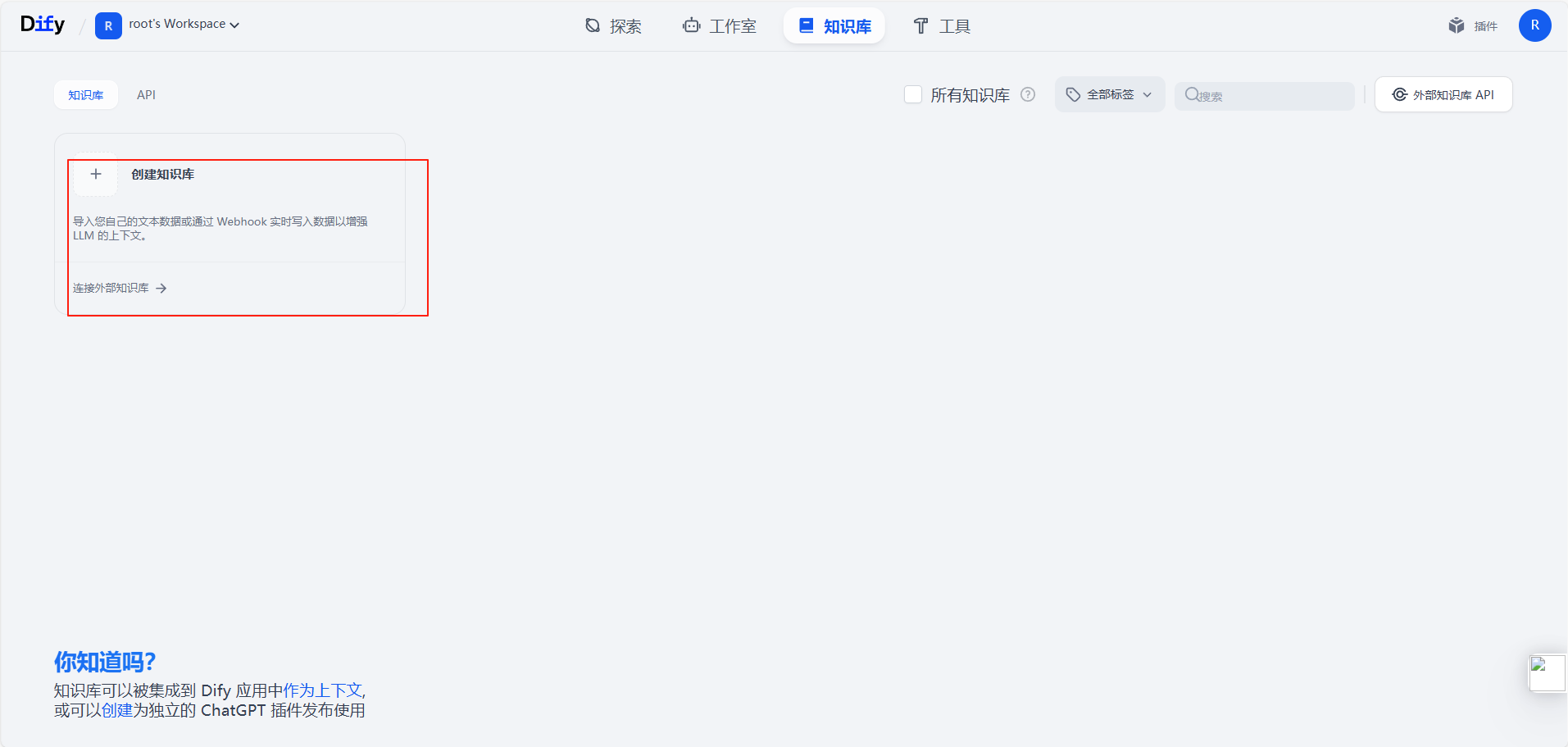


添加成功后的效果

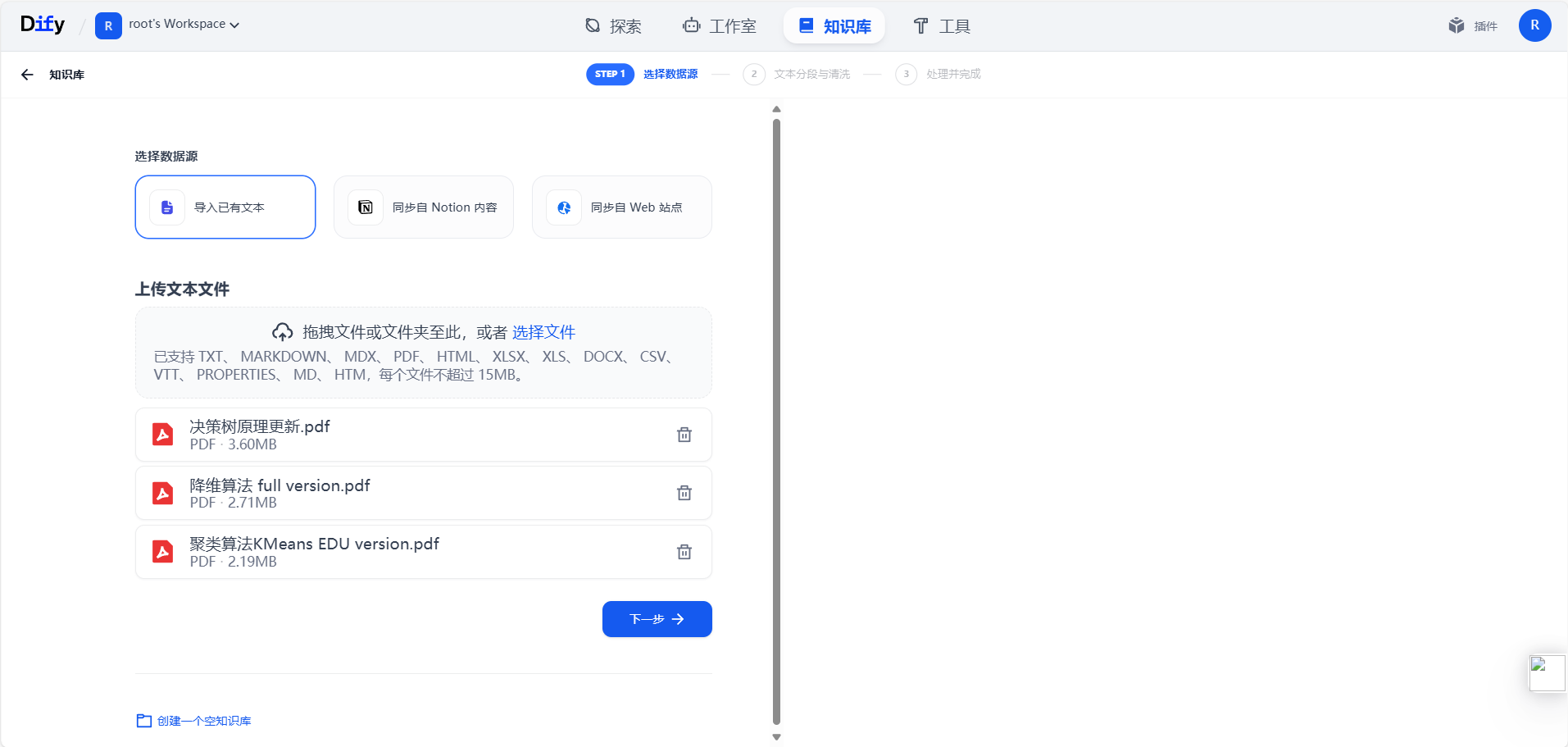


**5.2.3 创建知识库**

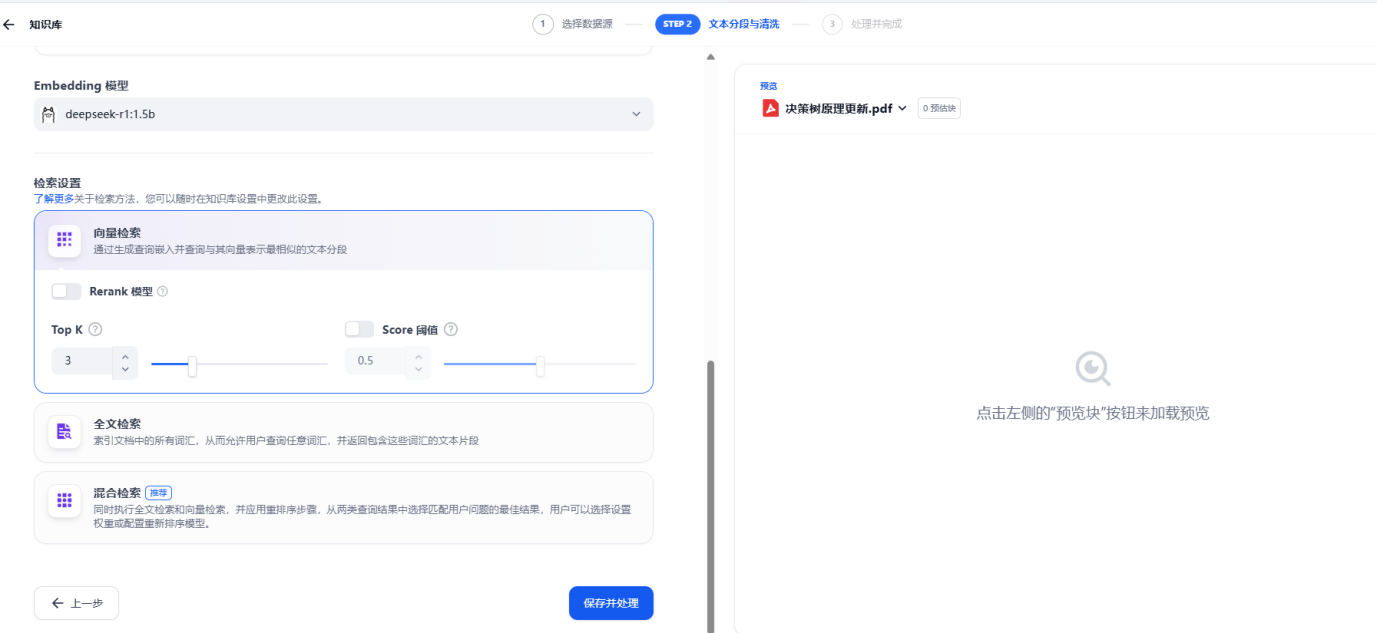
在Dify主界面，点击上方的【知识库】，点击【创建知识库】



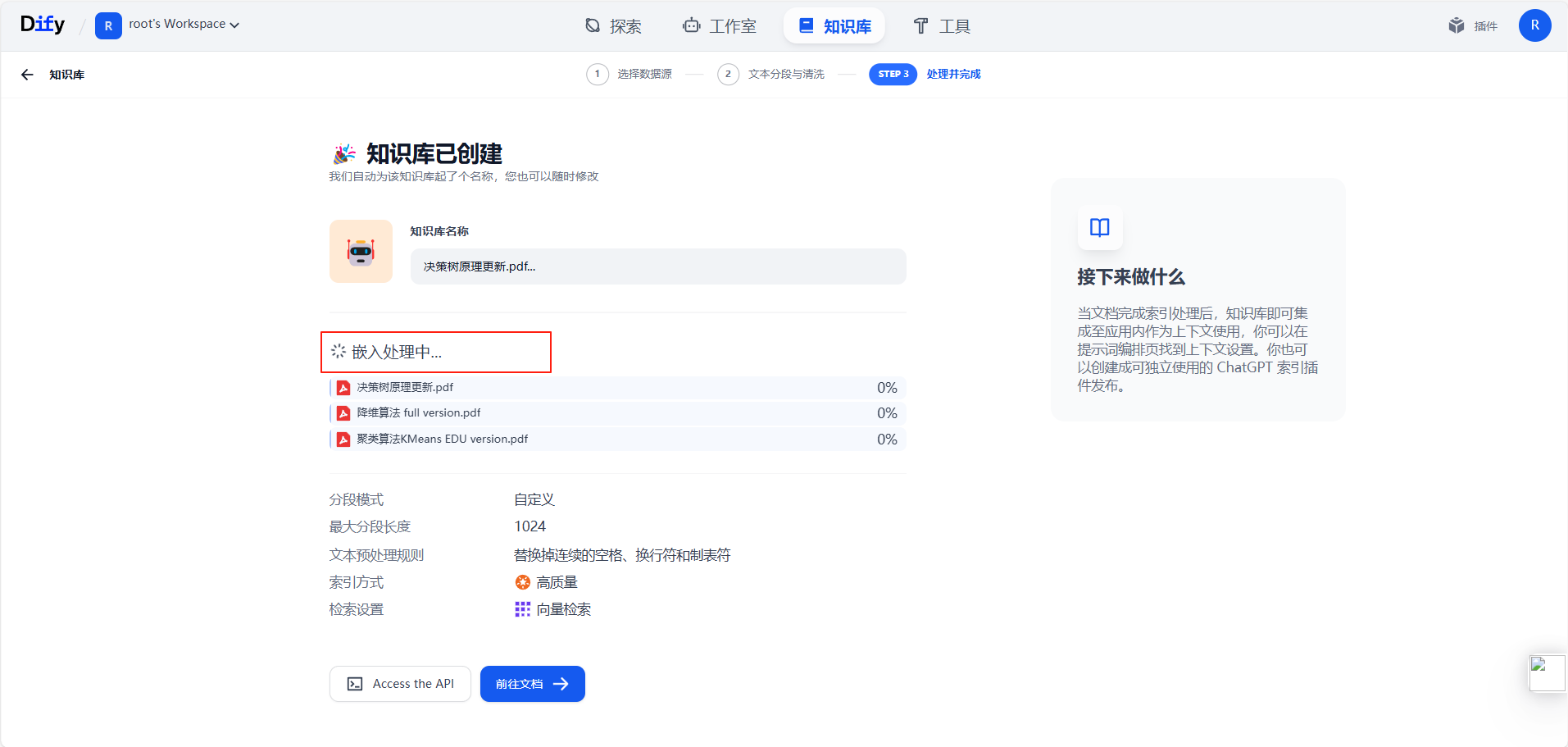
导入已有文本，上传资料，点击【下一步】



Embedding模型默认是前面配置的模型，参数信息配置完，点击保存即可



