在AutoDL中快速部署ChatGLM3-6B模型

本节内容介绍的是在AutoDL平台上租赁算力资源，并完成ChatGLM3-6B的部署调用的流程。根据官方介绍，ChatGLM3-6B 目前支持GPU运行（需要英伟达显卡）、CPU运行以及Apple M系列芯片运行。其中GPU运行需要至少6GB以上显存（4Bit精度运行模式下），而CPU运行则需要至少32G的内存。而由于Apple M系列芯片需要最少13G内存运行。在正式安装之前，需要先确保拥有足够的算力资源，以下推荐的是一种轻量化的部署方式，非常适合入门级的测试开发：AutoDL租赁服务器进行快速部署.

1. 物理机 or 云服务

- 完全小白，对大模型技术没有了解，建议用新人账号白嫖各大云服务平台的免费算力，再考虑购买或者租赁。
- 如果经常做微调实验，或实验室学生系统学习，有自己的物理机将更加方便，按照学习实践部分内容采购即可。
- 为用户提供相关的推理服务，首选云服务，有更大参数量，更好性能的模型选择，随用随停，按量计费。
- “独角兽”公司AI应用/大模型AI技术创新公司……，需要大规模大批量的微调训练或者对内/对外提供大量推理服务，按需配备高性能GPU服务器。

物理机部分大家可以按照前序了解的自行购买，但是这里再次强调，**购买需谨慎**，尤其在二手平台购买二手显卡需要更仔细专业的判断。目前国内市场也会有A、H系列显卡流通，可能是存货、二手、……渠道，但是这类高性能显卡要更专业细致的判断，谨防被骗。

云服务厂商

国内主流

- 阿里云：https://www.aliyun.com/product/ecs/gpu
- 腾讯云：https://cloud.tencent.com/product/gpu
- 火山引擎：https://www.volcengine.com/product/gpu

国外主流

- AWS：https://aws.amazon.com/cn/campaigns/aws-gpu/
- Vultr：https://www.vultr.com/products/cloud-gpu/
- TPU：https://cloud.google.com/gpu/?hl=zh_cn

算力平台

主要适用于学习和训练，不适用于企业级部署提供服务。

- ModelScope：阿里出品，中国的“HuggingFace”，模型开源社区，绑定阿里云有（24GB显存+36小时）GPU环境。https://www.modelscope.cn/home
- Colab：谷歌出品，升级服务仅需 9 美金。https://colab.research.google.com/
- Kaggle：免费，每周 30 小时 T4，P100 可用。https://www.kaggle.com/
- AutoDL：价格亲民，支持 Jupyter Notebook 及 ssh，国内可用。https://www.autodl.com/home

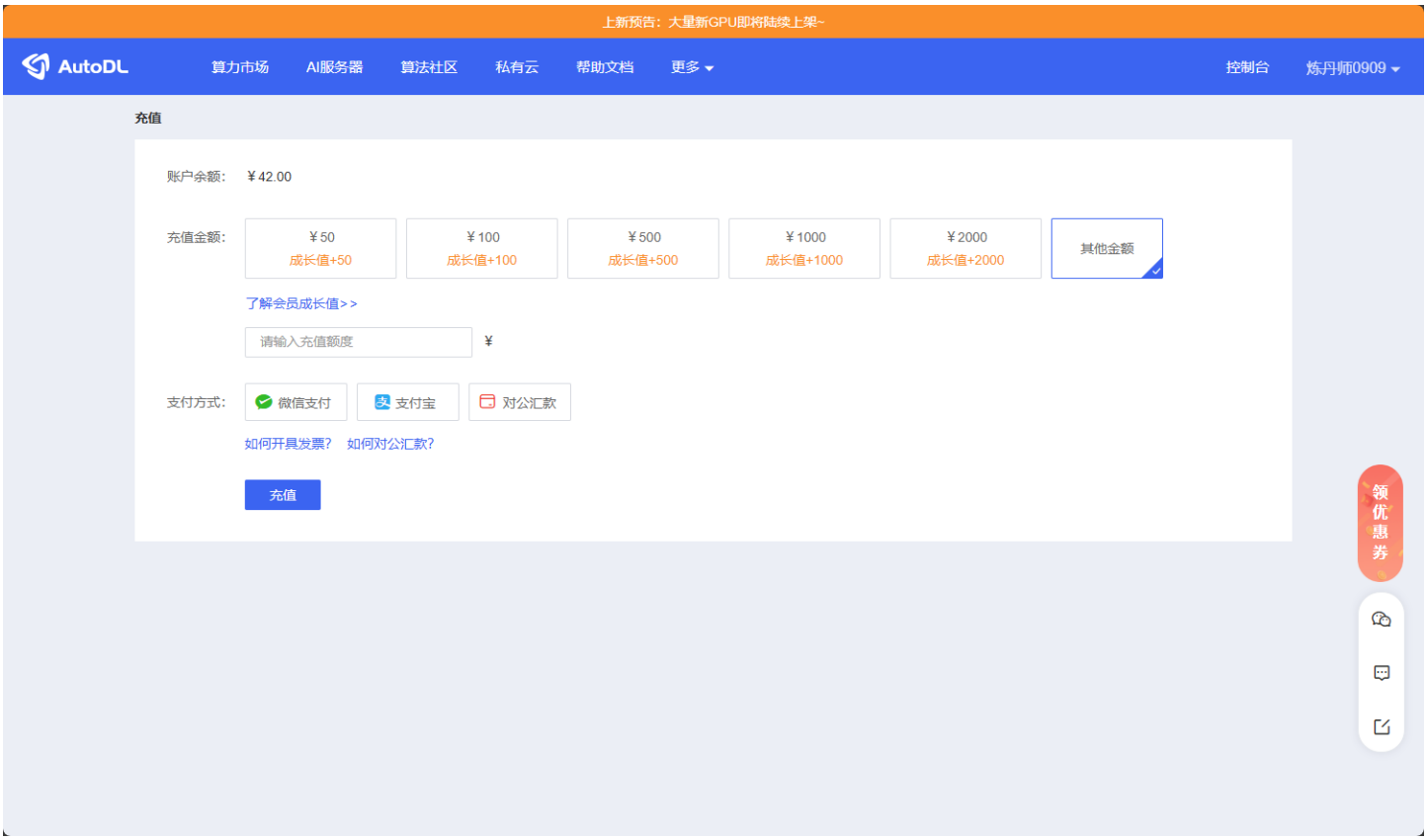
2. 算力准备

在正式安装之前，需要先确保拥有足够的算力资源，以下推荐的是一种轻量化的部署方式，非常适合入门级的测试开发：首先可以从以下链接中进入AutoDL的官方网址，在右上角的选项里可以注册/登录.

