PR #22651 完整报告

sgl-project/sglang

streaming session: spec v2 bonus accounting + comprehensive test matrix

合并时间：2026-04-16 08:12

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/22651

执行摘要

- 一句话：修复 spec v2 流式会话奖励槽会计问题，移除兼容性门并添加全面测试矩阵。
- 推荐动作：建议精读核心源码修改（特别是 eagle_info_v2.py 和 scheduler_output_processor_mixin.py），关注奖励槽会计的时序调整设计；同时浏览测试文件以理解覆盖范围，这对维护流式会话模块至关重要。

功能与动机

根据 PR body，Spec v2 (overlap + EAGLE) 在 _resolve_spec_overlap_token_ids 中事后增加 kv_committed_len，导致在完成轮次时出现一个令牌的竞争条件：cache_finished_req 在下一轮解析之前触发，使得 save_from_req 捕获的 committed 值少一个令牌，从而引起 EagleV2 继承测试失败。修复方案是模仿普通解码模式，在 prepare_for_decode 中预声明奖励槽，并在解析时调整。同时移除 tokenizer_communicator_mixin.py 中的不兼容门，因为相关会计 bug 已在其他 PR 中修复。

实现拆解

1. 修改 EagleDraftInput.prepare_for_decode：在 python/sglang/srt/speculative/eagle_info_v2.py 中，为每个请求预声明奖励槽（r.kv_committed_len += 1），确保在解码前占用额外槽位。
2. 调整 _resolve_spec_overlap_token_ids：在 python/sglang/srt/managers/scheduler_output_processor_mixin.py 中，将 req.kv_committed_len += accept_lens[i] 改为 req.kv_committed_len += accept_lens[i] - 1，减去已预声明的槽位，保持每轮总增量不变但时序提前。
3. 移除不兼容门：删除 python/sglang/srt/managers/tokenizer_communicator_mixin.py 中检查 spec v2 与流式会话不兼容的代码块，允许两者同时启用。
4. 添加测试覆盖：新建 test/registered/sessions/test_streaming_session_swa.py，包含 4 个 SWA 测试类；扩展 test/registered/sessions/test_streaming_session.py，更新常量、添加并发测试方法，覆盖 Llama、Eagle V1/V2、SWA 等配置。所有测试均启用 SGLANG_ENABLE_STRICT_MEM_CHECK_DURING_BUSY=2 以严格检查内存泄漏。
5. 配套调整：提交历史显示多次调试和修复，包括页面对齐、中止处理优化等，确保流式会话在各种场景下的正确性。

关键文件：

- python/sglang/srt/speculative/eagle_info_v2.py（模块 推测解码；类别 source；类型 core-logic；符号 prepare_for_decode）：核心逻辑文件，修改了 EagleDraftInput.prepare_for_decode 以预声明奖励槽，解决了 spec v2 时序问题。
- python/sglang/srt/managers/scheduler_output_processor_mixin.py（模块 调度器；类别 source；类型 core-logic；符号 _resolve_spec_overlap_token_ids）：核心逻辑文件，修改了 _resolve_spec_overlap_token_ids 以调整奖励槽会计，减去预声明的槽位。
- python/sglang/srt/managers/tokenizer_communicator_mixin.py（模块 令牌管理；类别 source；类型 configuration；符号 open_session）：重要配置文件，移除了 spec v2 与流式会话的不兼容检查，允许两者同时使用。
- test/registered/sessions/test_streaming_session_swa.py（模块 SWA 测试；类别 test；类型 test-coverage；符号 TestStreamingSessionSWA, TestStreamingSessionSWARetractLargePage, TestStreamingSessionSWARetractMixedChunk, TestStreamingSessionSWAAbortLeakRepro）：新增测试文件，为 SWA 模型添加流式会话测试覆盖，确保修复在滑动窗口注意力下的正确性。
- test/registered/sessions/test_streaming_session.py（模块 会话测试；类别 test；类型 test-coverage；符号 _logprob_generate, _logprob_run_one_session, _logprob_assert_no_leak, _leak_async_generate）：扩展测试文件，更新常量、添加并发测试方法，覆盖多种配置组合以验证修复。

