PR #32678 完整报告

sgl-project/sglang

[cuda_graph] Gate breakable-CG capture_inputs retention to DP-gather paths

合并时间：2026-08-04 17:23

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/32678

执行摘要

- 一句话：仅 DP 路径保留 CG 捕获输入，prefill 图内存减约 0.87 GB
- 推荐动作：值得花约 5 分钟精读。它展示了一个极小的条件门控如何消除 CUDA graph 捕获阶段不必要的对象生命周期管理，同时保持 DP 路径语义完全不变。值得关注的设计决策是用 global_num_tokens_gpu is not None 作为 capture-only 张量存在性的代理标志，避免了引入显式状态位；读者可结合 #31682 一起阅读，理解 SGLang 中 runner/backend 分层下参数传递契约的演进。

功能与动机

PR body 明确指出 #31682 引入的保留逻辑是无条件的：capture-only 张量（global_num_tokens_gpu / global_dp_buffer_len）只存在于 require_mlp_tp_gather / require_attn_tp_gather（DP attention 或 MoE/MLP 数据并行）路径下，对非 DP 模型“没有需要保活的东西”，却仍为每个 prefill 图桶 pin 一个 forward_batch，属于“纯开销，膨胀 prefill CUDA-graph capture memory”。作者希望通过 runner 侧的门控让内存保留只发生在真正需要的路径上，从而在部署资源受限（高 mem_fraction_static）时把内存归还给 KV cache 池。

实现拆解

变更共 1 个文件（+5/-1），核心是把 capture_one 调用的 capture_inputs 从无条件传递改为条件传递。

1. 定位变更入口：在 python/sglang/srt/model_executor/runner/prefill_cuda_graph_runner.py 的 CUDA graph 捕获流程中，self.backend.capture_one(...) 调用处原为 capture_inputs=forward_batch；该参数会被 BreakableCudaGraphBackend 按 shape_key 保留（self._capture_inputs[shape_key]），用于维持 DP-padding 安装的 capture-only 张量在重放时的地址稳定。
2. 增加门控条件：将参数改为 forward_batch if forward_batch.global_num_tokens_gpu is not None else None。global_num_tokens_gpu 是 DP gather 路径专属的 capture-only 张量，非 None 即代表当前批次存在需要保活的对象；非 DP 路径（如普通 TP=1）传 None，后端便不会 pin 该批次，从而消除每个 prefill 图桶的额外内存保留。
3. 断言与范围：此次改动只影响 runner 侧调用契约，BreakableCudaGraphBackend 的保留逻辑、重放逻辑均未改动；DP / MoE / MLP-DP 路径的行为与 #31682 保持一致。由于

未新增测试，回归验证依赖现有 CI 中的 DP 与非 DP prefill 测试覆盖；作者在 body 中给出了手工测量数据（8k+1k prefill：7.04 GB → 6.17 GB）作为效果证据。

关键文件：

- python/sglang/srt/model_executor/runner/prefill_cuda_graph_runner.py（模块 图捕获；类别 source；类型 data-contract）：唯一变更文件，在 capture_one 调用处将 capture_inputs 从无条件 forward_batch 改为按 global_num_tokens_gpu 存在性条件传递，决定是否让 BreakableCudaGraphBackend 保留批次对象，是本 PR 实现的核心。

关键符号：PrefillCudaGraphRunner.capture

关键源码片段

python/sglang/srt/model_executor/runner/prefill_cuda_graph_runner.py

唯一变更文件，在 capture_one 调用处将 capture_inputs 从无条件 forward_batch 改为按 global_num_tokens_gpu 存在性条件传递，决定是否让 BreakableCudaGraphBackend 保留批次对象，是本 PR 实现的核心。

```
# 以 shape_key 捕获一个 prefill CUDA graph。
# DP padding 可能在 dummy batch 上安装 capture-only 张量，
# 如 global_num_tokens_gpu / global_dp_buffer_len； BCG 后端会按
# shape_key 保留传入的 capture_inputs，以确保这些张量在重放时
# 的地址仍然有效。因此只有这类 gather 路径才需要传 forward_batch；
# 非 DP 模型没有此类张量，传 None 可避免无谓的内存保留。
self.backend.capture_one(
    shape_key,
    run_once,
    capture_inputs=(
        forward_batch
        if forward_batch.global_num_tokens_gpu is not None
        else None
    ),
    post_warmup_hook=post_warmup_hook,
)
```

评论区精华

该 PR 没有收到代码评审评论（review_comments_count=0）。Issue 评论仅包含 Gemini Code Assist 机器人关于其消费者版本服务终止的公告，以及作者的 /tag-and-rerun-ci 指令。技术上的设计权衡由作者在 PR body 中说明：以 global_num_tokens_gpu 是否非 None 作为门控，复用现有张量的存在性语义，不引入新标志位，且保持后端契约不变。

- 暂无高价值评论线程

风险与影响

- 风险：门控完全依赖 forward_batch.global_num_tokens_gpu 是否被设置：若未来非 gather 路径也设置该字段，门控会退化为原行为（无内存收益）；若某些 DP 路径未设置

该字段，则会导致需要保活的 capture-inputs 被释放，重放时可能出现地址失效。目前实现中该字段与 global_dp_buffer_len 在同一 gather 路径下成对安装，风险可控，但缺少针对性测试覆盖（本次未新增测试）。此外，该改动位于 prefill CUDA graph 捕获这一核心路径，虽然只影响内存保留，但在高 mem_fraction_static 部署下可能与 post-capture KVsizing 交互，需要观察内存回流是否带来预期的 batch 扩容。

- 影响：对非 DP 用户：prefill CUDA graph 捕获内存明显下降（实测 -0.87 GB），该内存可回流 KV cache，支撑更大 batch 或更长序列，尤其对高 mem_fraction_static 部署收益明显。对 DP / MoE / MLP-DP 用户：行为与 #31682 后保持一致，无功能变化。对团队：该 PR 明确了 runner 与 BCG backend 间 capture_inputs 的契约——仅在需要保活 capture-only 张量时传递，降低了后续维护对内存语义的认知负担。
- 风险标记：缺少测试覆盖，依赖 global_num_tokens_gpu 门控标志，prefill 图捕获内存敏感

关联脉络

- PR #31682 （材料未提供标题，PR body 中引用编号）：本 PR 正是为修复 #31682 引入的无条件 capture_inputs 保留导致的内存膨胀；#31682 让 runner 总是把 forward_batch 作为 capture_inputs 传入，本 PR 将其收紧为仅 DP-gather 路径传入。