Langchain-Chatchat智能对话系统使用指南

1 商品说明

Langchain-Chatchat是一个开源的大模型应用平台，旨在帮助用户快速构建、部署并管理自己的大语言模型应用。无论是企业内部的智能助手，还是个人的知识问答系统，都提供了一个简便的解决方案，支持高效的文档检索和多轮对话，提升工作效率。

本商品通过鲲鹏服务器+EulerOS2.0进行安装部署

2 商品购买

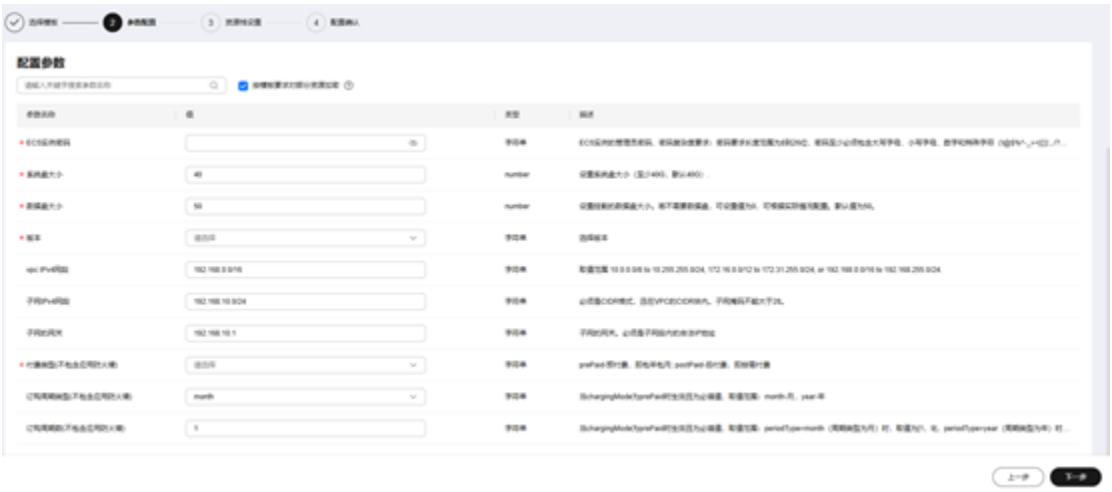
您可以在云商店搜索“Langchain-Chatchat智能对话系统”。

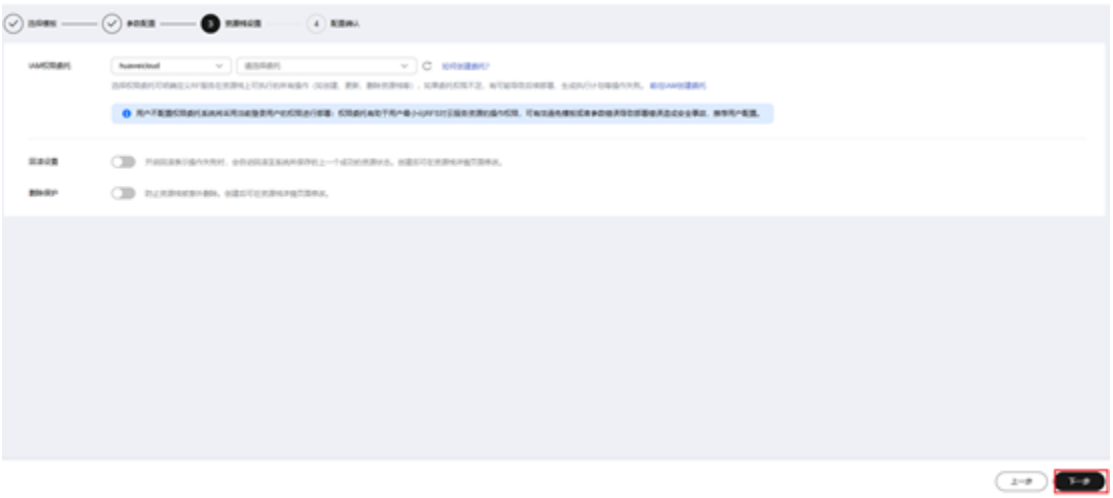
其中，地域、规格、推荐配置使用默认，购买方式根据您的需求选择按需/按月/按年，短期使用推荐按需，长期使用推荐按月/按年，确认配置后点击“立即购买”。