此时系统会自动对上传的文档进行解析和向量化处理，需要耐心等待几分钟。



创建成功以后，如下图，可以点击【前往文档】，查看分段信息，如下图：



点击具体的文档可以看到具体的分割信息



## 5.3 知识库应用

**5.3.1 添加知识库**

在Dify主界面，回到刚才的应用聊天页面，工作室--智能聊天机器人--添加知识库，如下图：



选择前面上面的知识库作为对话的上下文，保存当前应用设置，就可以进行测试了

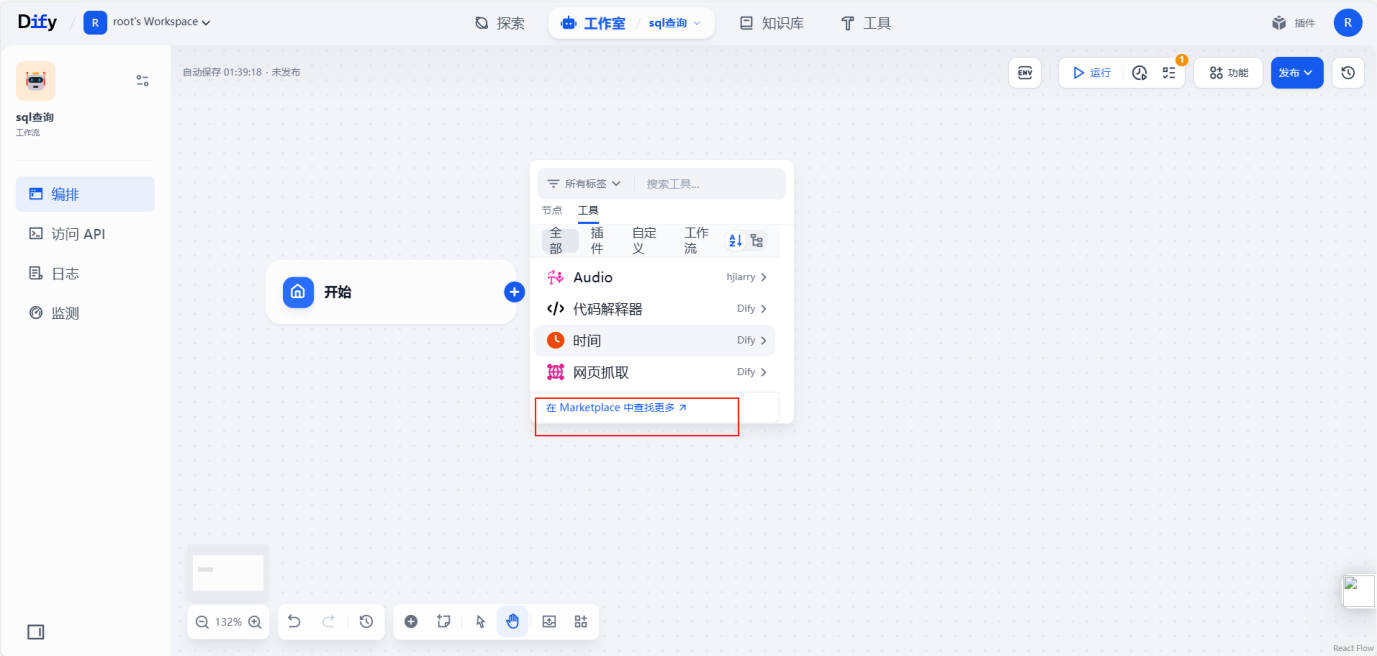
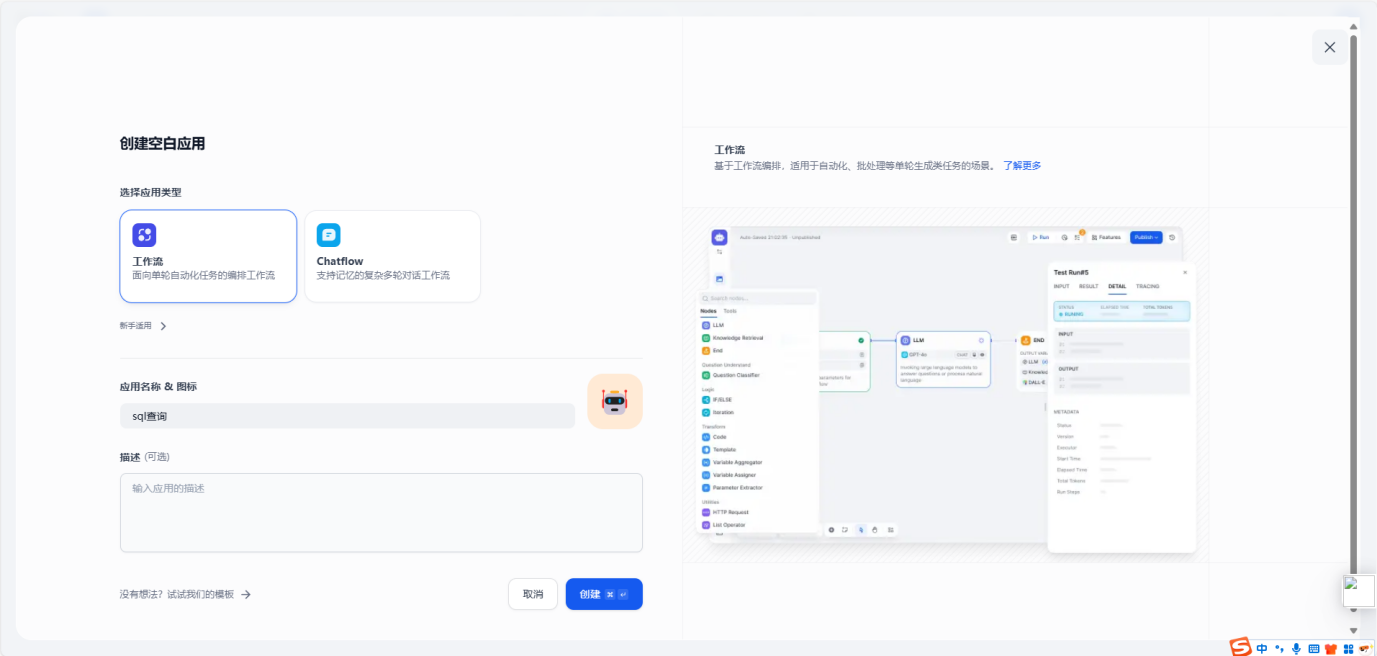
**5.3.2 测试**