关键符号：prepare_for_decode, _resolve_spec_overlap_token_ids, open_session

关键源码片段

python/sglang/srt/speculative/eagle_info_v2.py

核心逻辑文件，修改了 EagleDraftInput.prepare_for_decode 以预声明奖励槽，解决了 spec v2 时序问题。

```
def prepare_for_decode(self: EagleDraftInput, batch: ScheduleBatch):
    # 原有逻辑：计算所需令牌数并分配 KV 长度
    num_needed_tokens = 0
    cur_kv_lens_cpu = []
    nxt_kv_lens_cpu = []
    for r in batch.reqs:
        x = r.speculative_num_draft_tokens + 1 # 草案令牌数 + 奖励槽
        num_needed_tokens += x
        r.kv_allocated_len += x
        r.decode_batch_idx += 1
        # 新增：预声明奖励槽，模拟普通解码模式，避免时序竞争
        r.kv_committed_len += 1 # 关键修复：提前占用奖励槽位
        cur_kv_lens_cpu.append(r.kv_committed_len)
        nxt_kv_lens_cpu.append(r.kv_committed_len + x)
    # 后续张量转换和返回逻辑保持不变
    cur_kv_lens_cpu = torch.tensor(cur_kv_lens_cpu, dtype=torch.int32, device="cpu")
    nxt_kv_lens_cpu = torch.tensor(nxt_kv_lens_cpu, dtype=torch.int32, device="cpu")
    return num_needed_tokens, cur_kv_lens_cpu, nxt_kv_lens_cpu
```

python/sglang/srt/managers/scheduler_output_processor_mixin.py

核心逻辑文件，修改了 _resolve_spec_overlap_token_ids 以调整奖励槽会计，减去预声明的槽位。

```
def _resolve_spec_overlap_token_ids(
    self: Scheduler, result: GenerationBatchResult, batch: ScheduleBatch
) -> List[List[int]]:
    """解析推测解码重叠的令牌 ID，并更新提交长度。"""
    assert result.next_token_ids.is_cpu
    assert result.accept_lens.is_cpu
    next_token_ids = result.next_token_ids.tolist()
    accept_lens = result.accept_lens.tolist()
    result.num_accepted_tokens = sum(accept_lens) - len(batch.reqs)
    result.accept_length_per_req_cpu = [x - 1 for x in accept_lens]
    predict_tokens = []
    stride = self.draft_worker.speculative_num_draft_tokens
    for i, req in enumerate(batch.reqs):
        # 关键修复：减去已在 prepare_for_decode 中预声明的奖励槽
        # 这样每轮总增量为 accept_lens[i]，但时序提前，避免竞争
        req.kv_committed_len += accept_lens[i] - 1 # 调整会计逻辑
        predict_tokens.append(
            next_token_ids[i * stride : i * stride + accept_lens[i]]
        )
        req.spec_verify_ct += 1
        accepted_draft_tokens = result.accept_length_per_req_cpu[i]
        req.spec_accepted_tokens += accepted_draft_tokens
        req.update_spec_acceptance_histogram(accepted_draft_tokens)
    return predict_tokens
```

评论区精华

Review 讨论较少，仅有一个来自 gemini-code-assist[bot] 的自动化评论确认测试启用，无实质性技术交锋或争议。

- 测试启用确认 (other): 无实质争议，PR 被接受。

风险与影响

- 风险：- 回归风险：核心路径 _resolve_spec_overlap_token_ids 的修改可能影响其他推测解码逻辑，需确保无副作用。
- 测试覆盖风险：新增测试虽全面，但依赖于特定配置（如页面大小 256），可能未覆盖所有边缘情况。
- 兼容性风险：移除不兼容门后，spec v2 与流式会话的交互可能暴露未预见的 bug，尤其在高压并发场景。
- 性能风险：预声明奖励槽可能轻微增加内存占用，但整体影响应较小。
- 影响：- 用户影响：用户现在可在 spec v2 模式下使用流式会话，提升推测解码的会话支持，扩展了功能场景。

- 系统影响：修复了 KV 缓存继承错误，提高了流式会话在复杂解码模式下的正确性和稳定性。
- 团队影响：测试矩阵提供了全面的验证基准，便于后续维护和回归测试，减少了未来类似问题的调试成本。
- 风险标记：核心路径变更，测试覆盖依赖

关联脉络

- PR #22862 Streaming session: fix retract tail leak via _free_tail: 同属流式会话修复系列，解决了重试时的尾部内存泄漏问题，为本 PR 的测试覆盖提供基础。
- PR #22897 streaming session: trim spec v2 overshoot in cache_finished_req: 处理推测解码超限修剪，与本 PR 的奖励槽会计修复共同确保 spec v2 正确性。
- PR #22900 trim_overshoot: cap swa_evicted_seqlen + unit test: 扩展超限修剪以覆盖 SWA 场景，与本 PR 的测试矩阵互补。