2.1 商品支持自定义 ECS 购买，具体见章节 3.1.1

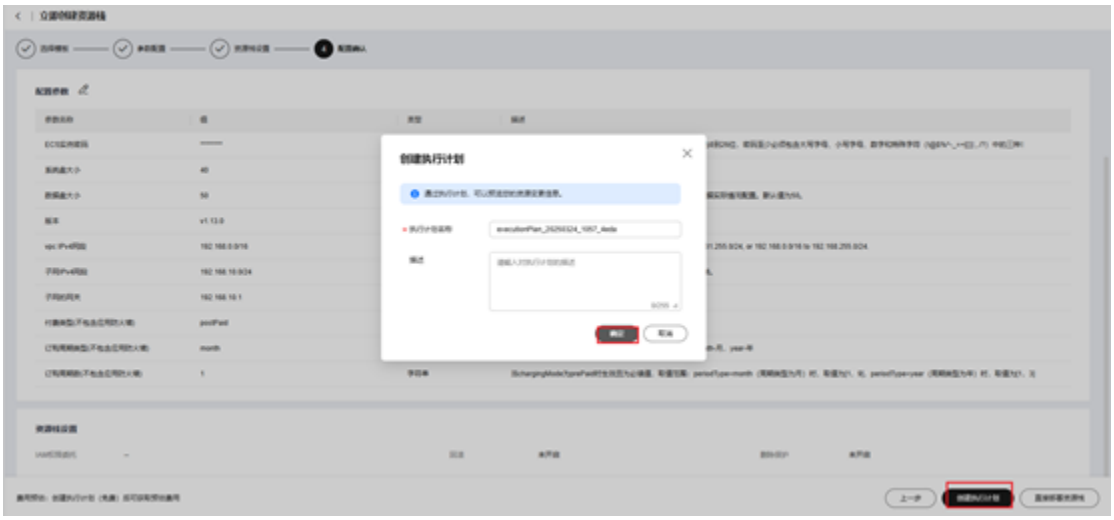
2.2 使用 RFS 模板直接部署

必填项填写后，点击 下一步



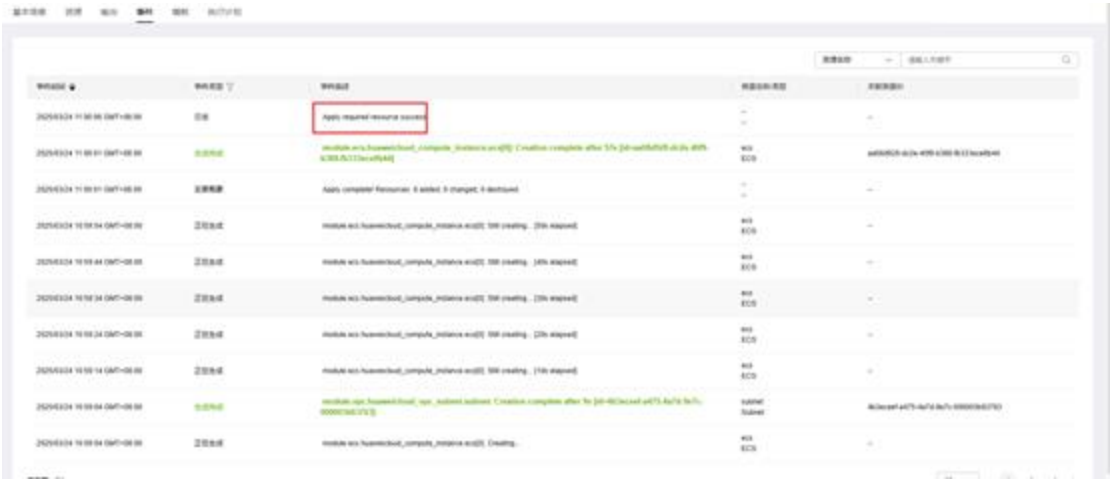


创建直接计划后，点击 确定



点击 部署

如下图“Apply required resource success. ”即为资源创建完成



3 商品资源配置

商品支持**ECS控制台配置**，下面对资源配置的方式进行介绍。

3.1 ECS 控制台配置

3.1.1 准备工作

在使用ECS控制台配置前，需要您提前配置好**安全组规则**。

安全组规则的配置如下：

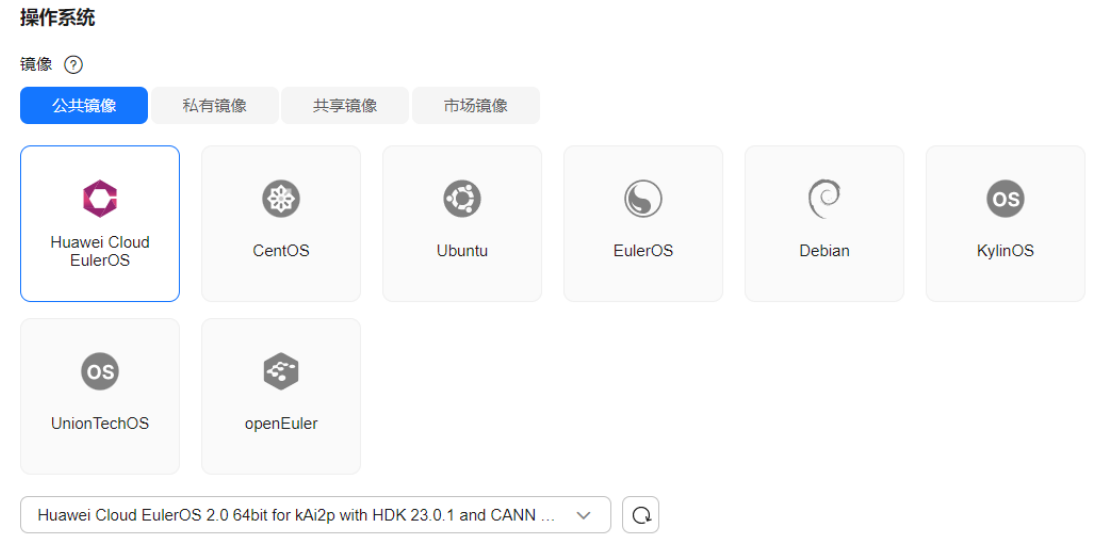
- 入方向规则放通端口chatchat的端口8501 7861，xinference的端口9997，必须包含这些端口才能正常访问使用。
- 入方向规则放通CloudShell连接实例使用的端口22，以便在控制台登录调试。
- 出方向规则一键放通。

