PR #16859 完整报告

sgl-project/sglang

[RL] DeepEP support for `--enable-return-routed-experts`

合并时间：2026-05-06 11:01

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/16859

执行摘要

- 一句话：DeepEP 支持 routed experts 捕获与 all-gather
- 推荐动作：值得精读，尤其是 capture 和 _get_local_slice 的设计权衡，以及测试如何构造有效覆盖。对从事分布式 MoE 和 RL 捕获的同学有参考价值。

功能与动机

从 slime 补丁迁移回上游（issue #1316），需要在 DeepEP a2a 模式下正确捕获 routed experts。之前的 tp=2 dp=2 测试配置导致 attn_tp_size=1，gather 路径从未被执行。

实现拆解

1导入新增与预分配gatherbuffer（python/sglang/srt/state_capturer/routed_experts.py）：新增 attn_tp_all_gather_into_tensor、get_attention_tp_size 和 get_moe_a2a_backend 导入；在 __init__ 中，若后端为 DeepEP，则预分配一个 gather_buffer，大小为 device_cache.buffer.shape[0] * attn_tp_size，用于容纳 all-gather 后的完整 topk_ids。

2. 重写 capture 方法：新增 capture 方法，当 DeepEP 启用时，先保存局部 topk_indices，然后从 gather_buffer 中切片目标区域，调用 attn_tp_all_gather_into_tensor 执行 all-gather，最后调用父类的 capture 将合并后的数据写入设备缓存。

3. 调整 _get_local_slice 条件：原来在 DP attention 下基于 DP rank 切片，现在对于 DeepEP 模式，capture 已将所有 rank 数据 gather 到 buffer 头部，因此直接使用 [0:N_local] 而不是全局偏移量，故增加 not get_moe_a2a_backend().is_deepep() 条件。

4. 测试重写（test/registered/rl/test_return_routed_experts.py）：将 baseline 和 reference 统一为 --tp 4 --dp 2 --enable-dp-attention --moe-a2a-backend deepep --deepep-mode normal，仅切换性能标志（overlap/cuda-graph/radix），确保 gather 路径被真正执行；模型换为 FP8 版本以兼容 DeepEP normal 模式（Bf16 断言过时）。

关键文件：

- python/sglang/srt/state_capturer/routed_experts.py（模块 状态捕获；类别 source；类型 entrypoint；符号 capture, init, _get_local_slice）：核心实现：新增 DeepEP 路径的 capture all-gather 与 DP 切片逻辑调整
- test/registered/rl/test_return_routed_experts.py（模块 测试；类别 test；类型 test-coverage）：测试重构：实际触发 DeepEP gather 路径，确保回归覆盖

关键符号：RoutedExpertsCapturer.create, RoutedExpertsCapturer.init, RoutedExpertsCapturer.capture, RoutedExpertsCapturer._get_local_slice

关键源码片段

python/sglang/srt/state_capturer/routed_experts.py

核心实现：新增 DeepEP 路径的 capture all-gather 与 DP 切片逻辑调整

```
# python/sglang/srt/state_capturer/routed_experts.py # head 版本

    def capture(self, layer_id: int, topk_indices: torch.Tensor):
        # 在 DeepEP 模式下，每个 attn-TP rank 只持有 topk_ids 的散列切片，
        # 需要在写入 device cache 之前跨 attn-TP 做 all-gather 恢复完整视图。
        if get_moe_a2a_backend().is_deepep():
            local_topk = topk_indices
            # gather_buffer 预分配的空间足够容纳所有 attn-TP rank 的拼接结果
            topk_indices = self.gather_buffer[
                : local_topk.size(0) * get_attention_tp_size()
            ]
            attn_tp_all_gather_into_tensor(topk_indices, local_topk)
        # 将（可能已 gather 的）topk_indices 写入设备缓存
        super().capture(layer_id, topk_indices)

    def _get_local_slice(
        self,
        forward_batch: ForwardBatch,
        can_run_graph: bool,
        cuda_graph_batch: Optional[int],
    ) -> torch.Tensor:
        # 在 DeepEP 路径下，capture() 已经将全局数据 gather 到 buffer 起始位置，
        # 每个 DP rank 的数据位于 [0:N_local] 而非全局偏移 [start_pos:end_pos]。
        # 因此仅在非 DeepEP 的 DP attention 场景才需要做 DP-rank 感知切片。
        if is_dp_attention_enabled() and not get_moe_a2a_backend().is_deepep():
            local_start_pos, local_num_tokens = get_dp_local_info(forward_batch)
            if can_run_graph:
                local_start_pos = get_attention_dp_rank() * cuda_graph_batch
            local_end_pos = local_start_pos + local_num_tokens
        else:
            local_start_pos, local_end_pos = 0, forward_batch.out_cache_loc.shape[0]
        return self.device_cache.buffer[
            local_start_pos:local_end_pos, :, : self.topk_size
        ]
```

评论区精华

PR body 中提及另一种 late-gather 实现（#17892 由 ocss884 提出），在 D2H 同步时才 gather；本 PR 保持 early-gather 方式（capture 时 gather）。关于测试配置，commit 历史显示逐步调整：先改为 tp=4 dp+deepep，然后固定 baseline/reference 仅变 perf 标志，最

后因 DeepEP 正常模式要求改为 FP8 模型并强制 --deepep-mode normal 以避免低延迟模式 buffer 不足。

- Early-gather vs late-gather for DeepEP all-gather (design): 本 PR 采用 early-gather 方案，已合并。

风险与影响

- 风险：仅测试了 DeepEP 后端，其他 MoE a2a 后端（如 libuv）上该代码路径不会触发，但未测试回归。gather_buffer 预分配会占用额外显存，显存开销随 attn_tp_size 线性增加。测试仅覆盖 H100 4-GPU 环境，AMD 和低端 GPU 未验证。另外，_get_local_slice 修改后，当 is_dp_attention_enabled() and not is_deepep() 时行为不变，但条件变化可能影响未来新后端引入时的正确性。
- 影响：直接影响使用 --enable-return-routed-experts 且配合 DeepEP a2a 后端的用户，现在能正确获取 routed experts 信息。不影响非 DeepEP 用户。测试覆盖增加，但需要 4 GPU H100 资源，CI 运行时间 400 秒。团队需维护 early-gather 实现，并与可能的 late-gather 方案保持一致性。
- 风险标记：仅覆盖 DeepEP 后端，显存开销增加，测试限于 FP8 模型，AMD CI 被禁用

关联脉络

- PR #12162 [RL] return routed experts support: 基础 PR，引入了 --enable-return-routed-experts，本 PR 是在其基础上增加 DeepEP 支持
- PR #17892 [alternative] late-gather for DeepEP routed experts: PR body 中提及的 late-gather 替代实现，与本 PR 的设计选择形成对比
- PR #24450 move topk capturers to srt/state_capturer/: 重构将 RoutedExpertsCapturer 移动到了 state_capturer/，本 PR rebase 在此之上