PR #47201 完整报告

vllm-project/vllm

[ROCm][Bugfix] Convert ModelOpt FP8 per-channel weights to e4m3fnuz on MI300/MI325

合并时间：2026-07-07 14:24

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/47201

执行摘要

- 一句话：修复 ROCm 平台 ModelOpt FP8 权重格式不兼容问题
- 推荐动作：值得阅读。该 PR 演示了如何利用平台抽象层（current_platform）和现有工具函数实现跨平台 FP8 dtype 兼容，是一个简洁的跨平台修复范例。建议关注其他 ModelOpt 方法是否需类似修复。

功能与动机

ROCm CI 中 ModelOpt FP8 测试因权重 dtype 硬编码为 *e4m3fn* 而非平台所需的 *e4m3fnuz* 导致 ValueError。PR body 指出：*We are seeing a CI failure due to the modelopt tests hardcoding the e4m3fn fp8 dtype rather than utilizing the appropriate format for the current platform*。

实现拆解

1. 在 vllm/model_executor/layers/quantization/modelopt.py 中导入 process_fp8_weight_channel_strategy 函数。
2. 修改 ModelOptFp8PcPtLinearMethod.process_weights_after_loading 方法，调用该函数处理权重和 scale 张量，利用其内部自动判断平台 FP8 格式（对 ROCm 执行 e4m3fn->e4m3fnuz 转换）。
3. 在 tests/quantization/test_modelopt.py 中将权重 dtype 断言从硬编码 torch.float8_e4m3fn 替换为 current_platform.fp8_dtype()，使测试兼容不同平台。
4. 验证：在 MI325 上执行 test_modelopt_fp8_pc_pt_checkpoint_setup 通过。

关键文件：

- vllm/model_executor/layers/quantization/modelopt.py（模块 量化层；类别 source；类型 data-contract；符号 process_weights_after_loading, process_fp8_weight_channel_strategy）：核心修复文件，修改 process_weights_after_loading 方法以支持 ROCm 平台的 e4m3fnuz 格式，导入并调用 process_fp8_weight_channel_strategy。
- tests/quantization/test_modelopt.py（模块 测试；类别 test；类型 test-coverage；符号 test_modelopt_fp8_pc_pt_checkpoint_setup）：测试配套，修改权重 dtype 断言为动态获取，确保 ROCm 兼容。

关键符号：process_weights_after_loading, test_modelopt_fp8_pc_pt_checkpoint_setup

关键源码片段

vllm/model_executor/layers/quantization/modelopt.py

核心修复文件，修改 process_weights_after_loading 方法以支持 ROCm 平台的 e4m3fnuz 格式，导入并调用 process_fp8_weight_channel_strategy。

```
def process_weights_after_loading(self, layer: torch.nn.Module) -> None:
    # 利用现有的 process_fp8_weight_channel_strategy 统一处理权重格式转换
    # 在 ROCm 上会自动将 e4m3fn 转换为 e4m3fnuz
    weight, weight_scale, _ = process_fp8_weight_channel_strategy(
        layer.weight, layer.weight_scale.data
    )
    layer.weight = Parameter(weight.t(), requires_grad=False)
    layer.weight_scale = Parameter(weight_scale, requires_grad=False)
    self.fp8_linear.process_weights_after_loading(layer)
```

评论区精华

Reviewer fxmarty-amd 建议复用现有的 process_fp8_weight_channel_strategy 抽象（引自 compressed_tensors 的使用），替代手动判断 is_fp8_fnuz() 并调用 normalize_e4m3fn_to_e4m3fnuz。作者采纳并修改代码。另外，fxmarty-amd 离线指出其他 ModelOpt 方法（如 ModelOptFp8LinearMethod、ModelOptFp8PbWoLinearMethod）也可能缺失类似转换，但未在当前 PR 中处理。

- 复用 process_fp8_weight_channel_strategy 替代手动转换 (design): 作者采纳建议，修改代码使用 process_fp8_weight_channel_strategy，并在之后更新提交。

风险与影响

- 风险：风险较低。核心变更仅影响 ModelOptFp8PcPtLinearMethod，复用已有抽象确保正确性。但其他 ModelOpt 方法（ModelOptFp8LinearMethod、ModelOptFp8PbWoLinearMethod）可能仍存在 dtype 不兼容问题，待后续覆盖。测试仅限于 checkpoint 加载和权重属性检查，未包含端到端推理准确性验证。对 NVIDIA 平台无副作用。
- 影响：对 ROCm 用户：修复了 ModelOpt FP8 per-channel per-token 量化在 MI300/MI325 上的 CI 失败，使该量化路径在 AMD GPU 上可正常运行。对 NVIDIA 用户无影响，因为 current_platform.fp8_dtype() 返回 torch.float8_e4m3fn，行为完全一致。变更影响范围小，仅涉及量化权重加载阶段。
- 风险标记：其他 ModelOpt 方法可能缺失类似修正

关联脉络

- PR #47318 [BugFix] Fix ModelOpt mixed-precision quantization for sparse quantized_layers configs.: 修改了同一个文件 vllm/model_executor/layers/quantization/modelopt.py，涉及 ModelOpt 量化逻辑