PR #23912 完整报告

sgl-project/sglang

feat: tiny improve fp8_gemm tune usage

合并时间：2026-04-28 19:47

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/23912

执行摘要

- 一句话：改善 FP8 GEMM 调优脚本可用性与负载均衡
- 推荐动作：可直接合并。该 PR 虽小，但参数命名和负载均衡等改进符合作者提出的动机。建议后续考虑添加单元测试覆盖调优脚本的边界情况。

功能与动机

PR body 指出当前调优脚本存在两个问题：

1) 默认将较大批次大小分配给最后一个 GPU，导致 GPU 间负载不均衡； 2) 调优结果未按批次大小组织，手动验证困难。

实现拆解

1. 导入调整与变量清理：新增 import random，移除未使用的 results 字典变量。
2. 掩码参数重构：将 needs_masking 从硬编码关键字参数改为通过 extra_kernel_args 字典传递，仅对 FP8 内核设置该参数，避免对 INT8 内核传递无关参数。
3. 批次大小分布优化：在 distribute_batch_sizes 函数中添加 random.shuffle(batch_sizes)，使批次大小随机打乱后再按张量并行大小均匀分配给各 GPU，从而平衡负载。
4. CLI 参数重命名：将 --batch-size 改为 --batch-sizes，支持指定多个批次大小；移除单一批次时强制使用单 GPU 的逻辑，允许多 GPU 处理多个批次。
5. 结果排序：在 save_configs 函数中保存配置前对字典按键（批次大小）进行排序，方便人工审查。

关键文件：

- benchmark/kernels/quantization/tuning_block_wise_kernel.py（模块 调优脚本；类别 source；类型 dependency-wiring；符号 run, save_configs, tune_on_gpu, distribute_batch_sizes）：唯一变更文件，包含所有改进：参数重命名、随机 shuffle、结果排序、掩码参数重构。

关键符号：run, save_configs, tune_on_gpu, distribute_batch_sizes, main

关键源码片段

benchmark/kernels/quantization/tuning_block_wise_kernel.py

唯一变更文件，包含所有改进：参数重命名、随机 shuffle、结果排序、掩码参数重构。

```
# 在 distribute_batch_sizes 中添加 shuffle 以实现负载均衡
def distribute_batch_sizes(batch_sizes, num_gpus):
    """Distribute batch sizes across available GPUs."""
    # shuffle 使各 GPU 分配到的任务计算量更均衡，避免最后一个 GPU 承担最大批次
    random.shuffle(batch_sizes)
    batches_per_gpu = []
    for i in range(num_gpus):
        start_idx = i * len(batch_sizes) // num_gpus
        end_idx = (i + 1) * len(batch_sizes) // num_gpus
        batches_per_gpu.append(batch_sizes[start_idx:end_idx])
    return batches_per_gpu


# 在 save_configs 中添加排序，使输出按批次大小有序
def save_configs(config_file_path, configs):
    existing_configs = {}
    if os.path.exists(config_file_path):
        with open(config_file_path, "r") as f:
            existing_configs = json.load(f)
    # 将键转换为整数并排序，便于人工审查
    existing_configs = {int(k): v for k, v in existing_configs.items()}
    existing_configs.update(configs)
    existing_configs = dict(sorted(existing_configs.items()))
    with open(config_file_path, "w") as f:
        json.dump(existing_configs, f, indent=4)


# 在 benchmark 函数中，通过 extra_kernel_args 按需传递 needs_masking
extra_kernel_args = {}
if A.dtype == torch.float8_e4m3fnuz or A.dtype == torch.float8_e4m3fn:
    kernel = (
        _w8a8_block_fp8_matmul_unrolledx4
        if (_is_hip == True and num_workgroups <= get_device_core_count())
        else _w8a8_block_fp8_matmul
    )
    # 仅 FP8 内核需要 masking 标志，避免 INT8 内核接收无关参数
    extra_kernel_args["needs_masking"] = needs_masking
else:
    kernel = _w8a8_block_int8_matmul
```

评论区精华

该 PR 有 1 条机器人评论（每日配额限制），无人工 review 讨论。review 由 b8zhong 直接审批通过，无争议。

- 暂无高价值评论线程

风险与影响

- 风险：风险极低。变更仅影响 benchmark 脚本，不涉及 SRT 核心运行时。参数名变更 --batch-size -> --batch-sizes 是 breaking change，但该脚本非公共 API，仅内部使用。random shuffle 可能使结果略有波动，但调优目的是寻找最优配置，不影响确定性。
- 影响：影响范围仅限于使用该调优脚本的开发者。变更后需使用新参数名 --batch-sizes 运行脚本。负载均衡改进可略微提升多 GPU 调优效率。结果排序便于人工审查。无用户或系统影响。
- 风险标记：暂无

关联脉络

- 暂无明显关联 PR