# 燧原S60测试报告

## 1. 环境规格

| 硬件 | 组件 | 详情 |
| --- | --- | --- |
| 服务器<br>（10.110.181.137） | 处理器 | Intel(R) Xeon(R) Gold 5218 CPU @ 2.30GHz |
| | 内存 | 125Gi |
| | 型号 | DELL R740 |
| | CPU核数 | 64 |
| | 操作系统 | Ubuntu 22.04.4 LTS |
| GPU | 型号 | S60 |
| | 显存 | GDDR6；48GB |
| | 显存带宽 | 672GB/s |
| | 接口规格 | PCIe Gen5 X16 |
| | 峰值算力 | 支持FP32、 FP16、BF16、 INT8四种精度<br>FP16：392TFLOPS？ |
| | TDP | 300W |
| | 最大操作温度 | 95°C |

## 2. 环境部署

### 2.1 驱动及软件包下载

```
sftp -o Port=2222 ftp_support@ftp.enflame-tech.com
password: 167jtnUco6
```

```
root@test-hpc-05:~# sftp -o Port=2222 ftp_support@ftp.enflame-tech.com
(ftp_support@ftp.enflame-tech.com) Password:
Connected to ftp.enflame-tech.com.
sftp> ls
RuiJie.tar  dataset
sftp> get RuiJie.tar
Fetching /RuiJie.tar to RuiJie.tar
RuiJie.tar                                                    0% 4512KB   2.9MB/s   42:53 ETA
```

```
# 下载完成之后解压:
tar -xvf RuiJie.tar
# 解压后文件如下图:
```

```
root@cse:/home/RuiJie# ll
total 3703952
drwxr-xr-x  4 root root       4096 Aug 15 15:07 ./
drwxr-xr-x 14 root root       4096 Aug 15 15:42 ../
drwxr-xr-x  3 root root       4096 Aug 15 15:05 TopsRider_i3x_3.1.2024081302_application_internal/
-rw-r--r--  1 root root 2218438224 Aug 15 15:05 TopsRider_i3x_3.1.8_deb_amd64.run
-rwxr-xr-x  1 root root        360 Aug 15 15:05 docker_run.sh*
drwxr-xr-x  2 root root       4096 Aug 15 15:07 models/
-rw-r--r--  1 root root     170765 Aug 15 15:07 sentence_transformers-2.7.0+gcu.3.2.20240805-py3-none-any.whl
-rw-------  1 root root 1574206976 Aug 15 15:06 ubuntu_amd64_20.04_dockerfile_images.tar
root@cse:/home/RuiJie#
```

## 2.2 硬件检查

服务器插卡后，可以通过以下命令检查加速卡是否安装正确。

```
lspci -d 1e36:
```

```
root@test-hpc-05:/home/workspace/inference_scripts# lspci -d 1e36:
3b:00.0 Processing accelerators: Shanghai Enflame Technology Co. Ltd S60 [Enflame] (rev 01)
```

## 2.3 驱动安装

```
cd Ruijie
bash TopsRider_i3x_3.1.8_deb_amd64.run -y
```

安装成功如下图所示：

```
root@cse:/home/Ruijie_test# bash TopsRider_i3x_3.1.2024081302_deb_amd64.run -y
Verifying archive integrity...  100%   MD5 checksums are OK. All good.
Uncompressing ENFLAME TOPSRIDER PACKAGE  100%
Logging file: /tmp/topsinstaller/20240814-173350.log
[1/3] Install TopsPlatform Package
[2/3] Install Dockerfile to /usr/local/topsrider/dockerfile
[3/3] Install Data Center Toolkit to /usr/local/topsrider/data_center_toolkit
Install Finished. 3 installed.

Please make sure that
 - PATH includes /opt/tops/bin
root@cse:/home/Ruijie_test#
```

检查驱动版本：

```
cat /sys/module/enflame/version
```

```
root@test-hpc-05:/home/workspace/inference_scripts# cat /sys/module/enflame/version
1.0.6.315
```

## 2.4 资源监控

安装驱动之后，在Host 环境下可以查看GCU 信息。

```
efsmi -dmon
```

```
root@test-hpc-05:~# efsmi -dmon
-------------------------------------------------------------------------------
------------------- Enflame System Management Interface --------------------
---------- Enflame Tech, All Rights Reserved. 2024 Copyright (C) -----------
-------------------------------------------------------------------------------

*Dev Logic  Pwr    DTemp  DUsed  Dpm     MUsed  Mem    Mclk
*Idx Id     W      C      %      L       %      MiB    MHz
0    0      96.0   40.0   0.0    Sleep   3.2    42976  7000
0    0      96.0   40.0   0.0    Sleep   3.2    42976  7000
0    0      96.0   40.0   0.0    Sleep   3.2    42976  7000
0    0      96.0   40.0   0.0    Sleep   3.2    42976  7000
0    0      96.0   40.0   0.0    Sleep   3.2    42976  7000
0    0      96.0   40.0   0.0    Sleep   3.2    42976  7000
0    0      96.0   40.0   0.0    Sleep   3.2    42976  7000
```

## 2.5 软件栈安装

1. 加载基础镜像并创建容器

```
cd Ruijie
 bash docker_run.sh
 cd /home/workspace
```

```
root@test-hpc-05:/home/lsc/RuiJie# cat docker_run.sh
#!/bin/bash

# 1. docker load image
docker load -i $PWD/ubuntu_amd64_20.04_dockerfile_images.tar   加载镜像


# 2. docker run a container with the special name "enflame_$version"
REPOSITORY=enflame
TAG=ubuntu_amd64_20.04_dockerfile_0731      容器名字

docker run -it --name docker_ruijie_test --network=host --ipc=host --privileged -v $PWD:/home/workspace $REPOSITORY:$TAG /bin/bash
```

2. 在容器内安装燧原软件包

```
# 安装vllm 框架+sentence-transformers
bash TopsRider_i3x_3.1.8_deb_amd64.run -C sentence-transformers,vllm -y
```

3. 重新安装新版本的sentence_transformers

```
# 卸载一键安装的sentence_transformers（临时处理方案，后续不需要重安装）
pip3 uninstall sentence-transformers
# 重新安装新的版本
pip3 install sentence_transformers-2.7.0+gcu.3.2.20240805-py3-none-any.whl
```

# 3. Qwen2-7B-Instruct推理用例测试

## 3.1 测试需求：

1. 输入参数：
   a. batch size = 1, 2, 4, 8
   b. seq length = 8K
   c. FP16/INT8
2. 指标:
   a. TTFT
   b. TPOT
   c. TPS

## 3.2 基于vLLM在线推理功能测试

1. 进入容器，并启动一个基于 OpenAI 的 API 服务器。

```
# 安装了 outlines 模块
pip3 install --upgrade outlines

cd /home/workspace
python3 -m vllm.entrypoints.openai.api_server \
        --model /home/workspace/models/Qwen2-7B-Instruct/ \
        --device gcu \
        --host 0.0.0.0 \
        --trust-remote-code \
        --port 8080 \
```

```
11          --enforce-eager
```

