PR #26931 完整报告

sgl-project/sglang

[AMD] dpsk-v4 swa loc cache support

合并时间：2026-06-02 13:37

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/26931

执行摘要

- 一句话：DeepSeek V4 SWA 位置缓存优化
- 推荐动作：建议仔细审查缓存失效逻辑的鲁棒性，尤其是 start_layer 假设和并发场景。推荐在合并前补充单元测试，验证不同层执行顺序和并发情况下的缓存行为。作者不需要对 PR 做额外操作。

功能与动机

DeepSeek V4 decode 阶段每一层都重复将完整的 KV 位置翻译为 SWA 位置，翻译操作调用两个小 kernel 约耗时 8us/layer。由于同一次 decode step 中各层的 SWA 位置不变，重复计算没有必要。PR 提出将翻译过程移到第一层并缓存结果供后续层复用。

实现拆解

1. 在 DeepSeekV4MemoryPool 类中新增 get_cached_swa_loc(raw_loc, layer_id) 方法（python/sglang/srt/mem_cache/deepseek_v4_memory_pool.py）。该方法检查环境变量 SGLANG_OPT_CACHE_SWA_TRANSLATION 是否启用缓存；若启用，则在 layer_id == self.start_layer 或 cached_loc 为空时执行 translate_loc_from_full_to_swa 并缓存，否则直接返回缓存结果。
2. 将 set_swa_key_buffer_radix_fused 和 set_swa_key_buffer_radix_fused_norm_rope 方法中的内联缓存逻辑替换为对 get_cached_swa_loc 的调用，消除重复代码。
3. 修改 DeepSeekV4DecoderLayer 中的两个 forward 路径（_forward_prepare_multi_stream_hip 和 _forward_prepare），将直接调用 translate_loc_from_full_to_swa 改为调用 get_cached_swa_loc 并传入 self.layer_id（python/sglang/srt/models/deepseek_v4.py）。
4. 添加 invalidate_loc_cache() 方法用于在映射变更时清空缓存，register_mapping 中已调用 self.cached_loc = None。

关键文件：

- python/sglang/srt/mem_cache/deepseek_v4_memory_pool.py（模块 内存池；类别 source；类型 core-logic；符号 get_cached_swa_loc, translate_loc_from_full_to_swa, invalidate_loc_cache, set_swa_key_buffer_radix_fused）：核心改动，新增 get_cached_swa_loc 缓存辅助函数，替换 set_swa_key_buffer_radix_fused 和 set_swa_key_buffer_radix_fused_norm_rope 中的内联缓存逻辑，并添加

invalidate_loc_cache 方法。

- python/sglang/srt/models/deepseek_v4.py（模块 模型层；类别 source；类型 data-contract；符号 _forward_prepare_multi_stream_hip, _forward_prepare）：修改了 DeepSeekV4DecoderLayer 的两个 forward 路径，将 translate_loc_from_full_to_swa 调用替换为 get_cached_swa_loc 并传入 self.layer_id，使模型层利用缓存机制。

关键符号：get_cached_swa_loc, invalidate_loc_cache, set_swa_key_buffer_radix_fused, set_swa_key_buffer_radix_fused_norm_rope, _forward_prepare_multi_stream_hip, _forward_prepare

关键源码片段

python/sglang/srt/mem_cache/deepseek_v4_memory_pool.py

核心改动，新增 get_cached_swa_loc 缓存辅助函数，替换 set_swa_key_buffer_radix_fused 和 set_swa_key_buffer_radix_fused_norm_rope 中的内联缓存逻辑，并添加 invalidate_loc_cache 方法。

```
def get_cached_swa_loc(self, raw_loc: torch.Tensor, layer_id: int) -> torch.Tensor:
    # 仅在环境变量开启缓存时使用缓存；否则每次都直接翻译
    if self._should_cache_swa:
        # 首次调用（缓存为空）或当前层为起始层时，执行翻译并缓存
        # 注意：依赖 layer_id == self.start_layer 来触发刷新，
        # 若首层并非 start_layer 则可能使用过期缓存（潜在风险）
        if layer_id == self.start_layer or self.cached_loc is None:
            self.cached_loc = self.translate_loc_from_full_to_swa(raw_loc)
        return self.cached_loc
    return self.translate_loc_from_full_to_swa(raw_loc)
```

python/sglang/srt/models/deepseek_v4.py

修改了 DeepSeekV4DecoderLayer 的两个 forward 路径，将 translate_loc_from_full_to_swa 调用替换为 get_cached_swa_loc 并传入 self.layer_id，使模型层利用缓存机制。

```
# 在 _forward_prepare_multi_stream_hip 和 _forward_prepare 中：
token_to_kv_pool = get_token_to_kv_pool()
# 替换前：swa_loc = token_to_kv_pool.translate_loc_from_full_to_swa(forward_batch.out_cache_
loc)
# 替换后：使用缓存辅助函数，传入当前层 ID 以触发或复用缓存
swa_loc = token_to_kv_pool.get_cached_swa_loc(
    forward_batch.out_cache_loc, self.layer_id
)
swa_cache = token_to_kv_pool.swa_kv_pool.kv_buffer[self.layer_id]
swa_page_size = token_to_kv_pool.swa_kv_pool.page_size
```

评论区精华

gemini-code-assist[bot] 指出缓存失效逻辑依赖 layer_id == self.start_layer 存在风险：

1) 若首次执行 SWA 的层级不是 self.start_layer（如某些层跳过或无需 SWA），缓存不会被更新，后续层会使用前一步骤的过期数据； 2) 在并发或交错执行场景下，cached_loc 作为实例变量可能导致竞态条件。该评论未被作者或合并者回复，但PR最终获得HaiShaw批准合并。

- 缓存失效逻辑的脆弱性 (correctness): 未获得作者或维护者回复，但 PR 最终被批准合并。

风险与影响

- 风险： 1) 缓存失效条件脆弱：layer_id == self.start_layer 假设第一个需要 SWA 的层级始终是 self.start_layer，若该假设不成立（如跳过部分层或执行顺序改变），将使用过期的 SWA 位置，导致译码错误。 2) 并发安全：cached_loc 作为类实例变量被多个 decode step 共享，若出现并发执行（如 speculative decoding 或并行推理），可能出现竞态条件。 3) 依赖环境变量：功能受 SGLANG_OPT_CACHE_SWA_TRANSLATION 控制，若该变量未正确设置或默认值未开启，优化不生效。
  - 影响：影响 DeepSeek V4 模型在 AMD GPU 上的 decode 性能，ITL 和吞吐量提升约 1.6%~2.3%（并发度 2~8）。改动范围限于两个文件，函数接口从 translate_loc_from_full_to_swa 改为 get_cached_swa_loc，对调用方透明。未引入测试覆盖，存在正确性隐患。
  - 风险标记：缓存失效逻辑脆弱，并发安全性未验证，缺少单元测试覆盖

关联脉络

- 暂无明显关联 PR