

昇腾AI创新大赛2024昇思赛题比赛指导文档

1 华为云申请代金券指南

在华为云平台训练需要使用代金券，领取方式见下文。注意代金券数量有限，先到先得，代金券金额有限，请节约使用，并及时关注余额（余额更新有延迟，发现低于100元就要及时申请代金券），避免欠费。操作方式如下。

1.1 代金券申请

首先登陆华为云，链接：<https://auth.huaweicloud.com/authui/login.html?locale=zh-cn&service=https%3A%2F%2Fwww.huaweicloud.com%2F#/login>，如果已经有华为云账号可直接登陆，如果没有需要先注册账号，然后实名认证。注册完华为云账号之后，需要进行全局配置，操作如下图：



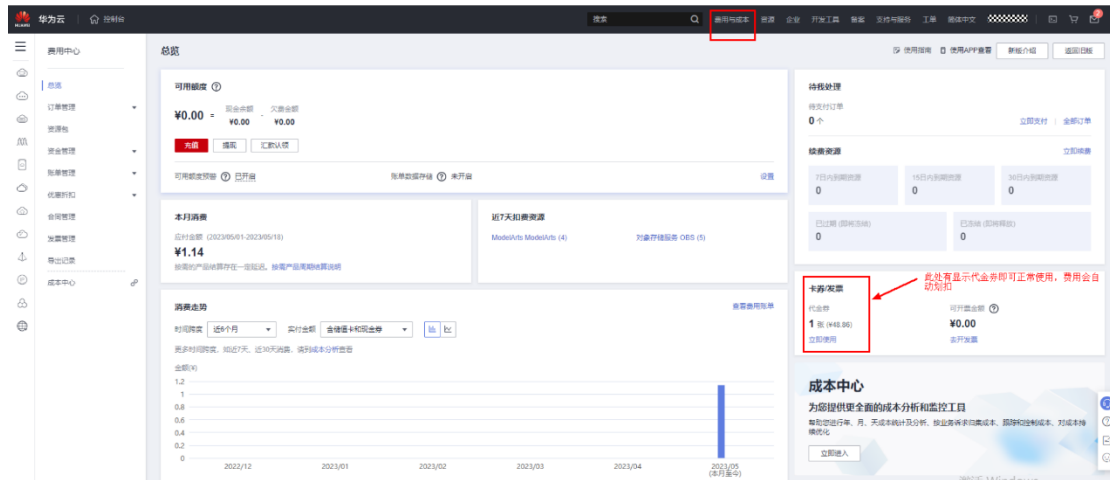
配置完成以后不要做其他操作（额外操作可能会收取费用导致账号欠费，需手动充值），去领取华为云代金券，注意代金券金额有限请谨慎使用。代金券领取链接详见比赛的各赛题官网页面，进入链接以后按照要求填写相关信息，提交申请。

1.2 代金券发放

审核标准：（1）选手需报名参加对应赛题；（2）申请选手需为队伍队长；

代金券到账时会进行短信提醒，同时可通过此链接查看代金券是否到账：<https://account.t.huaweicloud.com/usercenter/?agencyId=0e15c42d26c14cef994ead1af42648f9®ion=cn-north-4&locale=zh-cn#/userindex/allview>

打开界面如下图所示：



【特别提醒】

请参赛团队及时关注代金券额度，如发现额度较少，请先停止训练、删除服务。

- 1、由于比赛会用到昇腾算力、OBS存储等，会产生少量费用，因此在进行比赛操作前务必领取代金券，按照操作手册操作，以免账号欠费。代金券仅能在激活的账户上使用，参赛队员可与各自团队队长详细沟通代金券激活账户信息。
- 2、领取代金券资源后，请仔细了解代金券涵盖的资源类型，对于不包含的资源类型，或超出资源规格将会产生费用；
- 3、代金券到期后，如需继续使用相关服务，将产生相应费用。请在比赛结束后，及时删除不需要的项目，防止因资源到期产生不必要的扣费。释放资源请点击链接了解详情：
https://support.huaweicloud.com/usermanual-billing/renewals_topic_70000001.html
- 4、训练完成后，注意观察ModelArts首页是否还有计费中服务，并及时进行关闭；
- 5、您创建大赛所需资源时会优先扣除已领取的按需代金券，超出部分以按需付费的方式进行结算。如果您使用了其他类型规格的资源或其他云服务，将会产生费用。

2 华为云环境使用说明

2.1 注册镜像

赛题二模型微调 and 赛题三推理调优需选择指定镜像来进行开发。 镜像需要注册后使用，操

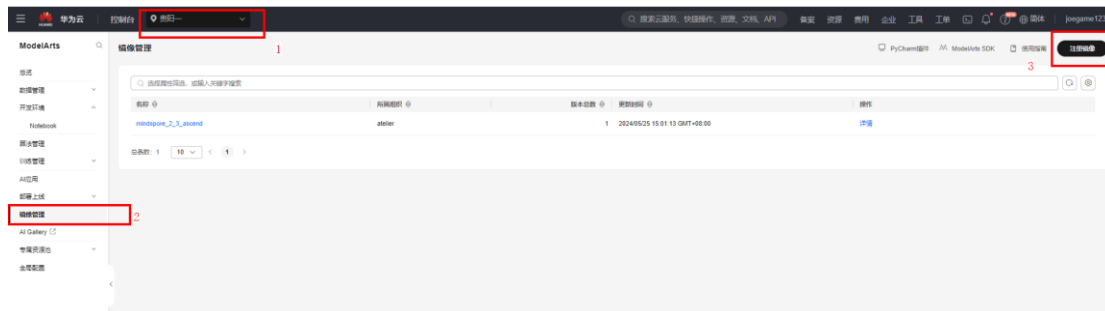
作流程如下：

2.1.1 进入 ModelArts 控制台

控制台链接：<https://console.huaweicloud.com/modelarts/?agencyId=0e15c42d26c14cef994ead1af42648f9®ion=cn-southwest-2&locale=zh-cn#/dev-container>，进入链接之后就会出现登录界面，如下图所示：