2. 打开另外一个终端，发送客户端请求。

注：下面IP需要替换为对应服务器IP

```
curl http://10.110.181.137:8080/v1/completions \
-H "Content-Type: application/json" \
-d '{
"model": "/home/workspace/models/Qwen2-7B-Instruct/",
"prompt": "如何制作月饼",
"max_tokens": 256,
"temperature": 0.01
}'
```

```
root@test-hpc-05:/home/lsc/RuiJie# curl http://10.110.181.137:8080/v1/completions \
-H "Content-Type: application/json" \
-d '{
"model": "/home/workspace/models/Qwen2-7B-Instruct/",
"prompt": "如何制作月饼",
"max_tokens": 256,
"temperature": 0.01
}'
{"id":"cmpl-a22db6d3011f408cba5d73e9a71e6f26","object":"text_completion","created":155607,"model":"/home/workspace/models/Qwen2-7B-Instruct/","choices":[{"index":0,"text":"皮和馅料？\n制作月饼皮和馅料需要以下材料和步骤：\n月饼皮：\n材料：\n- 中筋面粉 200克\n- 糖浆 100克\n- 猪油 100克\n- 水 50克\n步骤：\n1. 将中筋面粉过筛，放入大碗中。\n2. 将糖浆、猪油和水混合均匀，倒入面粉中。\n3. 用手揉成面团，揉至表面光滑，盖上湿布醒面30分钟。\n4. 将面团分成小份，每个约30克，搓成圆形。\n5. 将面团压扁，包入馅料，收口捏紧，搓成圆形。\n6. 将月饼放入烤盘中，用叉子在表面轻轻划几下，防止膨胀。\n7. 将烤盘放入预热至180度的烤箱中，烤约15-20分钟，至表面金黄即可。\n月饼馅料：\n材料：\n- 红豆沙 200克\n- 花生酱 100克\n- 糖","logprobs":null,"finish_reason":"length"}],"usage":{"prompt_tokens":3,"total_tokens":259,"completion_tokens":256}}root@test-hpc-05:/home/lsc/RuiJie#
```

## 3.3 基于vLLM在线推理功能测试（自定义数据集）

1. 继续沿用上面容器,启动服务端。

```
# 启动server端服务
cd inference_scripts
./run_openai_api_server_gn.sh
```

附录：查看run_openai_api_server_gn.sh

```
cat run_openai_api_server_gn.sh
```

内容如下：

```
# 进入工作目录
CUR_DIR=$(cd $(dirname $0);pwd)
pushd "${CUR_DIR}/.."
#设置模型路径与服务端口
export MODEL_PATH="/home/workspace/models/Qwen2-7B-Instruct/"
export SERVER_PORT=12345
export USE_PAGED=false
export MLU_VISIBLE_DEVICES="0,1"
export MODEL_Name="qwen2-7b-instruct"
export MLU_TP_NUM=1
seqlen_input=8192
seqlen_output=4000
#使用以下命令启动vLLM-API服务器，启动服务器后，vLLM将自动加载模型并启动API服务。
#使用命令 python -m vllm.entrypoints.api_server --help 可查看支持的脚本参数。
export MAX_TOTAL_TOKENS=$(expr ${seqlen_input} + ${seqlen_output})
max_num_batched_tokens=32768
max_model_len=32768
block_size=16
#启动服务
python3 -m vllm.entrypoints.openai.api_server  \
        --host 0.0.0.0 \
        --port ${SERVER_PORT} \
        --block-size ${block_size} \
        --trust-remote-code \
        --dtype float16 \
        --enforce-eager \
        --model ${MODEL_PATH}  \
        --gpu-memory-utilization 0.9 \
        --tensor-parallel-size ${MLU_TP_NUM} \
        --disable-log-requests \
        --max-model-len ${max_model_len} \
        --max-num-batched-tokens ${max_num_batched_tokens} \
        --max-num-seqs 1
```

2. 打开另一个终端，并进入同个容器，启动客户端发送请求。

```
# 重新启动容器
docker start docker_ruijie_test

# 使用 exec 命令进入一个正在运行的容器
docker exec -it docker_ruijie_test /bin/bash

# 执行client请求
cd home/workspace/inference_scripts
./run_openai_api_client_gn.sh
```

附录：查看run_openai_api_client_gn.sh

```
cat run_openai_api_client_gn.sh
```

内容如下：

```
CMD_TIME=$(date +%Y%m%d%H%M%S.%N)
# 进入工作目录
CUR_DIR=$(cd $(dirname $0);pwd)
pushd "${CUR_DIR}/.."
if [ ! -d "./log" ]; then mkdir -p "./log";fi
popd
export MODEL_PATH="/home/workspace/models/Qwen2-7B-Instruct/"
# export MLU_VISIBLE_DEVICES="0"
export MODEL_Name="qwen2-7b-instruct"
# export MLU_TP_NUM=1
export VLLM_LATENCY_DEBUG=FALSE
export CN_TASKTOPO_RESIDENT=FALSE
export USE_PAGED=false
export DATASET_PATH='/home/workspace/dataset/input.json'
#设置模型路径
MODEL_FULLNAME=${MODEL_PATH}
seqlen_input=8192
seqlen_output=4000

cd /home/workspace/vllm/benchmarks/
python3 benchmark_serving.py \
        --backend "vllm" \
        --model ${MODEL_FULLNAME} \
        --dataset ${DATASET_PATH} \
        --tokenizer ${MODEL_FULLNAME} \
        --trust-remote-code \
        --host "0.0.0.0" \
        --port 12345 \
        --endpoint '/v1/completions' \
        --sharegpt-output-len ${seqlen_output} \
        --num-prompts 10
```

