

PR #44679 完整报告

vllm-project/vllm

[ROCM][Bugfix] Make intermediate_pad TP-aware in rocm_aiter_fused_experts

合并时间: 2026-06-11 04:10

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/44679>

执行摘要

- 一句话: 修复 AITER MoE 在 $TP > 1$ 时精度严重下降
- 推荐动作: 推荐合并。该 PR 以极小的代码改动 (+12/-2) 修复了严重的精度回归, 且经过明确的基准测试验证。值得关注的是其设计决策模式——在跨后端、跨 TP 的场景下必须显式管理 padding 约定差异。建议在升级 AITER 后及时清理 TODO 注释。

功能与动机

PR#42098 为统一 padding 约定而修改了 `intermediate_pad` 的计算, 但未考虑 AITER 不同 MoE 后端 (CK vs FlyDSL) 在不同 TP 规模下对 padding 的不同处理。这导致 GPT-OSS-120B 在 $TP=8$ 时 GSM8K 准确率从 0.896 暴跌至 0.684, 被 bisect 确定为回归源, 亟需修复以恢复模型精度。

实现拆解

1. 提取 padding 计算逻辑: 在 `rocm_aiter_fused_experts` 函数中, 将原本内联的 `hidden_pad // 128 * 128` 和 `intermediate_pad // 64 * 64 * 2` 从 `fused_moe` 调用参数中拆出, 提前计算并存入变量。
2. 引入 TP 维度判断: 新增条件分支, 当 `moe_config.tp_size == 1` 时 `intermediate_pad` 使用 `intermediate_pad // 64 * 64 * 2`; 当 `tp_size > 1` 时使用 `intermediate_pad // 64 * 64 * 1`。这是因为 AITER 的 FlyDSL 后端 ($TP=1$) 和 CKTile 后端 ($TP=8$) 对 padding 的解释不同。
3. 传递计算后的 padding 值: 将预先计算好的 `hidden_pad` 和 `intermediate_pad` 直接传入 `rocm_aiter_ops.fused_moe`, 并添加详细注释说明 AITER 版本约束和待升级的 TODO。

关键文件:

- `vllm/model_executor/layers/fused_moe/experts/rocm_aiter_moe.py` (模块 MoE 专家; 类别 source; 类型 data-contract): 唯一修改文件, 核心修复: 将 `intermediate_pad` 的 padding 倍数从固定 `*2` 改为 TP-aware 条件分支。新增 padding 预处理代码和详细注释。

关键符号: `rocm_aiter_fused_experts`

关键源码片段

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

唯一修改文件，核心修复：将 intermediate_pad 的 padding 倍数从固定 *2 改为 TP-aware 条件分支。新增 padding 预处理代码和详细注释。

```
# File: vllm/model_executor/layers/fused_moe/experts/rocm_aiter_moe.py # 函数
rocm_aiter_fused_experts 中关键片段      # 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      )      # Round padding
values to match AITER's CK vs. FlyDSL MoE dispatch      # (currently pinned to
v0.1.13.post1).      # FlyDSL (used at TP=1) requires an extra *2 on
intermediate_pad,      # while CKTile (used at TP>1) does not.      # Ref:
https://github.com/ROCm/aiter/blob/v0.1.13.post1/aiter/fused_moe.py#L1073-L1099
    # TODO: Revisit after bumping AITER to 0.1.15 which includes padding fixes
hidden_pad = hidden_pad // 128 * 128      intermediate_pad = (
intermediate_pad // 64 * 64 * (2 if moe_config.tp_size == 1 else 1)      )      return
rocm_aiter_ops.fused_moe(      hidden_states, w1, w2, topk_weights, topk_ids,
    output_dtype=output_dtype,      hidden_pad=hidden_pad,
intermediate_pad=intermediate_pad,      # ... other arguments omitted for brevity
    )
```

（此片段展示了修复核心：将原先硬编码的 *2 改为基于 moe_config.tp_size 的条件表达式，并添加了详细注释说明不同后端的 padding 约定差异。）

评论区精华

审核人 dllehr-amd 指出这是 'quick fix'，并确认理解 padding 调整的原因，批准合并；计划在后续升级 AITER 到 0.1.15 时彻底解决。审核人 tjtanana 也批准。未出现技术争议。

- CI 预提交检查失败 (other): 开发者已按要求修复并推送，最终 CI 通过。

风险与影响

- 风险：回归风险：TP=1 throughput 在噪声范围内（从 3200 变为 3189 t/s），精度恢复至 0.900（接近 0.905 基线）。TP=8 时 throughput 从 5392 略微降至 5168（约 -4%），但精度从 0.684 恢复至 0.892，代价可接受。兼容性风险：补丁仅影响 AITER MoE 路径，且只修改了 padding 计算，不涉及其他模块。长期风险：代码中已有 TODO 注释提示升级 AITER 0.1.15 后需重新审视 padding 逻辑，若忘记清理可能引入技术债。
- 影响：用户：使用 ROCM AITER 且在 TP>1 下运行 MoE 模型的用户精度恢复正常。系统：仅修改 rocm_aiter_moe.py 中的 padding 逻辑，无配置 /API 变更。团队：为 ROCM/AITER 维护团队争取了时间，待 AITER 0.1.15 发布后可移除该 workaround。
- 风险标记：核心路径变更，缺少测试覆盖，平台特定风险（ROCM/AITER）

关联脉络

- PR #42098 Use hidden_pad and intermediate_pad from vLLM #34301: 本 PR 直接修复了 #42098 引入的精度回归, bisect 确认该 PR 更改了 intermediate_pad 的计算方式导致问题。