PR #24635 完整报告

sgl-project/sglang

Fix stuck when enabling MTP on DSA models

合并时间：2026-05-08 08:06

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/24635

执行摘要

- 一句话：修复 DSA 模型启用 MTP 时的死锁问题
- 推荐动作：此 PR 修复了高优先级 bug，改动集中、逻辑清晰，CI 已全部通过。建议尽快合并并回传到相关发布分支。值得关注的设计决策包括：frozen dataclass 在 CUDA graph replay 中的赋值模式，以及 _to_2d_context_lens 的布局规范方法。

功能与动机

根据 Issue #24571，深度求索 V3.2 及 GLM-5 等 DSA 模型在启用 MTP 后出现 hang 住的问题，CI 测试也被禁用。该 hang 由 deep_gemm 路径中 fp8_paged_mqa_logits 的 tensor 布局不匹配和 draft extend 模式未覆盖 v2 引起。

实现拆解

1. 规范 tensor 布局避免死锁：在 python/sglang/srt/layers/attention/nsa_backend.py 的 _to_2d_context_lens 函数中，将输入 seqlens 强制统一为 (N_total, 1) 形状，消除二义性使 deep_gemm.get_paged_mqa_logits_metadata 不再死锁。当输入为 2D 且列数不为 1 时先扁平化再 reshape，并保证 contiguous。
2. 扩展 draft extend 条件到 v2：在 init_forward_metadata、init_forward_metadata_capture_cuda_graph、init_forward_metadata_replay_cuda_graph 三处，将 is_draft_extend() 改为 is_draft_extend(include_v2=True)，确保 MTP v2 模式也能进入 deep_gemm 的 paged MQA logits 分支，避免因未走该分支导致数据不一致。
3. 修复冻结 dataclass 赋值错误：在 init_forward_metadata_replay_cuda_graph 中，将 metadata.paged_mqa_schedule_metadata = new_schedule 改为 object.__setattr__(metadata, "paged_mqa_schedule_metadata", new_schedule)，因为 NSAMetadata 是 frozen dataclass 直接赋值会抛 FrozenInstanceError，原代码在捕获异常后静默忽略实为隐晦 bug。
4. 重新启用 CI 测试：移除了三个测试文件中的 disabled="Disabled due to #24268. Should be fixed soon." 行，这些测试覆盖了 DSA 模型 MTP、DeepSeek V3.2 CP 单节点、以及 FP4 量化 MTP 场景，验证修复有效性。

关键文件：

- python/sglang/srt/layers/attention/nsa_backend.py（模块 注意力层；类别 source；类型 core-logic；符号 _to_2d_context_lens, init_forward_metadata, init_forward_metadata_capture_cuda_graph, init_forward_metadata_replay_cuda_graph）：核心修复文件：修改了 _to_2d_context_lens 避免死锁，扩展了 draft extend 条件到 v2，并修复了 frozen dataclass 在 CUDA graph replay 中的错误赋值。
- test/registered/8-gpu-models/test_dsa_models_mtp.py（模块 测试；类别 test；类型 test-coverage）：测试文件：移除 disabled 注释，重新启用 CI 测试覆盖 DSA 模型 MTP 场景（8 GPU H200）。
- test/registered/cp/test_deepseek_v32_cp_single_node.py（模块 测试；类别 test；类型 test-coverage）：测试文件：移除 disabled 注释，重新启用 DeepSeek V3.2 上下文并行单节点测试（8 GPU）。
- test/registered/quant/test_deepseek_v32_fp4_mtp_4gpu.py（模块 测试；类别 test；类型 test-coverage）：测试文件：移除 disabled 注释，重新启用 DeepSeek V3.2 FP4 MTP 4 GPU 测试。

关键符号：_to_2d_context_lens, init_forward_metadata, init_forward_metadata_capture_cuda_graph, init_forward_metadata_replay_cuda_graph

关键源码片段

python/sglang/srt/layers/attention/nsa_backend.py

核心修复文件：修改了 _to_2d_context_lens 避免死锁，扩展了 draft extend 条件到 v2，并修复了 frozen dataclass 在 CUDA graph replay 中的错误赋值。

```
def _to_2d_context_lens(seqlens_32: torch.Tensor, batch_size: int) -> torch.Tensor:
    # Always normalize to (N_total, 1) layout, to avoid deadlock at deep_gemm.fp8_paged_mqa_
    logits
    if seqlens_32.dim() == 2:
        if seqlens_32.size(1) == 1:
            # Already (batch, 1) — done
            return seqlens_32
        # Fall through and re-flatten if the caller already gave us a (bs, next_n)
        # view — we want (N_total, 1) regardless.
        seqlens_32 = seqlens_32.reshape(-1)
    return seqlens_32.contiguous().view(-1, 1)

# 调用处（示例来自 init_forward_metadata）：
if is_cuda() and (
    forward_batch.forward_mode.is_decode_or_idle()
    or forward_batch.forward_mode.is_target_verify()
    or forward_batch.forward_mode.is_draft_extend(include_v2=True) # 关键修复：从 include_v2=
    False 改为 True
):
    try:
        import deep_gemm
        # ...
```

```
            paged_mqa_schedule_metadata = deep_gemm.get_paged_mqa_logits_metadata(
                seqlens_32_2d, 64, deep_gemm.get_num_sms()
            )
        except (ImportError, ModuleNotFoundError):
            paged_mqa_schedule_metadata = None
```

评论区精华

无 reviewer 评论，作者自行触发 CI 并通过后合并。CI 显示测试 test_dsa_models_mtp.py（8-gpu-h200）、test_deepseek_v32_fp4_mtp_4gpu.py（4-gpu-b200）、test_deepseek_v32_cp_single_node.py（8-gpu-h200-deepep）均通过。

- 暂无高价值评论线程

风险与影响

- 风险：核心风险在于 _to_2d_context_lens 的 reshape 行为变更：原先如果输入为 2D 则直接返回，现在会检查列数并可能扁平化再 reshape，这可能改变下游消费该 tensor 的代码预期。但注释说明这是为了统一布局，且下游消费函数 deep_gemm.get_paged_mqa_logits_metadata 预期一个 (batch, 1) 形状，所以应该安全。另外 object.__setattr__ 绕过了 frozen dataclass 的不可变性，可能被其他代码误用，但这是已有模式（capture 中已有直接赋值），replay 中也用相同模式保持一致性。整体风险较低，但涉及 CUDA graph 捕获和重放路径，需防止回退。
- 影响：影响范围：修复了 DSA 模型（DeepSeek V3.2、GLM-5）在启用 MTP（含 v2）时的 hang 问题；重新激活了三个关键 CI 测试，覆盖多 GPU 场景（TP8 DP8、FP4 4GPU、CP 8GPU）。对使用 EAGLE 推测解码和 NSA attention 的用户是关键修复；不影响没有启用 MTP 的配置。
- 风险标记：冻结 dataclass 绕过，核心路径变更，CUDA graph 捕获重放

关联脉络

- PR #24571 [Bug] MTP causes hang on DSA models after rebasing deep_gemm: 此 PR 直接修复该 Issue 报告的问题。