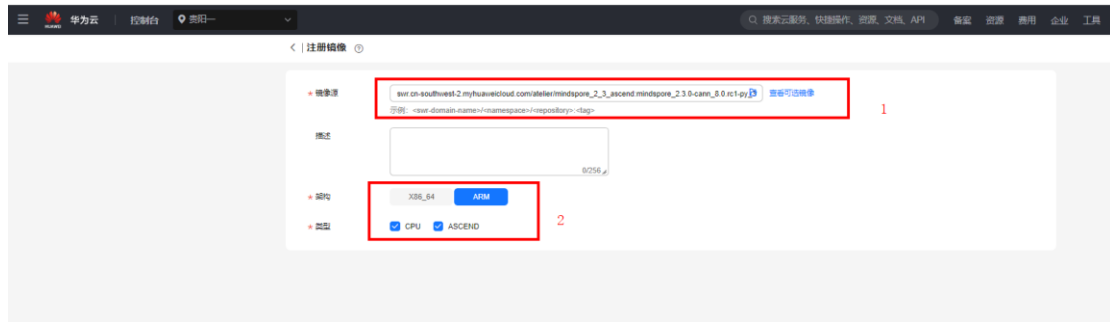


按照提示登录账号，进入ModelArts控制台如下图所示：



2.1.2 注册镜像

在上图1处，必须选择“贵阳一”节点，然后依次点击图中2“镜像管理”，图中3“注册镜像”，之后就会出现如下图所示界面：



上图1处需要填入镜像的SWR地址：`swr.cn-southwest-2.myhuaweicloud.com/atelier/mindspore_2_3_ascend:mindspore_2.3.0-cann_8.0.rc1-py_3.9-euler_2.10.7-aarch64-snt9b-20240525100222-259922e`；图中2处按截图“架构”和“类型”分别选择“ARM”和“CPU ASCEND”，然后点击界面右下角“立即注册”即可。

注意：

赛题二模型微调和赛题三模型推理都会用到这个镜像，同时需要额外安装指定的依赖（如MindSpore、MindFormers等），详细操作请见对应赛题的指导；

镜像注册过之后就无需注册了，否则会出现如下图所示的错误：

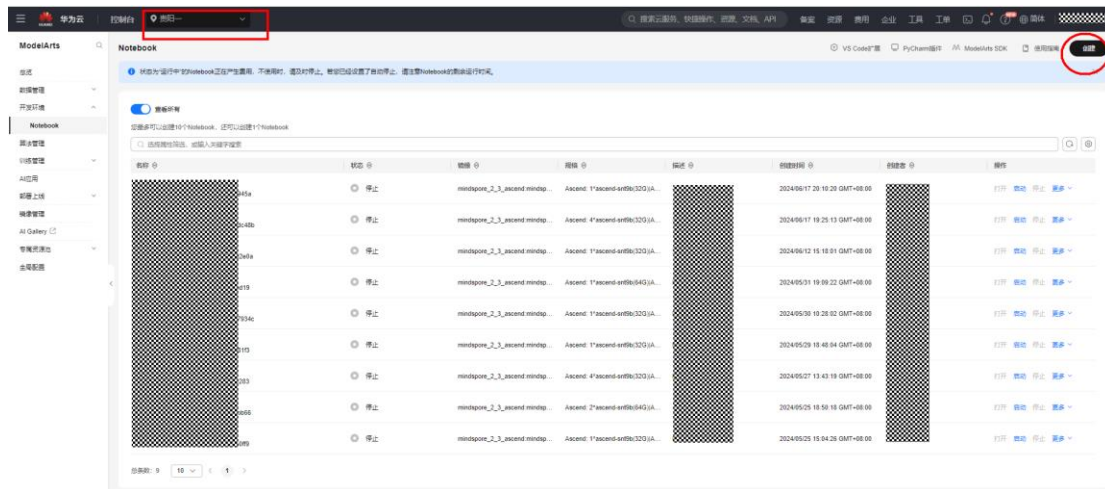


2.2 Notebook 环境

赛题一，二，三都可在华为云ModelArts的开发环境Notebook里面完成，进入该环境的操作如下所示。

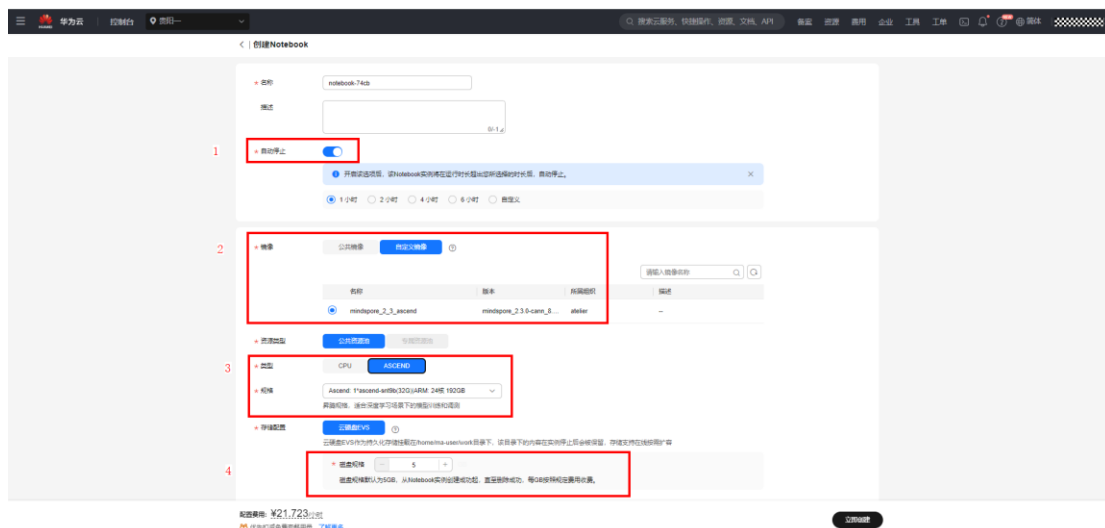
2.2.1 进入 ModelArts 控制台

按照1.2.1中“1. 进入ModelArts控制台”进入控制台，检查站点是否选择为“西南. 贵阳一”，然后选择“开发环境-Notebook”进入如下Notebook界面：



