PR #24314 完整报告

sgl-project/sglang

Deprecate record_nolora_graph dual MoE CUDA graph capture

合并时间：2026-05-16 10:20

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/24314

执行摘要

- 一句话：废弃 dual MoE CUDA graph 双路捕获功能
- 推荐动作：建议关注 LoRA CUDA 图捕获的开发者仔细阅读此 PR。它展示了面对特性复杂化时的回退决策过程，以及在 Triton kernel 中利用早期退出实现零开销条件化的技巧。

功能与动机

PR body 明确指出 dual-capture 路径是 "a bug factory and not that fast"，并且在 #23727 和 #23873 中持续暴露出混合 LoRA / 无 LoRA 重放时的正确性问题。因此决定回退该特性，回归更简洁的 CUDA 图捕获方案。

实现拆解

1. 移除配置项：在 ServerArgs 中删除 record_nolora_graph 字段及对应 CLI 参数 --record-nolora-graph。
2. 移除全局状态：在 python/sglang/srt/layers/moe/utils.py 中删除 RECORD_NOLORA_GRAPH 全局变量、should_record_nolora_graph() 函数及其在 initialize_moe_config() 中的赋值逻辑。
3. 简化 graph 管理：在 python/sglang/srt/model_executor/cuda_graph_runner.py 中移除模块级变量 _capture_lora_variant、访问函数 get_capture_lora_variant/_set_capture_lora_variant，以及 _default_make_graph_key/_make_graph_key/_resolve_lora_variant 方法，恢复 key 只包含 (bs, stream_idx)。
4. 统一 LoRA 录制：在 python/sglang/srt/lora/lora_moe_runners.py 中删除 _add_lora_gate_up_delta、_add_lora_down_delta、build_lora_hooks 三处的 nolora 早期退出，改为在 capture 模式中始终执行 LoRA 路径，利用 Triton kernel 在 adapter_enabled 全零时的零开销早期退出。
5. 测试无配套变更，依赖现有 LoRA CUDA graph CI（如 test_lora_cuda_graph）。

关键文件：

- python/sglang/srt/model_executor/cuda_graph_runner.py（模块 图运行器；类别 source；类型 data-contract；符号 get_capture_lora_variant, _set_capture_lora_variant, _default_make_graph_key, _make_graph_key）：核心变更文件，移除了双变体捕获的模块状态、访问函数和 graph key 构造方法，大幅简化 CUDA 图捕获逻辑。

- python/sglang/srt/layers/moe/utils.py（模块 MoE 配置；类别 source；类型 core-logic；符号 should_record_nolora_graph）：删除了 RECORD_NOLORA_GRAPH 全局变量、should_record_nolora_graph() 函数及其在 initialize_moe_config() 中的初始化逻辑，消除了双路捕获的配置入口。
- python/sglang/srt/lora/lora_moe_runners.py（模块 LoRA 运行器；类别 source；类型 dependency-wiring）：删除了三个 nolora 早期退出检查，改为始终录制 LoRA 路径；新增 has_active_lora 逻辑，在 capture 模式强制为 True，依赖 kernel 内部零开销退出。
- python/sglang/srt/server_args.py（模块 服务参数；类别 source；类型 core-logic）：移除了 record_nolora_graph 字段及其 CLI 参数注册，去掉了用户可配置该特性的入口。

关键符号：get_capture_lora_variant, _set_capture_lora_variant, _default_make_graph_key, _make_graph_key, _resolve_lora_variant, should_record_nolora_graph, _add_lora_gate_up_delta, _add_lora_down_delta, build_lora_hooks

关键源码片段

python/sglang/srt/model_executor/cuda_graph_runner.py

核心变更文件，移除了双变体捕获的模块状态、访问函数和 graph key 构造方法，大幅简化 CUDA 图捕获逻辑。

```
# 变更后：移除了 _capture_lora_variant 及其访问函数，graph key 恢复为仅包含 (bs, stream_idx)
# _capture_lora_variant: Optional[str] = None # 已删除
# def get_capture_lora_variant() ... # 已删除
# def _set_capture_lora_variant(...) ... # 已删除

def get_is_capture_mode():
    return is_capture_mode

# graph key 构造：不再包含 variant_label
# 原 _default_make_graph_key(bs, stream_idx, variant_label) 已移除
# 捕获循环中直接使用 (bs, stream_idx) 作为键
```

python/sglang/srt/lora/lora_moe_runners.py

删除了三个 nolora 早期退出检查，改为始终录制 LoRA 路径；新增 has_active_lora 逻辑，在 capture 模式强制为 True，依赖 kernel 内部零开销退出。

```
def _add_lora_gate_up_delta(
    hidden_states: torch.Tensor,
    intermediate_cache: torch.Tensor,
    ...
) -> None:
    """Add LoRA gate_up delta to intermediate_cache in-place."""
    from sglang.srt.lora.triton_ops import fused_moe_lora, merged_experts_fused_moe_lora_add

    if get_is_capture_mode():
        # After deprecation: always record LoRA path during capture.
```

```
        # adapter_enabled is all-zeros, so the Triton kernel early-exits
        # per program (zero overhead).
        has_active_lora = True
    else:
        num_loras = len(lora_info.lora_ranks)
        has_active_lora = (
            (
                lora_info.adapter_enabled[:num_loras]
                * (lora_info.lora_ranks > 0).to(lora_info.adapter_enabled.dtype)
            )
            .any()
            .item()
        )
    if not has_active_lora or lora_info is None or lora_info.max_lora_rank == 0:
        return
    # ... rest of the function unchanged
```

评论区精华

本 PR 未引发实质性讨论。gemini-code-assist[bot] 的自动评论无反馈意见，随后获得 yushengsu-thu 和 Fridge003 两位维护者的 Approve。

- 自动代码审查 (other): 无实质性讨论，PR 随后获得两位维护者批准。

风险与影响

- 风险：主要风险在于移除了 nolora 图形路径后，纯无 LoRA 推理场景可能因始终执行 LoRA kernel 而产生额外开销。但 PR 依赖 Triton kernel 在 adapter_enabled 全零时的程序级早期退出途径，理论上零开销。此外，cuda_graph_runner 中的 graph key 构造简化可能影响与子类（如 EAGLEDraftCudaGraphRunner）的兼容性，需确保子类不依赖已移除的方法。
- 影响：对用户：--record-nolora-graph CLI 选项失效，但默认值为 True，移除后不再产生双套 CUDA 图，可能略微降低显存占用和捕获时间。对系统：CUDA 图捕获逻辑简化，消除了双变体同步引入的正确性隐患。对团队：减少约 129 行有问题的代码，降低维护负担。
- 风险标记：移除优化路径，核心路径变更，回退设计

关联脉络

- PR #22809 [original dual MoE CUDA graph capture]: 本 PR 直接回退了 #22809 引入的 dual-capture 特性。
- PR #23727 [follow-up bug]: PR body 提及 #23727 暴露了混合 LoRA/ 无 LoRA 重放时的正确性问题。
- PR #23873 [follow-up bug]: PR body 提及 #23873 暴露了混合 LoRA/ 无 LoRA 重放时的正确性问题。