3.1.2 创建 ECS

前提工作准备好后，选择ECS控制台配置跳转到购买ECS页面，ECS资源的配置如下图所

示：


基础配置
计费模式
包年/包月
按需计费
竞价计费
按需计费实例不支持备案。了解备案限制
区域
华北-北京四
推荐区域 华北-北京四 华南-广州 华东-上海一 华北-乌兰察布一 西南-贵阳一
云服务器创建后无法更改区域；不同区域之间内网互不相通，请就近选择靠近您业务的区域，减少网络时延。如何选择区域
可用区
随机分配
可用区1
可用区2
可用区3
可用区7
随机至多可用区
实例
规格类型选型
业务场景选型
CPU架构
x86计算
鲲鹏计算
实例筛选
--请选择vCPUs--
--请选择内存--
请输入规格名称模糊搜索
隐藏售罄的规格
鲲鹏通用计算增强型
鲲鹏内存优化型
鲲鹏超高I/O型
CSDN @p_xcn



值得注意的是：

- VPC您可以自行创建
- 安全组选择3.1.1章节中配置的安全组
- 弹性公网IP选择现在购买，推荐选择“按流量计费”，带宽大小可设置为5Mbit/s
- 高级配置需要在高级选项支持注入自定义数据，所以登录凭证不能选择“密码”，选

择创建后设置

- 其余默认或按规则填写即可。

4 商品使用

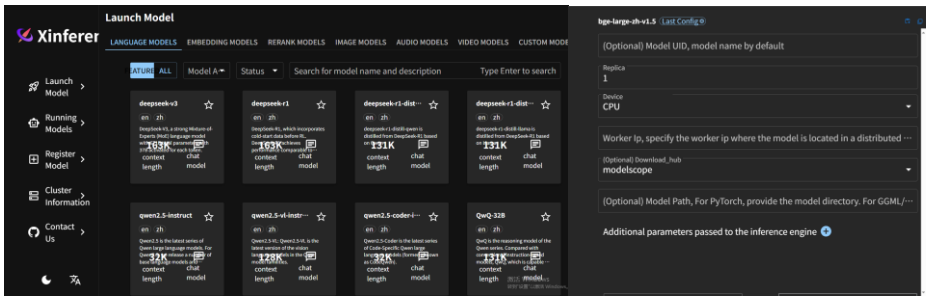
4.1 使用 xinference 拉取模型

4.1.1 启动 xinference

Xinference-local –host 0.0.0.0 –port 9997

4.1.2 拉取模型

使用网址公网ip+9997进入xinference网页，拉取自己需要的vllm、Embedding等模型。



使用modelscope的方式下载自己需要的模型。

4.2 修改模型配置文件启动 Langchain-Chatchat

4.2.1 修改配置文件并初始化

cd /root/chatchat_data目录下，vim model_setting.yaml

修改支持的大模型模型和xinference的模型

```
# 支持的Agent模型
SUPPORT_AGENT_MODELS:
  - chatglm3-6b
  - glm-4
  - openai-api
  - Qwen-2
  - qwen2-instruct
  - gpt-3.5-turbo
  - gpt-4o
  - gemma-3-it

MODEL_PLATFORMS:
  - platform_name: xinference
    platform_type: xinference
    api_base_url: http://1.92.77.123:9997/v1
    api_key: EMPTY
    api_proxy: ''
    api_concurrencies: 5
    auto_detect_model: true
    llm_models:
      - gemma-3-it
    embed_models:
      - bge-large-zh-v1.5
```

chatchat init 初始化

4.2.2 启动 Langchain-chatchat

cd /home/Langchain-chatchat/docker

docker compose up -d

然后使用公网ip+8501打开网页

4.3 上传文档进行知识问答

4.3.1 自建知识库上传文档

自建一个知识库上传自己的文档，然后添加文件到知识库。

4.3.2 进行 RAG 对话

选择大模型和知识库来进行RAG对话。

4.4 参考文档

- https://github.com/chatchat-space/Langchain-Chatchat