2.2.2 创建 Notebook 环境

点击上图右上角“创建”可新创建Notebook环境，会出现如下截图：



说明：

图中的1处：为了节省华为云代金券的使用，这里强烈建议打开“自动停止”。这个停止的时间在进入Notebook环境后也可自行设置，下文出现对应界面会进行说明；

图中的2处：镜像这里选择“自定义镜像”，就会看到1.2.1注册的自定义镜像；

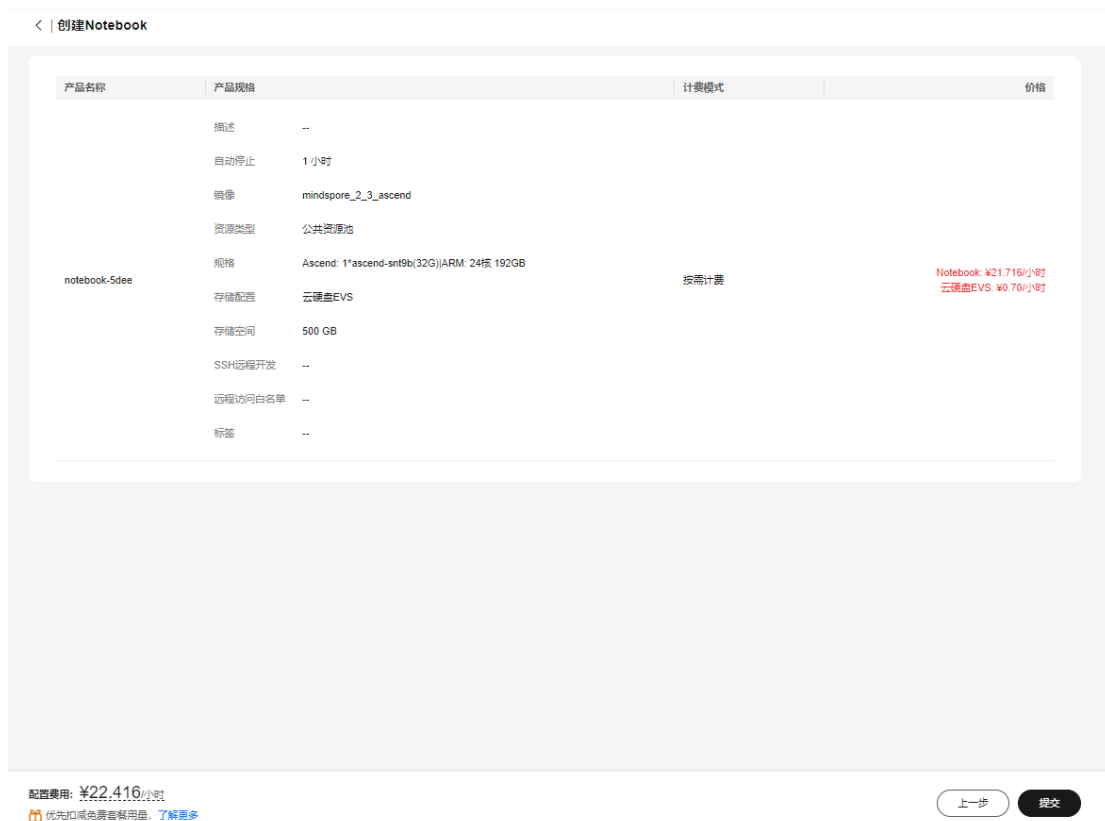
图中的4处：“磁盘规格”赛题二建议选择500G，赛题三可选择300G；

图中的3处：“类型”选择“Ascend”，“规格”点开可看到有8种选择，如下截图所示：

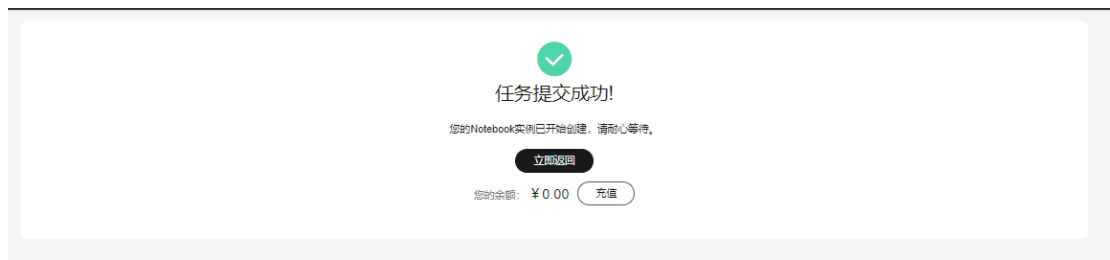


不同赛题的不同任务可有不同的选择，请选手选择32G显存的单卡或多卡资源，赛题二最低配置为“Ascend:4*ascend-snt9b(32G)ARM:96核768GB”，赛题三最低配置为第一个“Ascend:1*ascend-snt9b(32G)ARM:24核 192GB”，其他任务的最低配置会在后续描述中给出。不同规格对应的价格也有不同，选手可根据代金券使用情况酌情选择。

