PR #33431 完整报告

sgl-project/sglang

[Fix] Skip padded state slots in the chunked GDN kernel

合并时间：2026-08-19 04:07

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/33431

执行摘要

- 一句话：补 GDN chunk kernel 的 -1 哨兵索引回归测试，防越界崩溃
- 推荐动作：值得精读：PR body 的根因分析是理解线性注意力内核 padded 批处理边界的优质样本，尤其是 boundary_check 局限与 idle DP rank 重放场景的还原。测试设计有可借鉴处：mixed 与 all_padded 双形态 + 强制同步暴露越界 + pool 未触碰断言三者配合，能同时捕获崩溃与静默写坏两类回归。建议后续跟踪：#33810 的源码修复需保持与本测试同步；可考虑在 nightly 多 rank 套件中增加 DP + linear-attn 的真实重放用例，弥补 per-commit 单 GPU 无法覆盖的场景。

功能与动机

PR body 给出了完整的根因分析：_forward_metadata 会用 state index = -1 污染 padded request 行；decode kernel 已有 if idx >= 0 守卫，但 chunked extend kernel 没有，导致 -1 进入 initial_state + index * stride_init_state 的指针运算；boundary_check 只校验声明 block shape、不校验实际 allocation，因此拦不住。在 breakable-CUDA-graph prefill + DP attention 下，空闲 DP rank 会重放全 padded batch 的 extend，每一行都带 -1，kernel 以 cudaErrorIllegalAddress 崩溃；作者在 Qwen3-Next-80B 与 Qwen3.5-397B 上以单请求 --dp 4 即可复现，且指出 per-commit 套件没有覆盖 linear-attn hybrid 与 DP attention 的组合，所以问题此前未被发现。

实现拆解

1. 根因定位（来自 PR body 的分析）

_forward_metadata 对 padded request 行写入 state index = -1 作为哨兵；decode 路径的 fused_recurrent 已有 if idx >= 0 守卫，但 chunked extend 路径（chunk_delta_h.py）缺失，导致 -1 直达 initial_state + index * stride_init_state 指针运算，同时影响初始 state 读取与就地 final-state 写入。boundary_check 只对照声明的 block shape 校验，无法识别这种“值在类型上合法、但指向 allocation 之外”的访问。

2. 触发场景还原

breakable-CUDA-graph prefill 与 DP attention 组合下，空闲 DP rank 会重放一个全 padded batch 的 extend，每行均携带 -1，kernel 崩溃为 cudaErrorIllegalAddress。body 明确给出复现条件：Qwen3-Next-80B 与 Qwen3.5-397B，单请求配合 --dp 4 即可在 pristine main 上复现。

3. 修复与上游对齐

首个 commit 在源码侧加守卫；后续 merge 中作者发现 upstream 以 db8f3cdd11（PR #33810）落地了等效修复，将检查提升为 valid_state = index >= 0 并同时作用于 initial-state load 与 in-place store。为不与上游分叉，本分支对 chunk_delta_h.py 不再做改动，直接采用上游形式。

4. 回归测试落库（本 PR 最终变更）

在 test/registered/attention/test_chunk_gated_delta_rule.py 新增 test_padded_state_index_is_skipped（+47/-0）：构造 mixed [0, -1, 2, -1] 与 all_padded [-1] * B 两种 cache_indices 分别执行 chunked GDN kernel；通过 torch.isfinite(o.float()).all() 强制同步以暴露越界访问，并断言未命中的 state pool 行保持初值，防止 -1 被当作有效索引静默写坏 pool。

5. 配套验证

无配置、schema、部署配套改动；PR Test 与 PR Test Extra 均通过，作者在 issue 侧记录“CI Passed”。

关键文件：

- test/registered/attention/test_chunk_gated_delta_rule.py（模块 内核测试；类别 test；类型 test-coverage；符号 test_padded_state_index_is_skipped）：本 PR 的最终唯一变更文件，新增 test_padded_state_index_is_skipped 回归测试，验证 chunked GDN kernel 对 -1 哨兵 state 索引的跳过行为，防止越界崩溃与 state pool 静默写坏。