https://www.autodl.com/home

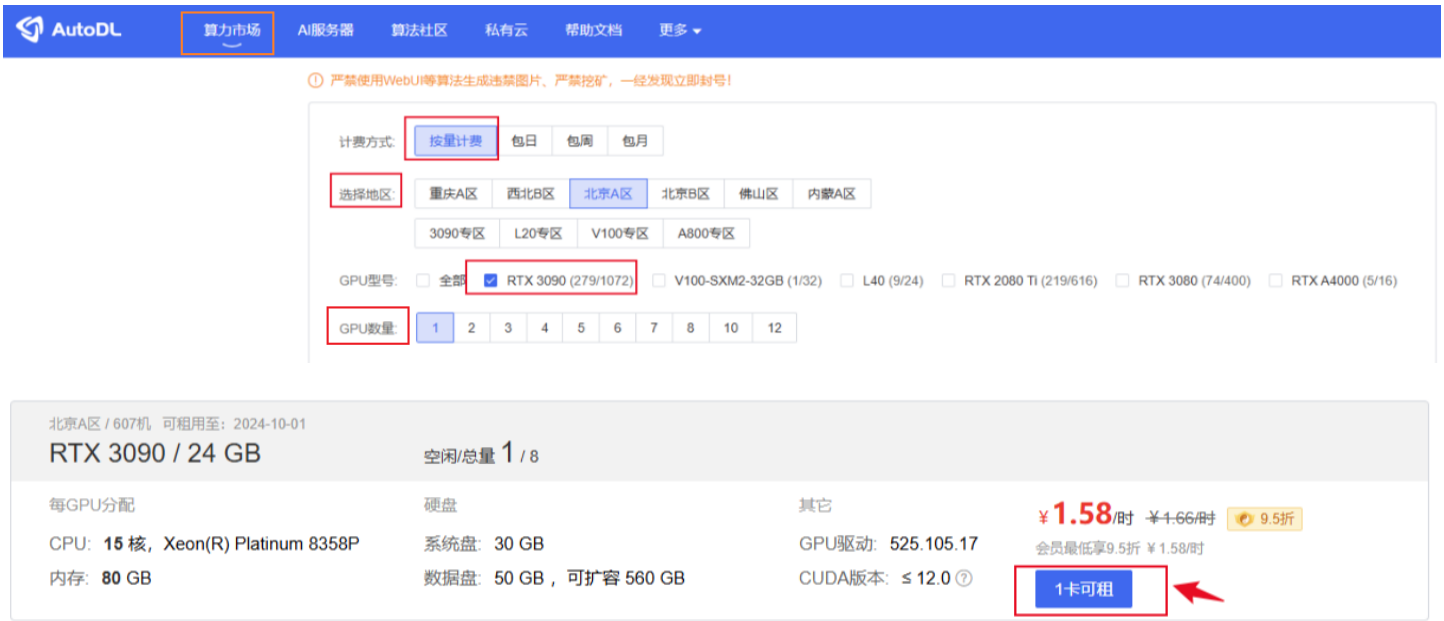
进入界面之后点击右上角的用户信息可以查看余额和进行充值，其金额可以自定义。*注意：只有账户有余额才能在后续算力市场租赁主机。*


控制台
炼丹师0909
炼丹师0909
未实名
ID：2bd2ceda-18bb-4d5c-a8c7-679b33394f82
炼丹会员
可用余额：￥42.00
充值
冻结余额：￥0.00
代金券：￥0.00
容器实例：1
退出登录
炼丹师0909
炼丹会员
认证学生升级炼
等级与会员福利
成长值
距离升级还需44
进入成长值主页
费用信息



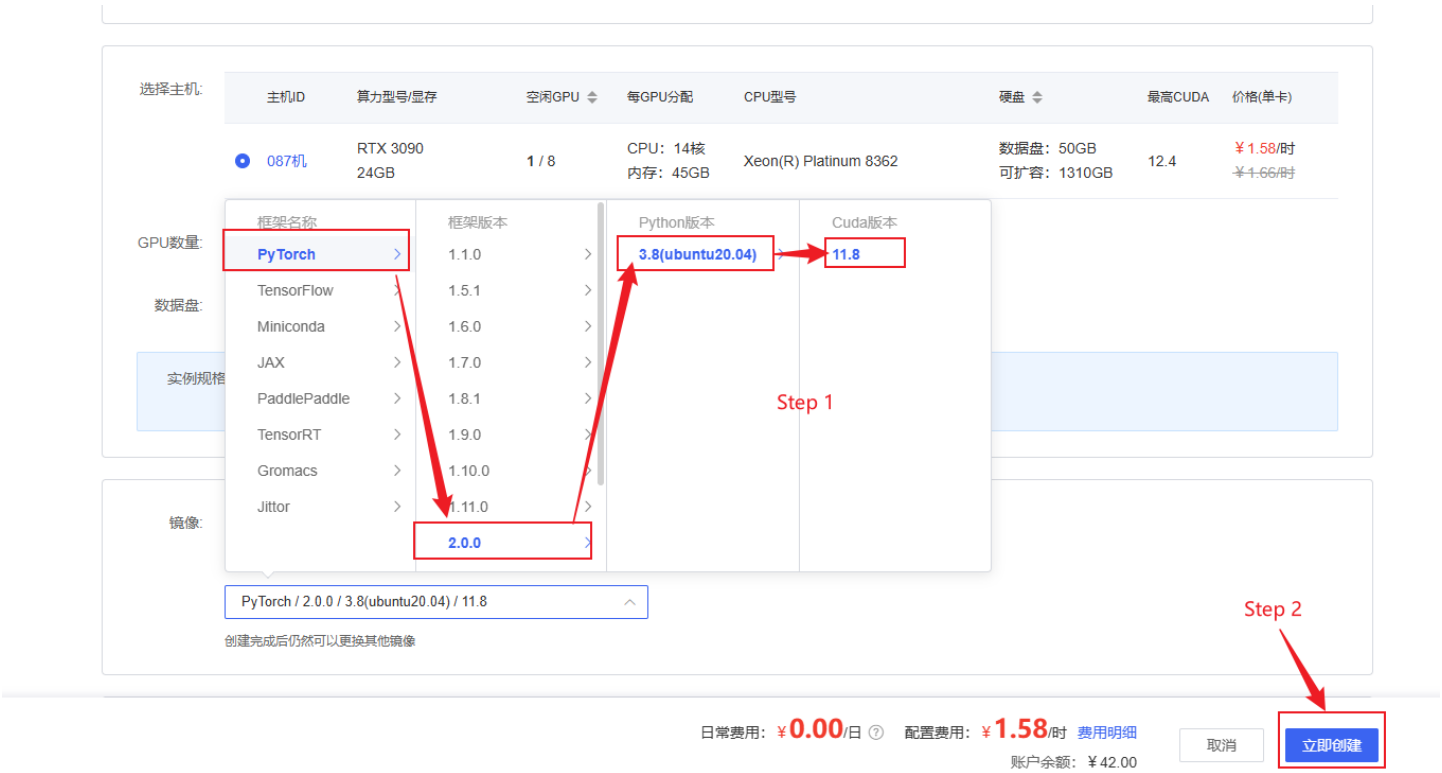
确认用户余额充裕后点击左上角的算力市场，租赁合适的主机，推荐的配置为：*计费方式选择按量计费、地区任选、GPU型号选择RTX3090/24GB、GPU数量选择为1。*