配置完成后点击“立即创建”，就会进入如下界面：

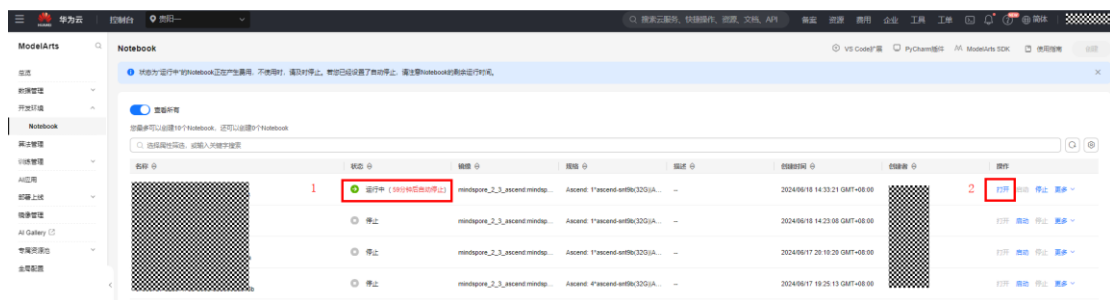


然后点击右下角的“提交”，出现下图：

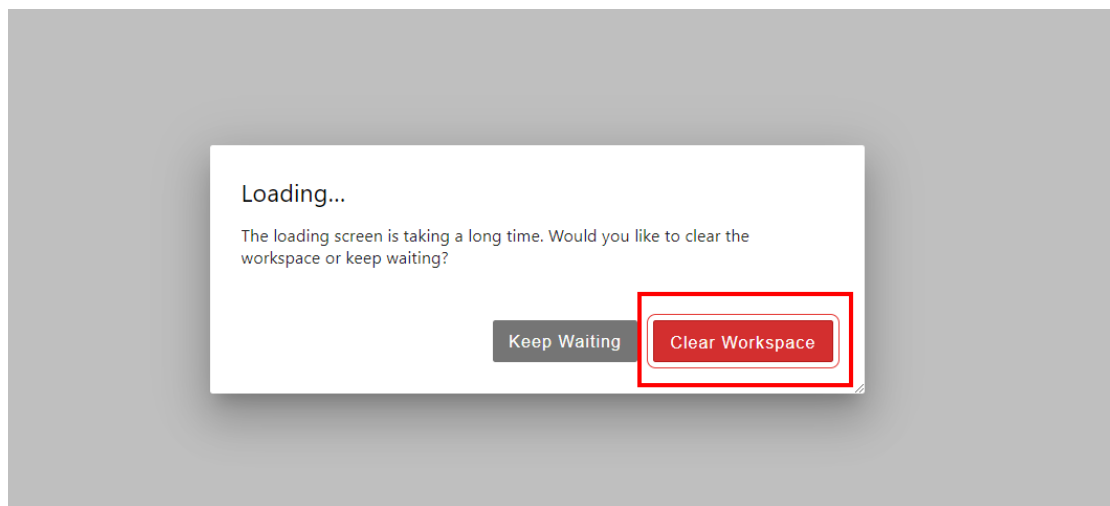


2.2.3 进入 Notebook 环境

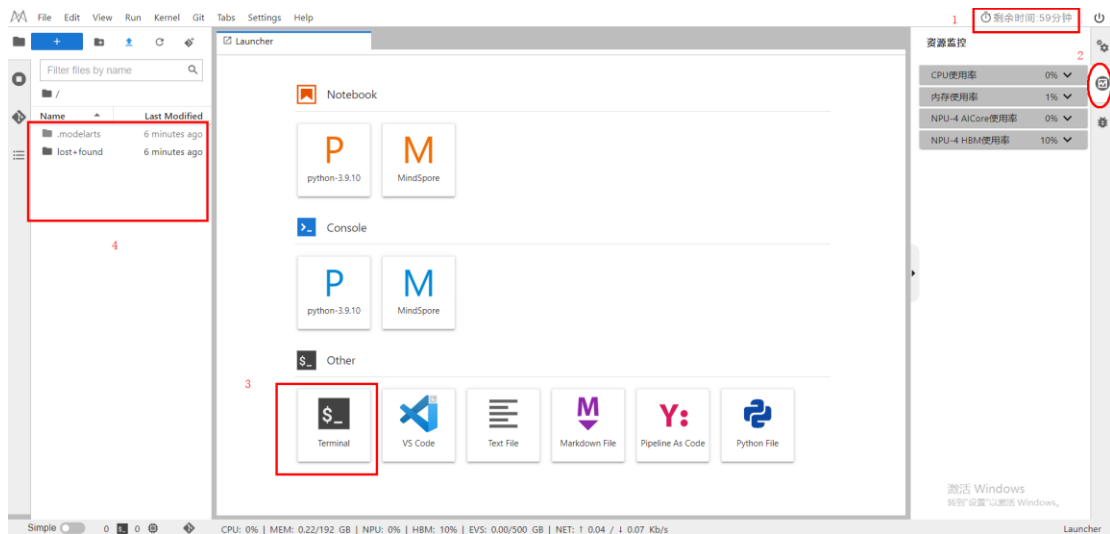
点击“立即返回”之后就会进入下面界面：



等待2分钟左右时间就会出现上图1处的“运行中”，然后点击图中2出的打开，等待1分钟左右时间，如果出现如下界面：



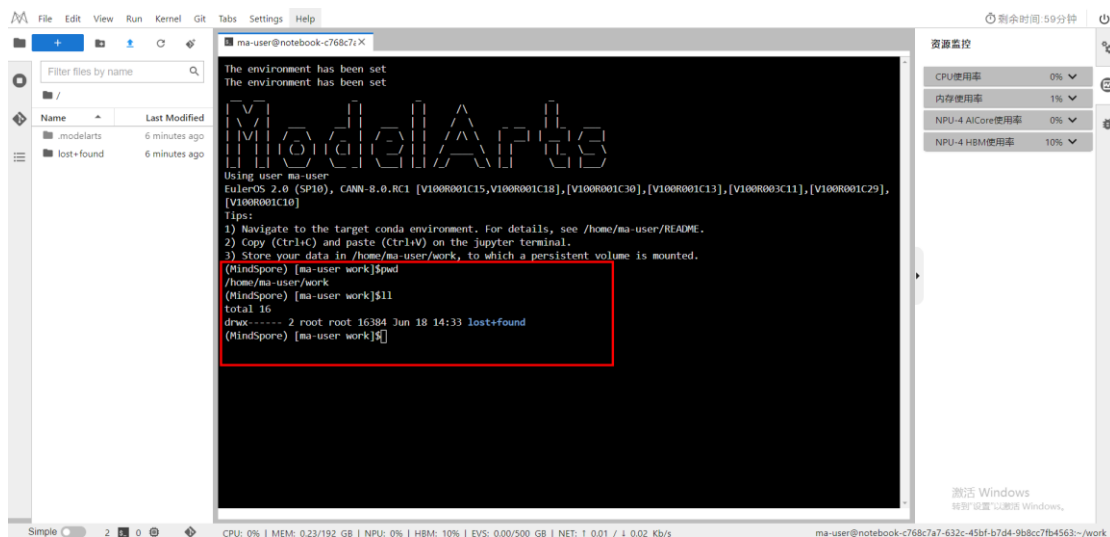
点击“Clear Workspace”，就会进入如下界面：



点击上图1处“剩余时间: XX”就可以手动修改自动停止Notebook的时间，在运行过程中可随时修改；

点击上图中2处可查看CPU和NPU的内存使用情况；

点击上图中3处可进入终端，如下图所示：



进入终端模型的虚拟环境是“MindSpore”，此为默认虚拟环境，必须使用这个。默认的目录位置是/home/ma-user/work，与截图左侧文件栏（上上张图中的4处）所在的目录位置一致。然后就可以在终端完成下面的赛题了。

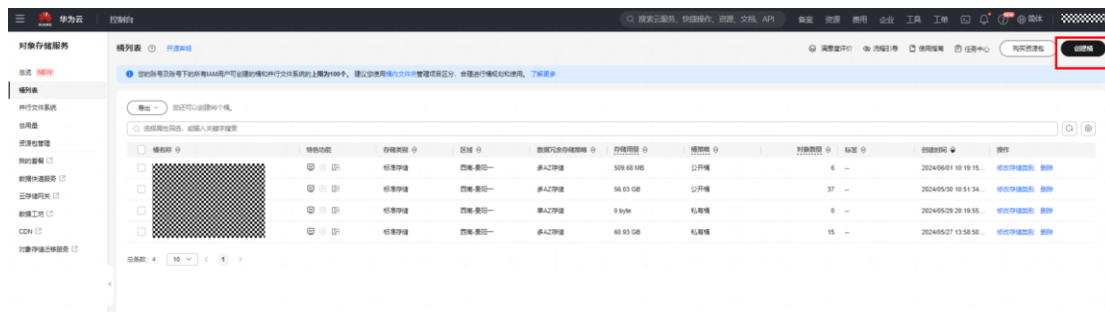
此外，华为云官方也提供了开发环境介绍，可参考链接：https://support.huaweicloud.com/devtool-modelarts/devtool-modelarts_0001.html；具体Notebook的使用可参考链接：https://support.huaweicloud.com/devtool-modelarts/devtool-modelarts_0004.html；有兴趣的开发者可以去浏览学习。

