PR #23062 完整报告

sgl-project/sglang

[bugfix]fix(qwen3_5): broadcast per-tensor scale in _make_packed_weight_loader for FP8 models

合并时间：2026-04-30 14:16

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/23062

执行摘要

- 一句话：修复 Qwen3.5 FP8 per-tensor 量化权重加载崩溃
- 推荐动作：值得精读。本 PR 虽改动量小，但精准解决了量化模型权重加载中的语义差异：per-tensor scale 是全局标量应当广播，而非常规张量的分割。Review 建议的代码合并方式（统一 numel()==1 分支）也值得借鉴，它消除了断言限制并提升了可读性。新增测试的设计思路（直接 mock 类方法和参数）可作为类似测试的模板。

功能与动机

加载 FP8 per-tensor 量化 Qwen3.5 模型时，由于 _make_packed_weight_loader 对 per-tensor scale 执行 torch.split 而非广播，导致 RuntimeError: split_with_sizes expects split_sizes to sum exactly to 1, but got 3。这是 0.5.10 中 GatedDeltaNet fused projection 重构引入的回归（关联 Issue #23051）。

实现拆解

1. 修改 _make_packed_weight_loader 内部 weight_loader：将原来针对标量的 len(shape)==0 分支替换为 loaded_weight.numel() == 1 通用分支，移除限制单 shard 的断言。使用 [loaded_weight.view(-1)] * len(loaded_shard_id) 将 per-tensor scale 广播至所有逻辑 shard。
2. 保持正常权重 split 逻辑不变：对于多维度的权重张量，仍按 split_sizes 在 output_dim 维度上分割。
3. 新增单元测试文件 test_qwen3_5_packed_weight_loader.py：通过 mock 模块和 PerTensorScaleParameter，覆盖标量广播、单元素张量广播、两 shard 广播以及正常权重分割等场景，并注册为 CPU-only CI 测试（suite stage-a-test-cpu）。

关键文件：

- python/sglang/srt/models/qwen3_5.py（模块 模型加载；类别 source；类型 core-logic；符号 Qwen3_5GatedDeltaNet._make_packed_weight_loader, weight_loader (inner)）：核心修改文件：修复 _make_packed_weight_loader 中 per-tensor scale 的广播逻辑，是本次 bugfix 的关键位置
- test/registered/unit/models/test_qwen3_5_packed_weight_loader.py（模块 单元测试；类别 test；类型 test-coverage；符号 _make_mock_module, _make_per_tensor_scale_param, TestMakePackedWeightLoader,

test_scalar_weight_broadcast）：新增的单元测试文件，覆盖了 broadcast 和 split 的主要场景，是保证代码质量的重要配套

关键符号：Qwen3_5GatedDeltaNet._make_packed_weight_loader, weight_loader (inner)

关键源码片段

python/sglang/srt/models/qwen3_5.py

核心修改文件：修复 _make_packed_weight_loader 中 per-tensor scale 的广播逻辑，是本次 bugfix 的关键位置

```
def weight_loader(param, loaded_weight, loaded_shard_id=None):
    # Only intercept tuple shard_ids (split checkpoint) ;
    # int or None pass through to original loader.
    if isinstance(loaded_shard_id, tuple):
        split_sizes = cls._get_split_sizes_for_param(
            module, param, loaded_shard_id
        )

        # Per-tensor scale (scalar or [1]) should be broadcast to every
        # logical shard, not split .
        if loaded_weight.numel() == 1:
            # view(-1) unifies scalar ( []) and single-element ( [1]) shapes
            # into a 1-D tensor of size 1 .
            chunks = [loaded_weight.view(-1)] * len(loaded_shard_id)
        else:
            # Normal multi-element weight: split along output dimension .
            split_dim = getattr(param, "output_dim", 0)
            if _is_cpu:
                # CPU padding handling ...
                pass
            else:
                chunks = loaded_weight.split(split_sizes, dim=split_dim)

        assert len(chunks) == len(loaded_shard_id), (
            f"Chunk/shard mismatch: {len(chunks)=}, "
            f"{len(loaded_shard_id)=}, {split_sizes=}"
        )

        for idx, chunk in zip(loaded_shard_id, chunks):
            original_weight_loader(param, chunk, idx)
        return

    # Fall through to original loader for int/None shard_id .
    return original_weight_loader(param, loaded_weight, loaded_shard_id)
```

评论区精华

Review 中 gemini-code-assist[bot] 建议将标量和单元素张量的处理统一为使用 numel()==1 和 view(-1)，以提高健壮性并避免形状问题。alexnails 表示同意该建议，并额外要求添加单元测试。kkyyxhll 采纳代码建议并提交了包含 7 个测试用例的单元测试文件，最终获得 Approved。

- 统一标量和单元素张量处理为 numel() == 1 (design): 采纳，kkyyxhll 提交新的 commit 实现了推荐方案。
- 请求添加单元测试 (testing): kkyyxhll 添加了 test/registered/unit/models/test_qwen3_5_packed_weight_loader.py，包含 7 个测试用例。

风险与影响

- 风险：核心风险较低：改动仅影响权重加载时进入 tuple shard_id 且 numel()==1 的分支，该分支原为标量专用（现已删除断言），非量化权重（多元素）不受影响。新增的 view(-1) 确保了统一形状输出，避免了之前标量分支对 assert len(split_sizes)==1 的依赖。潜在风险是未来若有其他单元素 checkpoint 张量（如 [1,1]）需 split 而非 broadcast，会误入该分支；但目前所有场景中 per-tensor scale 均需 broadcast，语义正确。测试覆盖了主要场景，CI 已通过。
- 影响：直接影响：修复了 FP8 per-tensor 静态量化 Qwen3.5 模型（dense 和 MoE）在 v0.5.10 后的加载失败，使其恢复正常运行。对非量化 BF16 模型无任何影响（权重矩阵多元素，不进入 numel()==1 分支）。间接影响：新增的单元测试可防止日后对该位置的修改引入回归。团队可参考此模式处理类似量化参数加载问题。
- 风险标记：回归修复，核心路径变更，测试覆盖已添加

关联脉络

- PR #24095 [misc] fix lint in main branch: 本 PR 合并 main 分支以包含该 lint 修复，确保 CI 通过。