3. 测试结果如下：

| Number | 问题 | S60 |
| --- | --- | --- |

| | | |
|---|---|---|
| 1 | RCDC远程唤醒开机终端功能介绍及配置方式 | RCDC远程唤醒开机终端功能' 是一种技术，允许管理员在办公桌面云V7.0R2版本和课堂桌面云V5.1_R1版本及以上版本中远程开机终端设备，以减少用户因终端开机占用的等待时间。此功能支持IDV、TCI（包括部分瘦终端和利旧PC）终端的局域网WOL唤醒和广域网WOW唤醒。在办公桌面云中，管理员可在RCDC界面的终端管理中操作单台或批量开机，并配置定时任务。在课堂桌面云中，支持学生机和教师机的唤醒，需要在教室设置开启VLAN并填写终端网段的VLAN号，同时要求终端的网卡支持WOL唤醒，交换机支持终端关机后协商接口工作为10M半双工。此功能需满足特定的硬件和网络条件，以确保远程唤醒的顺利进行。 |
| 2 | EG3220这个日志能保存六个月吗 | EG3220 在新版本11.1(6)B26 以后的版本默认存储内容审计日志时间为180天，即大约6个月。因此，EG3220 可以保存至少6个月的日志。要修改日志保存时间，可以通过命令行或Web界面进行调整。 |
| 3 | z5100支持\|2tpvpn吗 | 【提醒】知识库疑似无相关内容，请重新提问或咨询人工坐席。 |
| 4 | Z5100支持哪些VPN? | Z5100防火墙支持SSL VPN和IPsec VPN。具体配置请参考相关链接。 |
| 5 | WS6008默认可以管理几个AP | WS6008 默认可以管理 16 个 AP。 |
| 6 | WS6008是不是默认可以管理32个AP | 【提醒】知识库疑似无相关内容，请重新提问或咨询人工坐席。 |
| 7 | CS7015服务器,一个v520卡可以带多少终端 | CS7015服务器的每块V520和CG620显卡最多可以支持12个桌面使用。因此，一个V520卡可以带12个终端。 |
| 8 | RG-WALL 1600-Z3200-S,支持ipv6吗 | RG-WALL 1600-Z3200-S' 支持 IPv6。 |
| 9 | RG-CT7900硬盘内存 | RG-CT7900 系列终端的标准配置包括 8G 内存和 256G M.2 NVME 固态硬盘。最高配置可扩容至 4*8G 内存和 1T M.2 NVME 固态硬盘，或添加 2 个 2TB SATA 接口硬盘。适用于多种场景，包括内存和硬盘的扩容操作。具体扩容操作包括内存扩容、硬盘扩容及 PCIe 扩展卡添加。在进行拆装设备进行扩容操作前，请务必阅读安全说明。 |
| 10 | sam+最多可以注册多少用户 | 【提醒】知识库疑似无相关内容，请重新提问或咨询人工坐席。 |

```
============ Serving Benchmark Result ============
Successful requests:                     10
Benchmark duration (s):                  27.62
Total input tokens:                      35069
Total generated tokens:                  546
Request throughput (req/s):              0.36
Input token throughput (tok/s):          1269.89
Output token throughput (tok/s):         19.77
latency per token (ms):                  505.78
---------------Time to First Token----------------
Mean TTFT (ms):                          15596.60
Median TTFT (ms):                        16138.77
P99 TTFT (ms):                           26444.35
-----Time per Output Token (excl. 1st token)------
Mean TPOT (ms):                          42.81
Median TPOT (ms):                        43.02
P99 TPOT (ms):                           44.81
==================================================
```

对比测试结果详见：vllm测试用例

## 3.4 基于vLLM在线推理性能测试

1. 继续沿用上面容器,启动服务端。

```
# 启动server端服务
cd inference_scripts
./run_openai_api_server_xn.sh
```

附录：查看run_openai_api_server_xn.sh

```
cat run_openai_api_server_xn.sh
```

内容如下：

```
# 进入工作目录
CUR_DIR=$(cd $(dirname $0);pwd)
pushd "${CUR_DIR}/.."
#设置模型路径与服务端口
export MODEL_PATH="/home/workspace/models/Qwen2-7B-Instruct/"
# export MODEL_PATH="/home/workspace/models/Qwen2-7B-Instruct-GPTQ-Int4/"
export SERVER_PORT=12345
export USE_PAGED=false
```

```
export MODEL_Name="qwen2-7b-instruct"
export TP_NUM=1
seqlen_input=8192
seqlen_output=256
#使用以下命令启动vLLM-API服务器，启动服务器后，vLLM将自动加载模型并启动API服务。
#使用命令 python -m vllm.entrypoints.api_server --help 可查看支持的脚本参数。
export MAX_TOTAL_TOKENS=$(expr ${seqlen_input} + ${seqlen_output})
max_num_batched_tokens=$(expr ${MAX_TOTAL_TOKENS} \* 8)
max_model_len=32000
# block_size=$(expr ${seqlen_input} + 512)
block_size=16
#启动服务
python3 -m vllm.entrypoints.openai.api_server  \
        --host localhost \
        --port ${SERVER_PORT} \
        --block-size ${block_size} \
        --trust-remote-code \
        --dtype float16 \
        --enforce-eager \
        --model ${MODEL_PATH}  \
        --gpu-memory-utilization 0.9 \
        --tensor-parallel-size ${TP_NUM} \
        --disable-log-requests \
        --max-model-len ${max_model_len} \
        --max-num-batched-tokens ${max_num_batched_tokens} \
        --max-num-seqs 16
```

2. 打开另一个终端，并进入同个容器，启动客户端发送请求。

```
# 重新启动容器
docker start docker_ruijie_test

# 使用 exec 命令进入一个正在运行的容器
docker exec -it docker_ruijie_test /bin/bash

# 执行client请求
cd home/workspace/inference_scripts
./run_openai_api_client_xn.sh
```

附录：查看run_openai_api_client_xn.sh

```
cat run_openai_api_client_xn.sh
```

内容如下：

```
CMD_TIME=$(date +%Y%m%d%H%M%S.%N)
# 进入工作目录
CUR_DIR=$(cd $(dirname $0);pwd)
pushd "${CUR_DIR}/.."
if [ ! -d "./log" ]; then mkdir -p "./log";fi
popd
export MODEL_PATH="/home/workspace/models/Qwen2-7B-Instruct/"
# export MODEL_PATH="/home/workspace/models/Qwen2-7B-Instruct-GPTQ-Int4/"
# export MLU_VISIBLE_DEVICES="0"
export MODEL_Name="qwen2-7b-instruct"
# export TP_NUM=1
export VLLM_LATENCY_DEBUG=FALSE
export CN_TASKTOPO_RESIDENT=FALSE
export USE_PAGED=false
export DATASET_PATH='/home/workspace/dataset/ShareGPT_V3_unfiltered_cleaned_split.json'
#设置模型路径
MODEL_FULLNAME=${MODEL_PATH}
seqlen_output=256
seqlen_input=8192
max_model_len=32000
array_bs=(1 2 4)
array_seqlen=(8192)
for seqlen_input in "${array_seqlen[@]}"; do
    #不同条件不同赋值测试,减少部分测试
    case $seqlen_input in
        64)
            array_bs=(1 32 64 128 256)
            ;;
        128)
            #array_bs=(1 32 64 128)
            array_bs=(1 32 64 128 256)
            ;;
        256)
            #array_bs=(1 16 32 64)
            array_bs=(1 32 64 128 256)
            ;;
        512)
            #array_bs=(1 8 16 32)
            array_bs=(1 32 64 128 180 256)
            ;;
        1024)
            #array_bs=(1 4 8 16)
```

```
            array_bs=(1 16 32 64 90 128)
            ;;
        2048)
            array_bs=(1 8 16 32)
            ;;
        4096)
            array_bs=(1)
            ;;
        8192)
            array_bs=(1)
            # array_bs=(2)
            # array_bs=(4)
            # array_bs=(8)
            ;;
        *)
            array_bs=(1 2 4)
            ;;
    esac
    for bs in "${array_bs[@]}"; do
        # 0. 设置参数
        export MAX_TOTAL_TOKENS=$(expr ${seqlen_input} + ${seqlen_output})
        block_size=${MAX_TOTAL_TOKENS}
        max_num_batched_tokens=$(expr ${seqlen_input} \* $bs)
        if [ $max_model_len -gt $max_num_batched_tokens ]; then
            max_num_batched_tokens=$max_model_len
        fi
        LOG_FILENAME="${CUR_DIR}/../log/${MODEL_Name}_serving_${CMD_TIME}"
        # 1. 监控 cnmon 命令并将输出追加到日志文件中
        # while true; do cnmon -c ${MLU_VISIBLE_DEVICES} >>
"${LOG_FILENAME}_cnmon.log";sleep 1;done 2>&1 &
        PID_CNMONProcess=$!
        echo "PID_CNMONProcess: $PID_CNMONProcess"
        # 2. 性能测试
        cd /home/workspace/vllm/
        python3 benchmarks/benchmark_serving.py \
                --model ${MODEL_FULLNAME} \
                --dataset ${DATASET_PATH} \
                --tokenizer ${MODEL_FULLNAME} \
                --trust-remote-code \
                --host "localhost" \
                --port 12345 \
                --endpoint '/v1/completions' \
                --num-prompts ${bs} \
                --sharegpt-output-len ${seqlen_output} \
                --sharegpt-input-len ${seqlen_input} \
                2>&1 | tee "${LOG_FILENAME}.log"
```

```
        # 3. 删除【实时记录cnmon信息】的进程;# 使用 tail -f ./*_cnmon.log 可以实时查看日志文件的内容
        sleep 3
        kill -9 $PID_CNMONProcess
    done
done
sleep 1
#切换目录
pushd "${CUR_DIR}/.." && ls -la log/
```

