# PR #45047 完整报告

vllm-project/vllm

[Bugfix] Fix Llama4 weight loading

合并时间：2026-06-11 01:40

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/45047

## 执行摘要

- 一句话：修复 Llama4 等模型权重加载 KeyError
- 推荐动作：建议精读此 PR，尤其是 maybe_remap_moe_expert_param_name 的泛化技巧，以及如何通过单点修改覆盖多个模型。可作为 MoE 重构后兼容性补丁的典型范例。

## 功能与动机

MoE 重构后，部分模型权重加载因参数名不匹配而崩溃。PR body 明确指出 Nightly CI 构建 #70731 中 Llama-4-Scout-Fp8-ModelOpt 服务器因 KeyError: 'layers.0.feed_forward.experts.w2_input_scale' 崩溃，原因是 remap 函数只处理 .mlp.experts 路径，而 Llama4 使用 .feed_forward.experts。

## 实现拆解

1. 泛化重映射条件（weight_utils.py）：将 maybe_remap_moe_expert_param_name 中的条件从硬编码的 .mlp.experts 改为更通用的 .experts，并同步更新 docstring 示例。
2. 在 llama4.py 中添加重映射调用：在 load_weights 中处理 expert scale 参数的分支里，调用 maybe_remap_moe_expert_param_name 对 name 进行转换，避免 params_dict[name] 抛出 KeyError。
3. 在 lfm2_moe.py 中添加重映射调用：在 load_weights 的 else 分支中，对非 stacked 权重调用 maybe_remap_moe_expert_param_name 以正确映射名称。
4. 在 mllama4.py 中添加重映射调用：在 _handle_expert_scale_broadcasting 中，对包含 feed_forward.experts 的 scale 参数先执行重映射，再访问 params_dict。
5. 更新各文件的 import：在 lfm2_moe.py、mllama4.py、llama4.py 中增加 maybe_remap_moe_expert_param_name 的导入。无测试文件变更。

关键文件：

- vllm/model_executor/model_loader/weight_utils.py（模块 权重加载；类别 source；类型 data-contract）：核心修复文件：将重映射条件从 .mlp.experts 泛化为 .experts，使函数支持 feed_forward.experts 等变体，同时保留旧行为。
- vllm/model_executor/models/llama4.py（模块 模型定义；类别 source；类型 data-contract）：在 llama4.py 的 load_weights 中，处理 expert scale 的分支里添加了重映射调用，这是修复的主场景。

- vllm/model_executor/models/lfm2_moe.py（模块 模型定义；类别 source；类型 data-contract）：在 lfm2_moe.py 的 load_weights else 分支中添加重映射调用，确保非 stacked 权重正确加载。
- vllm/model_executor/models/mllama4.py（模块 模型定义；类别 source；类型 data-contract）：在 mllama4.py 的 _handle_expert_scale_broadcasting 中添加重映射调用，修复视觉语言模型 variant。

关键符号：未识别

## 关键源码片段

### vllm/model_executor/model_loader/weight_utils.py

核心修复文件：将重映射条件从 .mlp.experts 泛化为 .experts，使函数支持 feed_forward.experts 等变体，同时保留旧行为。

```
# vllm/model_executor/model_loader/weight_utils.py

def maybe_remap_moe_expert_param_name(
    name: str,
    params_dict: dict[str, torch.nn.Parameter],
) -> str:
    """
    将旧格式的 expert 参数名映射到新格式。
    旧格式: layers.0.feed_forward.experts.w2_input_scale
    新格式: layers.0.feed_forward.experts.routed_experts.w2_input_scale

    注意：这里使用 .experts. 而不是 .mlp.experts.，以适应
    feed_forward.experts 等其他前缀路径。
    """
    # 仅当名称包含 .experts. 时进行重映射
    if ".experts." not in name:
        return name

    # 如果已经包含 routed_experts，则跳过
    if ".experts.routed_experts." in name:
        return name

    # 检查是否为专家参数（根据常见后缀判断）
    expert_param_suffixes = [
        "w13_weight", "w2_weight",
        "w13_weight_scale", "w2_weight_scale",
        "w13_input_scale", "w2_input_scale",
        # ... 其他后缀
    ]
    is_expert_param = any(
        f".{suffix}" in name or name.endswith(suffix)
        for suffix in expert_param_suffixes
    )
```

```
    if not is_expert_param:
        return name

    # 在 .experts. 之后插入 routed_experts.
    # 使用 replace(..., 1) 仅替换第一个匹配项，避免多层嵌套
    new_name = name.replace(".experts.", ".experts.routed_experts.", 1)

    # 只有在目标名称存在于模型中时，才使用新名称
    if new_name in params_dict:
        return new_name

    return name
```

### vllm/model_executor/models/llama4.py

在 llama4.py 的 load_weights 中，处理 expert scale 的分支里添加了重映射调用，这是修复的主场景。

```
# vllm/model_executor/models/llama4.py
# 在 load_weights 方法中，处理 flat expert scale 的分支：

if "experts." in name and any(scale_name in name for scale_name in scale_names):
    # 在访问 params_dict 之前，先进行名称重映射
    name = maybe_remap_moe_expert_param_name(name, params_dict)
    param = params_dict[name] # 现在不会抛出 KeyError
    weight_loader = getattr(param, "weight_loader", default_weight_loader)
    if getattr(weight_loader, "supports_moe_loading", False):
        shard_id = "w2" if "w2_" in name else "w1"
        # ... 处理 FP8 转置
        weight_loader(param, loaded_weight, name, shard_id=shard_id, expert_id=0)
    else:
        weight_loader(param, loaded_weight)
    continue
```

## 评论区精华

该 PR 无 review 评论，但获得两名 reviewer 的快速批准。bnellnm 表示“正打算调查这个问题”，说明问题已被发现但尚未修复，本 PR 填补了空白。

- 暂无高价值评论线程

## 风险与影响

- 风险：无新增测试，回归风险较低但存在：修改了 maybe_remap_moe_expert_param_name 的匹配逻辑，若其他模型也使用 .experts 但路径结构不同（例如嵌套层级），可能导致意外重映射。不过由于新逻辑会检查重映射后的名称是否存在于 params_dict，不存在时返回原名，因此影响有限。
- 影响：直接影响 Llama4、mllama4、lfm2_moe 模型的权重加载，修复了它们因 MoE 重构导致的启动崩溃。对其他使用 .mlp.experts 的模型（如 DeepSeek 系列）无影响。整体修

复范围小，但用户可立即在 Nightly CI 中验证。

- 风险标记：核心路径变更，缺少测试覆盖

## 关联脉络

- PR #41184 [Refactor] MoE refactor: routed_experts submodule: 此 PR 是导致回归的源头，MoE 重构将 expert 参数移至 routed_experts 子模块。本 PR 修复了该重构引入的不兼容问题。