选择RTX3090/24GB卡的理由是ChatGLM3-6B的GPU运行需要至少6GB以上显存（4Bit精度运行模式下），而CPU运行则需要至少32G的内存。其中CPU运行模式下内存占用过大且运行效率较低，GPU模式部署才能有效的进行大模型的学习实践。基于性能和性价比进行考量，我们建议选择以上参数进行部署。



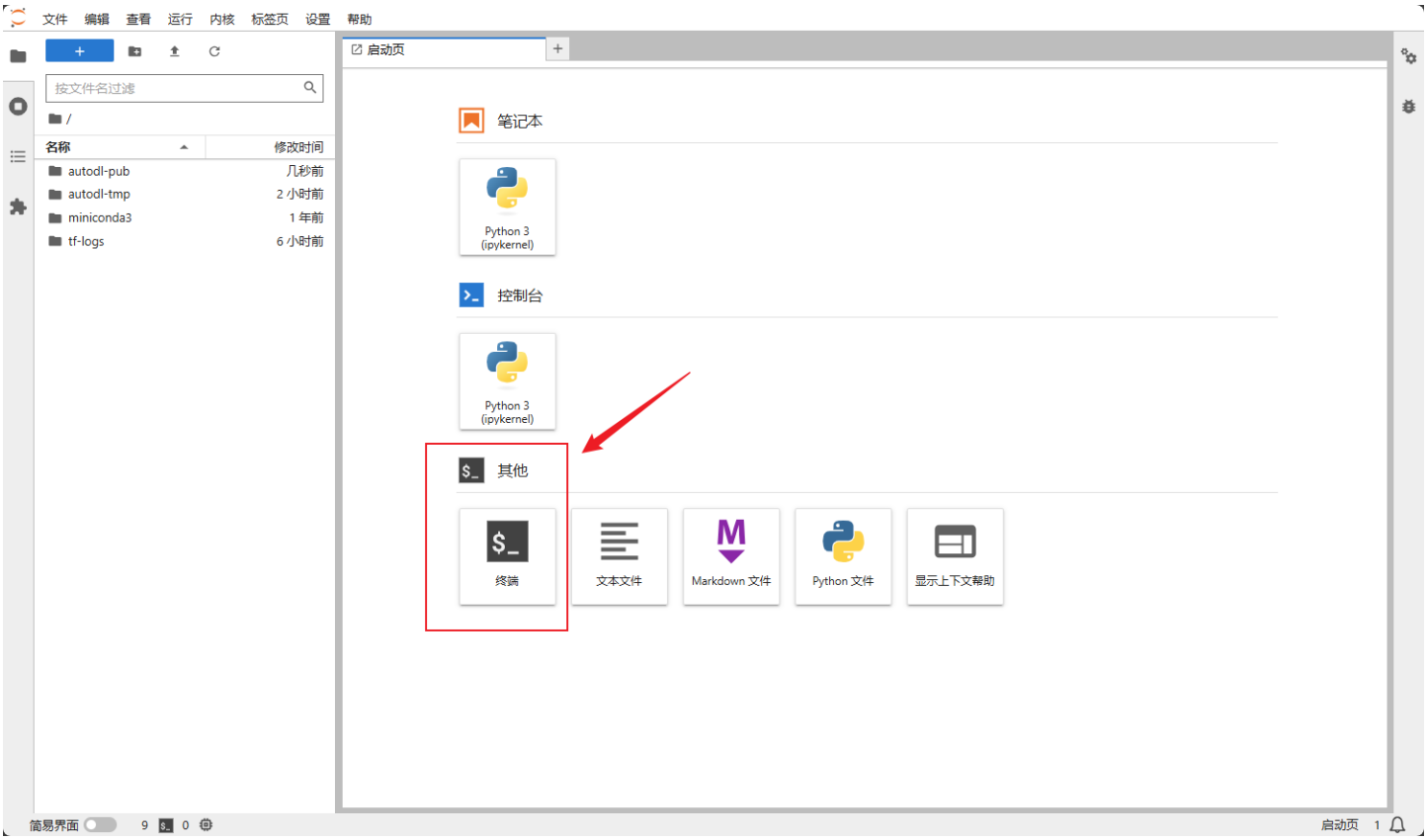
选择好合适的主机后需要在下方的镜像栏中选择适合的框架——框架名称：PyTorch，框架版本：2.0.0，Python版本：3.8（Ubuntu20.04），Cuda版本11.8.选择好之后点击右下角的立即创建便可完成配置。

其中 PyTorch 是一个流行的深度学习框架，支持大规模模型的训练和推理。Python 3.8 是一个稳定且常用的版本，兼容大多数机器学习库和工具。选择 Ubuntu 20.04 作为操作系统版本是因为其长期支持和广泛使用，特别适合在生产环境中部署。Cuda 是 NVIDIA 提供的并行计算平台和编程模型，支持 GPU 加速。选择 Cuda 11.8 版本是因为它与 PyTorch 2.0.0 兼容。