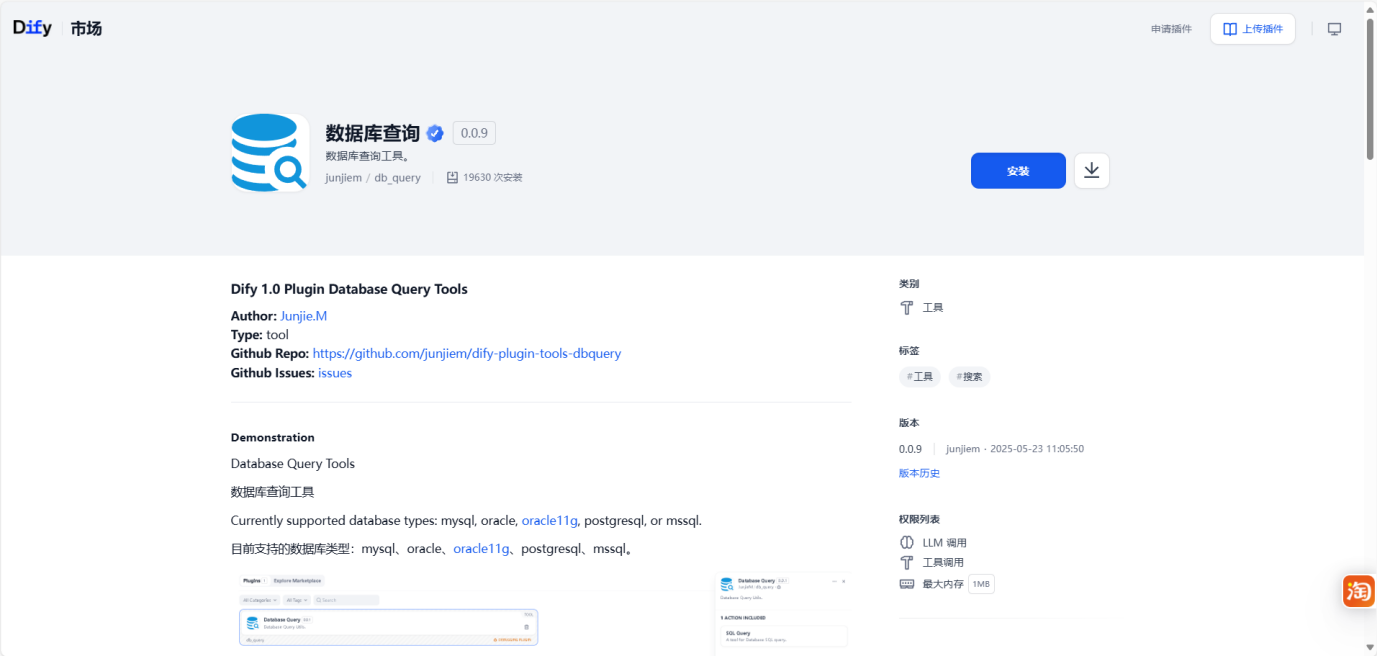
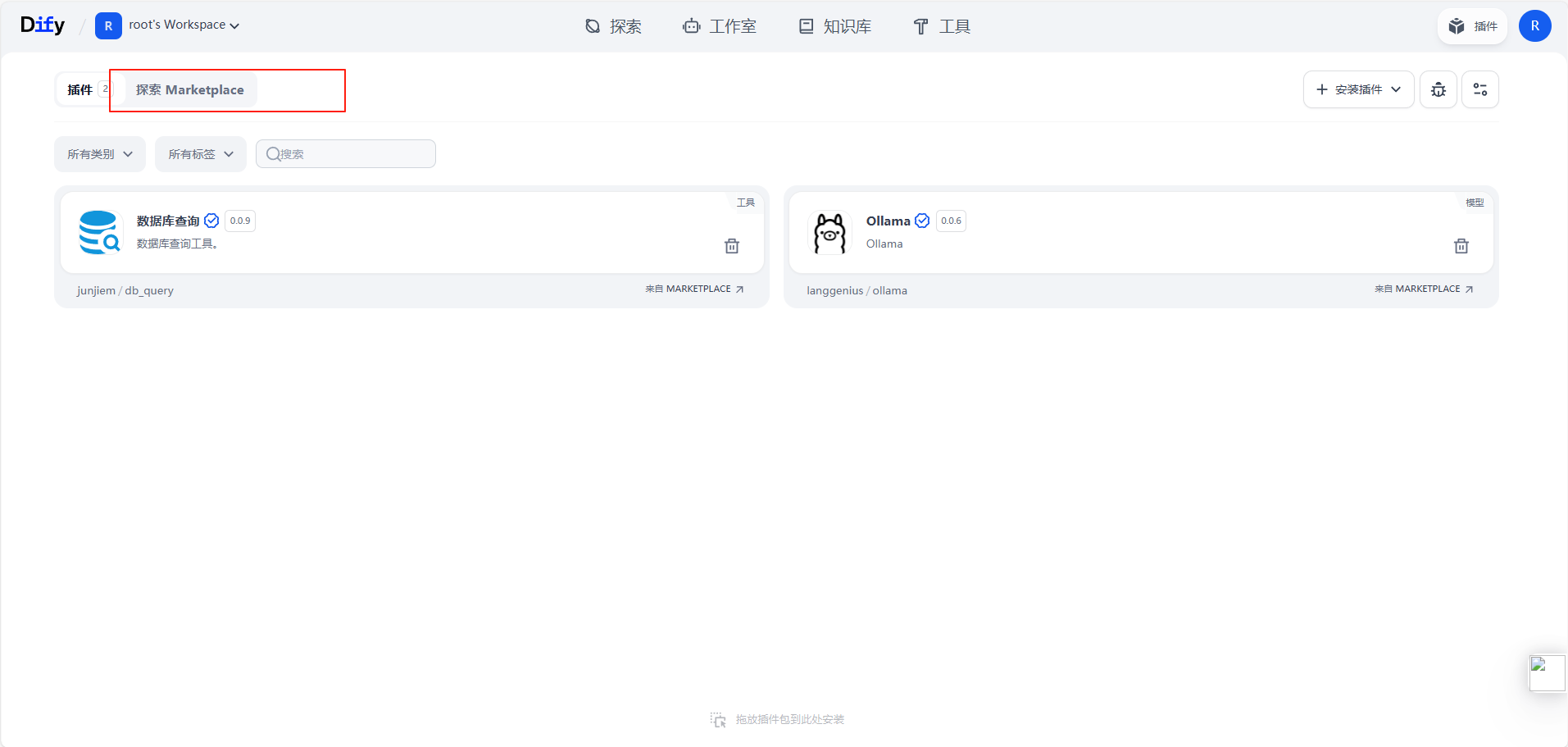
此时输入问题，就可以看到相关的回复了。



