PR #45054 完整报告

vllm-project/vllm

[Bugfix] Fix weight loading issues caused by #41184

合并时间：2026-06-10 13:20

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/45054

执行摘要

- 一句话：修复 MoE 权重加载路径映射，添加 .routed_experts 前缀
- 推荐动作：这是一个典型的适配型 bugfix，展示了多模型对单一底层重构的协同修改。值得建议相关模型维护者关注，确保自己的自定义加载代码与主干保持一致。设计决策：将前缀统一改为 routed_experts，保持了与 #41184 的一致性。

功能与动机

PR #41184 引入了 MoE 专家参数名的重构，将权重名从 '.moe.experts.' 改为 '.moe.experts.routed_experts.'，但 Qwen、Step 等模型的自定义 load_weights 代码未同步更新，导致权重加载时 KeyError。参见 PR body 说明与 issue 评论中报告的 deepseek v4 错误。

实现拆解

实现拆解分为四步：

1. 更新 expert_params_mapping 前缀：在 step3p5.py、step3_text.py、qwen3_vl_moe.py 的 load_weights 方法中，将 expert_params_mapping 列表中的路径前缀从 ".moe.experts." 改为 ".moe.experts.routed_experts."，覆盖权重、缩放和输入缩放的所有条目。
2. 同步 fused_expert_params_mapping：qwen3_vl_moe.py 中另有 fused_expert_params_mapping，也进行了同样的前缀修改。
3. 调整 aria.py 的 packed_modules_mapping：将 "experts.w13_weight" 改为 "experts.routed_experts.w13_weight"，确保参数合并路径正确。
4. 简化 deepseek_v4/quant_config.py：移除 MoERunner 的导入和 isinstance 分支，因为 #41184 后专家层统一由 RoutedExperts 表示；同时简化 is_mxfp4_quant 的条件表达式。

未添加测试变更，但作者测试了 Qwen3 MoE 和 Step 模型。

关键文件：

- vllm/model_executor/models/step3p5.py（模块 模型加载；类别 source；类型 data-contract；符号 load_weights）：核心修改：更新 MoE 专家参数映射路径，修复 Step3.5 权重加载失败

- vllm/model_executor/models/step3_text.py（模块 模型加载；类别 source；类型 data-contract；符号 load_weights）：同步更新 step3_text 的 expert_params_mapping 路径前缀
- vllm/model_executor/models/qwen3_vl_moe.py（模块 模型加载；类别 source；类型 data-contract；符号 load_weights）：更新 Qwen3 MoE 的 expert_params_mapping 和 fused_expert_params_mapping
- vllm/models/deepseek_v4/quant_config.py（模块 量化配置；类别 source；类型 data-contract；符号 get_quant_method, is_mxfp4_quant）：移除 MoERunner 引用，调整 isinstance 检查，修复 deepseek v4 权重加载
- vllm/model_executor/models/aria.py（模块 模型加载；类别 source；类型 data-contract；符号 packed_modules_mapping）：更新 packed_modules_mapping 中的专家参数键名

