MindFormers实验手册-v1.0

1. 运行环境及模型信息

目标：以LLaMA2-7B模型为例，熟悉Mindformers大模型套件的微调和推理流程

配置：Atlas800 A2 单节点8卡

标准镜像：http://mirrors.cn-central-221.ovaijisuan.com/detail/129.html

运行环境：

名称	版本
MindFormers	r1.1
MindPet	1.0.3
MindSpore	2.3.rc2
CANN	8.0.RC1.beta1
驱动和固件	8.0.RC1.beta1
Python	3.9

2. 相关材料与准备工作

实操模型：LLaMA2-7B

模型权重和相关文件放置在指定目录下，按照以下结构形式组织文件内容：

```
mindformers
    └── checkpoint_download
        └── llama2
            ├── llama2_7b.ckpt        # 模型权重
            ├── llama2_7b.yaml        # 模型配置文件
            └── tokenizer.model       # 模型tokenizer文件
```

数据集路径：/data01/datasets/

```
datasets
    ├── alpaca_data.json
    └── belle_chat_ramdon_10k.json
```

3. 微调

> 以下所有操作均在容器内进行

3.1 数据预处理

Step1. 执行`alpaca_converter.py`，使用fastchat工具添加prompts模板，将原始数据集转换为多轮对话格式

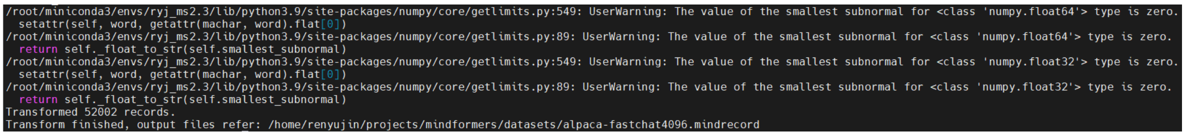
```
cd ./mindformers/tools/dataset_preprocess/llama/

python alpaca_converter.py \
--data_path /data01/datasets/alpaca_data.json \
--output_path /{path}/alpaca-data-conversation.json
```

```
# 参数说明
data_path: 存放alpaca数据的路径
output_path: 输出转换后对话格式的数据路径
```

Step2. 执行`llama_preprocess.py`，进行数据预处理、Mindrecord数据生成，将带有prompt模板的数据转换为mindrecord格式

```
python llama_preprocess.py \
--dataset_type qa \
--input_glob /{path}/alpaca-data-conversation.json \
--model_file /{path}/tokenizer.model \
--seq_length 4096 \
--output_file /{path}/alpaca-fastchat4096.mindrecord
```

转换成功日志：



```
/root/miniconda3/envs/ryj_ms2.3/lib/python3.9/site-packages/numpy/core/getlimits.py:549: UserWarning: The value of the smallest subnormal for <class 'numpy.float64'> type is zero.
  setattr(self, word, getattr(machar, word).flat[0])
/root/miniconda3/envs/ryj_ms2.3/lib/python3.9/site-packages/numpy/core/getlimits.py:89: UserWarning: The value of the smallest subnormal for <class 'numpy.float64'> type is zero.
  return self._float_to_str(self.smallest_subnormal)
/root/miniconda3/envs/ryj_ms2.3/lib/python3.9/site-packages/numpy/core/getlimits.py:549: UserWarning: The value of the smallest subnormal for <class 'numpy.float32'> type is zero.
  setattr(self, word, getattr(machar, word).flat[0])
/root/miniconda3/envs/ryj_ms2.3/lib/python3.9/site-packages/numpy/core/getlimits.py:89: UserWarning: The value of the smallest subnormal for <class 'numpy.float32'> type is zero.
  return self._float_to_str(self.smallest_subnormal)
Transformed 52002 records.
Transform finished, output files refer: /home/renyujin/projects/mindformers/datasets/alpaca-fastchat4096.mindrecord
```

3.2 全参微调

Step1. 打开`config/llama2/finetune_llama2_7b.yaml`，训练数据集路径设置为3.1中预处理好的微调数据集路径，并在`input_columns`中添加`labels`

```
train_dataset: &train_dataset
  data_loader:
    type: MindDataset
    dataset_dir: "/{path}/alpaca-fastchat4096.mindrecord"
    shuffle: True
  input_columns: ["input_ids", "labels"]
```

Step2. 修改运行相关配置

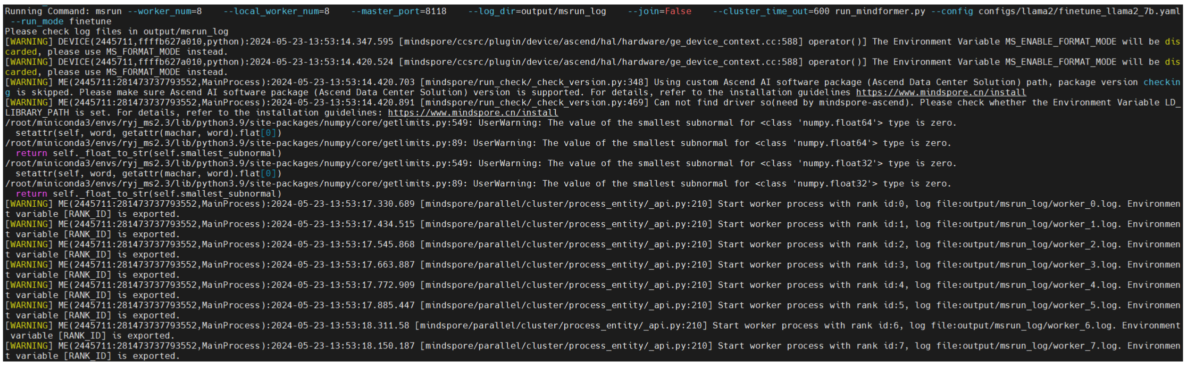
```
load_checkpoint: '/{path}/llama2_7b.ckpt'     # 使用权重的绝对路径
auto_trans_ckpt: True                         # 打开权重自动转换
use_parallel: True                            # 开启并行设置
run_mode: 'finetune'                          # 设置微调模式
```