3 obs 数据传输指南

赛题二模型微调及赛题三推理调优的依赖包，数据集等将存储在华为云的obs桶里面，获取链接（URL）在比赛官网对应赛题的赛事详情页面，以及本指导书的各个赛题详细指导中展示，大家可以在Notebook终端用wget+URL命令进行文件下载。

此外，赛题二模型微调及赛题三推理调优在作品提交环节，会涉及较大文件的提交（如代码文件，保存的模型输出等），同样可以通过将文件上传obs桶，然后在作品提交报告中提供obs下载链接（URL）的方式完成提交，上传及获取URL的指南如下所示。

华为云OBS桶链接：<https://console.huaweicloud.com/console/?agencyId=0e15c42d26c14cef994ead1af42648f9®ion=cn-southwest-2&locale=zh-cn#/obs/manager/buckets>，点击链接之后跟Notebook环境一样登录账号，然后进入到如下界面：



点击右上角的“创建桶”，会出现如下画面：

< 创建桶

复制桶配置 选择策略

该桶可选。选择后可复制源桶的以下配置信息：区域 / 数据冗余策略 / 存储类别 / 桶策略 / 服务端加密 / 归档数据直读 / 企业项目 / 标签。

区域 西南-贵阳一

不同区域的资源之间内网互不相通，请选择靠近您业务的区域，可以降低网络时延，提高访问速度。桶创建成功后不支持变更区域，请谨慎选择。 [如何选择区域](#) ①

桶名称 [查看命名规则](#) ①

① 不能和本用户已有桶重名 ① 不能和其他用户已有的桶重名 ① 创建成功后不支持修改

数据冗余策略 多AZ存储 单AZ存储 ①

数据在同区域的多个AZ中存储，可用性更高。

⚠ 启用后不支持修改。多AZ存储采用相对较高计费标准。 [价格详情](#)

默认存储类别

标准存储 低频访问存储 归档存储

适合高性能，高可靠，高可用，频繁访问场景 适合高可靠，低成本，较少访问场景 适合长期存储，平均一年访问一次

创建桶时选择的存储类别会作为上传对象的默认存储类别。 [了解存储类别差异](#) ①

桶策略 私有 公共读 公共读写 复制桶策略 ①

任何用户都可以对桶内对象进行读操作，仅桶所有者可以进行写操作。

归档数据直读 开启 关闭 ①

关闭归档直读，归档存储类别的数据要先恢复才能访问，归档存储数据恢复和访问会收取相应的费用。 [价格详情](#)

服务端加密 SSE-KMS SSE-OBS 不加密 ①

开启服务端加密后，上传到当前桶的对象会被加密，您也可以在桶创建完成之后在桶概览页面调整服务端加密配置。

⚠ 建议开启加密，核心数据更安全，如果您使用KMS加密模式，超过免费配额会收取相应费用。 [价格详情](#)

创建阶段 使用阶段

OBS桶：创建免费 按需/资源包计费 OBS计费说明 立即创建

注意：

上图的区域需要选择“西南-贵阳一”，就是跟创建notebook的区域选择一样的；桶策略需要选择“公共读”，不然里面的数据别人下载不了，桶的大小不用设置，桶是自动扩容的。

obs桶存储详细操作，可参考如下说明：

在Notebook中上传下载OBS文操作件参考链接：https://support.huaweicloud.com/modelarts_faq/modelarts_05_0024.html

一些常见的问题处理方法参考链接：

https://support.huaweicloud.com/modelarts_faq/modelarts_05_0067.html

也可使用obsutil工具将本地的文件上传到obs桶，参考链接：

https://support.huaweicloud.com/utiltg-obs/obs_11_0001.html

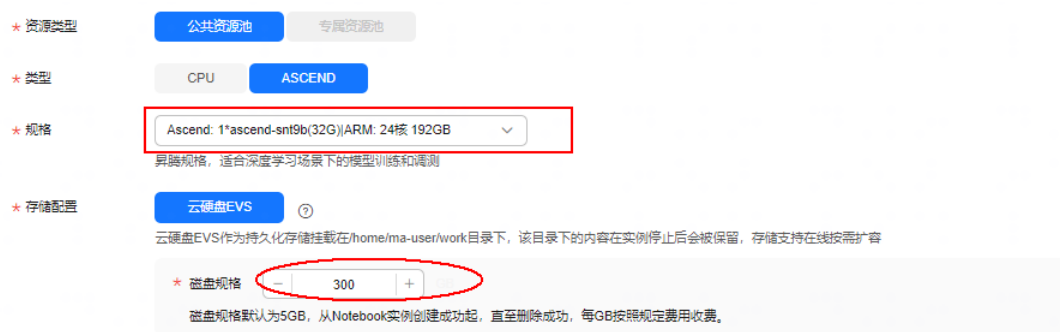
4 赛题三：推理调优指导

4.1 赛题介绍

本赛题基础流程共分为以下5个环节：环境准备、模型权重准备、启动llm-serving、启动推理及推理时长获取、logits文件保存，下方会针对每个环节进行完整说明。

4.2 环境准备

本赛题指定使用华为云modelarts-开发环境-Notebook，使用32G显存的NPU，硬盘规格推荐使用300G，如下图所示设置：



The screenshot shows the configuration interface for a Huawei Cloud ModelArts Notebook. The 'Resource Type' is set to 'Public Resource Pool'. The 'Type' is set to 'ASCEND'. The 'Specification' is set to 'Ascend: 1*ascend-sn19b(32G)/ARM: 24核 192GB'. The 'Storage Configuration' is set to 'Cloud Disk EVS'. The 'Disk Specification' is set to '300'.

在默认基础环境下，即指定的华为云自定义镜像下，需按照要求额外安装指定的MindSpore和MindFormers依赖。注意，以下的命令强烈建议在终端运行。

4.2.1 模块卸载

在安装之前需要手动卸载两个镜像自带的两个模块，卸载命令如下：

```
pip uninstall mindformers mindspore-lite
```

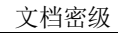
4.2.2 MindSpore 安装

MindSpore可用如下命令安装：

```
pip install mindspore==2.3.0rc2
```

如果上面安装命令出现问题，可通过如下命令安装：

```
wget https://2024-ascend-innovation-contest-mindspore.obs.cn-southwest-2.myhuaweicloud.com/topic2-finetune/mindspore-2.3.0rc2-cp39-cp39-linux\_aarch64.whl
```



第12页, 共22页

还有另外其他的依赖需要安装，安装命令如下：

```
cd llm-serving/  
pip install -r requirement.txt  
pip install tiktoken
```

注意：每次Notebook重新启动之后都需要重新卸载自带的mindformers和mindspore-lite包、安装MindSpore、设置环境变量一遍，依赖也需要重新安装一遍，之前下载过的文件会保留的。