关键符号：load_weights, get_quant_method, is_mxfp4_quant

关键源码片段

vllm/model_executor/models/step3p5.py

核心修改：更新 MoE 专家参数映射路径，修复 Step3.5 权重加载失败

```
# expert_params_mapping 将 checkpoint 中的旧格式权重名映射到模型定义的新格式。
# PR #41184 将专家参数名从 .moe.experts. 改为 .moe.experts.routed_experts.,
# 因此映射键都需要包含 routed_experts 前缀。
expert_params_mapping = [
    # 权重张量（gate/up/down）
    (f".moe.experts.routed_experts.{base_layer}w13_weight",
     ".moe.gate_proj.weight", "w1"),
    (f".moe.experts.routed_experts.{base_layer}w13_weight",
     ".moe.up_proj.weight", "w3"),
    (f".moe.experts.routed_experts.{base_layer}w2_weight",
     ".moe.down_proj.weight", "w2"),
    # 二级缩放（scale_2）
    (f".moe.experts.routed_experts.{base_layer}w13_weight_scale_2",
     ".moe.gate_proj.weight_scale_2", "w1"),
    (f".moe.experts.routed_experts.{base_layer}w13_weight_scale_2",
     ".moe.up_proj.weight_scale_2", "w3"),
    (f".moe.experts.routed_experts.{base_layer}w2_weight_scale_2",
     ".moe.down_proj.weight_scale_2", "w2"),
    # 一级缩放（scale）
    (f".moe.experts.routed_experts.{base_layer}w13_weight_scale",
     ".moe.gate_proj.weight_scale", "w1"),
    (f".moe.experts.routed_experts.{base_layer}w13_weight_scale",
     ".moe.up_proj.weight_scale", "w3"),
    (f".moe.experts.routed_experts.{base_layer}w2_weight_scale",
     ".moe.down_proj.weight_scale", "w2"),
    # 输入缩放（input_scale）
    (f".moe.experts.routed_experts.{base_layer}w13_input_scale",
     ".moe.gate_proj.input_scale", "w1"),
```

```
        (f".moe.experts.routed_experts.{base_layer}w13_input_scale",
         ".moe.up_proj.input_scale", "w3"),
        (f".moe.experts.routed_experts.{base_layer}w2_input_scale",
         ".moe.down_proj.input_scale", "w2"),
    ]
```

vllm/models/deepseek_v4/quant_config.py

移除 MoERunner 引用，调整 isinstance 检查，修复 deepseek v4 权重加载

```
    # get_quant_method 根据层类型返回合适的量化方法。
    # PR #41184 后，MoE 专家层统一由 RoutedExperts 表示，不再单独使用 MoERunner，
    # 因此移除了对 MoERunner 的 isinstance 检查。
    def get_quant_method(self, layer, prefix):
        if isinstance(layer, RoutedExperts):
            if is_layer_skipped(prefix=prefix,
                                ignored_layers=self.ignored_layers,
                                fused_mapping=self.packed_modules_mapping):
                return UnquantizedFusedMoEMethod(layer.moe_config)
            if self.expert_dtype == "fp4":
                if self.moe_quant_algo == "NVFP4":
                    return ModelOptNvFp4FusedMoE(
                        quant_config=self._get_nvfp4_config(),
                        moe_config=layer.moe_config)
                return Mxfp4MoEMethod(layer.moe_config)
            # expert_dtype == "fp8": fall through to Fp8Config
        return super().get_quant_method(layer, prefix)

    # is_mxfp4_quant 同样去掉了 MoERunner 分支。
    def is_mxfp4_quant(self, prefix, layer):
        if not isinstance(layer, RoutedExperts) or self.expert_dtype != "fp4":
            return False
        return self.moe_quant_algo != "NVFP4"
```

评论区精华

在 issue 评论中，用户 wzhao18 报告了 deepseek v4 的权重加载 KeyError，并提供了针对 quant_config.py 的补丁（移除 MoERunner import 和调整 isinstance 检查）。作者 bnellnm 确认该补丁解决了 deepseek v4 问题，并将其纳入 PR 的第三个 commit。这是一个典型的社区协作修复过程。

- DeepSeek V4 权重加载错误报告与修复 (correctness): 已通过 PR 修复。

风险与影响

- 风险：主要风险在于权重映射路径必须与模型定义完全一致。如果新的映射仍有遗漏（如某些量化参数未覆盖），可能导致权重加载失败或静默错误。此外，deepseek_v4 的修改移除了 MoERunner 分支，若某些检查点仍使用旧格式，可能会忽略部分专家层。但根据 PR 测试（Qwen3 MoE 和 Step3.5），风险较低。缺少直接的单元测试覆盖是短板。

- 影响：影响范围：此修复影响 Qwen3 MoE、Step3.5、Step3 Text、Aria 和 DeepSeek V4 模型的权重加载。这些模型之前可能因 #41184 而无法加载，修复后恢复正常。对其他模型无影响。未涉及运行时推理路径。
- 风险标记：缺少测试覆盖，多模型一致性风险，深度依赖 #41184

关联脉络

- PR #41184 MoE weight mapping refactor: 引出了本 PR 需要修复的权重名变化
- PR #45002 [Bugfix] fix qwen3.5 ep weight loading: 类似的权重加载修复，与本 PR 同系列