Step3. 修改并行配置，并行策略可以进行小范围修改进行尝试

```
parallel_config:
  data_parallel: 8
  model_parallel: 1
  pipeline_stage: 1
  micro_batch_num: 1
```

Step4. 使用msrun启动分布式微调任务

```
bash scripts/msrun_launcher.sh "run_mindformer.py \
--config configs/llama2/finetune_llama2_7b.yaml \
--run_mode finetune" 8
```

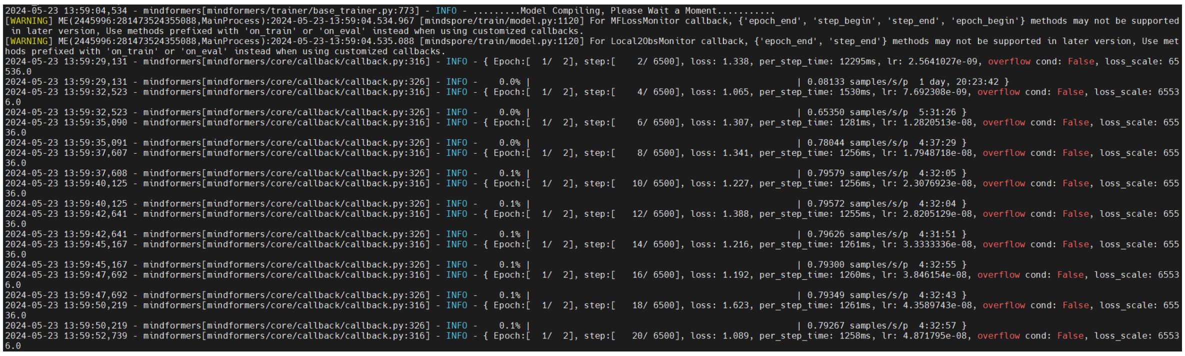


```
Running Command: msrun --worker_num=8    --local_worker_num=8    --master_port=8118    --log_dir=output/msrun_log    --join=False    --cluster_time_out=600 run_mindformer.py --config configs/llama2/finetune_llama2_7b.yaml
 --run_mode finetune
Please check log files in output/msrun_log
[WARNING] DEVICE(2445711,ffffb627a010,python):2024-05-23-13:53:14.347.595 [mindspore/ccsrc/plugin/device/ascend/hal/hardware/ge_device_context.cc:588] operator()] The Environment Variable MS_ENABLE_FORMAT_MODE will be dis
carded, please use MS_FORMAT_MODE instead.
[WARNING] DEVICE(2445711,ffffb627a010,python):2024-05-23-13:53:14.420.524 [mindspore/ccsrc/plugin/device/ascend/hal/hardware/ge_device_context.cc:588] operator()] The Environment Variable MS_ENABLE_FORMAT_MODE will be dis
carded, please use MS_FORMAT_MODE instead.
[WARNING] ME(2445711:281473737793552,MainProcess):2024-05-23-13:53:14.420.703 [mindspore/run_check/_check_version.py:348] Using custom Ascend AI software package (Ascend Data Center Solution) path, package version checkin
g is skipped. Please make sure Ascend AI software package (Ascend Data Center Solution) version is supported. For details, refer to the installation guidelines https://www.mindspore.cn/install
[WARNING] ME(2445711:281473737793552,MainProcess):2024-05-23-13:53:14.420.891 [mindspore/run_check/_check_version.py:469] Can not find driver so(need by mindspore-ascend). Please check whether the Environment Variable LD_
LIBRARY_PATH is set. For details, refer to the installation guidelines: https://www.mindspore.cn/install
/root/miniconda3/envs/ryj_ms2.3/lib/python3.9/site-packages/numpy/core/getlimits.py:549: UserWarning: The value of the smallest subnormal for <class 'numpy.float64'> type is zero.
  setattr(self, word, getattr(machar, word).flat[0])
/root/miniconda3/envs/ryj_ms2.3/lib/python3.9/site-packages/numpy/core/getlimits.py:89: UserWarning: The value of the smallest subnormal for <class 'numpy.float64'> type is zero.
  return self._float_to_str(self.smallest_subnormal)
/root/miniconda3/envs/ryj_ms2.3/lib/python3.9/site-packages/numpy/core/getlimits.py:549: UserWarning: The value of the smallest subnormal for <class 'numpy.float32'> type is zero.
  setattr(self, word, getattr(machar, word).flat[0])
/root/miniconda3/envs/ryj_ms2.3/lib/python3.9/site-packages/numpy/core/getlimits.py:89: UserWarning: The value of the smallest subnormal for <class 'numpy.float32'> type is zero.
  return self._float_to_str(self.smallest_subnormal)
[WARNING] ME(2445711:281473737793552,MainProcess):2024-05-23-13:53:17.330.689 [mindspore/parallel/cluster/process_entity/_api.py:210] Start worker process with rank id:0, log file:output/msrun_log/worker_0.log. Environmen
t variable [RANK_ID] is exported.
[WARNING] ME(2445711:281473737793552,MainProcess):2024-05-23-13:53:17.434.515 [mindspore/parallel/cluster/process_entity/_api.py:210] Start worker process with rank id:1, log file:output/msrun_log/worker_1.log. Environmen
t variable [RANK_ID] is exported.
[WARNING] ME(2445711:281473737793552,MainProcess):2024-05-23-13:53:17.545.868 [mindspore/parallel/cluster/process_entity/_api.py:210] Start worker process with rank id:2, log file:output/msrun_log/worker_2.log. Environmen
t variable [RANK_ID] is exported.
[WARNING] ME(2445711:281473737793552,MainProcess):2024-05-23-13:53:17.663.887 [mindspore/parallel/cluster/process_entity/_api.py:210] Start worker process with rank id:3, log file:output/msrun_log/worker_3.log. Environmen
t variable [RANK_ID] is exported.
[WARNING] ME(2445711:281473737793552,MainProcess):2024-05-23-13:53:17.772.909 [mindspore/parallel/cluster/process_entity/_api.py:210] Start worker process with rank id:4, log file:output/msrun_log/worker_4.log. Environmen
t variable [RANK_ID] is exported.
[WARNING] ME(2445711:281473737793552,MainProcess):2024-05-23-13:53:17.885.447 [mindspore/parallel/cluster/process_entity/_api.py:210] Start worker process with rank id:5, log file:output/msrun_log/worker_5.log. Environmen
t variable [RANK_ID] is exported.
[WARNING] ME(2445711:281473737793552,MainProcess):2024-05-23-13:53:18.311.58 [mindspore/parallel/cluster/process_entity/_api.py:210] Start worker process with rank id:6, log file:output/msrun_log/worker_6.log. Environment
 variable [RANK_ID] is exported.
[WARNING] ME(2445711:281473737793552,MainProcess):2024-05-23-13:53:18.150.187 [mindspore/parallel/cluster/process_entity/_api.py:210] Start worker process with rank id:7, log file:output/msrun_log/worker_7.log. Environmen
t variable [RANK_ID] is exported.
```