3. 测试结果截图及记录：

a. batch size = 1； input_len = 8192； output_len = 256; FP16

```
root@test-hpc-05:/home/workspace/inference_scripts# ./run_openai_api_client_xn.sh
/home/workspace /home/workspace/inference_scripts
/home/workspace/inference_scripts
PID_CNMONProcess:
benchmarks/benchmark_serving.py:634: UserWarning: The '--dataset' argument will be deprecated in the next release. Please use '--dataset-name' and '--dataset-path' in the future runs.
  main(args)
Namespace(backend='vllm', base_url=None, best_of=1, dataset='/home/workspace/dataset/ShareGPT_V3_unfiltered_cleaned_split.json', dataset_name='sharegpt', dataset_path=None, disable_tqdm=False, endpoint='/v1/completions', host='localhost', metadata=None, model='/home/workspace/models/Qwen2-7B-Instruct/', num_prompts=1, port=12345, request_rate=inf, result_dir=None, save_result=False, seed=0, sharegpt_input_len=8192, sharegpt_output_len=256, sonnet_input_len=550, sonnet_output_len=150, sonnet_prefix_len=200, tokenizer='/home/workspace/models/Qwen2-7B-Instruct/', trust_remote_code=True, use_beam_search=False)
first 8417
Traffic request rate: inf
100%|██████████| 1/1 [00:14<00:00, 14.01s/it]
256
============ Serving Benchmark Result ============
Successful requests:                     1
Benchmark duration (s):                  14.01
Total input tokens:                      8192
Total generated tokens:                  256
Request throughput (req/s):              0.07
Input token throughput (tok/s):          584.53
Output token throughput (tok/s):         18.27
latency per token (ms):                  54.75
---------------Time to First Token----------------
Mean TTFT (ms):                          1539.03
Median TTFT (ms):                        1539.03
P99 TTFT (ms):                           1539.03
-----Time per Output Token (excl. 1st token)------
Mean TPOT (ms):                          48.92
Median TPOT (ms):                        48.92
P99 TPOT (ms):                           48.92
==================================================
```

b. batch size = 2； input_len = 8192； output_len = 256; FP16

```
root@test-hpc-05:/home/workspace/inference_scripts# ./run_openai_api_client_xn.sh
/home/workspace /home/workspace/inference_scripts
/home/workspace/inference_scripts
PID_CNMONProcess:
benchmarks/benchmark_serving.py:634: UserWarning: The '--dataset' argument will be deprecated in the next release. Please use '--data
set-name' and '--dataset-path' in the future runs.
  main(args)
Token indices sequence length is longer than the specified maximum sequence length for this model (221059 > 131072). Running this seq
uence through the model will result in indexing errors
Namespace(backend='vllm', base_url=None, best_of=1, dataset='/home/workspace/dataset/ShareGPT_V3_unfiltered_cleaned_split.json', data
set_name='sharegpt', dataset_path=None, disable_tqdm=False, endpoint='/v1/completions', host='localhost', metadata=None, model='/home
/workspace/models/Qwen2-7B-Instruct/', num_prompts=2, port=12345, request_rate=inf, result_dir=None, save_result=False, seed=0, share
gpt_input_len=8192, sharegpt_output_len=256, sonnet_input_len=550, sonnet_output_len=150, sonnet_prefix_len=200, tokenizer='/home/wor
kspace/models/Qwen2-7B-Instruct/', trust_remote_code=True, use_beam_search=False)
first 8417
first 8417
Traffic request rate: inf
100%|██████████| 2/2 [00:22<00:00, 11.06s/it]
256
256
============ Serving Benchmark Result ============
Successful requests:                     2
Benchmark duration (s):                  22.11
Total input tokens:                      16384
Total generated tokens:                  512
Request throughput (req/s):              0.09
Input token throughput (tok/s):          740.88
Output token throughput (tok/s):         23.15
latency per token (ms):                  86.38
---------------Time to First Token----------------
Mean TTFT (ms):                          7234.27
Median TTFT (ms):                        7234.27
P99 TTFT (ms):                           7234.76
-----Time per Output Token (excl. 1st token)------
Mean TPOT (ms):                          58.34
Median TPOT (ms):                        58.34
P99 TPOT (ms):                           58.34
==================================================
```