创建完成后，点击左边栏的容器实例便可随时找到配置好的实例，在快捷工具栏中点击Jupyter lab开始模型的安装部署。

3. 换源和安装依赖包



进入Jupyter lab打开终端开始环境配置，首先要进行的是 pip 换源和安装依赖包。点击启动终端，在其中逐行输入以下代码以实现功能。

在终端通过命令升级 pip，确保使用的是最新版本的 pip，这样可以避免在安装库时出现兼容性问题。

更换 pip 的默认源为清华大学的镜像源，以加速 Python 库的下载和安装。

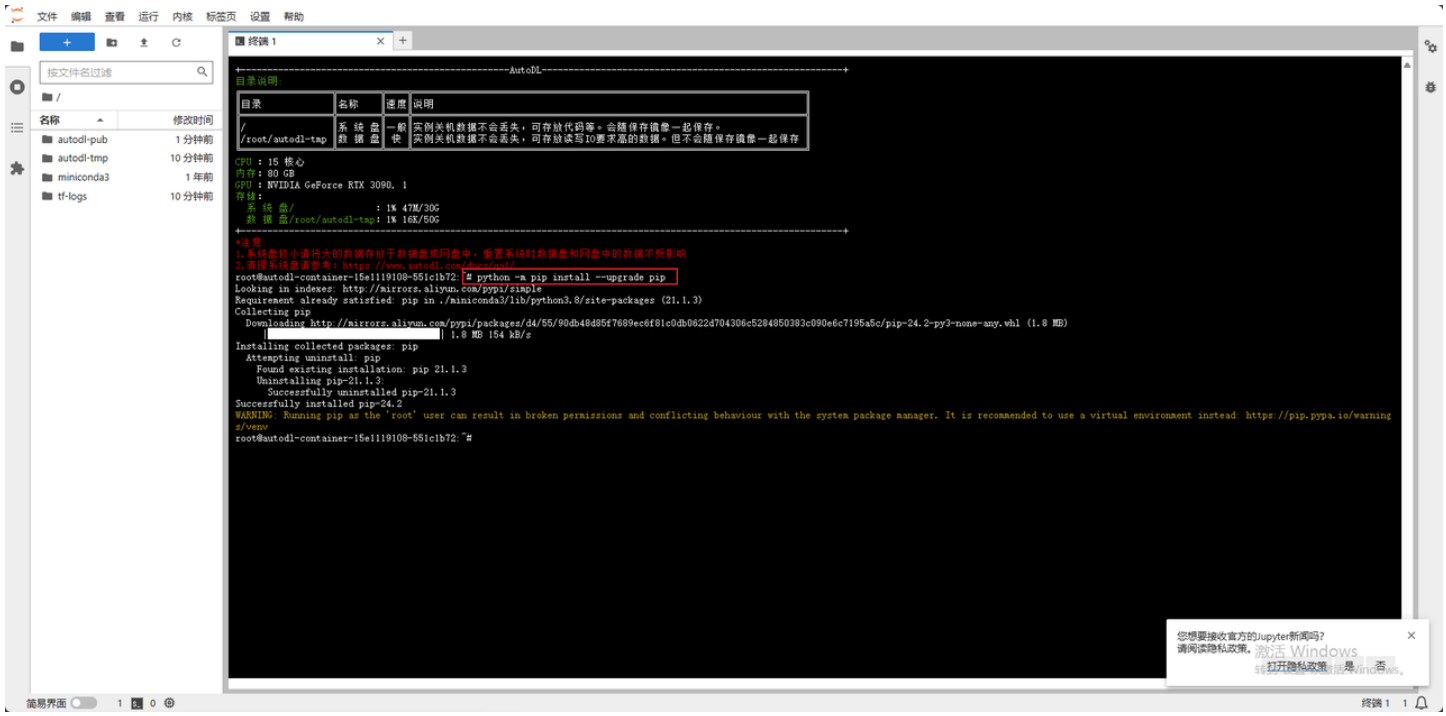
以下是安装的库的介绍：

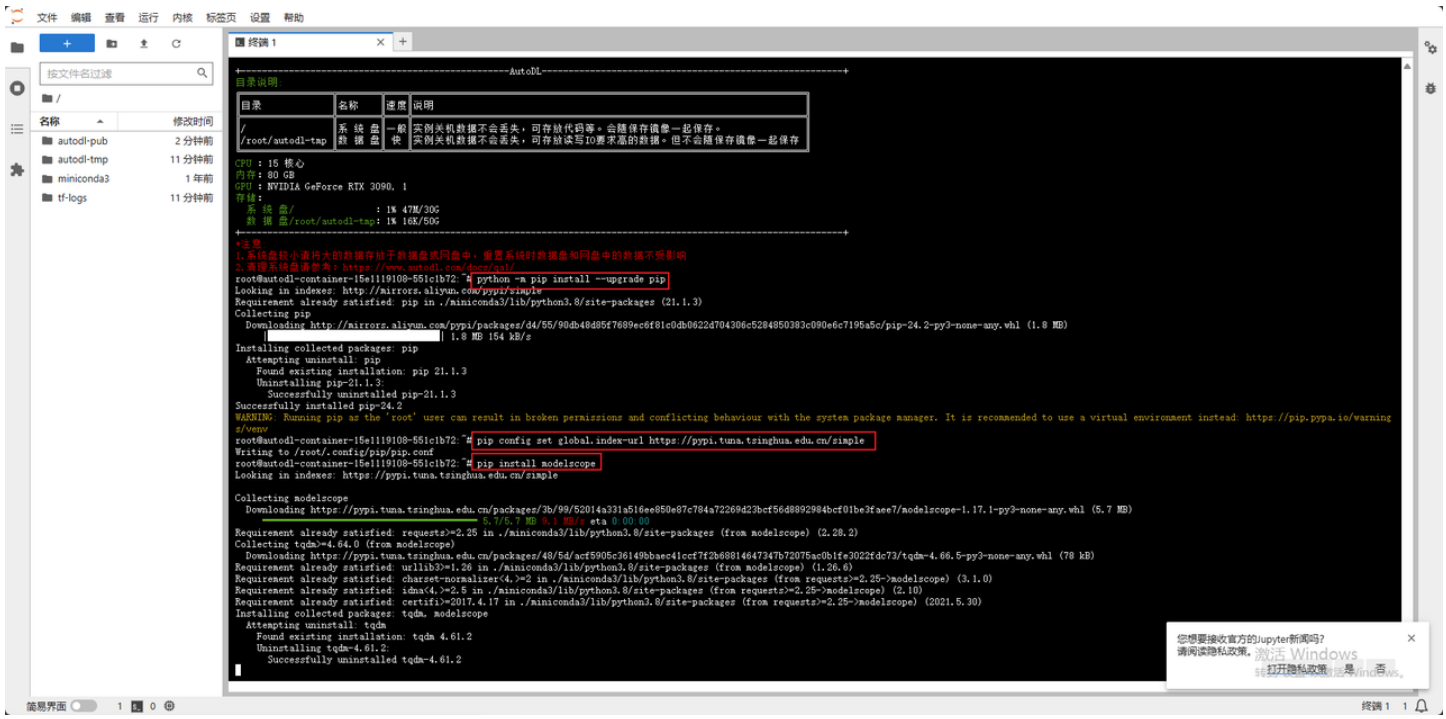
modelscope: 用于模型推理和部署的库，支持多种机器学习和深度学习模型。