## 5.4SQL执行器

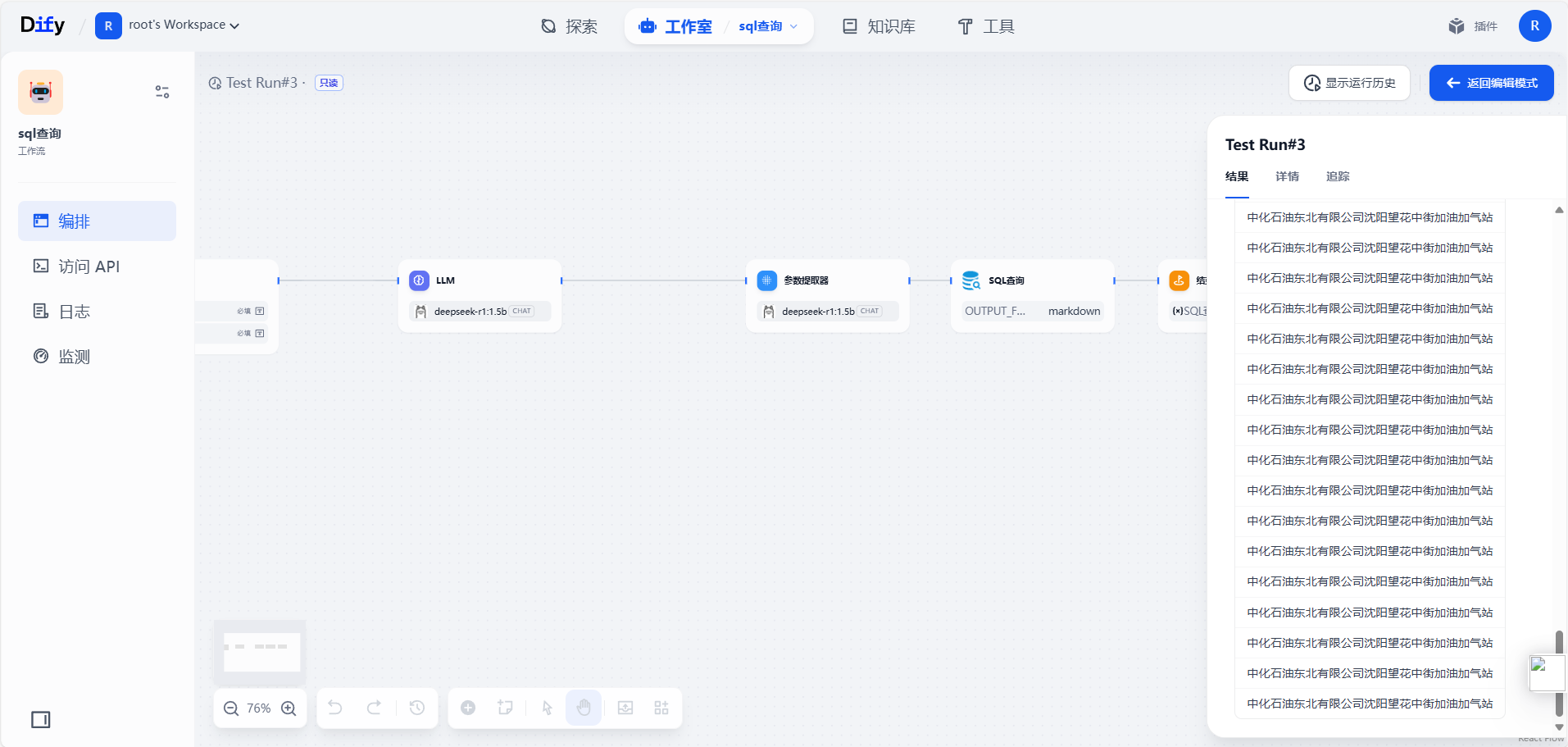
我们可以通过工作流来创建一个SQL语句的执行器，也就是我们可以输入相关的SQL语句然后通过工作流来连接



