# PR #37448 完整报告

vllm-project/vllm

Fix AttributeError in Qwen3.5 GDN layers with quantized models

合并时间：2026-03-20 07:21

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/37448

## 执行摘要

- 一句话：修复 Qwen3.5 GDN 层在量化模型下因 MergedColumnParallelLinear 无 weight 属性而抛出的 AttributeError。
- 推荐动作：建议关注 Qwen 模型维护和量化支持的开发者精读此 PR，以了解 MergedColumnParallelLinear 在量化时的属性访问差异和形状计算调整。变更虽小，但揭示了量化层与标准线性层之间的重要设计权衡。

## 功能与动机

Issue #37444 报告在使用量化模型 cyankiwi/Qwen3.5-9B-AWQ-4bit 时，模型加载失败并抛出 AttributeError: 'MergedColumnParallelLinear' object has no attribute 'weight'。这是由 PR #36795 引入的回归，该 PR 优化了 Qwen3 的输入投影但错误地假设 MergedColumnParallelLinear 始终有 .weight 属性，而量化时权重由内核管理，无此属性。

## 实现拆解

1. 定位问题文件：在 vllm/model_executor/models/qwen3_5.py 和 vllm/model_executor/models/qwen3_next.py 的 forward 方法中，torch.ops.vllm.gdn_in_proj 调用访问了 self.in_proj_qkvz.weight.shape[0] 和 self.in_proj_ba.weight.shape[0]。
2. 替换形状计算：将上述访问改为 sum(self.in_proj_qkvz.output_sizes) // self.tp_size 和 sum(self.in_proj_ba.output_sizes) // self.tp_size，因为 MergedColumnParallelLinear 的 output_sizes 属性在量化时仍可用，且除以 tp_size 是张量并行下的必要调整。
3. 原因与影响：修复了量化模型下的 AttributeError，同时保持非量化模型兼容；未引入测试或配置改动，但需确保形状计算与模型其他部分一致。

关键文件：

- vllm/model_executor/models/qwen3_5.py（模块 模型执行；类别 source；类型 core-logic；符号 forward）：Qwen3.5 模型的核心实现文件，包含导致 AttributeError 的 forward 方法，修复后确保量化兼容性。
- vllm/model_executor/models/qwen3_next.py（模块 模型执行；类别 source；类型 core-logic；符号 forward）：Qwen3 Next 模型的类似实现文件，同样修复了相同的 AttributeError，保持代码一致性。

关键符号：forward

## 关键源码片段

### vllm/model_executor/models/qwen3_5.py

Qwen3.5 模型的核心实现文件，包含导致 AttributeError 的 forward 方法，修复后确保量化兼容性。

```
def forward(
    self,
    hidden_states: torch.Tensor,
    output: torch.Tensor,
):
    # ... 其他代码省略
    mixed_qkvz, ba = torch.ops.vllm.gdn_in_proj(
        hidden_states,
        sum(self.in_proj_qkvz.output_sizes) // self.tp_size, # 修复：使用 output_sizes 替代 weight，并除以 tp_size 以适配张量并行
        sum(self.in_proj_ba.output_sizes) // self.tp_size, # 同上，确保量化模型下无 AttributeError
        self.prefix,
    )
    # ... 后续代码省略
```

### vllm/model_executor/models/qwen3_next.py

Qwen3 Next 模型的类似实现文件，同样修复了相同的 AttributeError，保持代码一致性。

```
def forward(
    self,
    hidden_states: torch.Tensor,
    output: torch.Tensor,
):
    # ... 其他代码省略
    projected_states_qkvz, projected_states_ba = torch.ops.vllm.gdn_in_proj(
        hidden_states,
        sum(self.in_proj_qkvz.output_sizes) // self.tp_size, # 修复：避免量化时无 weight 属性，使用 output_sizes 并除以 tp_size
        sum(self.in_proj_ba.output_sizes) // self.tp_size, # 同上，确保形状匹配和推理正确性
        self.prefix,
    )
    # ... 后续代码省略
```

## 评论区精华

Review 中 xyang16 指出原始修复中 sum(self.in_proj_qkvz.output_sizes) 需要除以 self.tp_size 以避免形状不匹配，参考了代码库类似实现。此建议被采纳，最终变更添加了 // self.tp_size，确保张量并行下的正确性。

- 形状参数需要除以 tp_size 以避免不匹配 (correctness): 建议被采纳，最终代码中添加了 // self.tp_size 来正确计算输出维度。

## 风险与影响

- 风险：风险较低，主要涉及形状计算：若 output_sizes 不正确或除以 tp_size 逻辑有误，可能导致张量形状不匹配或推理错误。由于是回归修复，风险是引入新的形状问题，但变更简单且基于现有属性，回归可能性小。
- 影响：直接影响是修复了使用 AWQ、FP8 等量化方法的 Qwen3.5 和 Qwen3 Next 模型的加载和推理崩溃，恢复功能。间接影响是提高了量化兼容性，确保优化后的 gdn_in_proj 操作在量化设置下正常工作，影响范围限于这些模型及其用户。
- 风险标记：形状计算错误，量化兼容性

## 关联脉络

- PR #36795 [Perf] Enable dual stream execution of input projection for Qwen3: 引入了 gdn_in_proj 自定义操作并错误地访问 .weight.shape[0]，导致本 PR 修复的回归问题。