transformers: 包含了大量预训练的 Transformer 模型，包括 BERT、GPT 等等。

sentencepiece: 一个用于处理文本的库，特别是对子词单元进行分词操作，常用于自然语言处理任务。

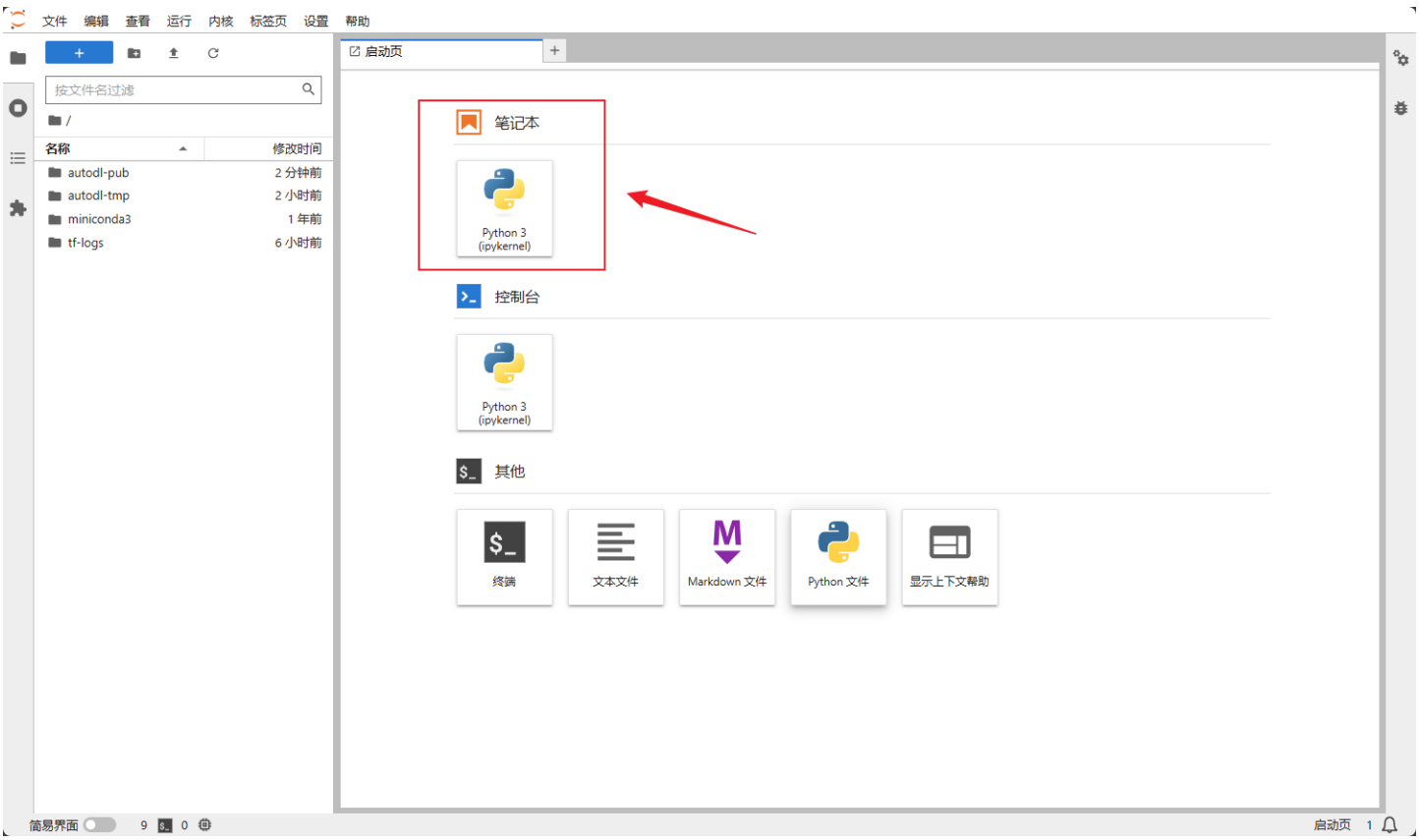
```
# 升级pip
python -m pip install --upgrade pip
# 更换 pypi 源加速库的安装
pip config set global.index-url https://pypi.tuna.tsinghua.edu.cn/simple

pip install modelscope
pip install transformers
pip install sentencepiece
```


文件 编辑 查看 运行 内核 标签页 设置 帮助
终端 1
按文件名过滤
/
名称
修改时间
autodl-pub 1 分钟前
autodl-tmp 10 分钟前
miniconda3 1 年前
tf-logs 10 分钟前
-----------AutoDL-----------
目录说明:
目录 名称 速度 说明
/ 系统盘 一般 实例关机数据不会丢失，可存放代码等。会随保存镜像一起保存。
/root/autodl-tmp 数据盘 快 实例关机数据不会丢失，可存放读写IO要求高的数据。但不会随保存镜像一起保存
CPU : 15 核心
内存: 80 GB
GPU : NVIDIA GeForce RTX 3090, 1
存储:
系统盘/ : 1% 47M/30G
数据盘/root/autodl-tmp: 1% 16K/50G
*注意
1.系统盘较小请将大的数据存放于数据盘或同盘中，重置系统时数据盘和同盘中的数据不受影响
2.查理系统盘请参考：https://www.autodl.com/docs/qa1/
root@autodl-container-15e1119108-551c1b72:~# python -m pip install --upgrade pip
Looking in indexes: http://mirrors.aliyun.com/pypi/simple
Requirement already satisfied: pip in ./miniconda3/lib/python3.8/site-packages (21.1.3)
Collecting pip
Downloading http://mirrors.aliyun.com/pypi/packages/d4/55/90db48d85f7689ec6f81c0db0622d704306c5284850383c090e6c7195a5c/pip-24.2-py3-none-any.whl (1.8 MB)
1.8 MB 154 kB/s
Installing collected packages: pip
Attempting uninstall: pip
Found existing installation: pip 21.1.3
Uninstalling pip-21.1.3:
Successfully uninstalled pip-21.1.3
Successfully installed pip-24.2
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
root@autodl-container-15e1119108-551c1b72:~#
您想要接收官方的Jupyter新闻吗?
请阅读隐私政策。
打开隐私政策 是 否
简易界面
终端 1


