PR #42373 完整报告

vllm-project/vllm

fix: MoE model using shared routed experts crashes on AMD GPUs

合并时间：2026-05-25 12:03

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/42373

执行摘要

- 一句话：修复 ROCm 上 MoE 模型导入路径崩溃
- 推荐动作：值得合并。这是由上游重构引入的回归 bug，修复简洁且经过验证。建议通知 CI 加入对类似 import 路径的静态检查，防止未来重构再次遗漏。

功能与动机

加载 MoE 模型（如 DeepSeek-V2/V3、Mixtral）时，若启用 AITER fused MoE 且使用共享 routed expert 路径，会在模型加载时因 ModuleNotFoundError: No module named 'vllm.model_executor.layers.fused_moe.rocm_aiter_fused_moe' 崩溃。根本原因是 PR #41979 重构 fused_moe 目录时将 rocm_aiter_fused_moe.py 移入 experts/ 子目录并重命名，但漏掉了两处 lazy import 的更新。

实现拆解

1. 更新 lazy import 路径：在 vllm/model_executor/layers/fused_moe/router/aiter_shared_routed_fused_moe_router.py 的 _compute_routing 方法中，将两处来自 vllm.model_executor.layers.fused_moe.rocm_aiter_fused_moe 的 import 改为 vllm.model_executor.layers.fused_moe.experts.rocm_aiter_moe。
2. 变更点：文件仅修改了两行代码，每处将旧模块路径替换为新的 expert 子模块路径，分别导入 aiter_topK_meta_data 和 inject_shared_expert_weights。
3. 测试覆盖：本次变更未新增或修改测试文件，但 review 中获得了 ROCm 团队成员的验证承诺（maeehart: We can validate that this works）。

关键文件：

- vllm/model_executor/layers/fused_moe/router/aiter_shared_routed_fused_moe_router.py（模块 MoE 路由器；类别 source；类型 data-contract）：包含两处 stale lazy import，导致 ROCm AITER MoE 模型加载时崩溃。修复将旧路径更新为新的 expert 子模块路径。

关键符号：_compute_routing

关键源码片段

vllm/model_executor/layers/fused_moe/router/aiter_shared_routed_fused_moe_router.py

包含两处 stale lazy import，导致 ROCm AITER MoE 模型加载时崩溃。修复将旧路径更新为新的 expert 子模块路径。

```
# 文件：vllm/model_executor/layers/fused_moe/router/aiter_shared_routed_fused_moe_router.py
# 在 _compute_routing 方法中，两次 lazy import 都从旧路径更新为新路径

def _compute_routing(self, hidden_states, router_logits, indices_type, *, input_ids=None):
    # ... 前面的逻辑 ...

    # 第一次 import：位于 fuse_sigmoid_in_kernel 分支之前（第 76 行）
    # 旧路径：from vllm.model_executor.layers.fused_moe.rocm_aiter_fused_moe import aiter_
    topK_meta_data
    # 新路径：
    from vllm.model_executor.layers.fused_moe.experts.rocm_aiter_moe import (
        aiter_topK_meta_data,
    )
    # 注意：该处位于函数顶部，但实际执行取决于后续 if 分支

    # ... 中间逻辑 ...

    # 如果 aiter_topK_meta_data 不为 None，则注入共享 expert 权重（第 129-131 行）
    if aiter_topK_meta_data is not None:
        # 第二次 import：位于另一个条件分支内
        # 旧路径：from vllm.model_executor.layers.fused_moe.rocm_aiter_fused_moe import inject_
        shared_expert_weights
        # 新路径：
        from vllm.model_executor.layers.fused_moe.experts.rocm_aiter_moe import (
            inject_shared_expert_weights,
        )
        # 使用注入逻辑 ...
        shared_weights = torch.sigmoid(shared_logits)
        topk_weights, topk_ids = inject_shared_expert_weights(
            topk_weights, topk_ids, topk=topk,
            num_fused_shared_experts=num_fse,
            shared_expert_weights=shared_weights,
        )
```

评论区精华

gemini-code-assist[bot] 提出了合并 import 以减少性能开销的建议，但作者 weizhoublue 以 not true 简洁驳回。由于两处 import 位于不同的执行分支（if rocm_aiter_ops.fuse_sigmoid_in_kernel(...) 和 if aiter_topK_meta_data is not None），它们不会同时执行，因此合并并无实际收益。该建议被正确驳回。

- 合并两处 import 以减少性能开销 (performance): 作者回复 "not true" 拒绝。由于两处 import 位于不同条件分支（一个在 fuse_sigmoid_in_kernel 为 true 的分支前，另一个在 aiter_topK_meta_data is not None 的分支），不会同时执行，因此合并无法带来收益。

风险与影响

- 风险：风险极低。变更仅涉及两处 import 路径字符串，逻辑完全不变，且已获得 bnellnm 和 tjtanaa 的 approve。主要风险是有人手动 reverting PR #41979 后该路径再次失效，但这是协调问题而非代码风险。
- 影响：对 ROCm 平台上使用 AITER fused MoE 且启用共享 routed expert 的所有 MoE 模型（DeepSeek-V2/V3、Mixtral 等）是关键 bugfix，直接影响这些模型能否正常加载。不涉及用户接口或 API 变更。
- 风险标记：缺失测试覆盖

关联脉络

- PR #41979 [MoE] Migrate ROCm Aiter to oracle kernel setup（推测）: 本 PR 修复的是 PR #41979 重构引入的回归，该重构将 rocm_aiter_fused_moe.py 移入 experts/ 子目录并重命名，但未更新所有引用。