4.3 模型权重准备

要运行起来需要先将权重文件和tokenizer文件下载到指定文件夹内，具体操作如下。

在与mindformers同级目录下（这里是 /home/ma-user/work/）创建目录，在终端输入命令如下：

```
cd /home/ma-user/work/  
mkdir -p checkpoint_download/llama2/
```

下载llama2-7b基础权重文件到该目录下，命令如下：

```
wget https://2024-ascend-innovation-contest-mindspore.obs.cn-southwest-2.myhuaweicloud.com/topic3-infer/llama2_7b.ckpt -P checkpoint_download/llama2/
```

下载llama2-7b的tokenizer文件到该目录下，命令如下：

```
wget https://2024-ascend-innovation-contest-mindspore.obs.cn-southwest-2.myhuaweicloud.com/topic3-infer/tokenizer.model -P checkpoint_download/llama2/
```

4.4 启动 llm-serving

llm-serving的使用方法可参考链接：<https://gitee.com/mindspore/llm-serving>，也可参考serving仓库，链接为：<https://gitee.com/mindspore/serving>，还有MindSpore官网的介绍教程，链接：<https://www.mindspore.cn/serving/docs/zh-CN/master/index.html>。具体使用指导如下步骤。

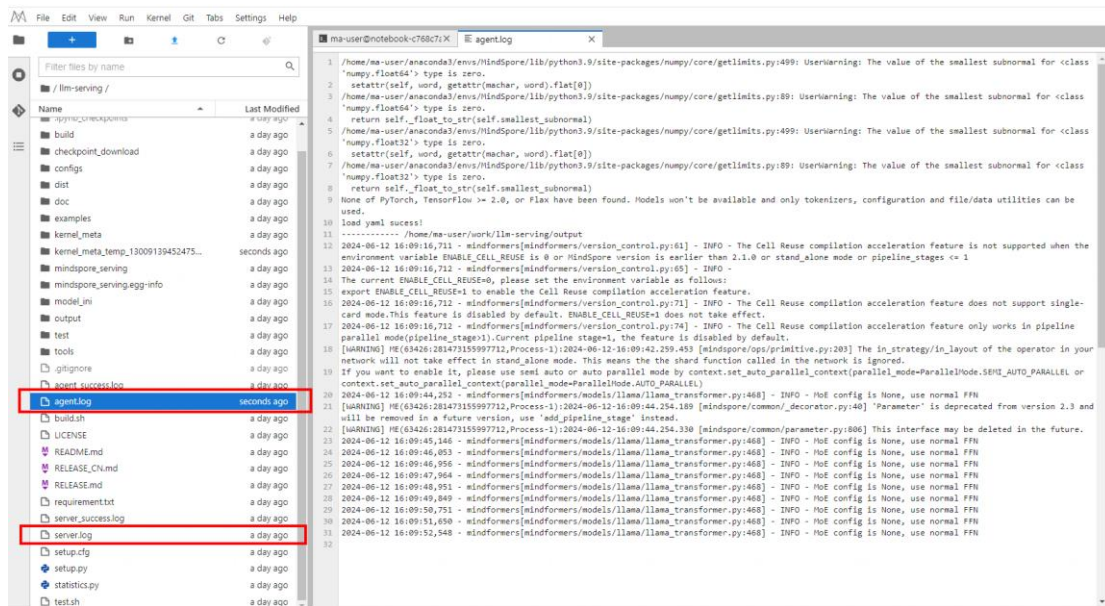
使用 start.py启动推理服务，命令如下：

```
cd /home/ma-user/work/llm-serving/  
python examples/start.py --config /home/ma-user/work/llm-serving/configs/llama/
```

llama_7b_kbk_pa_dyn.yaml

此处配置文件可使用包中自带配置文件，如需修改请谨慎，以上命令中的路径以你本地实际路径为准。

运行成功serving服务拉起一般需要5分钟左右，请耐心等待。如果时间过长可查看运行中的日志情况，运行过程的日志文件保存可在 /home/ma-user/work/llm-serving/ 目录下的 agent.log 和 server.log 文件里，具体如下截图：



运行成功之后终端显示如下图所示：

```
(MindSpore) [ma-user llm-serving]$python examples/start.py --config /home/ma-user/work/demo/llm-serving/configs/llama/llama_7b_kbk_pa_dyn.yaml
----starting agents----
/home/ma-user/anaconda3/envs/MindSpore/lib/python3.9/subprocess.py:941: RuntimeWarning: line buffering (buffering=1) isn't supported in binary mode, the default buffer size will be used
  self.stdout = io.open(c2pread, 'rb', bufsize)
----agents are ready----
----starting server----
----server is ready----
```

另外说明：后续如果有其他操作需要关闭服务可见1.6.6说明。

4.5 启动推理及推理时长获取

此处提供两种推理方式。

第一种是快速推理，主要用于测试能否正常推理，实际推理时间检测主要通过第二种方式。在serving服务启动成功的情况下，在终端运行如下代码可启动快速单条推理：

```
curl 127.0.0.1:8835/models/llama2/generate \
-X POST \
```

```
-d '{"inputs": " I love Beijing, because", "parameters": {"max_new_tokens": 56, "do_sample": "True", "return_full_text": "True"}, "stream": "True"}' \
-H 'Content-Type: application/json'
```

注意：此处的127.0.0.1:8835，中的8835要跟配置文件“/home/ma-user/work/demo/llm-serving/configs/llama/llama_7b_kbk_pa_dyn.yaml”中的 serving_config:下的 server_port:8835 一样；包中自带 llama_7b_kbk_pa_dyn.yaml 配置文件可直接运行。

成功之后如下图所示：

```
(MindSpore) [ma-user llm-serving]$
(MindSpore) [ma-user llm-serving]$ curl 127.0.0.1:8835/models/llama2/generate \
> -X POST \
> -d '{"inputs": " I love Beijing, because", "parameters": {"max_new_tokens": 16, "do_sample": "True", "return_full_text": "True"}, "stream": "True"}' \
> -H 'Content-Type: application/json'
{"generated_text": "it is the most beautiful city in the world. It is a city with", "finish_reason": "length", "generated_tokens": 16, "prefill": [{"id": 372, "logprob": 17.203125, "special": false, "text": "it"}], "seed": 0, "tokens": [{"id": 372, "logprob": 17.203125, "special": false, "text": "it"}, {"id": 338, "logprob": 16.25, "special": false, "text": " is"}, {"id": 278, "logprob": 13.0390625, "special": false, "text": " the"}, {"id": 1556, "logprob": 14.9296875, "special": false, "text": " most"}, {"id": 9560, "logprob": 14.890625, "special": false, "text": " beautiful"}, {"id": 4272, "logprob": 17.921875, "special": false, "text": " city"}, {"id": 297, "logprob": 21.96875, "special": false, "text": " in"}, {"id": 278, "logprob": 22.90625, "special": false, "text": " the"}, {"id": 3186, "logprob": 22.09375, "special": false, "text": " world"}, {"id": 29889, "logprob": 22.171875, "special": false, "text": "."}, {"id": 739, "logprob": 13.921875, "special": false, "text": " It"}, {"id": 338, "logprob": 17.515625, "special": false, "text": " is"}, {"id": 263, "logprob": 13.1953125, "special": false, "text": " a"}, {"id": 4272, "logprob": 15.8046875, "special": false, "text": " city"}, {"id": 411, "logprob": 13.828125, "special": false, "text": " with"}, {"id": 1784, "logprob": 12.328125, "special": true, "text": ""}], "top_tokens": [{"id": 372, "logprob": 17.203125, "special": false, "text": "it"}], "details": null}
(MindSpore) [ma-user llm-serving]$
```