查看微调执行情况

```
tail -f output/msrun_log/worker_0.log
```

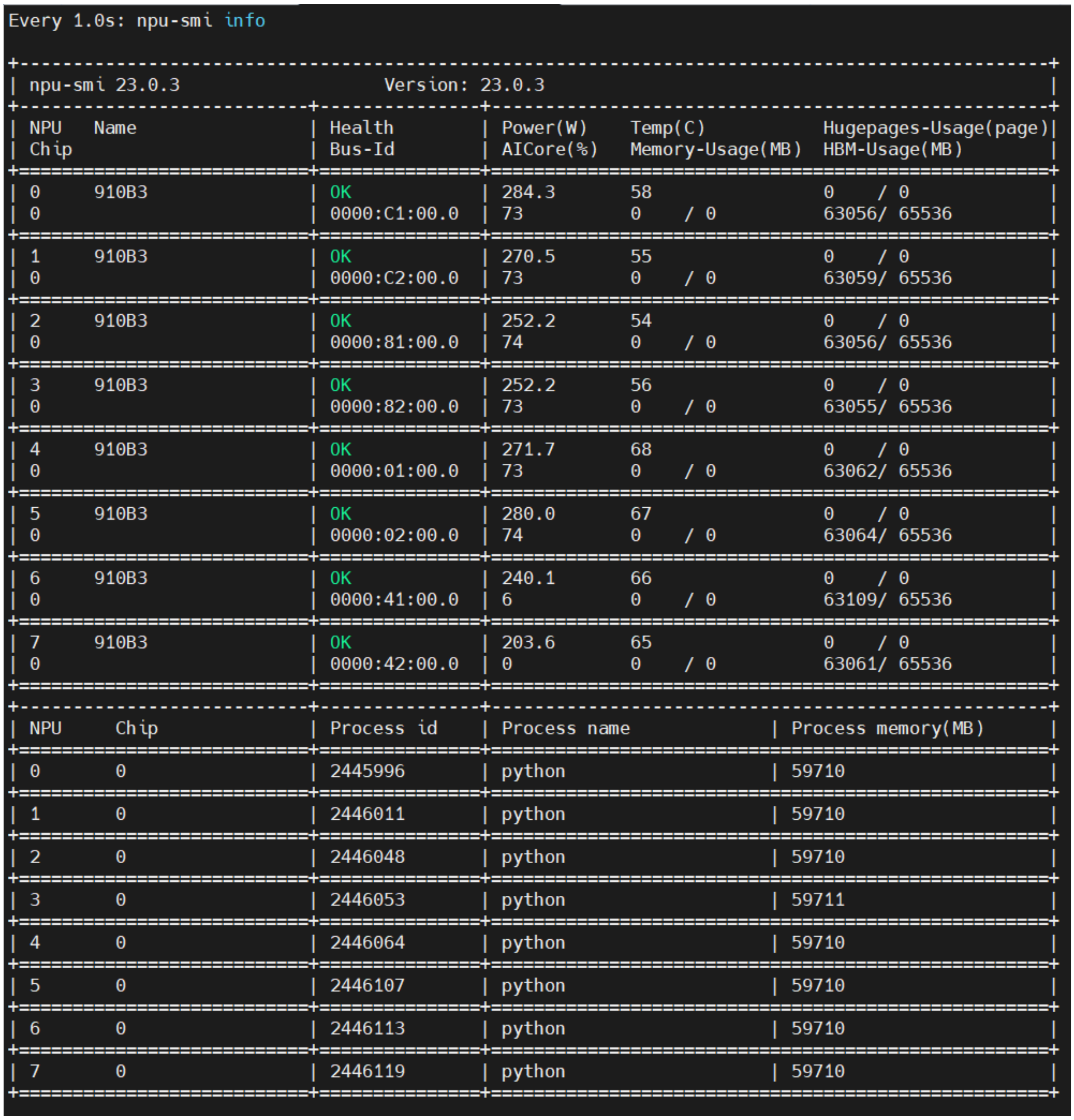
构建模型后，权重切分和网络编译需要等待10分钟左右。待出现loss则训练拉起成功



```
2024-05-23 13:59:04,534 - mindformers[mindformers/trainer/base_trainer.py:773] - INFO - .........Model Compiling, Please Wait a Moment...........
[WARNING] ME(2445996:281473524355088,MainProcess):2024-05-23-13:59:04.534.967 [mindspore/train/model.py:1120] For MFLossMonitor callback, {'epoch_end', 'step_begin', 'step_end', 'epoch_begin'} methods may not be supported
 in later version, Use methods prefixed with 'on_train' or 'on_eval' instead when using customized callbacks.
[WARNING] ME(2445996:281473524355088,MainProcess):2024-05-23-13:59:04.535.088 [mindspore/train/model.py:1120] For Local2ObsMonitor callback, {'epoch_end', 'step_end'} methods may not be supported in later version, Use met
hods prefixed with 'on_train' or 'on_eval' instead when using customized callbacks.
2024-05-23 13:59:29,131 - mindformers[mindformers/core/callback/callback.py:316] - INFO - { Epoch:[  1/  2], step:[    2/ 6500], loss: 1.338, per_step_time: 12295ms, lr: 2.5641027e-09, overflow cond: False, loss_scale: 65
536.0
2024-05-23 13:59:29,131 - mindformers[mindformers/core/callback/callback.py:326] - INFO -    0.0% |                                                  | 0.08133 samples/s/p  1 day, 20:23:42 }
2024-05-23 13:59:32,523 - mindformers[mindformers/core/callback/callback.py:316] - INFO - { Epoch:[  1/  2], step:[    4/ 6500], loss: 1.065, per_step_time: 1530ms, lr: 7.692308e-09, overflow cond: False, loss_scale: 6553
6.0
2024-05-23 13:59:32,523 - mindformers[mindformers/core/callback/callback.py:326] - INFO -    0.0% |                                                  | 0.65350 samples/s/p  5:31:26 }
2024-05-23 13:59:35,090 - mindformers[mindformers/core/callback/callback.py:316] - INFO - { Epoch:[  1/  2], step:[    6/ 6500], loss: 1.307, per_step_time: 1281ms, lr: 1.2820513e-08, overflow cond: False, loss_scale: 655
36.0
2024-05-23 13:59:35,091 - mindformers[mindformers/core/callback/callback.py:326] - INFO -    0.0% |                                                  | 0.78044 samples/s/p  4:37:29 }
2024-05-23 13:59:37,607 - mindformers[mindformers/core/callback/callback.py:316] - INFO - { Epoch:[  1/  2], step:[    8/ 6500], loss: 1.341, per_step_time: 1256ms, lr: 1.7948718e-08, overflow cond: False, loss_scale: 655
36.0
