PR #26423 完整报告

sgl-project/sglang

[RL] Fix crash when the reqs in a batch have a mix of `return_routed_experts` = True and False.

合并时间：2026-05-30 11:46

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/26423

执行摘要

- 一句话：修复混合 return_routed_experts 标志导致服务器崩溃
- 推荐动作：值得精读，尤其是 _GenerationStreamAccumulator 中对可选输出字段的 ' 全部填充 None' 策略。该模式可以推广到其他需要按批次位置对齐输出字段的场景，保持 batch_position 不变性。

功能与动机

当一批请求中同时包含 return_routed_experts=True 和 False 时，下游 dispatch（tokenizer_manager 和 multi_tokenizer_mixin）索引 recv_obj.routed_experts[i] 时会因为列表长度少于批次号而抛出 IndexError。该问题同样存在于 output_hidden_states 和 indexer_topk，因为它们的收集逻辑仅在请求选择了对应标志时才 append 数据，导致列表长度与批次号不对齐。

实现拆解

1. 在 output_streamer.py 的 _stream_output_generation 方法中，计算三个批处理级别标志 return_hidden_states、return_routed_experts、return_indexer_topk（均为 any(req.return_xxx for req in reqs if req is not skip_req)），并将它们传入 _GenerationStreamAccumulator 构造器。
2. 在 _GenerationStreamAccumulator 类中添加三个布尔字段，将三个可选输出列表字段类型改为 Optional[list]，默认 None；在 __post_init__ 中按标志初始化列表；在 accept 中当批处理标志为 True 时，为每个请求 append 其对应值（若请求未选择则为 None）。
3. 在 tokenizer_manager.py 的 _handle_batch_output 中，对 output_hidden_states 和 routed_experts 加上 if val is not None 保护，避免写入 None 到 meta_info。
4. 测试文件 test_return_routed_experts.py 延迟加载 baseline 数据（_ensure_comparison_results），新增 test_mixed_return_routed_experts_batch_alignment 方法，启动两个服务（默认和 multi-tokenizer），发送混合标志请求，验证响应。

关键文件：

- python/sglang/srt/managers/scheduler_components/output_streamer.py（模块 输出流；类别 source；类型 core-logic；符号 _GenerationStreamAccumulator, _stream_output_generation）：核心修复位置：添加批处理级别标志、修改列表初始化和

accept 逻辑，确保可选输出字段长度与批次号对齐。

- python/sglang/srt/managers/tokenizer_manager.py（模块 分词器管理；类别 source；类型 core-logic；符号 _handle_batch_output）：消费端适配：对 hidden_states 和 routed_experts 字段添加 None 检查，防止将 None 写入 meta_info 或索引错误。
- test/registered/rl/test_return_routed_experts.py（模块 测试；类别 test；类型 test-coverage；符号 _ensure_comparison_results, test_mixed_return_routed_experts_batch_alignment, _run_mixed_batch_alignment_case, _send_mixed_batch）：新增混合标志端到端测试，延迟加载 baseline 数据以支持单方法快速运行，验证默认和 multi-tokenizer 路径。

关键符号：_GenerationStreamAccumulator.post_init, _GenerationStreamAccumulator.accept, _stream_output_generation, _handle_batch_output

关键源码片段

test/registered/rl/test_return_routed_experts.py

新增混合标志端到端测试，延迟加载 baseline 数据以支持单方法快速运行，验证默认和 multi-tokenizer 路径。

```
def test_mixed_return_routed_experts_batch_alignment(self):
    # 覆盖默认 tokenizer manager 和 multi-tokenizer 路径
    self._run_mixed_batch_alignment_case([])
    self._run_mixed_batch_alignment_case(['--tokenizer-worker-num', 2])

@classmethod
def _run_mixed_batch_alignment_case(cls, other_args):
    process = popen_launch_server(
        MODEL_PATH, DEFAULT_URL_FOR_TEST,
        other_args=['--tp', 2, '--enable-return-routed-experts',
                    '--disable-cuda-graph', '--disable-piecewise-cuda-graph'] + other_args)
    try:
        responses = asyncio.run(cls._send_mixed_batch())
        cls._assert_mixed_batch_result(responses)
    finally:
        kill_process_tree(process.pid)

@classmethod
async def _send_mixed_batch(cls):
    # 两个请求：一个 return_routed_experts=False，另一个 True
    payload_no_rr = {'text': 'The quick brown fox jumps over the lazy dog.',
                     'sampling_params': {'temperature': 0, 'max_new_tokens': 16, 'ignore_eos': True}
                     ,
                     'return_routed_experts': False}
    payload_with_rr = {'text': 'The quick brown fox jumps over the lazy dog.',
                       'sampling_params': {'temperature': 0, 'max_new_tokens': 16, 'ignore_eos':
                       True},
```

```
                    'return_routed_experts': True}
    return await asyncio.gather(
        cls._post_generate(payload_no_rr),
        cls._post_generate(payload_with_rr),
    )

@classmethod
def _assert_mixed_batch_result(cls, responses):
    resp_no_rr, resp_with_rr = responses
    # 第一个请求的 routed_experts 应为 None（响应中不包含该字段）
    assert 'routed_experts' not in resp_no_rr['meta_info']
    assert 'routed_experts' in resp_with_rr['meta_info']
    assert resp_with_rr['meta_info']['routed_experts'] is not None
```

评论区精华

无（该 PR 没有 review 评论，仅由 ByronHsu 反复触发 CI 重跑）。

- 暂无高价值评论线程

风险与影响

- 风险：主要风险在于其他消费者直接访问 recv_obj.routed_experts[i] 而期望非 None 值。由于此前在混合批次下会崩溃，改为 None 后至少不会崩溃，但可能向下游传递 None 导致其他错误。不过 tokenizer_manager 已经跳过 None，且 openai 响应中不包含该字段。此外，新增的 any 扫描引入少量 CPU 开销，但批次大小通常不大，影响可忽略。
- 影响：用户：修复了启用 return_routed_experts、output_hidden_states 或 indexer_topk 且请求混合标志时的服务器崩溃，保证这些功能在混合场景下正常工作。系统：没有 API 变更，向下兼容。团队：核心逻辑集中在 output_streamer.py，新增测试覆盖关键场景。
- 风险标记：索引越界修复，None 值处理，批处理级别标志

关联脉络

- 暂无明显关联 PR