c. batch size = 4； input_len = 8192； output_len = 256; FP16

```
root@test-hpc-05:/home/workspace/inference_scripts# ./run_openai_api_client_xn.sh
/home/workspace /home/workspace/inference_scripts
/home/workspace/inference_scripts
PID_CNMONProcess:
benchmarks/benchmark_serving.py:634: UserWarning: The '--dataset' argument will be deprecated in the next release. Please use '--data
set-name' and '--dataset-path' in the future runs.
  main(args)
Token indices sequence length is longer than the specified maximum sequence length for this model (221059 > 131072). Running this seq
uence through the model will result in indexing errors
Namespace(backend='vllm', base_url=None, best_of=1, dataset='/home/workspace/dataset/ShareGPT_V3_unfiltered_cleaned_split.json', data
set_name='sharegpt', dataset_path=None, disable_tqdm=False, endpoint='/v1/completions', host='localhost', metadata=None, model='/home
/workspace/models/Qwen2-7B-Instruct/', num_prompts=4, port=12345, request_rate=inf, result_dir=None, save_result=False, seed=0, share
gpt_input_len=8192, sharegpt_output_len=256, sonnet_input_len=550, sonnet_output_len=150, sonnet_prefix_len=200, tokenizer='/home/wor
kspace/models/Qwen2-7B-Instruct/', trust_remote_code=True, use_beam_search=False)
first 8417
first 8417
Traffic request rate: inf
100%|██████████| 4/4 [00:33<00:00,  8.47s/it]
256
256
256
256
============ Serving Benchmark Result ============
Successful requests:                     4
Benchmark duration (s):                  33.88
Total input tokens:                      32768
Total generated tokens:                  1024
Request throughput (req/s):              0.12
Input token throughput (tok/s):          967.09
Output token throughput (tok/s):         30.22
latency per token (ms):                  132.36
---------------Time to First Token----------------
Mean TTFT (ms):                          12696.59
Median TTFT (ms):                        12335.23
P99 TTFT (ms):                           13738.15
-----Time per Output Token (excl. 1st token)------
Mean TPOT (ms):                          83.07
Median TPOT (ms):                        84.49
P99 TPOT (ms):                           84.49
==================================================
kill: usage: kill [-s sigspec | -n signum | -sigspec] pid | jobspec ... or kill -l [sigspec]
```

d. batch size = 8； input_len = 8192； output_len = 256; FP16

```
Token indices sequence length is longer than the specified maximum sequence length for this model (221059 > 131072). Running this seq
uence through the model will result in indexing errors
Namespace(backend='vllm', base_url=None, best_of=1, dataset='/home/workspace/dataset/ShareGPT_V3_unfiltered_cleaned_split.json', data
set_name='sharegpt', dataset_path=None, disable_tqdm=False, endpoint='/v1/completions', host='localhost', metadata=None, model='/home
/workspace/models/Qwen2-7B-Instruct/', num_prompts=8, port=12345, request_rate=inf, result_dir=None, save_result=False, seed=0, share
gpt_input_len=8192, sharegpt_output_len=256, sonnet_input_len=550, sonnet_output_len=150, sonnet_prefix_len=200, tokenizer='/home/wor
kspace/models/Qwen2-7B-Instruct/', trust_remote_code=True, use_beam_search=False)
first 8417
first 8417
Traffic request rate: inf
100%|██████████| 8/8 [01:02<00:00,  7.83s/it]
256
256
256
256
256
256
256
256
============ Serving Benchmark Result ============
Successful requests:                     8
Benchmark duration (s):                  62.65
Total input tokens:                      65536
Total generated tokens:                  2048
Request throughput (req/s):              0.13
Input token throughput (tok/s):          1046.01
Output token throughput (tok/s):         32.69
latency per token (ms):                  244.74
---------------Time to First Token----------------
Mean TTFT (ms):                          24935.95
Median TTFT (ms):                        32456.65
P99 TTFT (ms):                           32458.35
-----Time per Output Token (excl. 1st token)------
Mean TPOT (ms):                          147.89
Median TPOT (ms):                        118.39
P99 TPOT (ms):                           197.06
==================================================
kill: usage: kill [-s sigspec | -n signum | -sigspec] pid | jobspec ... or kill -l [sigspec]
```

e. batch size = 1； input_len = 8192； output_len = 256; INT4

S60对应的VLLM不支持GPTQ量化模型。

```
INFO 08-22 08:09:56 llm_engine.py:87] Initializing an LLM engine with config: model='/home/workspace/models/Qwen2-7B-Instruct-GPTQ-In
t4/', tokenizer='/home/workspace/models/Qwen2-7B-Instruct-GPTQ-Int4/', tokenizer_mode=auto, revision=None, tokenizer_revision=None, t
rust_remote_code=True, dtype=torch.float16, max_seq_len=32000, download_dir=None, load_format=auto, tensor_parallel_size=1, disable_c
ustom_all_reduce=False, quantization=gptq, enforce_eager=True, kv_cache_dtype=auto, device_config=gcu, seed=0)
00Traceback (most recent call last):
  File "/usr/lib/python3.8/runpy.py", line 194, in _run_module_as_main
    return _run_code(code, main_globals, None,
  File "/usr/lib/python3.8/runpy.py", line 87, in _run_code
    exec(code, run_globals)
  File "/usr/local/lib/python3.8/dist-packages/vllm/entrypoints/openai/api_server.py", line 236, in <module>
    engine = AsyncLLMEngine.from_engine_args(engine_args)
  File "/usr/local/lib/python3.8/dist-packages/vllm/engine/async_llm_engine.py", line 628, in from_engine_args
    engine = cls(parallel_config.worker_use_ray,
  File "/usr/local/lib/python3.8/dist-packages/vllm/engine/async_llm_engine.py", line 321, in __init__
    self.engine = self._init_engine(*args, **kwargs)
  File "/usr/local/lib/python3.8/dist-packages/vllm/engine/async_llm_engine.py", line 369, in _init_engine
    return engine_class(*args, **kwargs)
  File "/usr/local/lib/python3.8/dist-packages/vllm/engine/llm_engine.py", line 128, in __init__
    self._init_workers()
  File "/usr/local/lib/python3.8/dist-packages/vllm/engine/llm_engine.py", line 181, in _init_workers
    self._run_workers("load_model")
  File "/usr/local/lib/python3.8/dist-packages/vllm/engine/llm_engine.py", line 1041, in _run_workers
    driver_worker_output = getattr(self.driver_worker,
  File "/usr/local/lib/python3.8/dist-packages/vllm/worker/worker.py", line 101, in load_model
    self.model_runner.load_model()
  File "/usr/local/lib/python3.8/dist-packages/vllm/worker/model_runner.py", line 89, in load_model
    self.model = get_model(self.model_config,
  File "/usr/local/lib/python3.8/dist-packages/vllm/model_executor/utils.py", line 53, in get_model
    return get_model_fn(model_config, device_config, **kwargs)
  File "/usr/local/lib/python3.8/dist-packages/vllm/model_executor/model_loader.py", line 53, in get_model
    raise ValueError(
ValueError: The quantization method gptq is not supported for the current GPU. Minimum capability: 60. Current capability: 30.
```

