PR #42098 完整报告

vllm-project/vllm

Use hidden_pad and intermediate_pad from vLLM #34301

合并时间：2026-05-14 14:21

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/42098

执行摘要

- 一句话：修复 ROCm Aiter MoE padding 对齐问题提升性能
- 推荐动作：值得合并。改动小而精，有明确的性能收益和充分的 benchmark 数据支持。建议关注后续对非标准形状模型的进一步验证。

功能与动机

修复 ROCm Aiter 融合 MoE 路径中 hidden_pad 和 intermediate_pad 传参与内核栈 padding 约定不匹配的问题。该问题源自 PR#37128 引入的 padding 计算逻辑，并被 PR#37787 遗漏。在 GPT-OSS-120B / MI355X 上，修复后吞吐量几何平均提升约 8%。

实现拆解

1. 在 vllm/model_executor/layers/fused_moe/experts/rocm_aiter_moe.py 的 rocm_aiter_fused_experts 函数中，将传递给 rocm_aiter_ops.fused_moe 的 hidden_pad 参数由原始值改为 hidden_pad // 128 * 128（即向下对齐到 128 的倍数）。
2. 将 intermediate_pad 参数由原始值改为 intermediate_pad // 64 * 64 * 2（即向下对齐到 128 的倍数）。
3. 该对齐方式与 PR#34301 中引入的 padding 逻辑一致，确保内核栈能正确解释 padding 大小。
4. 仅修改了 2 行代码，属于数据契约层面的修正。

关键文件：

- vllm/model_executor/layers/fused_moe/experts/rocm_aiter_moe.py（模块 MoE 层；类别 source；类型 data-contract）：核心变更文件，修改了传递给 ROCm Aiter 融合 MoE 内核的 hidden_pad 和 intermediate_pad 对齐方式。

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

核心变更文件，修改了传递给 ROCm Aiter 融合 MoE 内核的 hidden_pad 和 intermediate_pad 对齐方式。

```
# vllm/model_executor/layers/fused_moe/experts/rocm_aiter_moe.py (line 356-362)
```

```
    return rocm_aiter_ops.fused_moe(
        hidden_states,
        w1,
        w2,
        topk_weights,
        topk_ids,
        expert_mask=expert_mask,
        quant_method=quant_method,
        activation_method=activation_method,
        w1_scale=quant_config.w1_scale,
        w2_scale=quant_config.w2_scale,
        a1_scale=quant_config.a1_scale if a1q_scale is None else a1q_scale,
        a2_scale=quant_config.a2_scale,
        doweight_stage1=apply_router_weight_on_input,
        num_local_tokens=num_local_tokens,
        output_dtype=output_dtype,
        hidden_pad=hidden_pad // 128 * 128, # 向下对齐到 128 的倍数
        intermediate_pad=intermediate_pad // 64 * 64 * 2, # 向下对齐到 128 的倍数
        bias1=quant_config.w1_bias if quant_config.use_mxfp4_w4a16 else None,
        bias2=quant_config.w2_bias if quant_config.use_mxfp4_w4a16 else None,
    )
```

评论区精华

gemini-code-assist[bot] 提出正确性风险：向下取整可能导致非标准形状模型的内存偏移错误，建议在张量分配时强制对齐或使用显式断言。 dllehr-amd 确认效果并说明具体影响：对于 GPT-OSS 模型，hidden_pad 从 192 变为 128，改变了 n_pad 值，计划进一步测试两阶段 MoE。 Rohan138 确认问题根源：该问题由 PR#37128 引入，PR#37787 遗漏。 最终结论：审核者均批准了该 PR，认为当前改动正确且有益。

- Padding 向下取整的正确性风险 (correctness): 审核者认为当前改动正确，实际测试精度仅下降 0.08pp，风险可控。

风险与影响

- 风险：正确性风险：向下取整可能导致非标准形状模型（padding 值不是 128 的倍数）接收到错误的 padding 计数，进而引起内存偏移错误，影响 SwiGLU 激活或不同专家的数据处理。该风险由 gemini-code-assist[bot] 指出，但实际测试显示精度仅下降 0.08pp（95.0% vs 94.92%），且审核者认为风险可控。 回归风险：低。改动仅 2 行，且已在特定模型上验证性能提升。
- 影响：用户影响：使用 ROCm Aiter 融合 MoE 后端的用户（如 AMD GPU 用户）将体验到显著的性能提升（约 8% 吞吐量提升），精度基本不变。 系统影响：仅影响 rocm_aiter_moe.py 一个文件，无外部接口或配置变更。 团队影响：修复了 PR#37128 引入的遗漏问题，降低了维护成本。
- 风险标记：核心路径变更，缺少测试覆盖

关联脉络

- PR #34301 引入 hidden_pad 和 intermediate_pad 对齐逻辑的原始 PR: 本 PR 的 padding 对齐方式正是参考了该 PR 中的逻辑。
- PR #37128 引入了 padding 计算逻辑但未对齐：本 PR 修复的问题正是由该 PR 引入的。
- PR #37787 应修复但遗漏了：本 PR 修复了该 PR 未能覆盖的问题。