第二种批量推理服务，这种方式也是主要用来测试推理时长的。

4.5.1 脚本获取

测试脚本下载解压命令如下：

```
cd /home/ma-user/work/
```

```
wget https://2024-ascend-innovation-contest-mindspore.obs.cn-southwest-2.myhuaweicloud.com/topic3-infer/performance\_serving.zip
```

```
unzip performance_serving.zip
```

在目录 llm-serving/mindspore_serving/agent/ 下有两个文件，一个命名为：agent_multi_post_method.py，一个命名为：agent_multi_post_method_save_logits.py，推理运行会默认使用命名为“agent_multi_post_method.py”的文件，此文件也是用来收集推理时长的。

4.5.2 推理数据集说明

为了比赛的公平公正，选手必须使用指定测试推理时长的数据集，此数据集为 performanc

e_serving/ 目录下的 alpaca_5010.json，此数据集是随 performance_serving.zip 包下载的，数据集路径的修改在 /home/ma-user/work/performance_serving 目录下的 test_serving_performance.py 文件的第211行，如下图所示：



```
204 if __name__ == '__main__':
205     parser = argparse.ArgumentParser(description="test serving performance")
206     parser.add_argument("-X", "--qps", help='x req/s', required=True, type=float)
207     parser.add_argument("-P", "--port", help='port, default is 8000', required=True)
208     parser.add_argument("-O", "--out_dir", help='dir for saving results', required=True)
209     parser.add_argument("-T", "--test_time", help='test all time, default 1h', required=False, type=int, default=3600)
210     args = parser.parse_args()
211     with open('./alpaca_5010.json') as f:
212         alpaca_data = json.loads(f.read())
213     INPUTS_DATA = []
214     OUTPUTS_DATA = []
215     for data in alpaca_data:
216         input_ = data["instruction"] + ":" + data["input"] if data["input"] else data["instruction"]
217         INPUTS_DATA.append(input_)
218         OUTPUTS_DATA.append(data["output"])
219     test_main(args.port, INPUTS_DATA, OUTPUTS_DATA, args.qps, args.out_dir, args.test_time)
220
```

运行之前请做好检查。

4.5.3 限定推理数据数目

为了比赛的公平公正，只需推理数据集的前1500条数据，这个设置是目录 /home/ma-user/work/performance_serving 下 test.sh 文件里面的代码：python test_serving_performance.py -X 1 -P 8835 -O "/" -T 5 中，参数说明如下：

-X 1：每秒发送1个请求；

-P 8835：此处端口号要跟配置文件 “/home/ma-user/work/demo/llm-serving/configs/llama/llama_7b_kbk_pa_dyn.yaml” 中的 serving_config:下的 server_port:8835 一样；

-T 5：表示发送请求的总时间为5s，具体代码可见 test_serving_performance.py；

上面命令的意思就是，总共发送请求的时间为 5s，每 1s 发送一个推理请求，就是要发送 5 个推理请求，就是推理5条测试数据集。

必须保证 -X 的设定值乘以 -T 的设定值等于1500，比如可设置为 -X 0.5 -T 3000；注意这两个参数的不同设置可能会造成推理时长的变化，也可能导致模型没法成功推理出1500条数据，具体情况可见 performance_serving/testLog/ 目录下日志。

此处给出基准推理时间：3551.9252s，此时间也是推理的基准时间，超过这个时间才算有

效作品，另外说明这个基准时间是在 $-X$ 和 $-T$ 设置的值为 0.5 和 3000 情况下跑出来的。

4.5.4 启动推理

推理启动可运行如下脚本：

```
cd /home/ma-user/work/performance_serving
```

```
nohup sh test.sh > test_sh.log 2>&1 &
```

注意：`> test_sh.log 2>&1 &`是用于日志重定向出来，便于保存推理的日志；

另外说明：

用于测试模型基础精度和推理的数据集已经内置在performance_serving文件中，请勿修改，如有修改可能导致模型基础精度测试不通过，后果选手自负。

推理运行完成以后，推理总时长是记录在 performance_serving/testLog/ 目录下日志文件的最后一行。

4.6 logits 文件保存

除了获取推理总时长之外，选手还需要提供调优以后模型推理生成的logits文件，目的是验证模型的精度，要求偏差在千分之五以内（即完成推理优化后的logits输出和优化前的标准logits输出绝对差值在千分之五以内），确保推理调优对模型推理的精度影响不会太大。具体操作流程如下：

4.6.1 修改配置文件

将目录 llm-serving/mindspore_serving/agent/ 下的 agent_multi_post_method.py 文件更改为其他名字做好备份，然后将 agent_multi_post_method_save_logits.py 文件改名为 agent_multi_post_method.py

