

PR #46419 完整报告

vllm-project/vllm

[ROCm]Enable AITER MoE backend for MiniMax-M3-MXFP4

合并时间: 2026-06-26 21:35

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/46419>

执行摘要

- 一句话: 为 MiniMax-M3-MXFP4 启用 AITER MoE 后端
- 推荐动作: 该 PR 展示了如何通过配置扩展 MoE 基础设施以支持新模型的特殊需求, 值得关注其临时兼容层的设计模式。建议在 AITER 修复 pad 问题后及时跟进清理。

功能与动机

MiniMax-M3-MXFP4 模型要求激活量化 (activation QDQ), 但现有的 TRITON_UNFUSED 后端未实现该功能, 导致推理精度不达标。AITER MoE 后端正确定支持该模型, 是目前 AMD ROCm 上唯二正确的后端之一 (另一为 EMULATION)。因此需要启用 AITER MoE 后端并修复与其行为匹配的配置。

实现拆解

1. 新增配置字段: 在 FusedMoEConfig (config.py) 中增加 intermediate_pad 可选字段, 允许模型层显式指定中间层 padding 值, 以覆盖默认的 (intermediate_size_per_partition - intermediate_size_per_partition_unpadded) 计算, 临时绕过 AITER 的 pad-skipping 精度问题。
2. 参数传递: 在 FusedMoE 函数入口 (layer.py) 新增 intermediate_pad 参数, 并将其透传到 FusedMoEConfig, 打通模型到配置的链路。
3. 激活与 Gate 模式扩展: 在 rocm_aiter_fused_experts (rocm_aiter_moe.py) 中添加对 MoEActivation.SWIGLUOAI_UNINTERLEAVE 激活类型的处理, 并分离 GateMode 选择逻辑: 当激活为 SWIGLUOAI (interleave) 或 SWIGLUOAI_UNINTERLEAVE (separated) 时, 分别设置 GateMode.INTERLEAVE 或 GateMode.SEPARATED, 以适应 MiniMax 模型非交错的权重布局。
4. 模型端适配: 在 MiniMaxM3MoE.__init__ (model.py) 中构造 FusedMoE 时传入 intermediate_pad=0, 禁用默认 padding 计算, 配合 AITER 当前版本行为。
5. 测试配套: 本次变更未新增测试, 依赖现有 CI 覆盖和 GSM8K 离线验证 (94% 准确率)。

关键文件:

- vllm/model_executor/layers/fused_moe/experts/rocm_aiter_moe.py (模块 混合专家; 类别 source; 类型 core-logic; 符号 rocm_aiter_fused_experts, _supports_activation, GateMode): 核心变更: 添加 SWIGLUOAI_UNINTERLEAVE 激活支持和 GateMode 分离, 是 MoE kernel 调度的关键。

- vllm/model_executor/layers/fused_moe/config.py (模块 混合专家; 类别 source; 类型 data-contract; 符号 FusedMoEConfig.intermediate_pad) : 新增 intermediate_pad 配置字段, 允许为特定模型指定中间层 padding 值, 以绕过 AITER 精度问题。
- vllm/model_executor/layers/fused_moe/layer.py (模块 混合专家; 类别 source; 类型 data-contract; 符号 FusedMoE, FusedMoEConfig) : 传递新增的 intermediate_pad 参数到 FusedMoEConfig, 打通配置链路。
- vllm/models/minimax_m3/amd/model.py (模块 模型定义; 类别 source; 类型 data-contract; 符号 MiniMaxM3MoE.init) : 模型层调用 FusedMoE 时传入 intermediate_pad=0, 启用 pad 绕过。

关键符号: rocm_aiter_fused_experts, _supports_activation, FusedMoE, MiniMaxM3MoE.init