f. batch size = 2； input_len = 8192； output_len = 256; INT4

S60对应的VLLM不支持GPTQ量化模型。

g. batch size = 4； input_len = 8192； output_len = 256; INT4

S60对应的VLLM不支持GPTQ量化模型。

h. batch size = 8；input_len = 8192；output_len = 256; INT4

S60对应的VLLM不支持GPTQ量化模型。

| | Output tokens_per_second(tps) | latency_per_token(ms) | first_token(ms) | Benchmark duration(s) |
|---|---|---|---|---|
| 1. batch size = 1<br>2. input_len = 8192<br>3. output_len = 256<br>4. FP16 | 18.27 | 48.92 | 1539.03 | 14.01 |
| 1. batch size = 2<br>2. input_len = 8192<br>3. output_len = 256<br>4. FP16 | 23.15 | 58.34 | 7234.27 | 22.11 |
| 1. batch size = 4<br>2. input_len = 8192<br>3. output_len = 256<br>4. FP16 | 30.22 | 83.07 | 12696.59 | 33.88 |
| 1. batch size = 8<br>2. input_len = 8192<br>3. output_len = 256<br>4. FP16 | 32.69 | 147.89 | 24935.95 | 62.65 |
| 1. batch size = 1<br>2. input_len = 8192<br>3. output_len = 256 | NA | NA | NA | NA |

| | | | | |
|---|---|---|---|---|
| 4. INT4 | | | | |
| 1. batch size = 2<br>2. input_len = 8192<br>3. output_len = 256<br>4. INT4 | NA | NA | NA | NA |
| 1. batch size = 4<br>2. input_len = 8192<br>3. output_len = 256<br>4. INT4 | NA | NA | NA | NA |
| 1. batch size = 8<br>2. input_len = 8192<br>3. output_len = 256<br>4. INT4 | NA | NA | NA | NA |

# 4. LLAMA2-7B拆解后的PyTorch子层的性能测试结果与分析

## 4.1 性能对比

该章节展示了，在batchsize=32, TP = 8, 4, 2的张量并行策略配置下，分别针对S60、K100_ai、V100和MLU370-X8单卡环境进行测试的结果。

### 4.1.1 TP = 8测试结果

下表展示了针对LLAMA2-7B大模型预训练中，TP = 8时不同平台下各个模块测试结果：

| 子层 | S60测试结果 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 平均时延(ms) | 标准差(ms) | 变异系数 | 浮点操作数(M) | 性能(TFLOPS) | 效率 | 平均时延(ms) | 标 |
| V2h | 5.242 | 0.278 | 5.31% | | | | 15.048 | |
| rmsNorm | 128.798 | 167.570 | 130.10% | | | | 16.728 | |
| Wqkv | 10.476 | 6.500 | 62.04% | 1649267.442 | 157.4274514 | 0.40 | 21.601 | |
| Rope | NA | NA | #VALUE! | | | | 1.727 | |
| QK | 3.392 | 0.124 | 3.65% | | | | 3.290 | |
| softmax | 23.879 | 5.764 | 24.14% | | | | 12.207 | |
| dropout | 24.181 | 5.936 | 24.55% | | | | 8.784 | |
| QKV | 3.202 | 0.111 | 3.46% | | | | 3.506 | |
| FC | 5.168 | 0.283 | 5.48% | 549755.8139 | 106.383445 | 0.27 | 15.982 | |
| Res add | 7.270 | 0.535 | 7.36% | | | | 5.291 | |
| rmsNorm | 128.879 | 167.783 | 130.19% | | | | 16.600 | |
| up | 10.232 | 5.942 | 58.08% | 1477468.75 | 144.3928317 | 0.37 | 20.806 | |
| gate | 10.291 | 6.398 | 62.18% | 1477468.75 | 143.5758529 | 0.37 | 20.848 | |
| silu | 1.787 | 0.034 | 1.93% | | | | 1.288 | |
| down | 9.295 | 3.139 | 33.77% | 1477468.75 | 158.9510996 | 0.41 | 18.326 | |
| h2V | 26.929 | 24.005 | 89.14% | 4294967.296 | 159.4938261 | 0.41 | 107.882 | 1 |
| SOFTMAX | 42.338 | 18.115 | 42.79% | | | | 45.596 | |
| Topk | 7.399 | 0.554 | 7.49% | | | | 2.358 | |

### 4.1.2 TP = 4测试结果

下表展示了针对LLAMA2-7B大模型预训练中，TP = 4时不同平台下各个模块测试结果：

| 子层 | S60测试结果 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 平均时延(ms) | 标准差(ms) | 变异系数 | 浮点操作数(M) | 性能(TFLOPS) | 效率 | 平均时延(ms) | 标准 |
| V2h | 5.331 | 0.288 | 5.41% | | | | 15.639 | 2 |
| rmsNorm | 128.874 | 167.767 | 130.18% | | | | 16.602 | 2 |
| Wqkv | 19.083 | 13.110 | 68.70% | 3298534.883 | 172.8503784 | 0.44 | 59.563 | 3 |
| Rope | NA | NA | #VALUE! | | | | 1.800 | 0 |
| QK | 3.403 | 0.124 | 3.65% | | | | 3.315 | 0 |
| softmax | 23.857 | 5.752 | 24.11% | | | | 12.194 | 1 |
| dropout | 24.164 | 5.925 | 24.52% | | | | 8.797 | 0 |
| QKV | 3.222 | 0.112 | 3.49% | | | | 3.483 | 0 |
| FC | 6.718 | 1.789 | 26.62% | 1099511.628 | 163.6555253 | 0.42 | 14.297 | 2 |
| Res add | 7.251 | 0.533 | 7.35% | | | | 5.187 | 0 |
| rmsNorm | 128.875 | 167.770 | 130.18% | | | | 16.605 | 2 |
| up | 21.160 | 46.761 | 220.99% | 2954937.5 | 139.6461551 | 0.36 | 41.374 | 1 |
| gate | 21.218 | 49.963 | 235.48% | 2954937.5 | 139.2687046 | 0.36 | 41.511 | 1 |
| silu | 3.598 | 0.144 | 4.00% | | | | 2.558 | 0 |
| down | 17.418 | 10.996 | 63.13% | 2954937.5 | 169.6527619 | 0.43 | 28.376 | 8 |
| h2V | 56.640 | 64.229 | 113.40% | 8589934.592 | 151.6595047 | 0.39 | 241.066 | 60 |
| SOFTMAX | 82.467 | 68.741 | 83.36% | | | | 91.104 | 8 |
| Topk | 9.786 | 0.969 | 9.90% | | | | 7.140 | 0 |

