

大模型]GLM-4-9B-chat FastApi 部署调用

FL1623863129 于 2024-06-11 06:04:59 发布 阅读量1.3k 收藏 22 点赞数 12

分类专栏: 大模型 文章标签: fastapi

大模型 专栏收录该内容 10 订阅 105 篇

境准备

AutoDL 平台中租一个 3090 等 24G 显存的显卡机器，如下图所示镜像选择 PyTorch -> 2.1.0 -> 3.10(ubuntu22.04) -> 12.1。

434机

RTX 3090
24GB

1 / 12

CPU: 18核
内存: 60GB

AMD EPYC 9754

框架名称	1.5.1	>	Python版本	Cuda版本
PyTorch	1.6.0	>	3.10(ubuntu22.04)	12.1
TensorFlow	1.7.0	>		
Miniconda	1.8.1	>		
JAX	1.9.0	>		
PaddlePaddle	1.10.0	>		
TensorRT	1.11.0	>		
Gromacs	2.0.0	>		
Jittor	2.1.0	>		
	2.3.0	>		

请选择框架名称/框架版本/Python版本/Cuda版本

创建完成后仍然可以更换其他镜像

下来打开刚刚租用 服务器 的 JupyterLab，并且打开其中的终端开始环境配置、模型下载和运行 demo。

换源和安装依赖包。

```
1 # 升级pip
2 python -m pip install --upgrade pip
3 # 更换 pypi 源加速库的安装
4 pip config set global.index-url https://pypi.tuna.tsinghua.edu.cn/simple
5
6 pip install fastapi==0.104.1
7 pip install uvicorn==0.24.0.post1
8 pip install requests==2.25.1
9 pip install modelscope==1.9.5
10 pip install transformers==4.37.2
11 pip install streamlit==1.24.0
12 pip install sentencepiece==0.1.99
13 pip install accelerate==0.24.1
14 pip install tiktoken==0.7.0
```

FL1623863129 关注

12

考虑到部分同学配置环境可能会遇到一些问题，我们在 AutoDL 平台准备了 GLM-4 的环境镜像，该镜像适用于本教程需要 GLM-4 的部署环境。点击下方 AutoDL 示例即可。（vLLM 对 torch 版本要求较高，且越高的版本对模型的支持更全，效果更好，所以新建一个全新的镜像。）

<https://www.codewithgpu.com/i/datawhalechina/self-llm/GLM-4>

型下载

用 `modelscope` 中的 `snapshot_download` 函数下载模型，第一个参数为模型名称，参数 `cache_dir` 为模型的下载路径。

`/root/autodl-tmp` 路径下新建 `download.py` 文件并在其中输入以下内容，粘贴代码后记得保存文件，如下图所示。并运行 `python /root/autodl-tmp/download.py`，模型大小为 18 GB，下载模型大概需要 10~20 分钟。

```
1 import torch
2 from modelscope import snapshot_download, AutoModel, AutoTokenizer
3 import os
4 model_dir = snapshot_download('ZhipuAI/glm-4-9b-chat', cache_dir='/root/autodl-tmp', revision='master')
```

端出现下图结果表示下载成功。

```
@autodl-container-48b1458428-65e58bb8 ~# python autodl-tmp/download.py
-06-05 13:37:15.846 - modelscope - INFO - PyTorch version 2.1.2+cu121 Found.
-06-05 13:37:15.846 - modelscope - INFO - Loading ast index from /root/.cache/modelscope/ast_indexer
-06-05 13:37:15.846 - modelscope - INFO - No valid ast index found from /root/.cache/modelscope/ast_indexer, generating ast index from prebuilt!
-06-05 13:37:15.953 - modelscope - INFO - Loading done! Current index file version is 1.9.5, with md5 9c6e7583447663c9cc232a819ce49422 and a total number of 945 components indexed
loading: 100%| 1.37k/1.37k
loading: 100%| 36.0/36.0
loading: 100%| 2.21k/2.21k
loading: 100%| 6.33k/6.33k
loading: 100%| 1.81G/1.81G
loading: 100%| 1.69G/1.69G
loading: 100%| 1.83G/1.83G
loading: 100%| 1.80G/1.80G
loading: 100%| 1.69G/1.69G
loading: 100%| 1.83G/1.83G
loading: 100%| 1.80G/1.80G
loading: 100%| 1.69G/1.69G
loading: 100%| 1.83G/1.83G
loading: 100%| 1.54G/1.54G
loading: 100%| 28.4k/28.4k
loading: 100%| 51.8k/51.8k
loading: 100%| 6.96k/6.96k
loading: 100%| 15.3k/15.3k
loading: 100%| 2.50M/2.50M
loading: 100%| 3.13k/3.13k
@autodl-container-48b1458428-65e58bb8 ~# cd autodl-tmp/
```

