PR #27764 完整报告

sgl-project/sglang

[Spec] Extract move_accept_tokens_to_target_kvcache into spec_utils

合并时间：2026-06-10 12:55

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/27764

执行摘要

- 一句话：提取 EAGLE v2 的 KV 缓存移动函数至 spec_utils
- 推荐动作：此 PR 展示了安全的提取共享工具模式，值得参考。若计划新增其他 spec worker（如 MTP、Medusa），可直接复用此函数。

功能与动机

PR body 指出："Extract the EAGLE v2 worker method into a shared spec_utils function so other spec workers can reuse it; no behavior change。"主要动机是提升代码复用性，降低未来添加新 spec worker 的集成成本。

实现拆解

1. 在 spec_utils.py 中新增函数 move_accept_tokens_to_target_kvcache，包含 batch、accept_index、num_correct_drafts、token_to_kv_pool_allocator 参数，函数体沿用原有逻辑。
2. 为 spec_utils.py 添加必要的导入：ScheduleBatch、BaseTokenToKVPoolAllocator、next_power_of_2、maybe_detect_oob、assign_extend_cache_locs、fill_accept_out_cache_loc。
3. 从 eagle_worker_v2.py 中删除原方法，在 _finalize_accept_tree_path 中改为调用 spec_utils.move_accept_tokens_to_target_kvcache，并传入 self.token_to_kv_pool_allocator。
4. 更新 eagle_worker_v2.py 的导入：从 spec_utils 导入新函数，移除不再需要的 assign_extend_cache_locs、fill_accept_out_cache_loc、next_power_of_2。
5. 从 eagle_info_v2.py 中移除未使用的导入（assign_extend_cache_locs、fill_accept_out_cache_loc）。
6. 通过重新运行 eagle 相关测试（test_spec_eagle 系列）验证功能等价，全部通过。

关键文件：

- python/sglang/srt/speculative/spec_utils.py（模块 推测解码；类别 source；类型 core-logic；符号 move_accept_tokens_to_target_kvcache）：核心变更文件：新增 move_accept_tokens_to_target_kvcache 函数并添加所需导入，成为共享函数宿主。

- python/sglang/srt/speculative/eagle_worker_v2.py（模块 推测解码；类别 source；类型 core-logic；符号 move_accept_tokens_to_target_kvcache）：从 EagleDraftWorker 中删除原方法，改为调用 spec_utils 版本，并调整导入。
- python/sglang/srt/speculative/eagle_info_v2.py（模块 推测解码；类别 source；类型 dependency-wiring）：移除未使用的导入，清理依赖。

关键符号：move_accept_tokens_to_target_kvcache

关键源码片段

python/sglang/srt/speculative/spec_utils.py

核心变更文件：新增 move_accept_tokens_to_target_kvcache 函数并添加所需导入，成为共享函数宿主。

```
def move_accept_tokens_to_target_kvcache(
    batch: ScheduleBatch,
    accept_index: torch.Tensor,
    num_correct_drafts: torch.Tensor,
    token_to_kv_pool_allocator: BaseTokenToKVPoolAllocator,
):
    """
    Move accepted tokens (drafts + bonus) to the target KV cache.

    Args:
        batch: The batch to run.
        accept_index: The index of the accepted tokens (incl. bonus).
        num_correct_drafts: Per-req count of correct drafts (excludes bonus);
            seq_lens is advanced by ``num_correct_drafts + 1`` to cover the bonus slot.
    """
    bs = len(batch.seq_lens)
    device = batch.seq_lens.device
    # accept_index element count, NOT bs * num_draft_tokens: for topk > 1 the
    # tree exceeds the accepted chain, over-reading accept_index (illegal memory).
    size = bs * accept_index.shape[1]

    # fill_accept_out_cache_loc reads out_cache_loc[accept_index]; -1 sentinel ok.
    maybe_detect_oob(
        accept_index,
        -1,
        batch.out_cache_loc.size(0),
        "spec v2 move_accept_tokens accept_index",
    )

    tgt_cache_loc = torch.zeros(
        size,
        dtype=torch.int64,
        device=device,
    )
```

```
accept_out_cache_loc = torch.zeros(size, dtype=torch.int64, device=device)
assign_extend_cache_locs[(bs,)](
    batch.req_pool_indices,
    batch.req_to_token_pool.req_to_token,
    batch.seq_lens,
    batch.seq_lens + num_correct_drafts + 1,
    tgt_cache_loc,
    batch.req_to_token_pool.req_to_token.shape[1],
    next_power_of_2(bs),
)
fill_accept_out_cache_loc[(size,)](
    accept_index,
    batch.out_cache_loc,
    accept_out_cache_loc,
    next_power_of_2(size),
)
token_to_kv_pool_allocator.get_kvcache().move_kv_cache(
    tgt_cache_loc, accept_out_cache_loc
)
```

评论区精华

PR 无实质 review 评论，但作者通过 /rerun-test 触发了测试重跑，所有 eagle 测试均通过，无回归。

- 暂无高价值评论线程

风险与影响

- 风险：纯重构，行为不变，风险低。但函数脱离类后调用者需显式传递 token_to_kv_pool_allocator，若未来其他 worker 误用可能引发问题。现有测试覆盖了主要路径，暂未新增测试。
- 影响：对用户无影响；对开发团队，spec_utils.py 成为共享函数宿主，减少重复代码，有利于新 spec worker 的统一维护。
- 风险标记：核心路径变更

关联脉络

- 暂无明显关联 PR