文件 编辑 查看 运行 内核 标签页 设置 帮助
终端 1
按文件名过滤
/
名称
修改时间
autodl-pub 2 分钟前
autodl-tmp 11 分钟前
miniconda3 1 年前
tf-logs 11 分钟前
-----------AutoDL-----------
目录说明:
目录 名称 速度 说明
/ 系统盘 一般 实例关机数据不会丢失，可存放代码等。会随保存镜像一起保存。
/root/autodl-tmp 数据盘 快 实例关机数据不会丢失，可存放读写IO要求高的数据。但不会随保存镜像一起保存
CPU : 15 核心
内存: 80 GB
GPU : NVIDIA GeForce RTX 3090, 1
存储:
系统盘/ : 1% 47M/30G
数据盘/root/autodl-tmp: 1% 16K/50G
*注意
1.系统盘较小请将大的数据存放于数据盘或同盘中，重置系统时数据盘和同盘中的数据不受影响
2.查理系统盘请参考：https://www.autodl.com/docs/qa1/
root@autodl-container-15e1119108-551c1b72:~# python -m pip install --upgrade pip
Looking in indexes: http://mirrors.aliyun.com/pypi/simple
Requirement already satisfied: pip in ./miniconda3/lib/python3.8/site-packages (21.1.3)
Collecting pip
Downloading http://mirrors.aliyun.com/pypi/packages/d4/55/90db48d85f7689ec6f81c0db0622d704306c5284850383c090e6c7195a5c/pip-24.2-py3-none-any.whl (1.8 MB)
1.8 MB 154 kB/s
Installing collected packages: pip
Attempting uninstall: pip
Found existing installation: pip 21.1.3
Uninstalling pip-21.1.3:
Successfully uninstalled pip-21.1.3
Successfully installed pip-24.2
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
root@autodl-container-15e1119108-551c1b72:~# pip config set global.index-url https://pypi.tuna.tsinghua.edu.cn/simple
Writing to /root/.config/pip/pip.conf
root@autodl-container-15e1119108-551c1b72:~# pip install modelscope
Looking in indexes: https://pypi.tuna.tsinghua.edu.cn/simple
Collecting modelscope
Downloading https://pypi.tuna.tsinghua.edu.cn/packages/3b/99/52014a331a516ee850e87c784a72269d23bcf56d8892984bcf01be3faee7/modelscope-1.17.1-py3-none-any.whl (5.7 MB)
5.7/5.7 MB 9.1 MB/s eta 0:00:00
Requirement already satisfied: requests>=2.25 in ./miniconda3/lib/python3.8/site-packages (from modelscope) (2.28.2)
Collecting tqdm>=4.64.0 (from modelscope)
Downloading https://pypi.tuna.tsinghua.edu.cn/packages/48/5d/acf5905c36149bbaec41ccf7f2b68814647347b72075ac0b1fe3022fdc73/tqdm-4.66.5-py3-none-any.whl (78 kB)
Requirement already satisfied: urllib3>=1.26 in ./miniconda3/lib/python3.8/site-packages (from modelscope) (1.26.6)
Requirement already satisfied: charset-normalizer<4,>=2 in ./miniconda3/lib/python3.8/site-packages (from requests>=2.25->modelscope) (3.1.0)
Requirement already satisfied: idna<4,>=2.5 in ./miniconda3/lib/python3.8/site-packages (from requests>=2.25->modelscope) (2.10)
Requirement already satisfied: certifi>=2017.4.17 in ./miniconda3/lib/python3.8/site-packages (from requests>=2.25->modelscope) (2021.5.30)
Installing collected packages: tqdm, modelscope
Attempting uninstall: tqdm
Found existing installation: tqdm 4.61.2
Uninstalling tqdm-4.61.2:
Successfully uninstalled tqdm-4.61.2
您想要接收官方的Jupyter新闻吗?
请阅读隐私政策。
打开隐私政策 是 否
简易界面
终端 1