2024-05-23 13:59:37,608 - mindformers[mindformers/core/callback/callback.py:326] - INFO -    0.1% |                                                  | 0.79579 samples/s/p  4:32:05 }
2024-05-23 13:59:40,125 - mindformers[mindformers/core/callback/callback.py:316] - INFO - { Epoch:[  1/  2], step:[   10/ 6500], loss: 1.227, per_step_time: 1256ms, lr: 2.3076923e-08, overflow cond: False, loss_scale: 655
36.0
2024-05-23 13:59:40,125 - mindformers[mindformers/core/callback/callback.py:326] - INFO -    0.1% |                                                  | 0.79572 samples/s/p  4:32:04 }
2024-05-23 13:59:42,641 - mindformers[mindformers/core/callback/callback.py:316] - INFO - { Epoch:[  1/  2], step:[   12/ 6500], loss: 1.388, per_step_time: 1255ms, lr: 2.8205129e-08, overflow cond: False, loss_scale: 655
36.0
2024-05-23 13:59:42,641 - mindformers[mindformers/core/callback/callback.py:326] - INFO -    0.1% |                                                  | 0.79626 samples/s/p  4:31:51 }
2024-05-23 13:59:45,167 - mindformers[mindformers/core/callback/callback.py:316] - INFO - { Epoch:[  1/  2], step:[   14/ 6500], loss: 1.216, per_step_time: 1261ms, lr: 3.3333336e-08, overflow cond: False, loss_scale: 655
36.0
2024-05-23 13:59:45,167 - mindformers[mindformers/core/callback/callback.py:326] - INFO -    0.1% |                                                  | 0.79300 samples/s/p  4:32:55 }
2024-05-23 13:59:47,692 - mindformers[mindformers/core/callback/callback.py:316] - INFO - { Epoch:[  1/  2], step:[   16/ 6500], loss: 1.192, per_step_time: 1260ms, lr: 3.846154e-08, overflow cond: False, loss_scale: 6553
6.0
2024-05-23 13:59:47,692 - mindformers[mindformers/core/callback/callback.py:326] - INFO -    0.1% |                                                  | 0.79349 samples/s/p  4:32:43 }
2024-05-23 13:59:50,219 - mindformers[mindformers/core/callback/callback.py:316] - INFO - { Epoch:[  1/  2], step:[   18/ 6500], loss: 1.623, per_step_time: 1261ms, lr: 4.3589743e-08, overflow cond: False, loss_scale: 655
36.0
2024-05-23 13:59:50,219 - mindformers[mindformers/core/callback/callback.py:326] - INFO -    0.1% |                                                  | 0.79267 samples/s/p  4:32:57 }
2024-05-23 13:59:52,739 - mindformers[mindformers/core/callback/callback.py:316] - INFO - { Epoch:[  1/  2], step:[   20/ 6500], loss: 1.089, per_step_time: 1258ms, lr: 4.871795e-08, overflow cond: False, loss_scale: 6553
6.0
```

