PR #22952 完整报告

sgl-project/sglang

[AMD] Add SGLANG_MORI_MOE_MAX_INPUT_TOKENS to truncate dispatch before MoE.

合并时间：2026-04-17 14:40

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/22952

执行摘要

- 一句话：新增环境变量 SGLANG_MORI_MOE_MAX_INPUT_TOKENS，在 MoE 计算前截断调度缓冲区以减少填充开销。
- 推荐动作：该 PR 值得关注其设计权衡：通过环境变量控制性能优化，但牺牲了部分安全性。建议精读 run_moe_core 中的截断逻辑，理解其与 mori_op.combine 的交互，并注意 review 中提到的潜在改进点。

功能与动机

根据 PR body 描述，在 MoriEP MoE 路径中，dispatch_a1（调度输出的隐藏状态）的第一个维度固定为 SGLANG_MORI_NUM_MAX_DISPATCH_TOKENS_PER_RANK * *ep_size*（或 SGLANG_MORI_PREALLOC_MAX_RECV_TOKENS）的缓冲区大小，但实际有效令牌数（totalRecvTokenNum）通常远小于此缓冲区大小。aiter.fused_moe 使用输入张量的第一个维度（dispatch_a1.shape[0]）来选择内核调度策略。由于填充的缓冲区大小远大于实际的 totalRecvTokenNum，aiter.fused_moe 会选择一个针对大令牌数设计的次优内核，导致 MoE 性能下降。

实现拆解

1. 导入依赖调整：在 python/sglang/srt/layers/moe/ep_moe/layer.py 中，从 sglang.srt.utils 导入新增 get_int_env_var 函数，用于读取整数环境变量。
2. 初始化环境变量：在 MoriEPMoE.__init__ 方法中，添加 self.mori_moe_max_input_tokens = get_int_env_var("SGLANG_MORI_MOE_MAX_INPUT_TOKENS", 0)，默认值 0 表示禁用截断。
3. 核心截断逻辑：在 run_moe_core 方法中，添加条件判断 if self.mori_moe_max_input_tokens > 0:，若启用则使用 limit = self.mori_moe_max_input_tokens 截断 dispatch_a1、dispatch_scale、dispatch_ids 和 dispatch_weights 张量为 [:limit]，然后传递给 fused_moe。由于 mori_op.combine 只读取 [0, totalRecvTokenNum) 范围内的位置，当环境变量设置正确（≥ totalRecvTokenNum）时，截断后的输出可直接使用，无需填充回原始大小。
4. 文档更新：在 docs/references/environment_variables.md 中添加新环境变量 SGLANG_MORI_MOE_MAX_INPUT_TOKENS 的说明，包括其作用、默认值和重要警告：值必须 ≥ 实际接收令牌数，否则会导致不正确的结果。

关键文件：

- python/sglang/srt/layers/moe/ep_moe/layer.py（模块 MoE 层；类别 source；类型 core-logic；符号 MoriEPMoE.init, MoriEPMoE.run_moe_core）：核心实现文件，负责读取环境变量并在 MoE 计算前截断调度张量。
- docs/references/environment_variables.md（模块 环境变量；类别 docs；类型 documentation）：文档文件，新增了环境变量 SGLANG_MORI_MOE_MAX_INPUT_TOKENS 的说明。

关键符号：MoriEPMoE.init, MoriEPMoE.run_moe_core

关键源码片段

python/sglang/srt/layers/moe/ep_moe/layer.py

核心实现文件，负责读取环境变量并在 MoE 计算前截断调度张量。

```
class MoriEPMoE:
    def __init__(self, ...):
        # ... 其他初始化代码 ...
        self.expert_mask[expert_start_idx:expert_end_idx] = 1

        # 新增：读取环境变量 SGLANG_MORI_MOE_MAX_INPUT_TOKENS，默认值为 0（禁用）
        self.mori_moe_max_input_tokens = get_int_env_var(
            "SGLANG_MORI_MOE_MAX_INPUT_TOKENS", 0
        )

    def run_moe_core(self, dispatch_output: DispatchOutput):
        # 解包调度输出张量
        dispatch_a1 = dispatch_output.hidden_states # 形状为 (M, hidden_size)，M
        是完整缓冲区大小
        dispatch_scale = dispatch_output.hidden_states_scale
        dispatch_ids = dispatch_output.topk_ids
        dispatch_weights = dispatch_output.topk_weights
        dispatch_recv_token_num = dispatch_output.num_recv_tokens_per_expert #
        实际有效令牌数

        # 截断调度张量以减少 MoE 计算中的填充行开销
        # 只有前 dispatch_recv_token_num 行是有效的，但缓冲区 M 通常更大
        # mori combine 操作只读取 [0, totalRecvTokenNum)
        范围内的数据，因此截断后的输出可直接传递，无需填充回原始大小
        if self.mori_moe_max_input_tokens > 0:
            limit = self.mori_moe_max_input_tokens
            dispatch_a1 = dispatch_a1[:limit] # 截断隐藏状态
            if dispatch_scale is not None:
                dispatch_scale = dispatch_scale[:limit] # 截断缩放因子（如果存在）
            dispatch_ids = dispatch_ids[:limit] # 截断 topk IDs
            dispatch_weights = dispatch_weights[:limit] # 截断 topk 权重

        # 后续 MoE 计算使用截断后的张量
```

```
    # ...
```

评论区精华

review 中，gemini-code-assist[bot] 提出了两个关键建议：

1. 安全性检查：当前实现中，如果用户设置的环境变量值小于实际接收令牌数，会导致静默数据损坏。建议在代码中使用 max(self.mori_moe_max_input_tokens, sum(dispatch_recv_token_num)) 作为截断限制，确保即使环境变量配置错误也能保持正确性，同时仍能从减少输入大小中获益。
2. 文档更新：相应地更新文档，说明该环境变量会取最大值以确保安全。 这些建议未被采纳，PR 按原方案合并，但突出了潜在风险。

- SGLANG_MORI_MOE_MAX_INPUT_TOKENS 的安全性检查 (correctness): 建议未被采纳，PR 按原方案合并，但突出了潜在风险。
- 环境变量文档更新 (documentation): 文档未按建议更新，保留了原警告内容。

风险与影响

- 风险：正确性风险：如果 SGLANG_MORI_MOE_MAX_INPUT_TOKENS 设置的值小于实际的 totalRecvTokenNum，截断会丢失有效数据，导致 MoE 计算结果错误，且无运行时检查或报错，可能引发静默数据损坏。 性能风险：若环境变量设置不当（如过大），可能无法有效优化内核选择；若过小，则导致错误。 兼容性风险：新增环境变量默认禁用，不影响现有行为，但用户需正确配置才能获益，且文档警告了错误配置的后果。
- 影响：对用户：高级用户可通过设置该环境变量优化 AMD GPU 上 MoriEP MoE 性能，但需谨慎配置以避免错误。 对系统：在正确配置下，可减少 MoE 计算中的填充开销，提升推理性能；错误配置可能导致模型输出不正确。 对团队：引入了新的调优参数，增加了 MoE 模块的复杂性，但未改变核心接口，影响范围有限。
- 风险标记：静默数据损坏风险，缺少运行时检查，环境变量配置敏感

关联脉络

- PR #22924 [UnifiedRadixTree]: Add HiCache hook interface for TreeComponent: 同属 MoE 相关功能增强，涉及缓存和性能优化。
- PR #22842 [CPU] Add gemma4_rmsnorm_cpu kernel: 同属内核优化类 PR，关注性能提升。