码准备

`/root/autodl-tmp` 路径下新建 `api.py` 文件并在其中输入以下内容，粘贴代码后记得保存文件。下面的代码有很详细的注释，大家如有不理解的地方，欢

```
1 from fastapi import FastAPI, Request
2 from transformers import AutoTokenizer, AutoModelForCausalLM
3 import uvicorn
4 import json
5 import datetime
6 import torch
7
8 # 设置设备参数
9 DEVICE = "cuda" # 使用CUDA
10 DEVICE_ID = "0" # CUDA设备ID, 如果未设置则为空
11 CUDA_DEVICE = f"{DEVICE}:{DEVICE_ID}" if DEVICE_ID else DEVICE # 组合CUDA设备信息
12
13 # 清理GPU内存函数
14 def torch_gc():
15     if torch.cuda.is_available(): # 检查是否可用CUDA
16         with torch.cuda.device(CUDA_DEVICE): # 指定CUDA设备
17             torch.cuda.empty_cache() # 清空CUDA缓存
18             torch.cuda.ipc_collect() # 收集CUDA内存碎片
19
20 # 创建FastAPI应用
21 app = FastAPI()
22
23 # 处理POST请求的端点
24 @app.post("/")
25 async def create_item(request: Request):
26     global model, tokenizer # 声明全局变量以便在函数内部使用模型和分词器
27     json_post_raw = await request.json() # 获取POST请求的JSON数据
28     json_post = json.dumps(json_post_raw) # 将JSON数据转换为字符串
29     json_post_list = json.loads(json_post) # 将字符串转换为Python对象
30     prompt = json_post_list.get('prompt') # 获取请求中的提示
31     history = json_post_list.get('history')
32     max_length = json_post_list.get('max_length')
33
```



FL1623863129

关注

12

私信

关注



博主文章

🔍

文章

- 0-11全版下载地址MSDN纯净版ISO-20217更新 🔒 124245
- 常见60种野生中草药 🔒 70434
- manager打不开闪退问题完美解决2017新方法 🔒 65884
- 商品类目查询方法怎样查看别人商品的淘宝类目查询工具软件 🔒 40625
- 原创VSCode C++怎么让运行的时候cmd窗口，而不是在VSCode调试输出 30896

专栏

- 环境配置 254篇
- 数据集 445篇
- 深度学习 216篇
- 软件工具 43篇
- Python 201篇
- Pytorch 15篇
- ▼

评论

- 数据集[目标检测]猪数据集VOC-2856张
云端: 优质好文，博主的文章细节很到文章思路清晰，图文并茂，排版整...
- yolov8的辣椒缺陷检测系统python源...
奇才李先生: 基于yolov8的辣椒缺陷检...
统python源码 onnx模型 评估指标曲 ...
- 数据集[目标检测]海上红外目标检测检测...
云端: 优秀，干货就是干货，字字精辟...
, 已收藏，博主的文章总是如一盏明 ...
- 数据集[目标检测]Udacity交通目标检测...
云端: 阅读这篇博文真是一种享受！作...
文字流畅自然，吸引了我的目光。👍 ...
- yolov5的中国交通标志TT100K检测...
黯然.: 😊 The explanation of 基于yolo...
中国交通标志TT100K检测系统pyth ...

在看

- 空间安全基础（三） 🔒 708
- ensp的telnet登录防火墙具体配置
- 【VA开源】基于Vue和SpringBoot古典舞交流平台

FL1623863129

关注

👍 12

💬

🔥

code刷题]面试经典150题之9python哈
详解（知识点+题合集） 673
者怎样在游戏业生存？

文章

tensorflow报错InternalError: libdevice
ound at ./libdevice.10.bc解决方法
集[目标检测]猪数据集VOC-2856张
[学习]基于YOLO高质量项目源码+模型
ll界面汇总

4		
3月	08月	07月
5篇	114篇	55篇
3月	05月	04月
4篇	124篇	136篇
3月	02月	01月
3篇	51篇	98篇
3年	464篇	2022年 180篇
1年	180篇	2020年 133篇

准备

下载

准备

部署



FL1623863129

关注

12