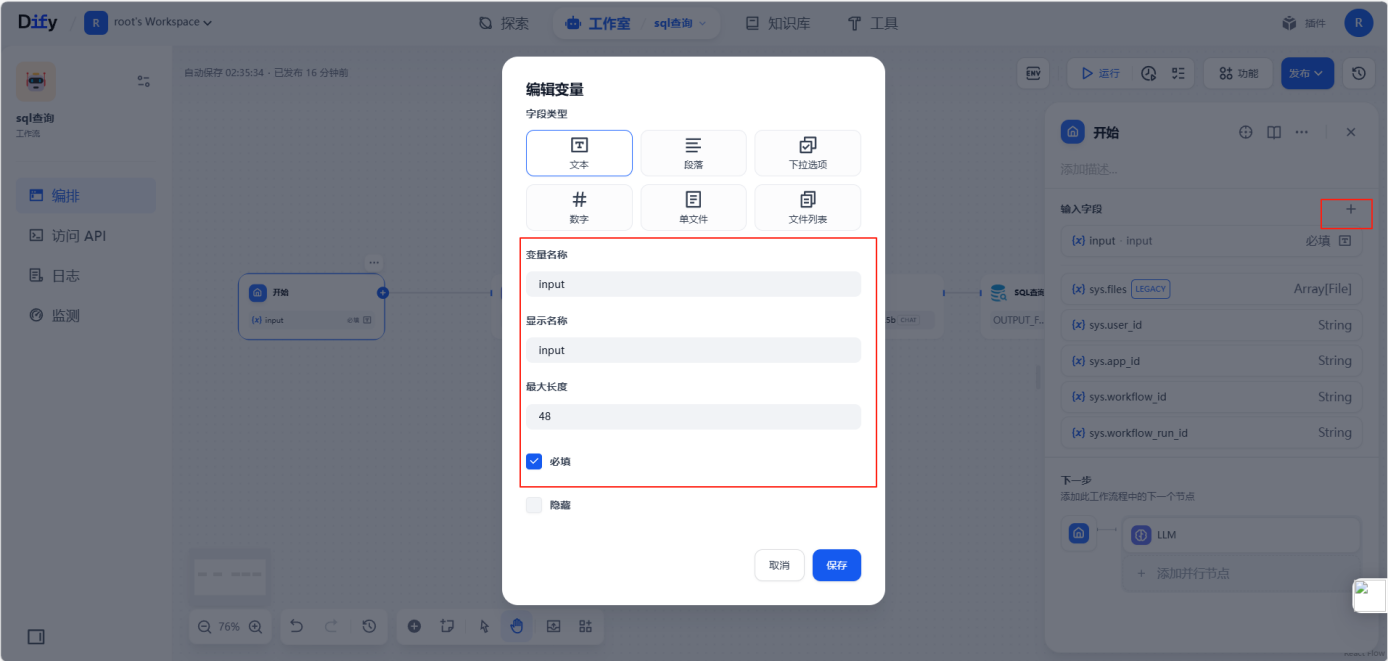


按照上方就可以安装成功数据库查询组件，之后数据库执行对应的SQL代码，具体的设计如下：

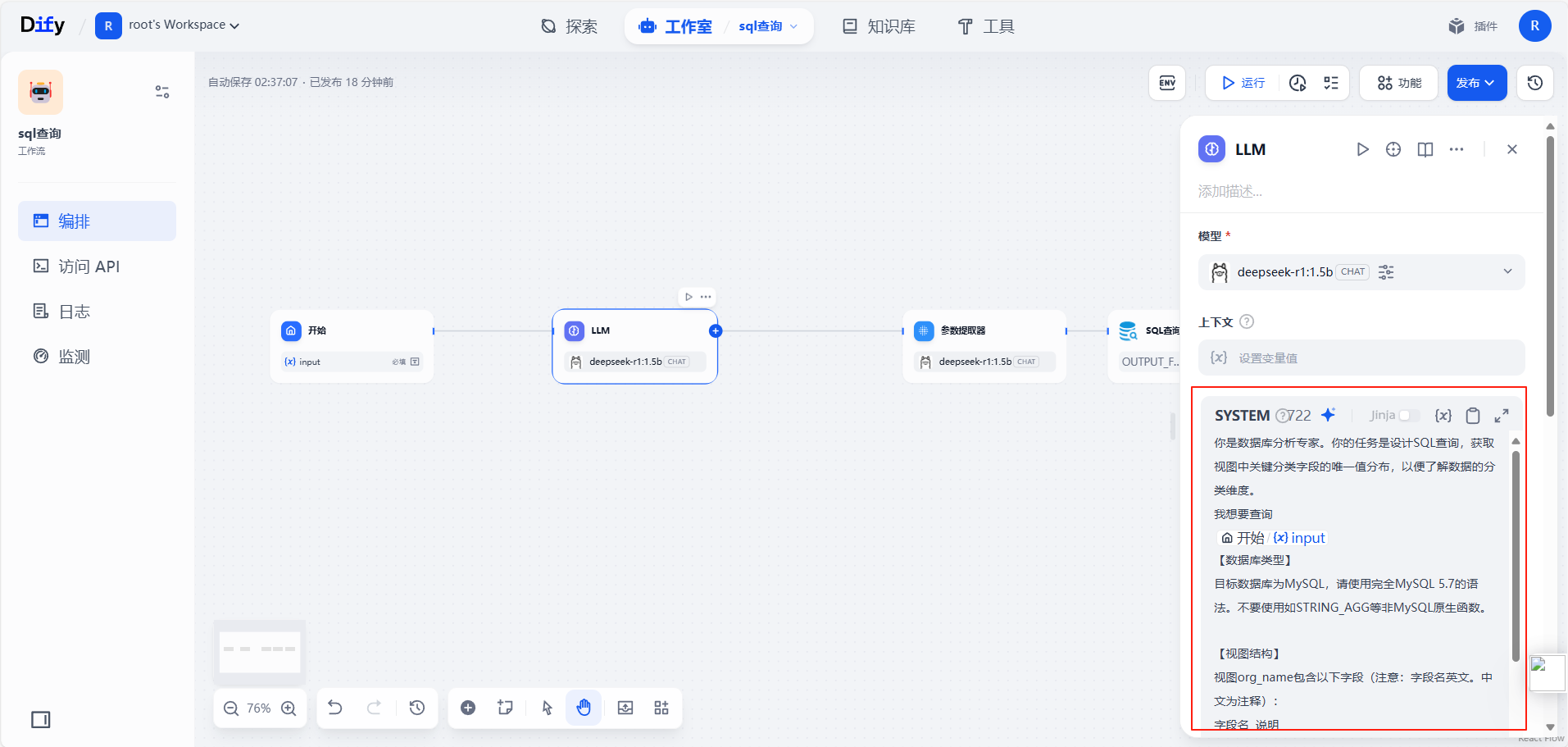
编辑相应的大模型。参数提取和sql查询成功查询出数据



具体设计如下，开始模块编辑用户的输入框



之后增加llm组件并添加提示词使大模型了解你要查询的数据库表明和字段名，还有要输出的结构，



具体提示词如下  
你是数据库分析专家。你的任务是设计SQL查询，获取视图中关键分类字段的唯一值分布，以便了解数据的分类维度。

我想要查询

{{#1749706731307.input#}}

【数据库类型】

目标数据库为MySQL，请使用完全MySQL 5.7的语法。不要使用如STRING\_AGG等非MySQL原生函数。

【视图结构】

视图org\_name包含以下字段（注意：字段名英文。中文为注释）：

字段名 说明

id 加气记录的唯一标识

license\_plate 车辆车牌号

formatted\_real\_time 加气日期（格式为YYYY-MM-DD）

now\_gas 加气量

pressure\_begin 加气起始压力

org\_code 加气组织编码

full\_name 加气站全称

【任务】

仅设计SQL查询来获取关键分类字段的唯一值列表，不包括加气量统计或详细组织编码。重点关注：

license\_plate的唯一值列表

formatted\_real\_time的唯一值列表

now\_gas的唯一值列表

pressure\_begin的唯一值列表

full\_name的唯一值列表

【输出格式】

{

"field\_info\_queries": [

{

"id": 1,

"title": "加气站名称分布",

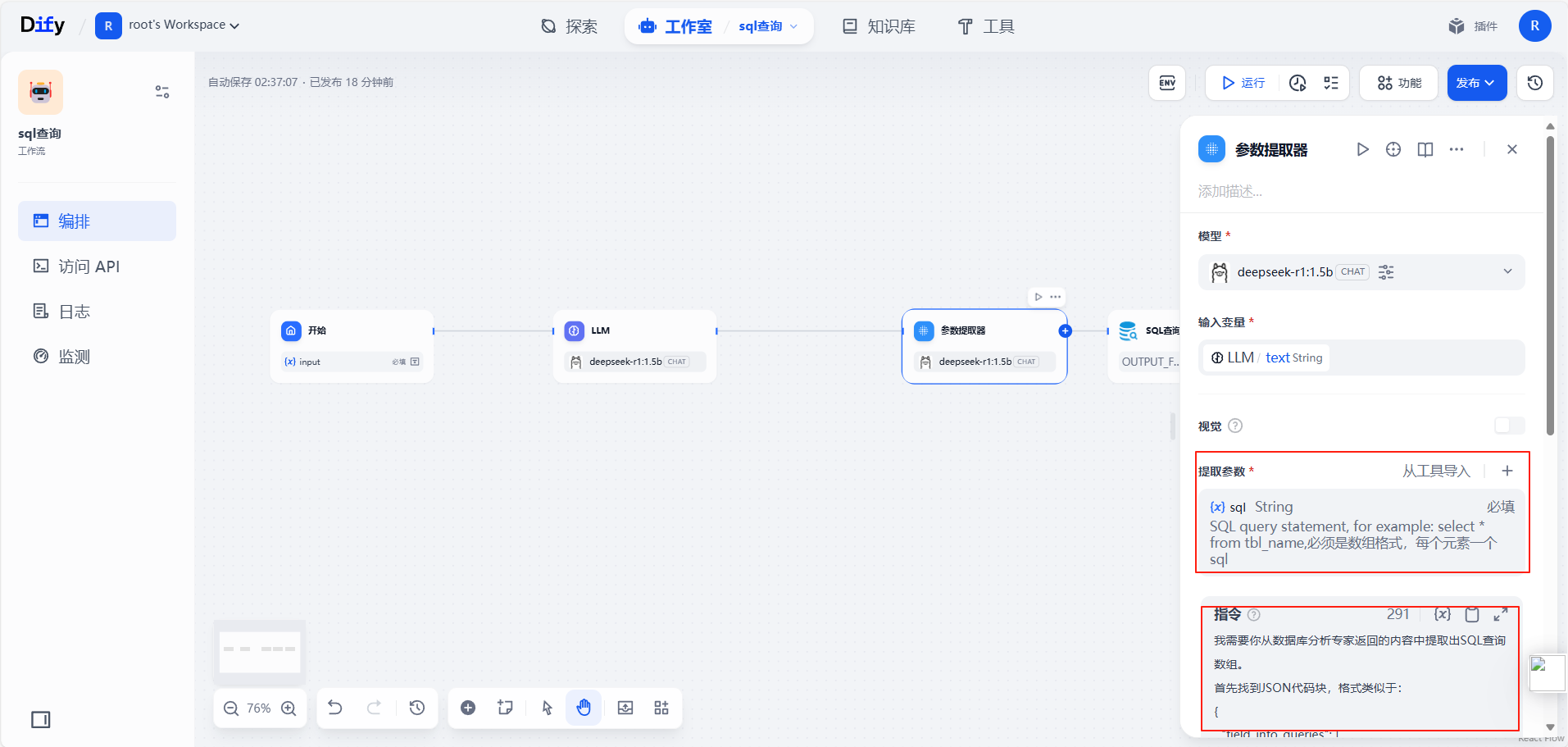
"purpose": "了解有哪些加气站",

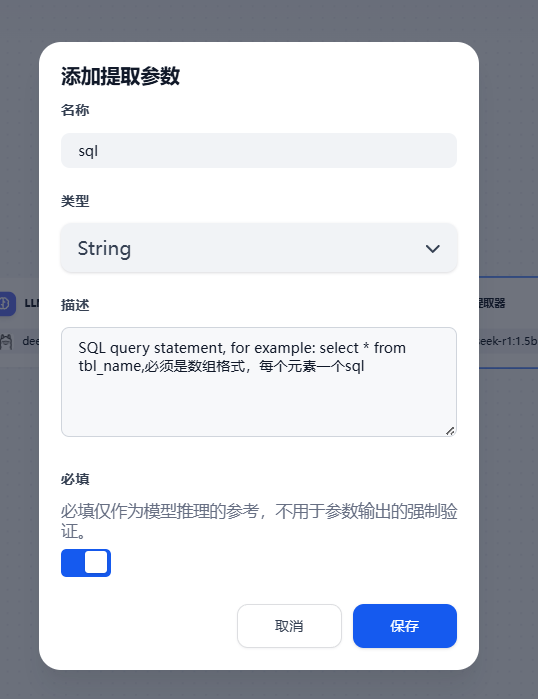
"sql": "SELECT DISTINCT full\_name FROM `gas\_station\_ai\_pads`.`org\_name` ORDER BY full\_name LIMIT 100"

}

]

}

deepseek大模型具有思考的能力，它输出的内容我们需要将sql语句提取出来，这就需要我们使用参数提取器组件将sql语句提取出来。  


提取参数如下，  
SQL query statement, for example: select \* from tbl\_name,必须是数组格式，每个元素一个sql

提取指令如下，告诉大模型我们需要的格式和内容。  
我需要你从数据库分析专家返回的内容中提取出SQL查询数组。

首先找到JSON代码块，格式类似于：

{

"field\_info\_queries": [

{

"id": 1,

"title": "门店名称分布",

"purpose": "了解有哪些门店",

"sql": "SELECT DISTINCT 门店名称 FROM `gas\_station\_ai\_pads`.`org\_name` ORDER BY 门店名称 LIMIT 100"