### 4.1.3 TP = 2测试结果

下表展示了针对LLAMA2-7B大模型预训练中，TP = 2时不同平台下各个模块测试结果：

| 子层 | S60测试结果 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 平均时延(ms) | 标准差(ms) | 变异系数 | 浮点操作数(M) | 性能(TFLOPS) | 效率 | 平均时延(ms) | 标准差 |
| V2h | 5.324 | 0.288 | 5.42% | | | | 16.050 | 2.6 |
| rmsNorm | 128.882 | 167.789 | 130.19% | | | | 16.603 | 2.8 |
| Wqkv | 37.631 | 42.340 | 112.52% | 6597069.767 | 175.3110627 | 0.45 | 124.383 | 161. |
| Rope | NA | NA | #VALUE! | | | | 1.680 | 0.0 |
| QK | 3.390 | 0.124 | 3.66% | | | | 3.320 | 0.1 |
| softmax | 23.879 | 5.763 | 24.13% | | | | 12.187 | 1.5 |
| dropout | 24.164 | 5.923 | 24.51% | | | | 8.811 | 0.7 |
| QKV | 3.218 | 0.112 | 3.48% | | | | 3.475 | 0.1 |
| FC | 13.134 | 6.290 | 47.89% | 2199023.256 | 167.4339249 | 0.43 | 36.019 | 13. |
| Res add | 7.256 | 0.534 | 7.35% | | | | 5.186 | 0.2 |
| rmsNorm | 128.866 | 167.749 | 130.17% | | | | 16.597 | 2.8 |
| up | 35.597 | 35.273 | 99.09% | 5909874.999 | 166.0229186 | 0.42 | 110.885 | 127. |
| gate | 35.608 | 38.437 | 107.94% | 5909874.999 | 165.9711691 | 0.42 | 111.053 | 127. |
| silu | 7.020 | 0.521 | 7.43% | | | | 5.108 | 0.3 |
| down | 33.701 | 32.383 | 96.09% | 5909874.999 | 175.3619238 | 0.45 | 58.101 | 36. |
| h2V | 110.739 | 136.769 | 123.51% | 17179869.18 | 155.1379076 | 0.40 | 325.373 | 1073 |
| SOFTMAX | 155.850 | 245.368 | 157.44% | | | | 176.441 | 316. |
| Topk | 14.250 | 2.052 | 14.40% | | | | 7.318 | 0.6 |

## 4.2 时延对比

1. 不同切分方式时延变化相同，下图以TP = 8为例，对比时延（时延比 = 燧原时延 除以 英伟达时延）

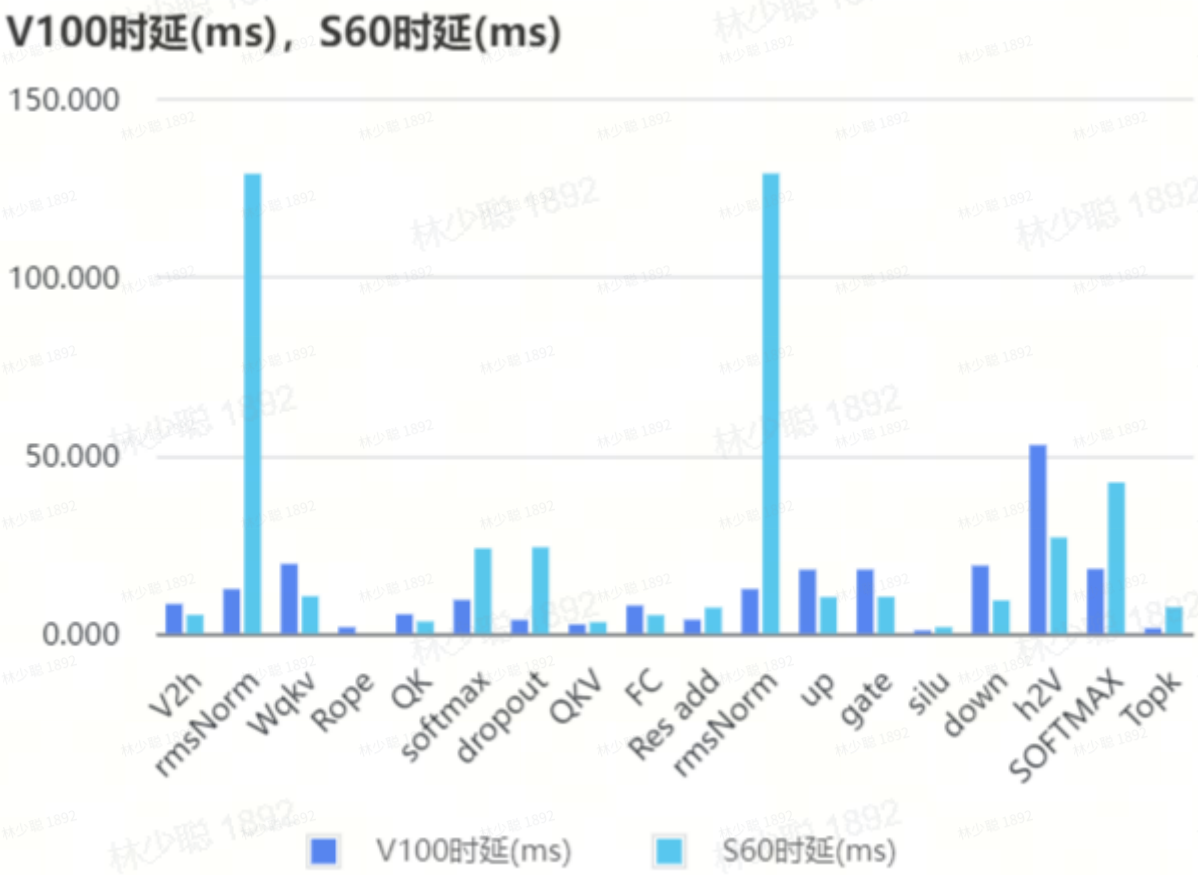
考虑V100算力为S60算力的112/392= 0.285倍，下表展示时延比并标注了大于1倍的子层：

| 子层名 | V100时延(ms) | S60时延(ms) | 时延比(燧原) |
| --- | --- | --- | --- |
| V2h | 8.287 | 5.242 | 0.633 |
| rmsNorm | 12.469 | 128.798 | 10.330 |
| Wqkv | 19.518 | 10.476 | 0.537 |
| Rope | 1.767 | NA | #VALUE! |
| QK | 5.406 | 3.392 | 0.627 |
| softmax | 9.435 | 23.879 | 2.531 |
| dropout | 3.831 | 24.181 | 6.312 |
| QKV | 2.587 | 3.202 | 1.238 |
| FC | 7.865 | 5.168 | 0.657 |
| Res add | 3.897 | 7.270 | 1.865 |
| rmsNorm | 12.469 | 128.879 | 10.336 |
| up | 17.894 | 10.232 | 0.572 |
| gate | 17.929 | 10.291 | 0.574 |
| silu | 0.913 | 1.787 | 1.958 |
| down | 19.045 | 9.295 | 0.488 |
| h2V | 52.805 | 26.929 | 0.510 |
| SOFTMAX | 18.108 | 42.338 | 2.338 |
| Topk | 1.521 | 7.399 | 4.866 |

备注：NA表示目前不支持该子层测试。

2. 可以看出，在batchsize=32, TP = 8测试条件下，S60在rmsNorm（10.3倍）、softmax（2.5倍）、droupout（6.3倍）、QKV（1.2倍）、Res add（1.8倍）、silu（1.9倍）、Topk（4.8倍）这些子层的时延比V100大。其他子层虽然比V100时延低，但考虑到2者之间的算力比差距，S60在各个子层的测试性能效率都低于V100。