关键源码片段

vllm/model_executor/layers/fused_moe/experts/rocm_aiter_moe.py

核心变更: 添加 SWIGLUOAI_UNINTERLEAVE 激活支持和 GateMode 分离, 是 MoE kernel 调度的关键。

```
def rocm_aiter_fused_experts(
    hidden_states: torch.Tensor,
    w1: torch.Tensor,
    w2: torch.Tensor,
    topk_weights: torch.Tensor,
    topk_ids: torch.Tensor,
    moe_config: FusedMoEConfig,
    activation: MoEActivation = MoEActivation.SILU,
    # ... 其他参数省略
) -> torch.Tensor:
    if quant_config is None:
        quant_config = FUSED_MOE_UNQUANTIZED_CONFIG

    # Gate/up interleave hint; only the SWIGLUOAI activations override it.
    activation_interleave = None
    if activation == MoEActivation.SILU:
        activation_method = ActivationMethod.SILU
    elif activation == MoEActivation.GELU:
        activation_method = ActivationMethod.GELU
    elif activation == MoEActivation.SWIGLUOAI:
        activation_method = rocm_aiter_ops.get_aiter_activation_type("swiglu")
    elif activation == MoEActivation.SWIGLUOAI_UNINTERLEAVE:
        # MiniMax-M3-MXFP4 使用非交错 (separated) gate/up 权重布局
        activation_method = rocm_aiter_ops.get_aiter_activation_type("swiglu")
        activation_interleave = False
    else:
        raise ValueError(f"Unsupported activation: {activation}")

    # ... 早期返回和量化分支 ...
```

```

# Compute padding on-the-fly for CK MXFP4 kernels
hidden_pad = 0
intermediate_pad = 0
assert moe_config.hidden_dim_unpadded is not None
assert moe_config.intermediate_size_per_partition_unpadded is not None
hidden_pad = hidden_states.shape[1] - moe_config.hidden_dim_unpadded
intermediate_pad = (
    (
        moe_config.intermediate_size_per_partition
        - moe_config.intermediate_size_per_partition_unpadded
    )
    if moe_config.intermediate_pad is None
    else moe_config.intermediate_pad
)
# Round padding to match AITER's alignment
hidden_pad = hidden_pad // 128 * 128
intermediate_pad = (
    intermediate_pad // 64 * 64 * (2 if moe_config.tp_size == 1 else 1)
)

# gate mode selection
from aiter.ops.flydsl.moe_common import GateMode
gate_mode = ""
if quant_config.use_mxfp4_w4a16:
    # MXFP4 权重始终为 interleave 布局
    gate_mode = GateMode.INTERLEAVE.value
elif activation_interleave is not None:
    # 根据激活类型设置 gate 模式
    gate_mode = (
        GateMode.INTERLEAVE.value
        if activation_interleave
        else GateMode.SEPARATED.value
    )

return rocm_aiter_ops.fused_moe(
    hidden_states,
    w1,
    w2,
    # ... 其他参数
    gate_mode=gate_mode,
)

```

vllm/model_executor/layers/fused_moe/config.py

新增 `intermediate_pad` 配置字段，允许为特定模型指定中间层 padding 值，以绕过 AITER 精度问题。

```

@dataclass
class FusedMoEConfig:
    # ... 其他字段 ...

```

```
hidden_dim_unpadded: int | None = None
# Defaults to intermediate_size_per_partition if not specified.
intermediate_size_per_partition_unpadded: int | None = None
# Model specific override: 用于临时绕过 AITER 的 pad-skipping 精度问题
intermediate_pad: int | None = None

moe_backend: MoEBackend = "auto"
# ... 后续字段 ...
```

评论区精华

- `intermediate_pad` 必要性: BowenBao 质疑为何需要特殊值, qli88 解释为临时规避 AITER `pad-skipping` 精度问题, 待 AITER 修复后移除。
- AITER 版本依赖: tjtaana 指出新导入的 `GateMode` 路径在 AITER `v0.1.13.post1` 中不存在, PR 须等 AITER 升级至 `v0.1.15.post3` 后方可合并。
- 导入作用域: tjtaana 要求将 `from aiter.ops.flydsl.moe_common import GateMode` 放在函数内部, 防止非 ROCm 平台 `ImportError`。最终代码已遵守。
- `hidden_pad` 冗余: tjtaana 发现最初引入的 `hidden_pad` 参数未被使用, 建议删除, 最终仅保留 `intermediate_pad`。
 - `intermediate_pad` 必要性和临时性 (design): 同意保留 `intermediate_pad` 作为临时机制, 后续待 AITER 修复后清理。
 - AITER 版本依赖 (correctness): PR 在 AITER 新版本发布后通过, 合并前 AITER 版本已升级。
 - 导入作用域 (design): 最终代码将 `from aiter.ops.flydsl.moe_common import GateMode` 置于 `rocm_aiter_fused_experts` 函数体内, 满足要求。

风险与影响

- 风险:
 1. AITER 版本绑定: 代码依赖 AITER `v0.1.15+`, 用户若使用旧版会因导入失败或符号缺失而报错。需保证环境文档和 CI 对齐。
 2. 临时逻辑遗留: `intermediate_pad=0` 是临时绕过方案, 若 AITER 未来修复 `pad` 问题而 `vLLM` 未及时清理, 将增加配置复杂性和潜在混淆。
 3. 平台隔离: 虽然导入位于函数内部, 但若 AITER 在其他平台通过条件编译已不可用, 风险较低; 但仍建议显式检查 AITER 可用性。
- 影响:
 - 用户影响: AMD ROCm 用户现可正确使用 `MiniMax-M3-MXFP4` 模型, `GSM8K` 准确率达 94%。此前唯一选择是 `EMULATION` 后端, AITER 提供更高性能。
 - 系统影响: 新增 `intermediate_pad` 配置参数, 默认 `None` 保持向后兼容。
 - 团队影响: 维护者需跟踪 AITER 版本升级, 及时发起清理 PR 移除临时参数。
 - 风险标记: 依赖 AITER 版本升级, 临时绕过需后续清理, 导入路径平台风险

关联脉络

- PR #46780 [ROCm] Fix AITER_UNIFIED_ATTN Dispatching After AITER Bump: 同为 ROCm AITER 后端调整，处理类似 AITER 版本兼容和导入问题。