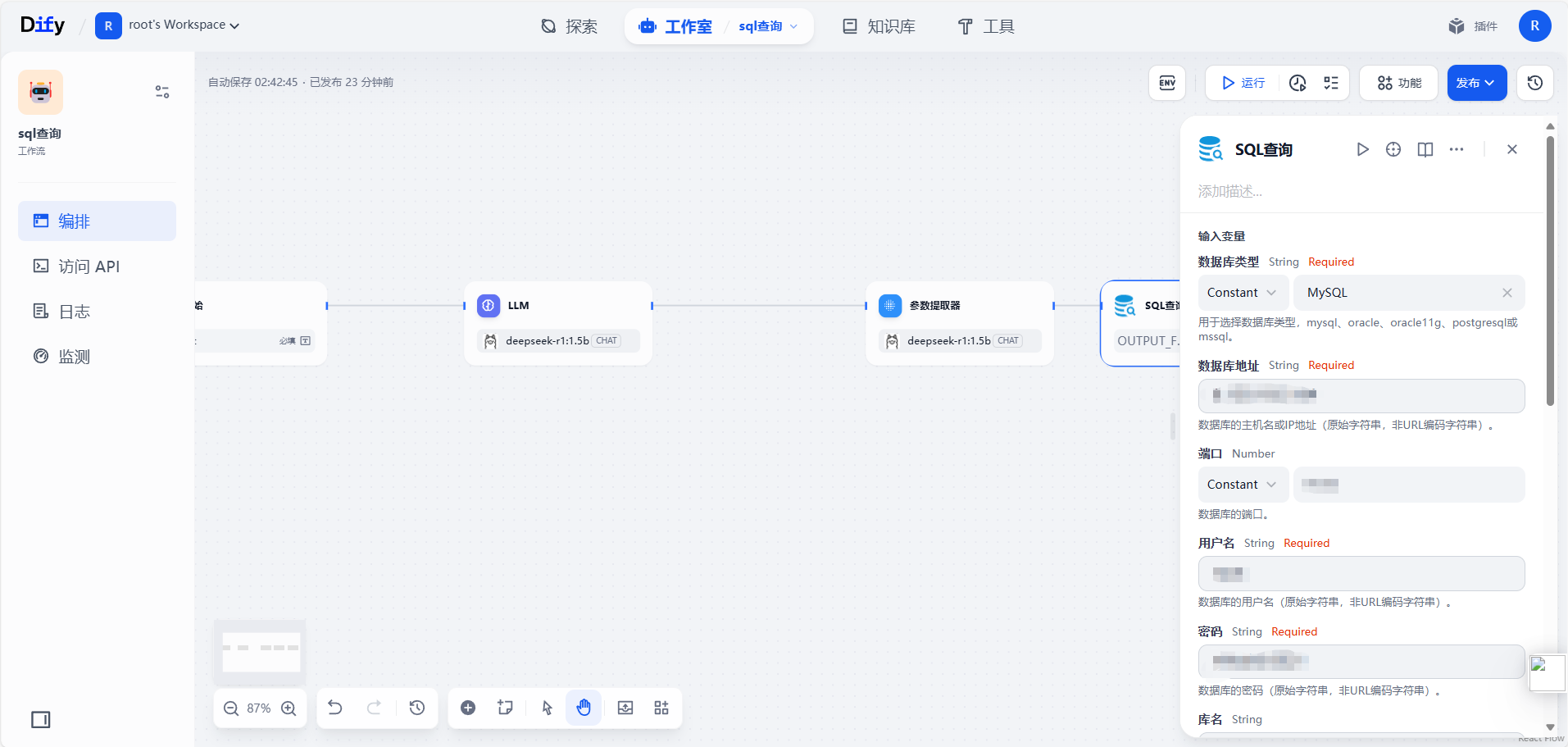
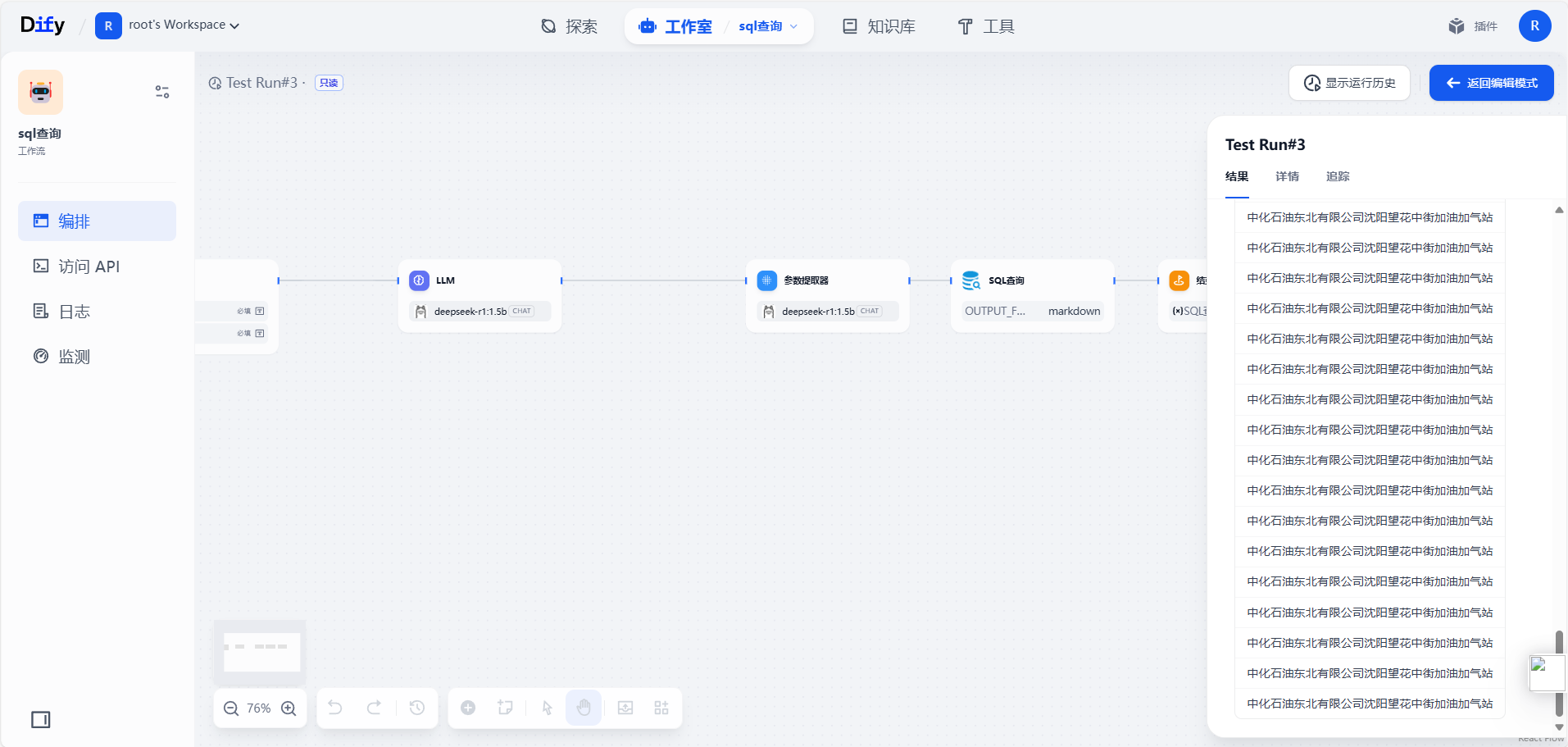
},

...

]

}

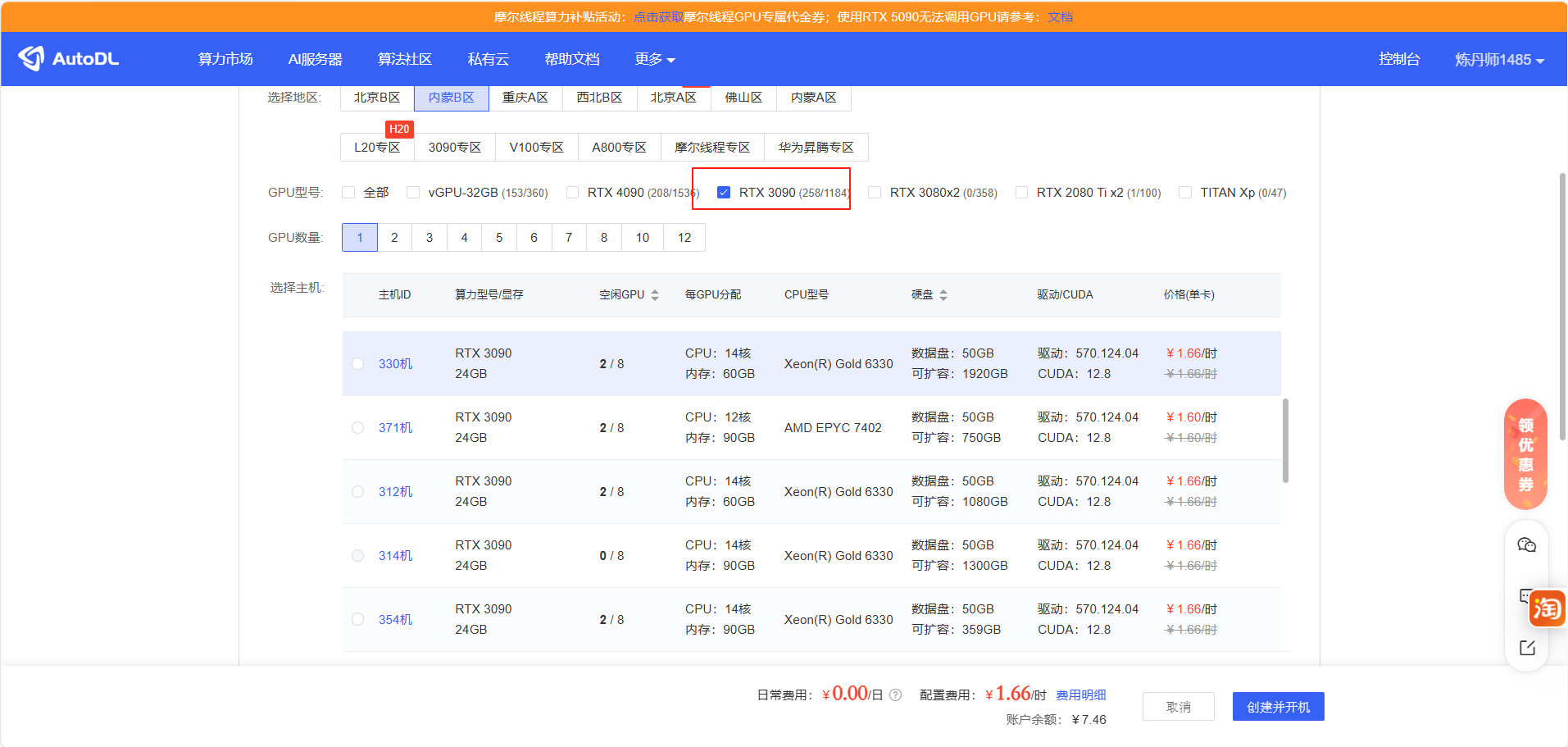
最后添加sql查询组件，将参数提取器的输出sql语句传输给sql查询组件，组件里需要我们填入sql库的连接信息。

