PR #42250 完整报告

vllm-project/vllm

[Bugfix][Model] Gemma4 MoE routing closure captures per_expert_scale, breaking functional_call substitution

合并时间：2026-05-14 01:43

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/42250

执行摘要

- 一句话：修复 Gemma4 MoE 路由闭包捕获参数问题
- 推荐动作：值得精读，特别是了解 Python 闭包捕获与 PyTorch functional API 交互的常见陷阱。该 PR 展示了如何通过避免变量捕获来确保参数替换正确工作。

功能与动机

Issue #42239 报告：Gemma4MoE.__init__中的 routing_function 闭包捕获了 per_expert_scale Parameter 的本地引用，导致 torch.func.functional_call 无法替换该参数，影响依赖 functional_call 的模块替换场景（如测试或模型优化）。

实现拆解

1. 移除本地捕获：删除 Gemma4MoE.__init__ 中的 per_expert_scale = self.per_expert_scale 行。
2. 运行时读取：将闭包 routing_function 内部所有对 per_expert_scale 的引用替换为 self.per_expert_scale。
3. 添加注释：在闭包定义前增加说明，解释为什么不能捕获本地变量，方便后续维护。 仅涉及文件 vllm/model_executor/models/gemma4.py，总差异 +7 / -4 行。

关键文件：

- vllm/model_executor/models/gemma4.py（模块 模型层；类别 source；类型 bugfix；符号 Gemma4MoE, init, routing_function）：核心修复文件，修改了 Gemma4MoE 类的 __init__方法，调整了 routing 闭包的参数捕获方式。

关键符号：Gemma4MoE.init, routing_function

关键源码片段

vllm/model_executor/models/gemma4.py

核心修复文件，修改了 Gemma4MoE 类的 __init__方法，调整了 routing 闭包的参数捕获方式。

```
class Gemma4MoE(nn.Module):
    def __init__(self, config, quant_config=None, prefix=""):
        super().__init__()
```

```
self.hidden_size = config.hidden_size
self.num_experts = config.num_experts
# 每个专家的输出缩放因子，融合到路由权重中
self.per_expert_scale = nn.Parameter(torch.ones(config.num_experts))

# 注意：此处直接在闭包内通过 self 访问 per_expert_scale，
# 而不是先捕获到本地变量。这样做是为了确保
# torch.func.functional_call 的参数替换能生效。
def routing_function(
    hidden_states, gating_output, topk, renormalize
):
    if current_platform.is_cuda_alike() or current_platform.is_xpu():
        return gemma4_fused_routing_kernel_triton(
            gating_output, topk, self.per_expert_scale
        )
    return gemma4_routing_function_torch(
        gating_output, topk, self.per_expert_scale
    )

self.experts = FusedMoE(
    num_experts=config.num_experts,
    top_k=config.top_k_experts,
    hidden_size=config.hidden_size,
    intermediate_size=getattr(config, "moe_intermediate_size", None),
    renormalize=True,
    quant_config=quant_config,
    prefix=f"{prefix}.experts",
    custom_routing_function=routing_function,
    activation="gelu_tanh",
)
```

评论区精华

- yewentao256：认为改动较小，无需添加专用单元测试。作者同意并移除了测试文件中的新增测试（最终合并仅包含源码改动）。
- Copilot：检测到测试文件中存在未使用的变量 orig_ids，可能触发 lint 错误。作者已修复。
  - Gemini Code Assist和 Copilot的自动 review 均确认了修改的正确性。
- 是否需要新增单元测试 (testing): 移除了测试，仅保留源码修改。
- 未使用变量 orig_ids (style): 作者已修复该问题。

风险与影响

- 风险：风险极低。改动本质上将闭包捕获的变量引用改为从 self 属性读取，逻辑等价。唯一的假设是 self.per_expert_scale 在闭包调用时不会变化（与原始捕获语义一致）。被移除的测试原本用于验证 functional_call 替换，回归风险较小，但 reviewer 认为改动足够简单直接。

- 影响：影响范围小，仅影响 Gemma4 MoE 模块中使用 torch.func.functional_call 的场景。普通推理路径不受任何影响。
- 风险标记：闭包捕获行为改变，无新增测试覆盖

关联脉络

- 暂无明显关联 PR