关键符号：test_padded_state_index_is_skipped

关键源码片段

test/registered/attention/test_chunk_gated_delta_rule.py

本 PR 的最终唯一变更文件，新增 test_padded_state_index_is_skipped 回归测试，验证 chunked GDN kernel 对 -1 哨兵 state 索引的跳过行为，防止越界崩溃与 state pool 静默写坏。

```
def test_padded_state_index_is_skipped(self):
    """Rows carrying state index -1 must be skipped, not addressed.

    Without the guard the sentinel reaches pointer arithmetic and
    addresses before the state pool.
    """
    device = get_device()
    dtype = torch.bfloat16
    B, T_per_seq, H, K, V, pool_size = 4, 64, 4, 128, 128, 8
    T = B * T_per_seq
    torch.manual_seed(0)

    # 初始化 state pool：chunked extend kernel 会按 cache_indices 逐行做
    # initial_state + index * stride_init_state 的指针运算，-1 会指向 pool 之前的
```

```
    # 内存，修复前表现为 cudaErrorIllegalAddress 或静默写坏相邻 allocation。
    pool_init = (
        torch.randn(pool_size, H, V, K, dtype=torch.float32, device=device) * 0.1
    )

    # B 条等长序列，cu_seqlens 用于让 kernel 按块切分元组。
    cu_seqlens = torch.zeros(B + 1, dtype=torch.long, device=device)
    cu_seqlens[1:] = (
        torch.arange(1, B + 1, dtype=torch.long, device=device) * T_per_seq
    )

    # 随机输入：q/k/v 为注意力张量，g 与 beta 为 GDN 的门控系数。
    q = torch.randn(1, T, H, K, dtype=dtype, device=device)
    k = torch.randn(1, T, H, K, dtype=dtype, device=device)
    v = torch.randn(1, T, H, V, dtype=dtype, device=device)
    g = torch.nn.functional.logsigmoid(
        torch.randn(1, T, H, dtype=dtype, device=device)
    )
    beta = torch.sigmoid(torch.randn(1, T, H, dtype=dtype, device=device))

    # mixed 覆盖部分 padded 的常规形态；all_padded 对应 idle DP rank 在
    # breakable-CUDA-graph 重放时的极端形态，修复前必现崩溃。
    for label, indices in (("mixed", [0, -1, 2, -1]), ("all_padded", [-1] * B)):
        with self.subTest(label):
            cache_indices = torch.tensor(indices, dtype=torch.int32, device=device)
            o, pool = self._run_chunk(
                pool_init, cache_indices, q, k, v, g, beta, cu_seqlens
            )

            # 强制同步：越界访问会在这一行以 CUDA error 暴露。
            self.assertTrue(torch.isfinite(o.float()).all())

            # 关键断言：padded 行既不能通过 -1 索引读写 pool，
            # 未命中的行必须保持 pool_init 的原始值。
            untouched = [s for s in range(pool_size) if s not in indices]
            self.assertTrue(
                torch.equal(pool[untouched], pool_init[untouched]),
                f"{label}: padded rows wrote into the state pool",
            )
```

评论区精华

本 PR 没有实质 review 讨论（仅 BBuf 的 APPROVED 空评，review_comments = 0），最有价值的信息分散在 commit 消息与 body 中：

1) commit 4183cce 明确说明作者发现 upstream db8f3cdd11（#33810）已落地等效 guard（valid_state = index >= 0，同时作用于 initial-state load 与 in-place store），因此主动放弃本地源码补丁、采用上游形式，这是"上游已修、本地收敛为回归测试"的典型协作模式； 2) body 完整解释了 boundary_check 为什么拦不住——它只校验声明的 block shape，不校验实

际 allocation； 3) issue 侧只有 CI 触发与结果链接，最终 “CI Passed”。

- 上游 #33810 已落地等效 guard，本 PR 收敛为纯回归测试 (other): 源码修复统一采用上游实现，本 PR 最终只提交 +47/-0 的测试变更。

风险与影响

- 风险：原始缺陷（已由上游 #33810 修复）的风险集中在 chunk_delta_h.py：padded 行的 -1 哨兵直接参与 initial_state + index * stride_init_state 指针运算，同时影响 initial-state load 与就地 final-state store，会寻址到 state pool 之前的内存，表现为 cudaErrorIllegalAddress 或静默写坏相邻 allocation；boundary_check 只对照声明的 block shape 校验，无法防护 allocation 之外的访问。残余风险：本 PR 的回归测试运行在 base-b（1-GPU）per-commit 套件中，无法直接还原 idle DP rank 重放全 padded extend 的多 rank 真实验证路径，DP attention 与 linear-attn 的组合覆盖仍是盲区。本 PR 自身为纯测试新增（+47/-0），无运行时风险，但依赖 get_device() 提供的 CUDA/bfloat16 环境。另需注意：由于本 PR 最终不含源码改动，我无法在此 PR 内直接核对 #33810 守卫的实现细节，建议结合该 PR 确认守卫同时覆盖 initial load 与 in-place store。
- 影响：用户影响：为 Qwen3-Next-80B、Qwen3.5-397B 等 GDN/ 线性注意力模型在 --dp 4 等 DP attention 配置下的稳定性提供回归防线，防止单请求即崩溃的故障复发。系统影响：避免 padded 行以 -1 索引静默写坏 state pool，保护后续 decode 阶段的中间状态正确性。团队与 CI 影响：新增测试进入 base-b（1-GPU，per-commit）内核测试套件，补上此前“linear-attn hybrid + DP attention 无任何 per-commit 覆盖”的空白；影响面集中在 attention 内核测试领域，不涉及 serve 路径、配置或部署。
- 风险标记：-1 哨兵进入指针运算，boundary_check 不校验 allocation，DP + linear-attn CI 覆盖盲区，breakable-CUDA-graph 重放路径，上游已修复，本 PR 仅补测试

关联脉络

- PR #33810 上游等效 guard 修复（标题未在材料中提供，commit db8f3cdd11）：同一修复线：上游先落地 valid_state = index >= 0 的源码守卫，本 PR 采用其形式并补齐 main 缺失的回归测试。