  
发布后输入数据问答成功。  


### **额外：云GPU部署**

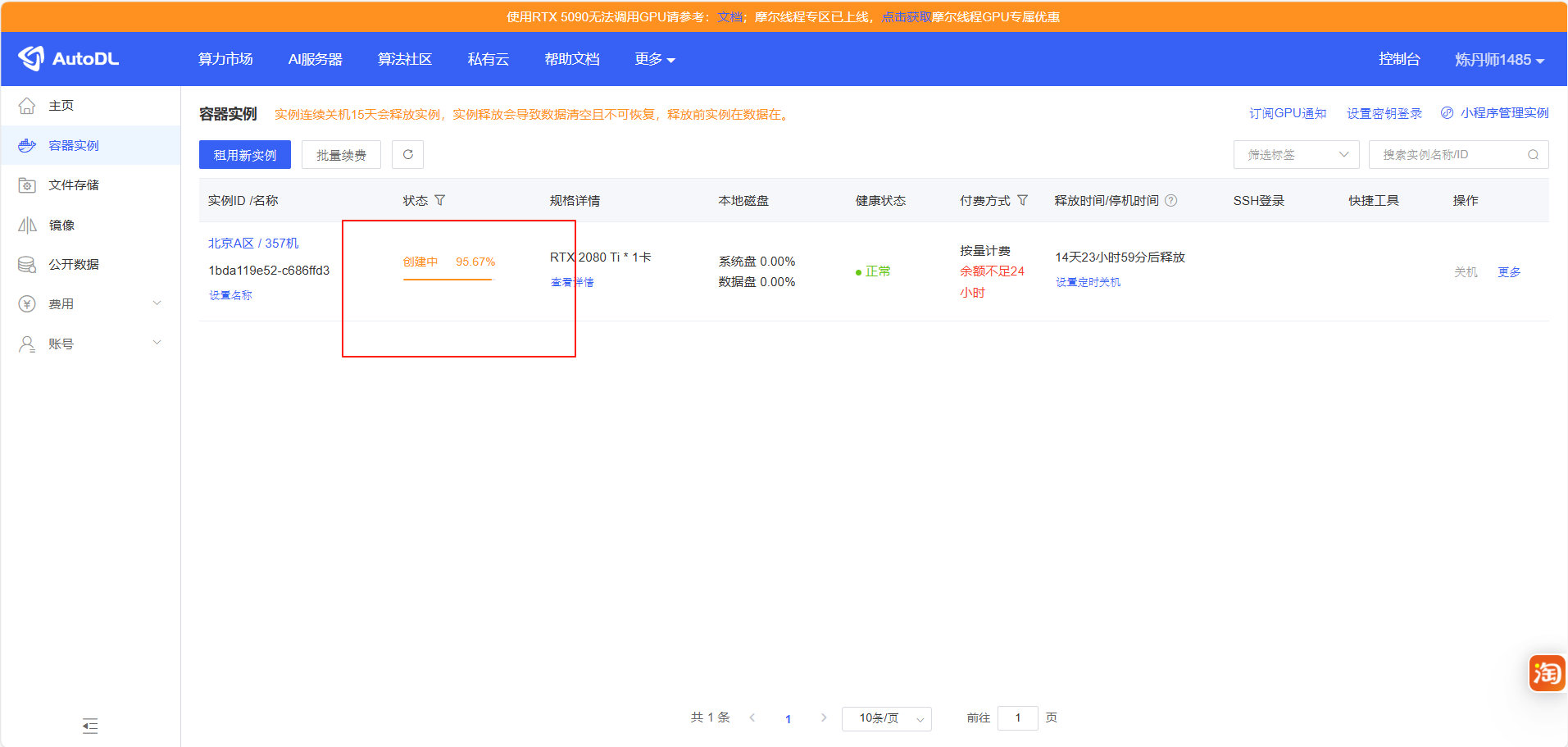
登录AutoDL网址<https://www.autodl.com/home>

点击算力市场找到显卡进行部署





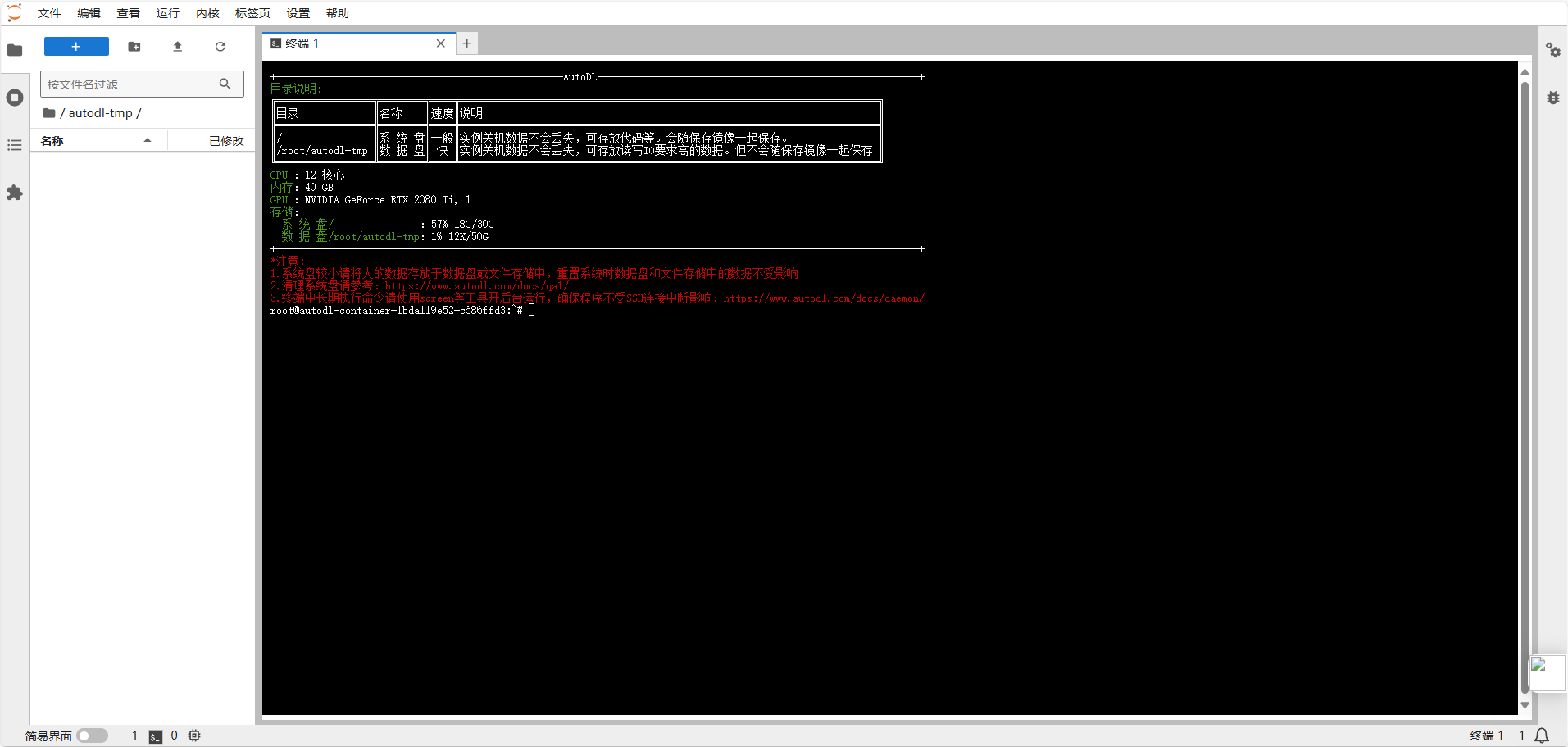
等待创建



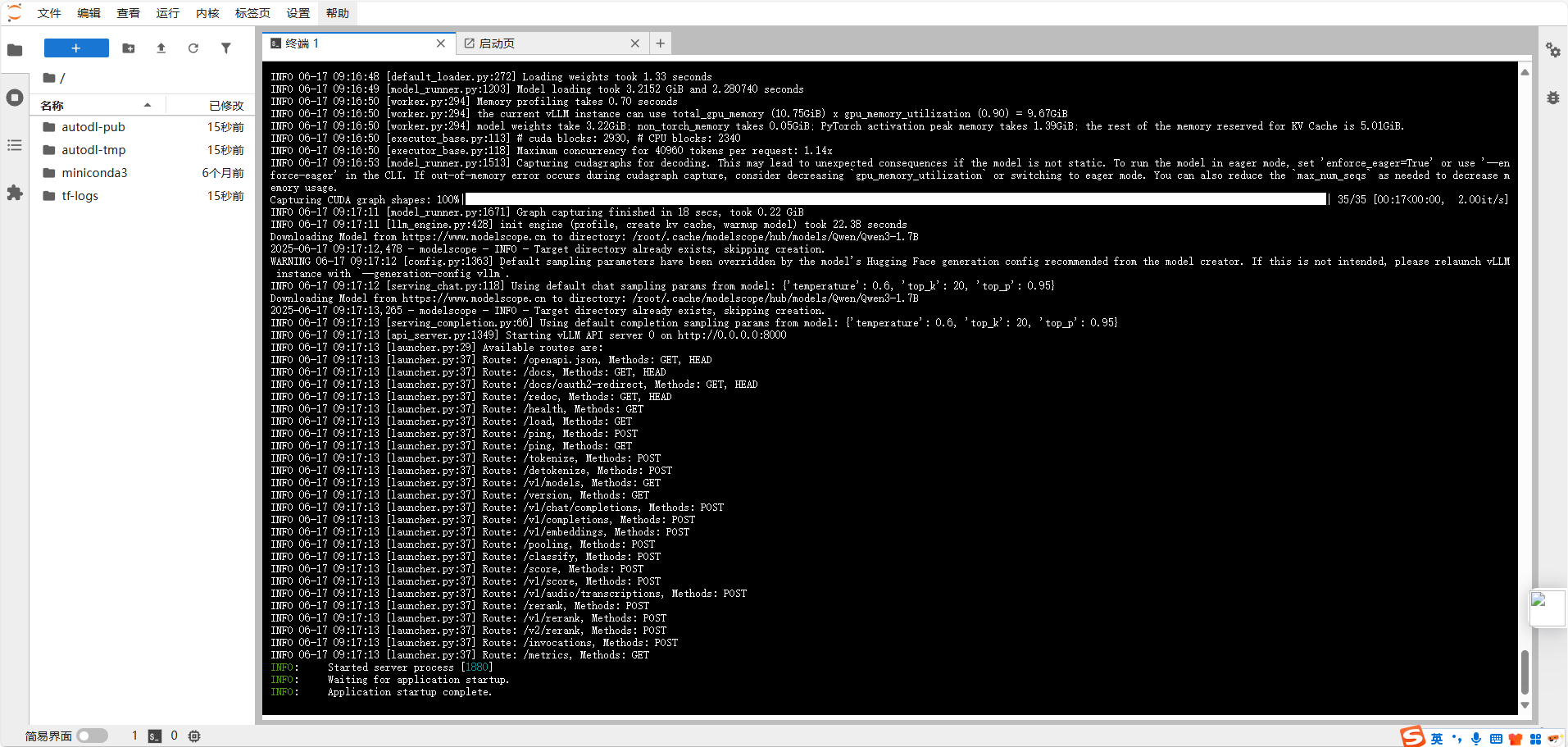
快捷工具显示jupyter就可以进去了