实时查看NPU使用情况

```
watch -n 1 npu-smi info
```



```
Every 1.0s: npu-smi info

+------------------------------------------------------------------------------------------------+
| npu-smi 23.0.3                   Version: 23.0.3                                               |
+---------------------------+---------------+----------------------------------------------------+
| NPU   Name                | Health        | Power(W)    Temp(C)           Hugepages-Usage(page)|
| Chip                      | Bus-Id        | AICore(%)   Memory-Usage(MB)  HBM-Usage(MB)        |
+===========================+===============+====================================================+
| 0     910B3               | OK            | 284.3       58                0    / 0             |
| 0                         | 0000:C1:00.0  | 73          0    / 0          63056/ 65536         |
+===========================+===============+====================================================+
| 1     910B3               | OK            | 270.5       55                0    / 0             |
| 0                         | 0000:C2:00.0  | 73          0    / 0          63059/ 65536         |
+===========================+===============+====================================================+
| 2     910B3               | OK            | 252.2       54                0    / 0             |
| 0                         | 0000:81:00.0  | 74          0    / 0          63056/ 65536         |
+===========================+===============+====================================================+
| 3     910B3               | OK            | 252.2       56                0    / 0             |
| 0                         | 0000:82:00.0  | 73          0    / 0          63055/ 65536         |
+===========================+===============+====================================================+
| 4     910B3               | OK            | 271.7       68                0    / 0             |
| 0                         | 0000:01:00.0  | 73          0    / 0          63062/ 65536         |
+===========================+===============+====================================================+
| 5     910B3               | OK            | 280.0       67                0    / 0             |
| 0                         | 0000:02:00.0  | 74          0    / 0          63064/ 65536         |
+===========================+===============+====================================================+
| 6     910B3               | OK            | 240.1       66                0    / 0             |
| 0                         | 0000:41:00.0  | 6           0    / 0          63109/ 65536         |
+===========================+===============+====================================================+
| 7     910B3               | OK            | 203.6       65                0    / 0             |
| 0                         | 0000:42:00.0  | 0           0    / 0          63061/ 65536         |
+===========================+===============+====================================================+
+---------------------------+---------------+----------------------------------------------------+
| NPU     Chip              | Process id    | Process name             | Process memory(MB)      |
+===========================+===============+====================================================+
| 0       0                 | 2445996       | python                   | 59710                   |
+===========================+===============+====================================================+
| 1       0                 | 2446011       | python                   | 59710                   |
+===========================+===============+====================================================+
| 2       0                 | 2446048       | python                   | 59710                   |
+===========================+===============+====================================================+
| 3       0                 | 2446053       | python                   | 59711                   |
+===========================+===============+====================================================+
| 4       0                 | 2446064       | python                   | 59710                   |
+===========================+===============+====================================================+
| 5       0                 | 2446107       | python                   | 59710                   |
+===========================+===============+====================================================+
| 6       0                 | 2446113       | python                   | 59710                   |
+===========================+===============+====================================================+
| 7       0                 | 2446119       | python                   | 59710                   |
+===========================+===============+====================================================+
```

3.3 LoRA微调（课后练习）

参考MindFormers开源仓中的：LLaMA2微调文档

4. 推理

> 以下所有操作均在容器内进行

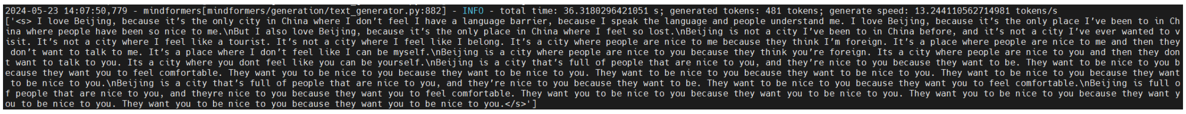
4.1 自回归推理

启动python终端，输入以下内容以完成自回归推理：

```python
# 设置MindSpore图模式并指定使用的device_id
import mindspore as ms
ms.set_context(mode=0, device_target="Ascend", device_id=0)
from mindformers import AutoModel, AutoTokenizer

# 通过AutoModel接口实例化模型
model = AutoModel.from_pretrained("llama2_7b", use_past=False, seq_length=512)
# 通过AutoTokenizer接口实例化tokenizer
tokenizer = AutoTokenizer.from_pretrained("llama2_7b")
```

```python
# 生成输入
input_ids = tokenizer("I love Beijing, because")["input_ids"]

# 调用model.generate()接口执行文本生成推理，多次执行推理，规避首次编图耗时
for i in range(5):
    output = model.generate(input_ids, do_sample=True, top_k=3)
    # 解码并打印输出
    print(tokenizer.decode(output))
```