   同时，画图展示两个平台的各子层时延对比：




# 5. 问题记录

## 5.1 问题1：S60对应的VLLM不支持GPTQ量化模型。

1. 问题描述：

```
INFO 08-22 08:09:56 llm_engine.py:87] Initializing an LLM engine with config: model='/home/workspace/models/Qwen2-7B-Instruct-GPTQ-In
t4/', tokenizer='/home/workspace/models/Qwen2-7B-Instruct-GPTQ-Int4/', tokenizer_mode=auto, revision=None, tokenizer_revision=None, t
rust_remote_code=True, dtype=torch.float16, max_seq_len=32000, download_dir=None, load_format=auto, tensor_parallel_size=1, disable_c
ustom_all_reduce=False, quantization=gptq, enforce_eager=True, kv_cache_dtype=auto, device_config=gcu, seed=0)
00Traceback (most recent call last):
  File "/usr/lib/python3.8/runpy.py", line 194, in _run_module_as_main
    return _run_code(code, main_globals, None,
  File "/usr/lib/python3.8/runpy.py", line 87, in _run_code
    exec(code, run_globals)
  File "/usr/local/lib/python3.8/dist-packages/vllm/entrypoints/openai/api_server.py", line 236, in <module>
    engine = AsyncLLMEngine.from_engine_args(engine_args)
  File "/usr/local/lib/python3.8/dist-packages/vllm/engine/async_llm_engine.py", line 628, in from_engine_args
    engine = cls(parallel_config.worker_use_ray,
  File "/usr/local/lib/python3.8/dist-packages/vllm/engine/async_llm_engine.py", line 321, in __init__
    self.engine = self._init_engine(*args, **kwargs)
  File "/usr/local/lib/python3.8/dist-packages/vllm/engine/async_llm_engine.py", line 369, in _init_engine
    return engine_class(*args, **kwargs)
  File "/usr/local/lib/python3.8/dist-packages/vllm/engine/llm_engine.py", line 128, in __init__
    self._init_workers()
  File "/usr/local/lib/python3.8/dist-packages/vllm/engine/llm_engine.py", line 181, in _init_workers
    self._run_workers("load_model")
  File "/usr/local/lib/python3.8/dist-packages/vllm/engine/llm_engine.py", line 1041, in _run_workers
    driver_worker_output = getattr(self.driver_worker,
  File "/usr/local/lib/python3.8/dist-packages/vllm/worker/worker.py", line 101, in load_model
    self.model_runner.load_model()
  File "/usr/local/lib/python3.8/dist-packages/vllm/worker/model_runner.py", line 89, in load_model
    self.model = get_model(self.model_config,
  File "/usr/local/lib/python3.8/dist-packages/vllm/model_executor/utils.py", line 53, in get_model
    return get_model_fn(model_config, device_config, **kwargs)
  File "/usr/local/lib/python3.8/dist-packages/vllm/model_executor/model_loader.py", line 53, in get_model
    raise ValueError(
ValueError: The quantization method gptq is not supported for the current GPU. Minimum capability: 60. Current capability: 30.
```

2. 问题回复：

后续版本会支持。

## 5.2 问题2：S60对应的pytorch不兼容cuda框架。

1. 问题描述：

```
root@test-hpc-05:/home/workspace/inference_scripts# python3 test_sublayer_czz.py
cpu 2.1.0+cpu
============batch test LLAMA2 7B --> TP =  8 ============================
Traceback (most recent call last):
  File "test_sublayer_czz.py", line 489, in <module>
    time_elapsed = call.layerTest()
  File "test_sublayer_czz.py", line 481, in layerTest
    op = v2h(self)
  File "test_sublayer_czz.py", line 22, in v2h
    start = tick()
  File "test_sublayer_czz.py", line 13, in tick
    torch.cuda.synchronize()
  File "/usr/local/lib/python3.8/dist-packages/torch/cuda/__init__.py", line 781, in synchronize
    _lazy_init()
  File "/usr/local/lib/python3.8/dist-packages/torch/cuda/__init__.py", line 289, in _lazy_init
    raise AssertionError("Torch not compiled with CUDA enabled")
AssertionError: Torch not compiled with CUDA enabled
root@test-hpc-05:/home/workspace/inference_scripts# python4
bash: python4: command not found
root@test-hpc-05:/home/workspace/inference_scripts# python3
Python 3.8.10 (default, Jul 29 2024, 17:02:10)
[GCC 9.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import os, time
>>> import numpy as np
>>> import torch
>>>
>>> import math
>>> import torch.cuda.nvtx as nvtx
>>> device = torch.device("cuda:0" if torch.cuda.is_available() else "cpu")
>>> print(device, torch.__version__)
cpu 2.1.0+cpu
```

2. 问题回复：

### 1. torch native NV->S60

SDXL 为例，从 NV GPU code 移植到 GCU，主要做了以下 2 处修改：

1. + import torch_gcu
2. .to("cuda") → .to("gcu")

```
import torch
import torch_gcu
from diffusers import DiffusionPipeline

pipe = DiffusionPipeline.from_pretrained(
      "./stable-diffusion-xl-base-1.0/",
       torch_dtype=torch.float16).to("gcu")

images = pipe(
    prompt="The collision of two black holes in the center of a galaxy.",
    height=512,
    width=512,
    num_images_per_prompt=1,
    num_inference_steps=30,
    guidance_scale=5.0
).images
```

| | | |
|---|---|---|
| torch.cuda.synchronize | Yes | replaced with torch.gcu.synchronize |

3. 问题解决

如下图所示：

a. 增加import torch_gcu

b. "cuda"改成"gcu"

```python
import os, time
import numpy as np
import torch
import math
# import torch.cuda.nvtx as nvtx
import torch_gcu

#from cuda import cudart
#stream = cudart.cudaProfilerStart()
# use cuda API
# device = torch.device("cuda:0" if torch.cuda.is_available() else "cpu")
device = "gcu"
print(device, torch.__version__)

def tick():
    #torch.cuda.synchronize()
    torch.gcu.synchronize()
    return time.time()

```

## 5.3 S60对应的pytorch版本无法将复数类型转换为某种内部数据类型。

1. 问题描述：

```
Traceback (most recent call last):
  File "test_sublayer_czz.py", line 497, in <module>
    time_elapsed = callLlama2.layerTest()
  File "test_sublayer_czz.py", line 347, in layerTest
    op = Rope(self)
  File "test_sublayer_czz.py", line 95, in Rope
    xq_out = torch.view_as_real(xq_ * freqs_cis).flatten(idx)
RuntimeError: false INTERNAL ASSERT FAILED at "/home/ci_build/jenkins/workspace/torch-gcu-release/torch_gcu/torch_gcu/csrc/pytorchbridge/gcu_utils.cpp":138, please report a bug to PyTorch. Cannot convert ScalarType ComplexFloat to topsatenDataType_t.
```

2. 问题回复：

FAE暂时未提供解决方案。

3. 问题解决：

通过屏蔽测试Rope函数解决。