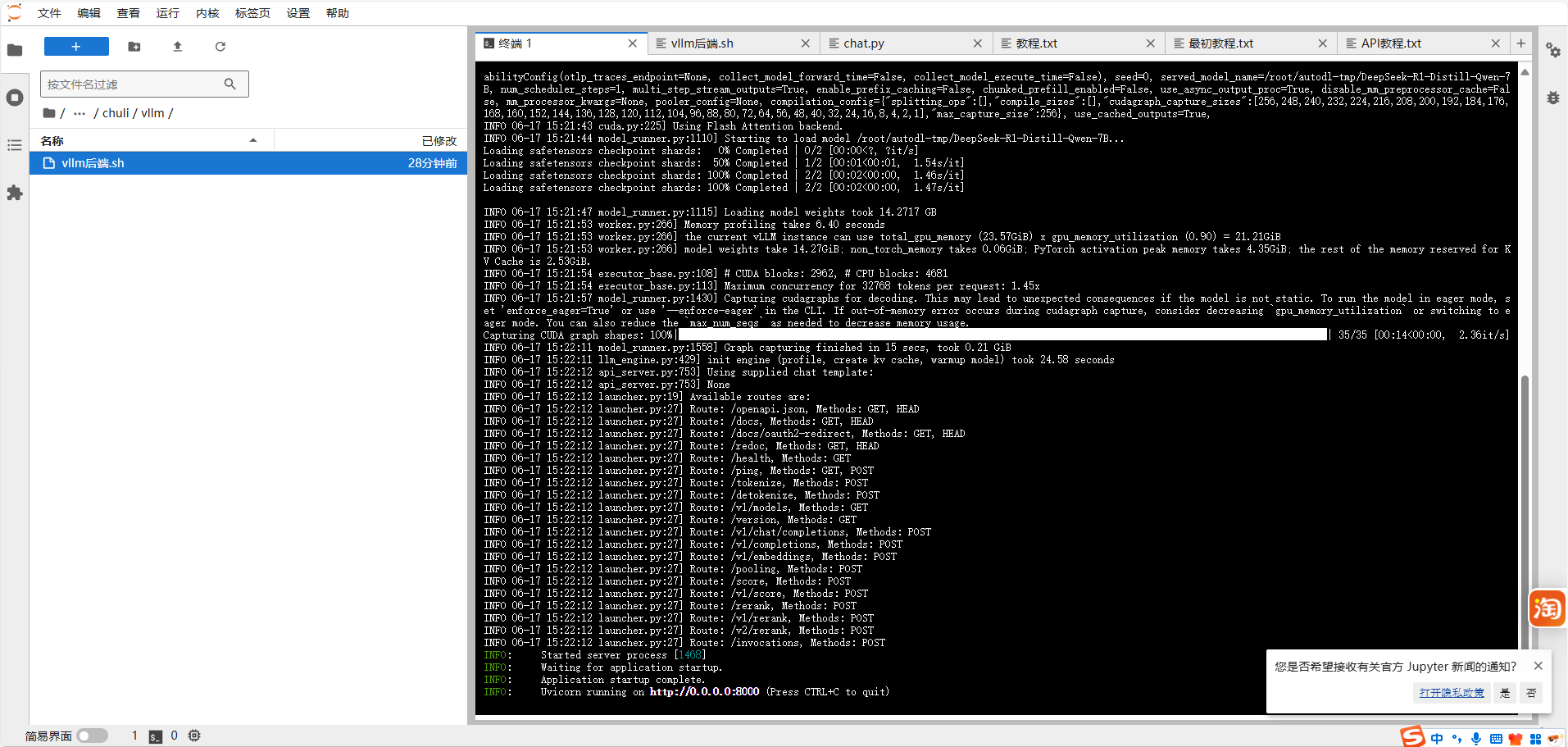
点击jupyter后进入操作界面



将端口改为8000后启动脚本



运行sh脚本 bash /root/deepseek/chuli/vllm/vllm后端.sh

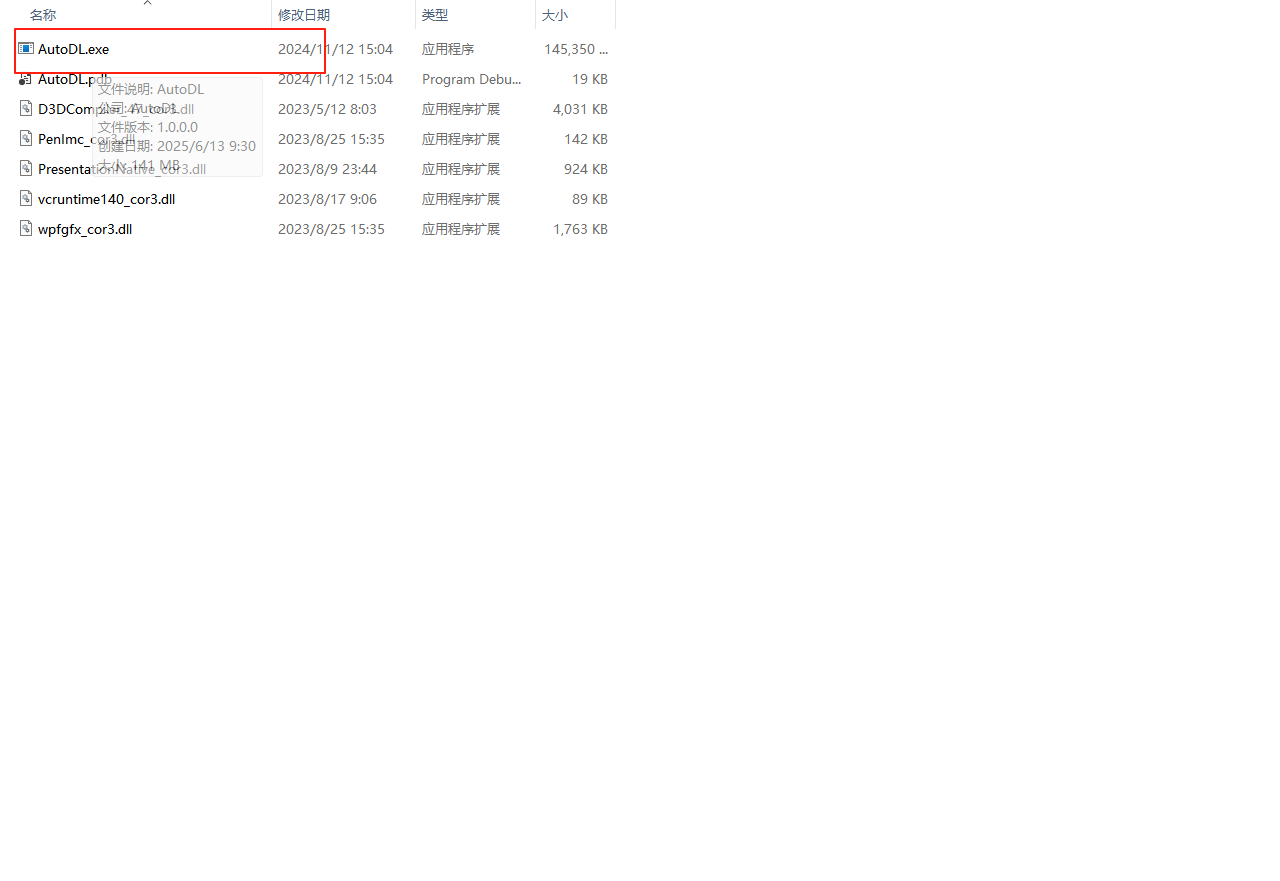


显示下列页面则成功部署

接下来我们要将api端口穿透出来，下载官方工具包



安装包解压后点击AutoDL.exe





复制页面的ssh相关参数到应用中