推理结果：



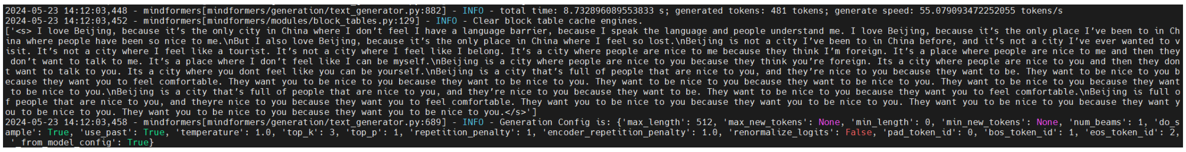
```
2024-05-23 14:07:50,779 - mindformers[mindformers/generation/text_generator.py:882] - INFO - total time: 36.3180296421051 s; generated tokens: 481 tokens; generate speed: 13.244110562714981 tokens/s
['<s> I love Beijing, because it’s the only city in China where I don’t feel I have a language barrier, because I speak the language and people understand me. I love Beijing, because it’s the only place I’ve been to in Ch
ina where people have been so nice to me.\nBut I also love Beijing, because it’s the only place in China where I feel so lost.\nBeijing is not a city I’ve been to in China before, and it’s not a city I’ve ever wanted to v
isit. It’s not a city where I feel like a tourist. It’s not a city where I feel like I belong. It’s a city where people are nice to me because they think I’m foreign. It’s a place where people are nice to me and then they
 don’t want to talk to me. It’s a place where I don’t feel like I can be myself.\nBeijing is a city where people are nice to you because they think you’re foreign. Its a city where people are nice to you and then they don
t want to talk to you. Its a city where you dont feel like you can be yourself.\nBeijing is a city that’s full of people that are nice to you, and they’re nice to you because they want to be. They want to be nice to you b
ecause they want you to feel comfortable. They want you to be nice to you because they want to be nice to you. They want to be nice to you because they want to be nice to you. They want to be nice to you because they want
 to be nice to you.\nBeijing is a city that’s full of people that are nice to you, and they’re nice to you because they want to be. They want to be nice to you because they want you to feel comfortable.\nBeijing is full o
f people that are nice to you, and theyre nice to you because they want you to feel comfortable. They want you to be nice to you because they want you to be nice to you. They want you to be nice to you because they want y
ou to be nice to you. They want you to be nice to you because they want you to be nice to you.</s>']
```

4.2 增量推理

启动python终端，输入以下内容以完成增量推理：

```python
# 设置MindSpore图模式并指定使用的device_id
import mindspore as ms
ms.set_context(mode=0, device_target="Ascend", device_id=0)
from mindformers import AutoModel, AutoTokenizer

# 通过AutoModel接口实例化模型
model = AutoModel.from_pretrained("llama2_7b", use_past=True, seq_length=512)
# 通过AutoTokenizer接口实例化tokenizer
tokenizer = AutoTokenizer.from_pretrained("llama2_7b")

# 生成输入
input_ids = tokenizer("I love Beijing, because")["input_ids"]

# 调用model.generate()接口执行文本生成推理，多次执行推理，规避首次编图耗时
for i in range(5):
    output = model.generate(input_ids, do_sample=True, top_k=3)
    # 解码并打印输出
    print(tokenizer.decode(output))
```

推理结果：



```
2024-05-23 14:12:03,448 - mindformers[mindformers/generation/text_generator.py:882] - INFO - total time: 8.732896089553833 s; generated tokens: 481 tokens; generate speed: 55.079093472252055 tokens/s
2024-05-23 14:12:03,452 - mindformers[mindformers/modules/block_tables.py:129] - INFO - Clear block table cache engines.
['<s> I love Beijing, because it’s the only city in China where I don’t feel I have a language barrier, because I speak the language and people understand me. I love Beijing, because it’s the only place I’ve been to in Ch
ina where people have been so nice to me.\nBut I also love Beijing, because it’s the only place in China where I feel so lost.\nBeijing is not a city I’ve been to in China before, and it’s not a city I’ve ever wanted to v
isit. It’s not a city where I feel like a tourist. It’s not a city where I feel like I belong. It’s a city where people are nice to me because they think I’m foreign. It’s a place where people are nice to me and then they
 don’t want to talk to me. It’s a place where I don’t feel like I can be myself.\nBeijing is a city where people are nice to you because they think you’re foreign. Its a city where people are nice to you and then they don
t want to talk to you. Its a city where you dont feel like you can be yourself.\nBeijing is a city that’s full of people that are nice to you, and they’re nice to you because they want to be. They want to be nice to you b
ecause they want you to feel comfortable. They want you to be nice to you because they want to be nice to you. They want to be nice to you because they want to be nice to you. They want to be nice to you because they want
 to be nice to you.\nBeijing is a city that’s full of people that are nice to you, and they’re nice to you because they want to be. They want to be nice to you because they want you to feel comfortable.\nBeijing is full o
f people that are nice to you, and theyre nice to you because they want you to feel comfortable. They want you to be nice to you because they want you to be nice to you. They want you to be nice to you because they want y
ou to be nice to you. They want you to be nice to you because they want you to be nice to you.</s>']
2024-05-23 14:12:03,458 - mindformers[mindformers/generation/text_generator.py:689] - INFO - Generation Config is: {'max_length': 512, 'max_new_tokens': None, 'min_length': 0, 'min_new_tokens': None, 'num_beams': 1, 'do_s
ample': True, 'use_past': True, 'temperature': 1.0, 'top_k': 3, 'top_p': 1, 'repetition_penalty': 1, 'encoder_repetition_penalty': 1.0, 'renormalize_logits': False, 'pad_token_id': 0, 'bos_token_id': 1, 'eos_token_id': 2,
 '_from_model_config': True}
```

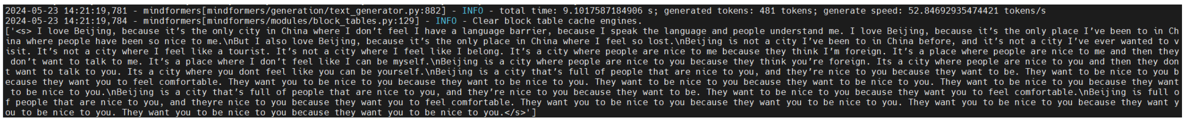
4.3 流式推理

启动python终端，输入以下内容以完成流式推理：

```python
# 设置MindSpore图模式并指定使用的device_id
import mindspore as ms
ms.set_context(mode=0, device_target="Ascend", device_id=0)
from mindformers import AutoModel, AutoTokenizer

# 通过AutoModel接口实例化模型
model = AutoModel.from_pretrained("llama2_7b", use_past=True, seq_length=512)
# 通过AutoTokenizer接口实例化tokenizer
```

```python
tokenizer = AutoTokenizer.from_pretrained("llama2_7b")

# 生成输入
input_ids = tokenizer("I love Beijing, because")["input_ids"]

# 标准输出流
from mindformers import TextStreamer
streamer = TextStreamer(tokenizer)

# 调用model.generate()接口执行文本生成推理，多次执行推理，规避首次编图耗时
for i in range(5):
    output = model.generate(input_ids, do_sample=True, top_k=3,
streamer=streamer)
    # 解码并打印输出
    print(tokenizer.decode(output))
```

