PR #46845 完整报告

vllm-project/vllm

[Bugfix] Fix MiniMax-M3 compressed-tensors FP8 MoE SwiGLU params

合并时间：2026-08-12 18:52

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/46845

执行摘要

- 一句话：修复 MiniMax-M3 FP8 MoE 的 SwiGLU 参数转发
- 推荐动作：建议精读。这是一个小而精准的 bugfix，展示了非默认激活参数在量化 MoE 路径中容易遗漏转发的问题，并附带针对性单元测试与端到端精度数据作为验收标准。同时建议跟进 #49473，排查其他量化路径是否有同类遗漏。

功能与动机

PR body 明确指出：MiniMax-M3 使用 hidden_act="swigluoai" 和非默认 SwiGLU 参数（swiglu_alpha=1.702、swiglu_beta=1.0、swiglu_limit=7.0），而 compressed-tensors FP8 MoE 路径此前只转发了 swiglu_limit，导致 fused MoE 使用了错误的激活参数，输出严重降级。tjtanaa 在评论中要求提供验证模型链接以帮助社区复现；prakhar-prakash-juspay 指出 ModelOptFp8MoEMethod 有完全相同遗漏并给出配套修复 #49473。

实现拆解

1. 修改 vllm/model_executor/layers/quantization/compressed_tensors/compressed_tensors_moe/compressed_tensors_moe_w8a8_fp8.py 的 get_fused_moe_quant_config，在调用 make_fp8_moe_quant_config 时新增 gemm1_alpha=getattr(layer, "swiglu_alpha", None) 与 gemm1_beta=getattr(layer, "swiglu_beta", None)。
2. 在 tests/quantization/test_compressed_tensors.py 中新增单元测试 test_compressed_tensors_w8a8_fp8_moe_forwards_swiglu_params，通过 object.__new__ 构造 CompressedTensorsW8A8Fp8MoEMethod 实例，用 Mock 模拟 layer 包含 swiglu_alpha=1.702、swiglu_beta=None、swiglu_limit=7.0，断言 gemm1_alpha == 1.702、gemm1_beta is None、gemm1_clamp_limit == 7.0，并补充 Fp8MoeBackend 与 CompressedTensorsW8A8Fp8MoEMethod 的导入。
3. 端到端验证使用 EmbeddedLLM/MiniMax-M3-FP8-dynamic 检查点，修复后 GSM8K Platinum 95.92（BF16 95.81）、GPQA Diamond 77.95（BF16 77.78），恢复至接近 BF16 精度；同时引入兼容性设计，None 默认值保留默认激活路径。

无配置或部署配套改动。

关键文件：

- vllm/model_executor/layers/quantization/compressed_tensors/compressed_tensors_moe/compressed_tensors_moe_w8a8_fp8.py（模块 量化实现；类别 source；类型

data-contract；符号 get_fused_moe_quant_config）：修复核心路径：get_fused_moe_quant_config 新增 gemm1_alpha 与 gemm1_beta 的转发，使 MiniMax-M3 的非默认 SwiGLU 参数能正确传入 make_fp8_moe_quant_config。

- tests/quantization/test_compressed_tensors.py（模块 量化测试；类别 test；类型 test-coverage；符号 test_compressed_tensors_w8a8_fp8_moe_forwards_swiglu_params）：新增单元测试覆盖转发行为，直接断言 gemm1_alpha、gemm1_beta、gemm1_clamp_limit 被正确传递，防止回归。

关键符号：CompressedTensorsW8A8Fp8MoEMethod.get_fused_moe_quant_config, test_compressed_tensors_w8a8_fp8_moe_forwards_swiglu_params

关键源码片段

vllm/model_executor/layers/quantization/compressed_tensors/compressed_tensors_moe/compressed_tensors_moe_w8a8_fp8.py

修复核心路径：get_fused_moe_quant_config 新增 gemm1_alpha 与 gemm1_beta 的转发，使 MiniMax-M3 的非默认 SwiGLU 参数能正确传入 make_fp8_moe_quant_config。

```
def get_fused_moe_quant_config(self, layer: torch.nn.Module) -> FusedMoEQuantConfig:
    is_per_token = self.input_quant.strategy == QuantizationStrategy.TOKEN
    # 关键修复：转发非默认 SwiGLU-OAI 参数（如 MiniMax-M3 的 swiglu_alpha=1.702）
    # 缺省为 None 以兼容默认 SwiGLU 激活路径，不影响现有模型
    return make_fp8_moe_quant_config(
        fp8_backend=self.fp8_backend,
        w1_scale=layer.w13_weight_scale,
        w2_scale=layer.w2_weight_scale,
        a1_scale=getattr(layer, "w13_input_scale", None),
        a2_scale=getattr(layer, "w2_input_scale", None),
        per_act_token_quant=is_per_token,
        per_out_ch_quant=is_per_token,
        block_shape=self.weight_block_size,
        gemm1_alpha=getattr(layer, "swiglu_alpha", None),
        gemm1_beta=getattr(layer, "swiglu_beta", None),
        swiglu_limit=getattr(layer, "swiglu_limit", None),
        layer=layer,
    )
```