4.6.2 关闭 llm-serving 服务

修改配置文件后，需要关闭后重启serving服务，保存npz文件的脚本才会生效。

关闭服务的具体操作截图如下：

```
(MindSpore) [ma-user performance_serving]$  
(MindSpore) [ma-user performance_serving]$ps -elf |grep python  
4 S ma-user 232 1 0 80 0 - 75393 ep_pol 09:31 ? 00:00:04 /modelarts/authoring/notebook-conda/bin/python /modelar  
4 S ma-user 249 247 2 80 0 - 7236327 ep_pol 09:31 ? 00:02:20 /modelarts/authoring/notebook-conda/bin/python /modelar  
4 S ma-user 33965 1 0 85 5 - 2954443 futex_ 09:58 pts/0 00:00:19 python examples/start_agent.py --config /home/ma-user/w  
5 S ma-user 34446 33965 27 85 5 - 2157416650 wait_w 09:58 pts/0 00:19:34 python examples/start_agent.py --config /home/ma-user  
1 S ma-user 34522 34446 0 85 5 - 2817842 futex_ 09:58 pts/0 00:00:13 python examples/start_agent.py --config /home/ma-user/w  
4 S ma-user 34638 34446 0 85 5 - 55966 pipe_w 09:58 pts/0 00:00:00 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
4 S ma-user 34648 34446 0 85 5 - 56338 ep_pol 09:58 pts/0 00:00:00 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 34657 34648 0 85 5 - 181700 unix_s 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 34658 34648 0 85 5 - 181871 unix_s 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 34659 34648 0 85 5 - 181871 unix_s 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 34660 34648 0 85 5 - 181871 unix_s 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 34661 34648 0 85 5 - 181433 unix_s 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 34662 34648 0 85 5 - 181869 unix_s 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 34663 34648 0 85 5 - 181438 unix_s 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 34664 34648 0 85 5 - 181438 unix_s 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 34693 34648 0 85 5 - 181682 do_sel 09:58 pts/0 00:00:09 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 35112 34648 0 85 5 - 182030 futex_ 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 35113 34648 0 85 5 - 182030 futex_ 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 35114 34648 0 85 5 - 181886 futex_ 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 35115 34648 0 85 5 - 181886 do_sys 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 35116 34648 0 85 5 - 181886 futex_ 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 35117 34648 0 85 5 - 181886 futex_ 09:58 pts/0 00:00:09 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 35118 34648 0 85 5 - 181886 futex_ 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
5 S ma-user 35119 34648 0 85 5 - 181886 futex_ 09:58 pts/0 00:00:08 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
4 R ma-user 39221 1 78 85 5 - 5285542 - 10:02 pts/0 00:54:02 python examples/server_app_post.py --config /home/ma-us  
4 S ma-user 39697 39221 0 85 5 - 55965 pipe_w 10:02 pts/0 00:00:00 /home/ma-user/anaconda3/envs/MindSpore/bin/python -c fr  
4 R ma-user 123639 3638 0 85 5 - 53360 - 11:10 pts/0 00:00:00 grep --color=auto python  
(MindSpore) [ma-user performance_serving]$  
(MindSpore) [ma-user performance_serving]$kill -9 33965 34446 34522 39221
```

命令如下：

```
ps -elf | grep python
```

```
kill -9
```

4.6.3 重启 llm-serving

关闭完成之后需要按照1.6.4 启动llm-serving 章节重启serving服务。

4.6.4 指定数据集

保存 logits 文件需要用到的推理数据集为 performance_serving/ 目录下的 alpaca_52

1.json，数据集路径的修改在 /home/ma-user/work/performance_serving 目录下的 test_serving_performance.py 文件的第211行，具体可见上文“6.5启动推理及时长获取”中的“第二种批量推理服务”下面的“2.推理数据集说明”。

4.6.5 调整参数配置

因为在推理过程中需要保存模型输出的 logits 文件，所以相比 1.6.4 每条的推理时长会更久，为了比赛的公平公正，也为了方便验证精度，此处 -X 和 -T 的值必须设置为 0.1 和 5000，即选手推理500条数据。参数修改完成之后就可以使用1.6.5中的第4条启动推理里面命令启动推理生成logits文件。

4.6.6 配置 npy 文件保存路径

numpy文件的保存路径可见目录 llm-serving/mindspore_serving/agent/ 下的 agent_multi_post_method_save_logits.py 文件的第824行，如下图所示：

[illegible]

为了比赛的顺利进行，这份保存 logits 文件的配置代码不可修改，如发现选手擅自修改代码导致精度评测不通过，后果选手自负。

4.6.7 查看保存结果

生成的 `npz` 文件的命名以 “_” 为分隔，有三部分组成，如 `1718418290.5565388_1.020355.0915.0278.0_100.npz`，第一部分 `1718418290.5565388` 为时间用于区分生成logits的前后顺序；第二部分 `1.020355.0915.0278.0` 是为了区别不同句子，相同表示生成的logits在同一个句子里；第三部分 `100` 表示每句话里面不同的token，具体情况可见`agent_multi_post_method_save_logits.py` 文件。

4.6.8 精度测试

赛事组这边为选手提供一份基准npz文件和精度测试代码（获取如下）。选手可在推理调优以后生成一份新的npz文件，然后使用精度验证代码将两份npz文件进行比对，以验证精

度。该精度测试方法基本思路就是读取相对应的npz文件，然后使用numpy中的allclose方法比对每个元素的绝对精度，如果绝对精度在千分之五以内方法就会返回True，否则就是False，所有文件比对都返回True即可算是合格，具体见代码。除了修改输入npz文件的路径，精度测试代码其他部分选手请勿修改，如发现问题可向赛事组反馈。

精度测试的环境可在华为云Notebook环境，选手也可在自己本地CPU环境运行，为了节省代金券，建议选手下载代码到本地运行。

基准npz文件获取命令：

```
cd /home/ma-user/work/
```

```
wget https://2024-ascend-innovation-contest-mindspore.obs.cn-southwest-2.myhuaweicloud.com/topic3-infer/file\_npz\_base.zip
```

```
unzip file_npz_base.zip
```

精度测试文件获取命令：

```
cd /home/ma-user/work/
```

```
wget https://2024-ascend-innovation-contest-mindspore.obs.cn-southwest-2.myhuaweicloud.com/topic3-infer/acc\_allclose.py
```

精度测试运行命令（如果命令中涉及到绝对路径，仅供参考，请确认自己实际路径是否正确）：

```
cd /home/ma-user/work/
```

```
python acc_allclose.py \
```

```
--base_path /home/ma-user/work/file_npz_base \
```

```
--new_path /home/ma-user/work/file_npz_new
```

4.7 作品提交

所有作答文件汇总后打包成zip压缩包，以团队名称命名压缩包（如：团队名称.zip）参赛者可多次提交，最新一次提交将覆盖上一次提交作品，以最后提交的作品为准。

作答文件需包含以下内容：

提供作品报告(word、pdf、markdown等格式)，模板如下：

业界推理优化算法调研

本作品使用的推理优化算法介绍



传至自己的obs桶，并在作品报告中附上获取链接）；

5 相关官方链接

MindSpore官网: <https://www.mindspore.cn/tutorials/zh-CN/r2.3.0rc2/index.html>

MindSpore代码仓: <https://gitee.com/mindspore/mindspore>

mindnlp: <https://github.com/mindspore-lab/mindnlp>

mindformers: https://gitee.com/mindspore/mindformers?from=gitee_search

mindformers使用说明文档: <https://mindformers.readthedocs.io/zh-cn/latest/>

llm serving: <https://gitee.com/mindspore/llm-serving>

serving: <https://gitee.com/mindspore/serving>

MindSpore Serving 文档: <https://www.mindspore.cn/serving/docs/zh-CN/master/index.html>