推理结果：



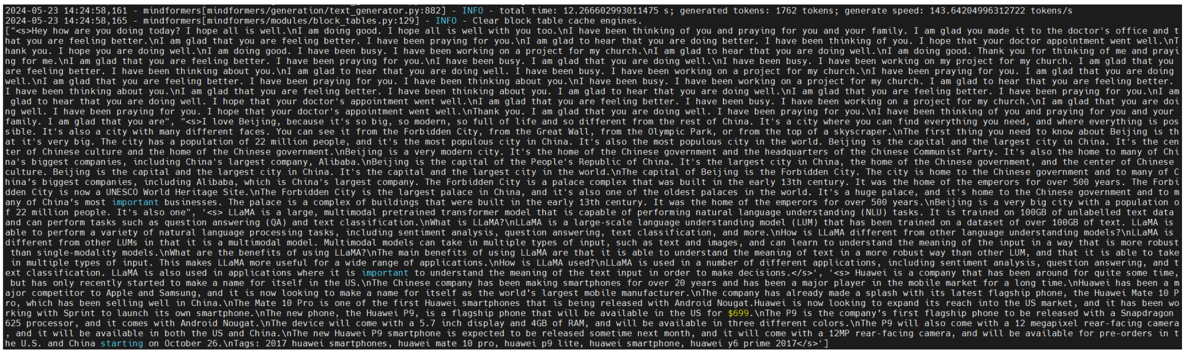
```
2024-05-23 14:21:19,781 - mindformers[mindformers/generation/text_generator.py:882] - INFO - total time: 9.1017587184906 s; generated tokens: 481 tokens; generate speed: 52.84692935474421 tokens/s
2024-05-23 14:21:19,784 - mindformers[mindformers/modules/block_tables.py:129] - INFO - Clear block table cache engines.
['<s> I love Beijing, because it’s the only city in China where I don’t feel I have a language barrier, because I speak the language and people understand me. I love Beijing, because it’s the only place I’ve been to in Ch
ina where people have been so nice to me.\nBut I also love Beijing, because it’s the only place in China where I feel so lost.\nBeijing is not a city I’ve been to in China before, and it’s not a city I’ve ever wanted to v
isit. It’s not a city where I feel like a tourist. It’s not a city where I feel like I belong. It’s a city where people are nice to me because they think I’m foreign. It’s a place where people are nice to me and then they
 don’t want to talk to me. It’s a place where I don’t feel like I can be myself.\nBeijing is a city where people are nice to you because they think you’re foreign. Its a city where people are nice to you and then they don
t want to talk to you. Its a city where you dont feel like you can be yourself.\nBeijing is a city that’s full of people that are nice to you, and they’re nice to you because they want to be. They want to be nice to you b
ecause they want you to feel comfortable. They want you to be nice to you because they want to be nice to you. They want to be nice to you because they want to be nice to you. They want to be nice to you because they want
 to be nice to you.\nBeijing is a city that’s full of people that are nice to you, and they’re nice to you because they want to be. They want to be nice to you because they want you to feel comfortable.\nBeijing is full o
f people that are nice to you, and theyre nice to you because they want you to feel comfortable. They want you to be nice to you because they want you to be nice to you. They want you to be nice to you because they want y
ou to be nice to you. They want you to be nice to you because they want you to be nice to you.</s>']
```

4.4 Batch推理

启动python终端，输入以下内容以完成batch推理：

```python
# 设置MindSpore图模式并指定使用的device_id
import mindspore as ms
ms.set_context(mode=0, device_target="Ascend", device_id=0)
from mindformers import AutoModel, AutoTokenizer

# 通过AutoModel接口实例化模型
# 多batch推理时模型实例化时batch_size设置为对应值
model = AutoModel.from_pretrained("llama2_7b", batch_size=4, use_past=True,
seq_length=512)
# 通过AutoTokenizer接口实例化tokenizer
tokenizer = AutoTokenizer.from_pretrained("llama2_7b")

# 生成多batch输入
input_list = ["Hey how are you doing today?",
              "I love Beijing, because",
              "LLaMA is a",
              "Huawei is a company that"]
input_ids = tokenizer(input_list, max_length=64, padding="max_length")
["input_ids"]

# 调用model.generate()接口执行文本生成推理，多次执行推理，规避首次编图耗时
for i in range(5):
    output = model.generate(input_ids, do_sample=True, top_k=3)
    # 解码并打印输出
    print(tokenizer.decode(output))
```