4. 模型下载

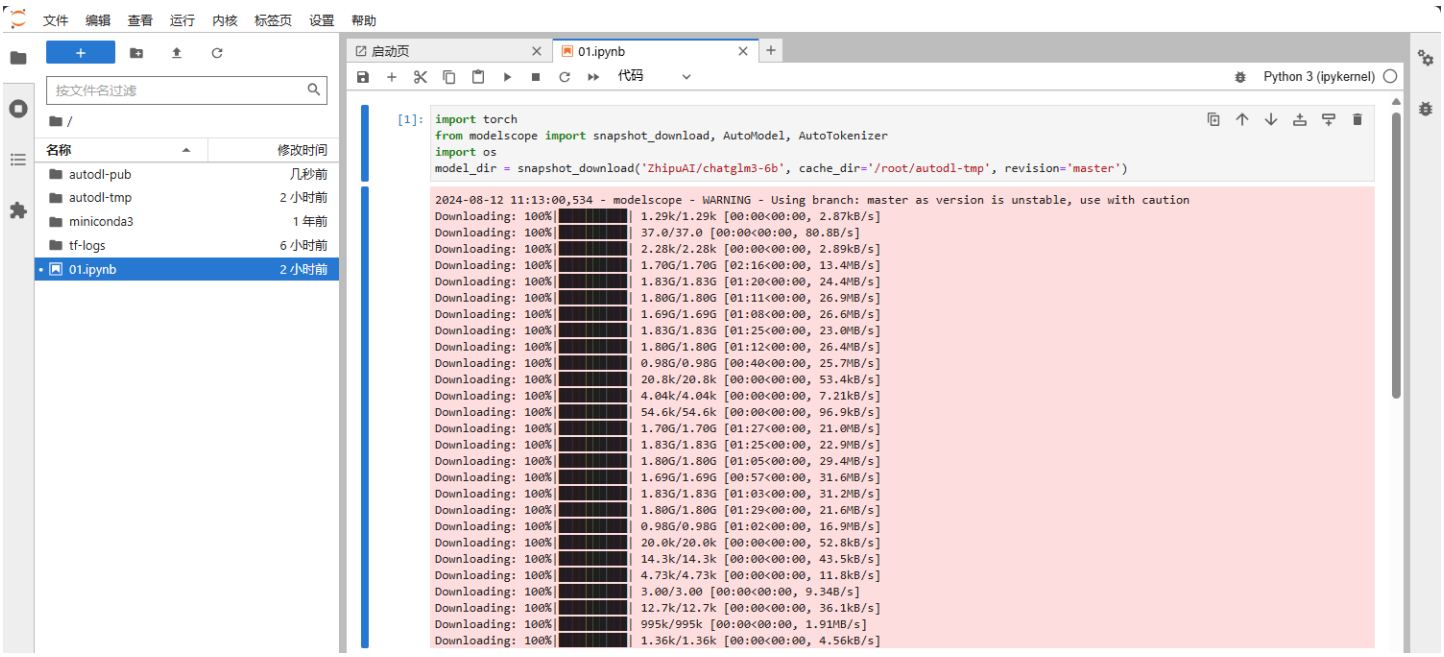
在启动页打开新的Jupiter notebook进行模型的下载。



这里选择的是使用 modelscope 中的snapshot_download函数下载模型，这个函数中的第一个参数为模型名称，第二个参数 cache_dir 为模型的下载路径。

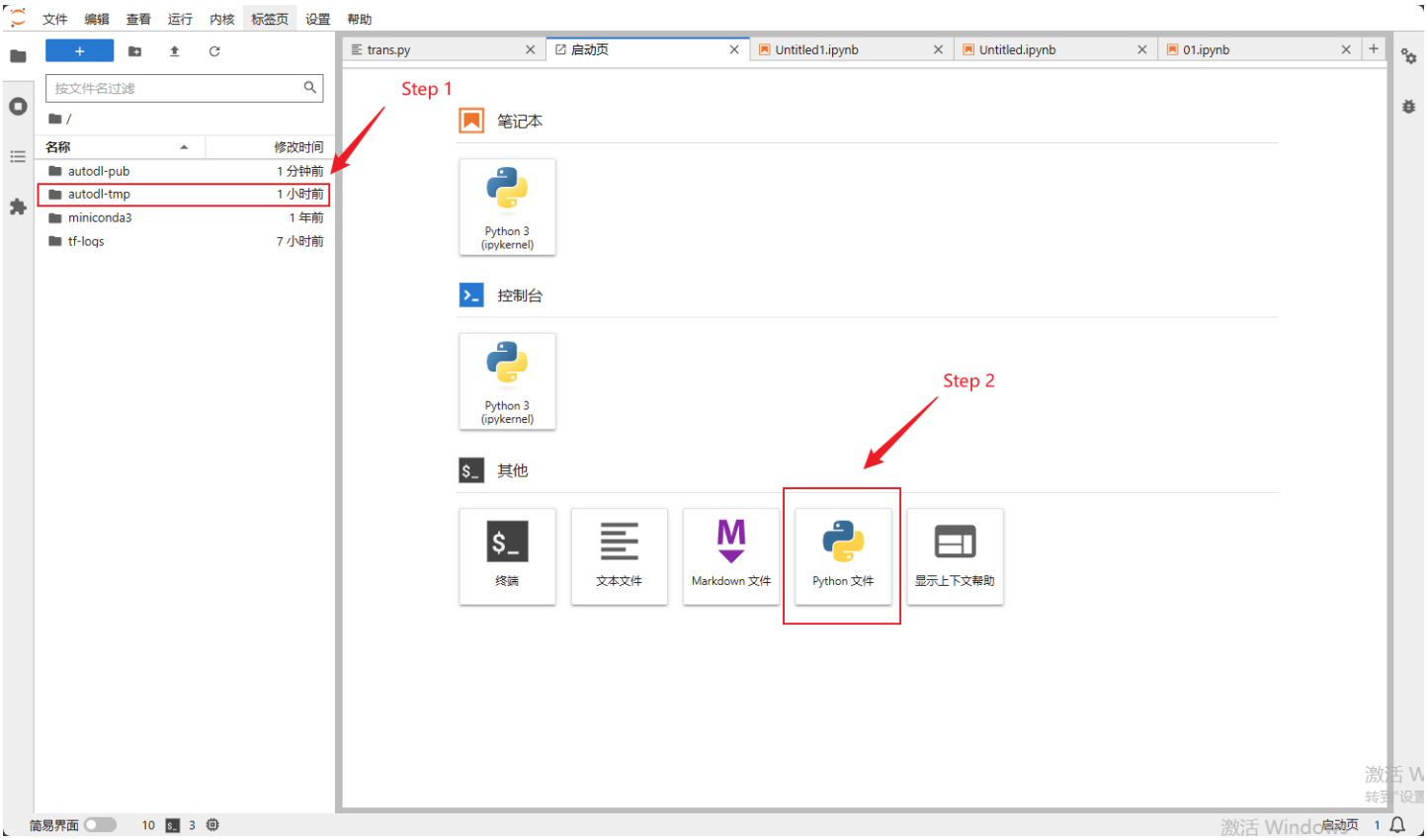
打开Jupyter Lab文件执行以下代码进行下载，ChatGLM3-6B模型大小为 14 GB，下载模型大概需要 20~25 分钟。

