PR #28091 完整报告

sgl-project/sglang

[LoRA] Fix experimental fast-path multi-adapter correctness + flashinfer 0.6.12 compatibility

合并时间：2026-06-20 07:20

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/28091

执行摘要

- 一句话：修复实验 LoRA 多适配器正确性与 flashinfer 0.6.12 兼容
- 推荐动作：此 PR 值得精读，尤其是双流重叠的设计权衡和 CUDA 图捕获的保护策略。展示了如何在不破坏性能的前提下修复并发正确性 bug。对于使用实验 LoRA 路径的团队必须合入，对于其他开发者可作为 CUDA 图安全的参考。

功能与动机

来自 issue #27329 的实验路径在 flashinfer 升级后出现编译失败，且用户报告多适配器时输出乱码。三个独立但都影响正确性的 bug 需要修复。具体问题包括：flashinfer 0.6.12 修改了内部 API；cuBLAS 快速路径错误地使用适配器槽 0；双流分配在 CUDA 图捕获时导致重用损坏。

实现拆解

1. flashinfer 0.6.12 兼容：在 trtllm_fused_moe_kernel_launcher.cu 中新增 RoutingInputMode 枚举，调整 get_sf_out_offset_* 调用，新增 expertIds 参数传递为 nullptr，设置 supported_major_versions=[10, 12] 以支持 sm103。同时更新 runner.h 和 fused_permute_quant.cuh 中符号调用。
2. cuBLAS 路径单适配器限制：在 7 个 cuBLAS 分发点（sgemm_lora_a、sgemm_lora_b、qkv_lora_b、gate_up_lora_b、kv_b_lora_absorbed×3）增加 weights.shape[0]==1 检查，多适配器时自动回退到 Triton 内核。
3. 双流重叠分配提升：在 moe_overlap.py 的 fused_experts_none_to_experimental_sgl_trtllm_fp8_lora_two_stream 中，在 side-stream fork 前调用 merged_experts_fused_moe_lora_add 的 stage='routing' 预布置路由缓存，并预分配 gate_up_lora_intermediate，使 side-stream 在 CUDA 图捕获时零分配。
4. 共享加法图捕获保护：在 shared_add_overlap.py 的 maybe_overlap_staged_shared_add 中增加 torch.cuda.is_current_stream_capturing() 检测，捕获时禁止跨流 add_，回退到串行加法。
5. 导入修复：将 _pack_topk_for_flashinfer_routed 替换为 fused_pack_topk，该函数已移入 jit_kernel/trtllm_lora_temp/topk_pack.py，影响 lora_dispatch.py、sgl_fp8_moe.py、moe_overlap.py。

关键文件：

- python/sglang/srt/lora/trtllm_lora_temp/moe_overlap.py（模块 双流 LoRA；类别 source；类型 core-logic；符号 fused_experts_none_to_experimental_sgl_trtllm_fp8_lora_two_stream）：核心双流修复：提升路由分配至主流，预分配中间缓冲区，side-stream 仅启动内核，解决多适配器下 CUDA 图捕获崩溃
- python/sglang/jit_kernel/trtllm_lora_temp/data/csrc/trtllm_fused_moe_kernel_launcher.cu（模块 JIT 内核；类别 other；类型 core-logic；符号 FusedMoeLauncher, Fp8BlockScaleLauncher）：flashinfer 0.6.12 兼容性核心修改：新增 RoutingInputMode 枚举，调整 get_sf_out_offset 调用，添加 expertIds 参数，支持 sm103
- python/sglang/srt/lora/trtllm_lora_temp/shared_add_overlap.py（模块 共享叠加；类别 source；类型 core-logic；符号 maybe_overlap_staged_shared_add）：增加 CUDA 图捕获检测，禁止跨流 shared-add 重叠，防止 replay 时 output 损坏
- python/sglang/srt/lora/trtllm_lora_temp/triton_ops/virtual_experts.py（模块 路由虚专家；类别 infra；类型 infrastructure；符号 _get_routing）：新增 stage='routing' 分支，在主流预布置路由缓存，避免 side-stream 图捕获时分配

关键符号：fused_experts_none_to_experimental_sgl_trtllm_fp8_lora_two_stream, maybe_overlap_staged_shared_add, _get_routing, fused_experts_none_to_experimental_sgl_trtllm_fp8_lora, prepare_moe_common

关键源码片段

python/sglang/srt/lora/trtllm_lora_temp/moe_overlap.py

核心双流修复：提升路由分配至主流，预分配中间缓冲区，side-stream 仅启动内核，解决多适配器下 CUDA 图捕获崩溃

```
# 提升 side-chain 分配至主流，确保 CUDA 图安全
# 预布置路由缓存（stage="routing"）并预分配 shrink intermediate
merged_experts_fused_moe_lora_add(
    output=gate_up_delta,
    hidden_states=hidden_states,
    lora_a=lora_info.gate_up_lora_a_weights,
    lora_b=lora_info.gate_up_lora_b_weights,
    topk_ids=topk_ids,
    topk_weights=topk_weights,
    token_lora_mapping=token_lora_mapping,
    mul_routed_weight=False,
    experts_shared_outer_loras_a=lora_info.experts_shared_outer_loras,
    experts_shared_outer_loras_b=False,
    routing_cache=fused_lora_routing_cache,
    stage="routing",
    local_expert_offset=quant_info.local_expert_offset,
    local_num_experts=quant_info.local_num_experts,
)
# 预分配 side-stream 所需的 intermediate buffer
gate_up_lora_intermediate = hidden_states.new_empty(
```

```
        (hidden_states.shape[0],
         topk_ids.shape[1],
         lora_info.gate_up_lora_a_weights.shape[2],
        )
    )
```

python/sglang/srt/lora/trtllm_lora_temp/shared_add_overlap.py

增加 CUDA 图捕获检测，禁止跨流 shared-add 重叠，防止 replay 时 output 损坏

```
    # 防止在 CUDA 图捕获期间进行跨流 add_
    # 跨流 add_ 事件会损坏 output，因此回退到串行加法
    if torch.cuda.is_current_stream_capturing():
        # 让模型层通过 unstage_shared_expert_add 回收 staged tensor
        # 在 current_stream.wait_stream(alt_stream) 之后安全执行加法
        return None
```

评论区精华

无 Review 评论，仅两名 reviewer 批准。PR body 已详尽说明问题原因和修复方案。

- 暂无高价值评论线程

风险与影响

- 风险：修复涉及实验路径（默认关闭），风险可控。但 JIT 内核编译依赖 flashinfer 版本，sm103 支持可能需额外验证。cuBLAS 路径限制单适配器可能引入小精度变化（多适配器回退到 Triton 路径，该路径已证明正确）。双流提升和 shared-add 保护仅影响捕获模式，不影响正常执行。无新增测试，需依赖现有 CI 覆盖。
- 影响：影响范围限于使用 --moe-runner-backend experimental_sgl_trtllm 和 SGLANG_EXPERIMENTAL_LORA_OPTI=1 的用户。修复后多适配器（≥2）输出正常，flashinfer 0.6.12+ 环境可编译，CUDA 图捕获下解码稳定。TP/EP>1 场景的崩溃问题消除。对于单适配器用户，cuBLAS 路径保持不变，性能无影响。
- 风险标记：实验路径默认关闭，JIT 内核兼容性依赖 flashinfer 版本，无新增测试覆盖

关联脉络

- PR #27329 [LoRA] Experimental fast LoRA path with experimental_sgl_trtllm MoE backend for FP8 and NVFP4 models: 引入此 PR 所修复的实验路径；本 PR 修复了该路径中多个正确性问题。