tests/quantization/test_compressed_tensors.py

新增单元测试覆盖转发行为，直接断言 gemm1_alpha、gemm1_beta、gemm1_clamp_limit 被正确传递，防止回归。

```
def test_compressed_tensors_w8a8_fp8_moe_forwards_swiglu_params():
    # 通过 object.__new__ 绕过 __init__ 构造实例，减少依赖
    quant_method = object.__new__(CompressedTensorsW8A8Fp8MoEMethod)
    quant_method.input_quant = QuantizationArgs(
        num_bits=8,
        type=QuantizationType.FLOAT,
        strategy=QuantizationStrategy.TOKEN,
```

```
        dynamic=True,
        symmetric=True,
    )
    quant_method.weight_block_size = None
    quant_method.fp8_backend = Fp8MoeBackend.TRITON

    # 用 Mock 模拟 layer，只暴露 get_fused_moe_quant_config 需要的字段
    layer = Mock()
    layer.w13_weight_scale = torch.ones(2, 4, 1)
    layer.w2_weight_scale = torch.ones(2, 8, 1)
    layer.w13_input_scale = None
    layer.w2_input_scale = None
    layer.swiglu_alpha = 1.702
    layer.swiglu_beta = None
    layer.swiglu_limit = 7.0

    quant_config = quant_method.get_fused_moe_quant_config(layer)

    # 验证转发：alpha 为 1.702，beta 为 None（未设置），clamp_limit 为 7.0
    assert quant_config.gemm1_alpha == 1.702
    assert quant_config.gemm1_beta is None
    assert quant_config.gemm1_clamp_limit == 7.0
```

评论区精华

> tjtanaa：@tanpinsiang please provide the link to the model that you used to validate this. It will be useful for the community... tanpinsiang：https://huggingface.co/EmbeddedLLM/MiniMax-M3-FP8-dynamic This is the target model for FP8 hardware. prakhar-prakash-juspay：ModelOptFp8MoEMethod has the identical omission — companion fix in #49473. Independently confirmed on a ModelOpt per-tensor FP8 MiniMax-M3 checkpoint: per-expert-layer cosine vs a pure-torch reference went 0.966 → 0.99+ once gemm1_alpha/gemm1_beta are forwarded... Thanks for the CT-side fix — it confirmed the pattern. tjtanaa：/ci run ... /ci retry（CI 重试后通过）

- 验证模型链接询问 (question)：社区可通过该链接复现 FP8 硬件上的效果。
- ModelOptFp8MoEMethod 同类遗漏 (design)：确认了参数转发的通用模式，催生 #49473 配套修复。
- CI 与重试 (other)：CI 通过后 PR 获得批准并合入。

风险与影响

- 风险：风险点：
  - make_fp8_moe_quant_config 对 gemm1_alpha/gemm1_beta 为 None 时的默认行为未在本次 PR 中显式验证，但 getattr 默认 None 与之前行为一致，默认 SwiGLU 模型不受影响；

    - 同类转发遗漏可能存在于其他量化路径（如 ModelOptFp8MoEMethod，已在 #49473 中修复），需排查 gemm1_alpha/gemm1_beta 在所有 MoE 量化入口的转发情况；
    - 变更仅影响 compressed-tensors W8A8 FP8 MoE 路径，改动面小，回归风险低。
- 影响：影响范围：
    - 用户侧：MiniMax-M3 使用 compressed-tensors FP8 动态检查点的用户将获得正确输出质量（GSM8K 从 0.00 恢复至 95.92）；使用默认 SwiGLU 的模型行为不变。
    - 系统侧：修改局限于单个量化方法与对应测试，无性能、安全影响。
    - 团队侧：验证了 swiglu_alpha/swiglu_beta 转发的通用模式，推动 #49473 配套修复落地，有助于后续统一 MoE 激活参数转发契约。
    - 风险标记：同类遗漏未排查完，需确认 None 兼容性，局部变更，影响面小

关联脉络

- PR #49473 Companion fix for ModelOpt FP8 MoE SwiGLU params: 讨论中明确提出 ModelOptFp8MoEMethod 有相同参数转发遗漏，本 PR 是 compressed-tensors 侧的配套修复，两者共同确立 gemm1_alpha/gemm1_beta 转发模式。