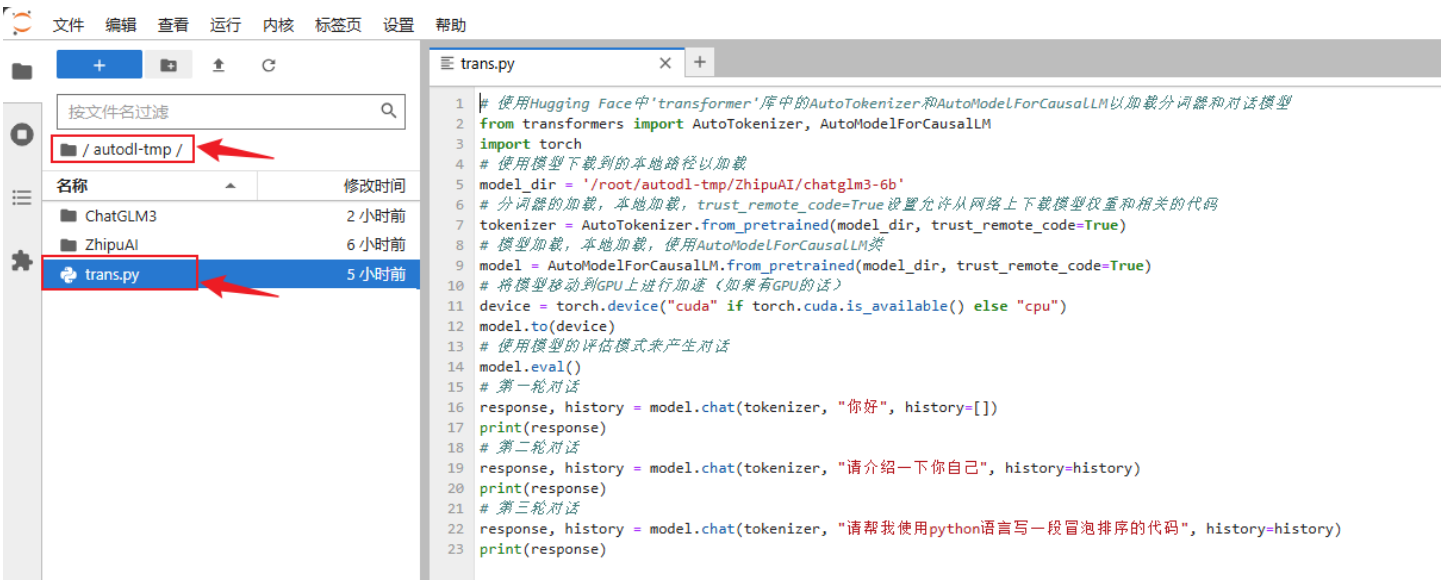
```
import torch
from modelscope import snapshot_download, AutoModel, AutoTokenizer
import os
model_dir = snapshot_download('ZhipuAI/chatglm3-6b', cache_dir='/root/autodl-tmp', revision='master')
```



5. 启动模型的代码

在/root/autodl-tmp路径下新建trans.py文件并在其中输入以下内容





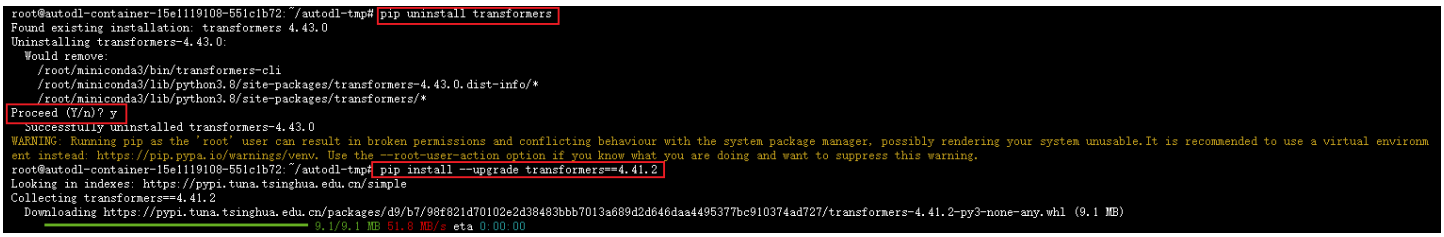

```python
from transformers import AutoTokenizer, AutoModelForCausalLM # 使用Hugging Face中'transformer'库中的AutoTokenizer和AutoModelForCausalLM以加载分词器和对话模型
import torch
model_dir = '/root/autodl-tmp/ZhipuAI/chatglm3-6b'  # 使用模型下载到的本地路径以加载
tokenizer = AutoTokenizer.from_pretrained(model_dir, trust_remote_code=True) # 分词器的加载，本地加载，trust_remote_code=True设置允许从网络上下载模型权重和相关的代码
model = AutoModelForCausalLM.from_pretrained(model_dir, trust_remote_code=True)  # 模型加载，本地加载，使用AutoModelForCausalLM类
device = torch.device("cuda" if torch.cuda.is_available() else "cpu") # 将模型移动到GPU上进行加速（如果有GPU的话）
```

```
model.to(device)
model.eval()  # 使用模型的评估模式来产生对话
# 第一轮对话
response, history = model.chat(tokenizer, "你好", history=[])
print(response)
# 第二轮对话
response, history = model.chat(tokenizer, "请介绍一下你自己", history=history)
print(response)
# 第三轮对话
response, history = model.chat(tokenizer, "请帮我使用python语言写一段冒泡排序的代码", history=history)
print(response)
```

6. 部署运行

需要注意的是，如果transfomers的版本不匹配会导致报错，因此我们需要先降其版本。回到启动页打开终端分别输入以下指令将版本确定至4.37.2：

```
pip uninstall transformers  #卸载当前版本
pip install --upgrade transformers==4.37.2  #安装指定版本
```



随后在终端输入以下指令，可以发现在平台上部署成功。可以看到，终端返回了前面trans.py文件提出的三个问题。

```
cd /root/autodl-tmp #将路径导向指定位置
Python trans.py  #执行对应文件
```

```
root@autodl-container-15e1119108-551c1b72: /autodl-tmp# python trans.py
Setting eos_token is not supported, use the default one.
Setting pad_token is not supported, use the default one.
Setting unk_token is not supported, use the default one.
Loading checkpoint shards: 100%|████████████████████████████████████████████████████████████| 7/7 [00:01<00:00,  4.47it/s]
你好 👋 我是人工智能助手 ChatGLM3-6B，很高兴见到你，欢迎问我任何问题。
我是一个人工智能助手 ChatGLM3-6B，是清华大学 KEG 实验室和智谱 AI 公司于 2023 年共同训练的语言模型。我的任务是针对用户的问题和要求提供适当的答复和支持。我是一段程序，并没有物理意义上的实体，所以不能像真正的中国人那样和你进行交流。
当然可以，以下是使用 Python 编写的冒泡排序代码：

```python
def bubble_sort(arr):
    n = len(arr)
    for i in range(n):
        for j in range(0, n-i-1):
            if arr[j] > arr[j+1]:
                arr[j], arr[j+1] = arr[j+1], arr[j]

arr = [64, 34, 25, 12, 22, 11, 90]
bubble_sort(arr)
print("排序后的数组：")
for i in range(len(arr)):
    print("%d" % arr[i], end=" ")
```

你可以将这段代码复制粘贴到你的 Python 编辑器中，并运行它。这将输出排序后的数组。
```