推理结果



```
2024-05-23 14:24:58,161 - mindformers[mindformers/generation/text_generator.py:882] - INFO - total time: 12.266602993011475 s; generated tokens: 1762 tokens; generate speed: 143.64204996312722 tokens/s
2024-05-23 14:24:58,165 - mindformers[mindformers/modules/block_tables.py:129] - INFO - Clear block table cache engines.
["<s>Hey how are you doing today? I hope all is well.\nI am doing good. I hope all is well with you too.\nI have been thinking of you and praying for you and your family. I am glad you made it to the doctor's office and t
hat you are feeling better.\nI am glad that you are feeling better. I have been praying for you.\nI am glad to hear that you are doing better. I have been thinking of you. I hope that your doctor appointment went well.\nT
hank you. I hope you are doing well.\nI am doing good. I have been busy. I have been working on a project for my church.\nI am glad to hear that you are doing well.\nI am doing good. Thank you for thinking of me and prayi
ng for me.\nI am glad that you are feeling better. I have been praying for you.\nI have been busy. I am glad that you are doing well.\nI have been busy. I have been working on my project for my church. I am glad that you
are feeling better. I have been thinking about you.\nI am glad to hear that you are doing well. I have been busy. I have been busy. I have been working on a project for my church.\nI have been praying for you. I am glad that you are doing
well.\nI am glad that you are feeling better. I have been praying for you. I have been thinking about you.\nI have been busy. I have been working on a project for my church. I am glad to hear that you are feeling better.
I have been thinking about you.\nI am glad that you are feeling better. I have been thinking about you. I am glad to hear that you are doing well.\nI am glad that you are feeling better. I have been praying for you.\nI am
 glad to hear that you are doing well. I hope that your doctor's appointment went well.\nI am glad that you are feeling better. I have been busy. I have been working on a project for my church.\nI am glad that you are doi
ng well. I have been praying for you. I hope that your doctor's appointment went well.\nThank you. I am glad that you are doing well. I have been praying for you.\nI have been thinking of you and praying for you and your
family. I am glad that you are", "<s>I love Beijing, because it's so big, so modern, so full of life and so different from the rest of China. It's a city where you can find everything you need, and where everything is pos
sible. It's also a city with many different faces. You can see it from the Forbidden City, from the Great Wall, from the Olympic Park, or from the top of a skyscraper.\nThe first thing you need to know about Beijing is th
at it's very big. The city has a population of 22 million people, and it's the most populous city in China. It's also the most populous city in the world. Beijing is the capital and the largest city in China. It's the cen
ter of Chinese culture and the home of the Chinese government.\nBeijing is a very modern city. It's the home of the Chinese government and the headquarters of the Chinese Communist Party. It's also the home to many of Chi
na's biggest companies, including China's largest company, Alibaba.\nBeijing is the capital of the People's Republic of China. It's the largest city in China, the home of the Chinese government, and the center of Chinese
culture. Beijing is the capital and the largest city in China. It's the capital and the largest city in the world.\nThe capital of Beijing is the Forbidden City. The city is home to the Chinese government and to many of C
hina’s biggest companies, including Alibaba, which is China's largest company. The Forbidden City is a palace complex that was built in the early 13th century. It was the home of the emperors for over 500 years. The Forbi
dden City is now a UNESCO World Heritage Site.\nThe Forbidden City is the largest palace in China, and it's also one of the oldest palaces in the world. It's a huge palace, and it's home to the Chinese government and to m
any of China’s most important businesses. The palace is a complex of buildings that were built in the early 13th century. It was the home of the emperors for over 500 years.\nBeijing is a very big city with a population o
f 22 million people. It's also one", '<s> LLaMA is a large, multimodal pretrained transformer model that is capable of performing natural language understanding (NLU) tasks. It is trained on 100GB of unlabelled text data
and can perform tasks such as question answering (QA) and text classification.\nWhat is LLaMA?\nLLaMA is a large-scale language understanding model (LUM) that has been trained on a dataset of over 100GB of text. LLaMA is
able to perform a variety of natural language processing tasks, including sentiment analysis, question answering, text classification, and more.\nHow is LLaMA different from other language understanding models?\nLLaMA is
different from other LUMs in that it is a multimodal model. Multimodal models can take in multiple types of input, such as text and images, and can learn to understand the meaning of the input in a way that is more robust
 than single-modality models.\nWhat are the benefits of using LLaMA?\nThe main benefits of using LLaMA are that it is able to understand the meaning of text in a more robust way than other LUM, and that it is able to take
 in multiple types of input. This makes LLaMA more useful for a wide range of applications.\nHow is LLaMA used?\nLLaMA is used in a number of different applications, including sentiment analysis, question answering, and t
ext classification. LLaMA is also used in applications where it is important to understand the meaning of the text input in order to make decisions.</s>', '<s> Huawei is a company that has been around for quite some time,
 but has only recently started to make a name for itself in the US.\nThe Chinese company has been making smartphones for over 20 years and has been a major player in the mobile market for a long time.\nHuawei has been a m
ajor competitor to Apple and Samsung, and it is now looking to make a name for itself as the world’s largest mobile manufacturer.\nThe company has already made a splash with its latest flagship phone, the Huawei Mate 10 P
ro, which has been selling well in China.\nThe Mate 10 Pro is one of the first Huawei smartphones that is being released with Android Nougat.Huawei is now looking to expand its reach into the US market, and it has been wo
rking with Sprint to launch its own smartphone.\nThe new phone, the Huawei P9, is a flagship phone that will be available in the US for $699.\nThe P9 is the company’s first flagship phone to be released with a Snapdragon
625 processor, and it comes with Android Nougat.\nThe device will come with a 5.7 inch display and 4GB of RAM, and will be available in three different colors.\nThe P9 will also come with a 12 megapixel rear-facing camera
, and it will be available in both the US and China.\nThe new Huawei P9 smartphone is expected to be released sometime next month, and it will come with a 12MP rear-facing camera, and will be available for pre-orders in t
he U.S. and China starting on October 26.\nTags: 2017 huawei smartphones, huawei mate 10 pro, huawei p9 lite, huawei smartphone, huawei y6 prime 2017</s>']
```

4.5 分布式推理（课后练习）

请参照分布式推理教程和LLaMA2多卡推理文档)进行学习