PR #26845 完整报告

sgl-project/sglang

[Qwen3.5][AMD] Fix shared-expert xep_size over-count under allreduce-EP

合并时间：2026-06-04 14:04

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/26845

执行摘要

- 一句话：修复 Qwen3.5 在 AMD 上 EP 模式下共享专家权重重复累加
- 推荐动作：该 PR 是 AMD 平台的关键 bugfix，值得精读其根因分析方法和跨后端比较思维。建议后续为 Qwen2MoE 添加针对 EP 缩放系数的单元测试以避免回归。

功能与动机

Qwen3.5-397B-A17B-FP8 在 MI355X 上使用 TP=2 EP=2 时 GSM8K 准确率从 94.2%（纯 TP）骤降至 10.8%，且模型在 中陷入无限循环。PR body 通过实验定位为 fused shared expert 在 allreduce-EP 下被每个 rank 完整计算并 allreduce 求和导致。

实现拆解

1. 添加 is_deepep_class_backend 导入：在 qwen2_moe.py 中从 sglang.srt.layers.moe.utils 导入 is_deepep_class_backend。
2. 修改 _get_shared_expert_weights 方法：计算 sigmoid 权重后，若 moe_ep_size > 1 且不是 DeepEP 类后端，则将权重除以 moe_ep_size。DeepEP 后端（DeepEP、Mooncake、Mori）每个 rank 分配独立共享槽位，无 over-count，故豁免。
3. 注释说明：在代码中添加详尽注释解释根因与修正逻辑，引用 FusedMoE 初始化和 DeepSeek-V2 的先例。
4. 测试配套：未新增单元测试，但 PR body 引用已有的 Qwen3.5 GSM8K 测试覆盖 EP 组合。

关键文件：

- python/sglang/srt/models/qwen2_moe.py（模块 模型层；类别 source；类型 data-contract；符号 _get_shared_expert_weights, Qwen2MoeSparseMoeBlock）：模型前向入口，实现 shared expert 权重缩放的核心修复

关键符号：_get_shared_expert_weights

关键源码片段

python/sglang/srt/models/qwen2_moe.py

模型前向入口，实现 shared expert 权重缩放的核心修复

```
def _get_shared_expert_weights(self, hidden_states: torch.Tensor) -> torch.Tensor:
    """Return sigmoid(shared_expert_gate) for fused shared expert weights."""
```

```
    if not self.enable_shared_expert_fusion or self.shared_expert_gate is None:
        return None
    shared_out = self.shared_expert_gate(hidden_states)
    shared_logits = shared_out[0] if isinstance(shared_out, tuple) else shared_out
    w = F.sigmoid(shared_logits)

    # 当使用 allreduce-EP 且 fused shared expert 为全局单 slot 时
    # （is_deepep_class_backend() == False），每个 EP rank 独立
    # 计算完整共享输出，后续 allreduce 会求和 ep_size 次。
    # 此处将权重除以 ep_size 以抵消该重复累加，同 DeepSeek-V2 的
    # fused_shared_experts_scaling_factor 模式。
    moe_ep_size = get_moe_expert_parallel_world_size()
    if moe_ep_size > 1 and not is_deepep_class_backend():
        w = w / float(moe_ep_size)
    return w
```

评论区精华

HaiShaw 询问是否还有其他模型需要修正，alexsun07 回应仅影响 Qwen3.5，因为 Kimi/GLM/DeepSeek 已使用 deepseek_v2 路径且早已修复。gemini-code-assist[bot] 建议使用 get_moe_expert_parallel_world_size() 替代 self.experts.moe_ep_size 以提高健壮性——实际实现已采用该建议。

- 是否其他模型也需要修复 (question): alexsun07 回应仅 Qwen3.5 受影响，因为 Kimi/GLM/DeepSeek 使用 deepseek_v2 路径且已修复
- 使用 get_moe_expert_parallel_world_size 替代 self.experts.moe_ep_size (design): 已采纳，实际实现使用 get_moe_expert_parallel_world_size()

风险与影响

- 风险：该修复作用于 shared expert 权重缩放，仅在 moe_ep_size > 1 且非 DeepEP 时生效，对纯 TP 模式无影响。缺少针对非 AMD 后端（CUDA）的测试验证，但逻辑与 DeepSeek-V2 已有模式对齐，回归风险低。未新增单元测试，依赖已有集成测试（GSM8K），可能遗漏边界条件。
- 影响：直接影响：修复 Qwen3.5 在 AMD GPU 上使用 allreduce-EP 时生成质量崩溃。间接影响：Qwen2MoE 系列模型（Qwen3.5 基于此）同样受益。系统影响：无 API 或部署变更。性能：略高于 disable shared-expert-fusion 方案（706 vs 656 tok/s）。
- 风险标记：核心路径变更，缺少测试覆盖

关联脉络

- PR #20736 [AMD] Enable shared expert fusion for Qwen3.5: 引入了该问题的前一版本，使得 shared expert 在 AMD AITER 路径上可用