PR #31050 完整报告

sgl-project/sglang

[FullCG] Preserve attention LSE through the custom-op boundary

合并时间：2026-07-21 09:01

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/31050

执行摘要

- 一句话：保留自定义 op 边界的注意力 LSE 以支持分段前缀 MHA
- 推荐动作：此 PR 实现了 FullCG 中 LSE 传递的关键机制，是 Chunked-prefix MHA 功能的基础。建议相关开发者精读 radix_attention.py 中的操作注册和 forward 逻辑。值得关注的设计决策是将 LSE 操作作为独立注册操作，避免破坏现有接口。

功能与动机

Chunked-prefix MHA needs per-token LSE to merge suffix and cached-prefix attention states. The piecewise CUDA-graph custom op exposed only attention output and inferred MHA identity from HIP-specific state.

实现拆解

1. 重构统一注意力操作：将原来的 unified_attention_with_output 拆分为内部实现 _unified_attention_with_output_impl，新增 return_lse 和 use_mha_companion 参数。保留原有注册操作名称，但将其作为新实现的包装。
2. 新增 LSE 返回操作：注册 unified_attention_with_output_and_lse、breakable_unified_attention_with_output_and_lse 等操作，它们调用 _unified_attention_with_output_impl，根据参数决定是否返回 LSE。
3. 调整 RadixAttention.forward：根据 forward_batch.mha_return_lse 和 context.mha_companion_layers 选择 LSE 版本操作，传递 use_mha_companion 和 key_value_num_tokens。当 return_lse 为 True 时返回 (output, lse) 元组。
4. 修 DeepSeek MHA 分段前缀路径：在 _chunked_prefix_attn_mha 中显式传递 key_value_num_tokens，确保 FullCG 下 K/V token 范围正确。
5. 新增 CPU 单元测试：添加 test_radix_attention.py，通过 mock 覆盖所有四种组合，验证调度正确性。

关键文件：

- python/sglang/srt/layers/radix_attention.py（模块 注意力层；类别 source；类型 core-logic；符号 unified_attention_with_output, _unified_attention_with_output_impl, _unified_attention_with_output_and_lse_fake, unified_attention_with_output_and_lse）：核心变更文件：新增 LSE 返回的统一注意力操作，重构 forward 方法，调整操作注册。

- test/registered/unit/layers/test_radix_attention.py（模块 单元测试；类别 test；类型 test-coverage；符号 _RecordingAttentionBackend, TestRadixAttentionGraphInterface, test_forward_dispatches_all_graph_and_lse_variants）：新增 CPU 单元测试，全面覆盖所有 LSE 与 graph 组合的调度路径。
- python/sglang/srt/models/deepseek_common/attention_forward_methods/forward_mha.py（模块 DeepSeek MHA；类别 source；类型 data-contract；符号 DeepseekMHAForwardMixin._chunked_prefix_attn_mha）：在 chunked prefix 注意力调用中传递 key_value_num_tokens，确保 K/V token 范围正确。

关键符号：unified_attention_with_output, _unified_attention_with_output_impl, _unified_attention_with_output_and_lse_fake, unified_attention_with_output_and_lse, RadixAttention.forward, DeepseekMHAForwardMixin._chunked_prefix_attn_mha

关键源码片段

python/sglang/srt/layers/radix_attention.py

核心变更文件：新增 LSE 返回的统一注意力操作，重构 forward 方法，调整操作注册。

```
# RadixAttention.forward (partial)
# 根据是否返回 LSE 和是否在 breakable graph 中选择操作
return_lse = bool(forward_batch.mha_return_lse)
mha_companion_layers = context.mha_companion_layers
use_mha_companion = (
    mha_companion_layers is not None
    and mha_companion_layers[self.layer_id] is self
)
if is_in_breakable_cuda_graph():
    # Breakable graph 下的自定义 op 选择
    op = (
        breakable_unified_attention_with_output_and_lse # 注册的 LSE 操作
        if return_lse
        else breakable_unified_attention_with_output
    )
else:
    # 非 breakable graph 下的自定义 op 选择
    op = (
        unified_attention_with_output_and_lse
        if return_lse
        else unified_attention_with_output
    )
# 执行选中的操作，并捕获 LSE（若没有 LSE 则为 None）
lse = op(
    q, k, v, output, save_kv_cache, self.layer_id,
    use_mha_companion=use_mha_companion,
    key_value_num_tokens=key_value_num_tokens,
    **kwargs,
)
if return_lse:
```

```
        return output.view(-1, self.tp_q_head_num, self.v_head_dim), lse
    return output
```

test/registered/unit/layers/test_radix_attention.py

新增 CPU 单元测试，全面覆盖所有 LSE 与 graph 组合的调度路径。

```
# 模拟注意力后端，记录调用并返回固定输出 /LSE
class _RecordingAttentionBackend:
    def __init__(self, *, return_lse=True):
        self.calls = []
        self.return_lse = return_lse

    def forward(
        self, query, key, value, attention_layer, forward_batch,
        save_kv_cache, **kwargs,
    ):
        self.calls.append(SimpleNamespace(
            query=query, key=key, value=value,
            attention_layer=attention_layer,
            output=forward_batch._attn_output,
            out_cache_loc=forward_batch.out_cache_loc.clone(),
            save_kv_cache=save_kv_cache, kwargs=kwargs,
        ))
        output = torch.full_like(query, 3)
        lse = torch.full((query.shape[0], query.shape[1]), 7, dtype=torch.float32)
        return (output, lse) if self.return_lse else output

# 测试方法片段：遍历所有组合并检查调度到正确操作
def test_forward_dispatches_all_graph_and_lse_variants(self):
    layer = self._new_layer()
    query = torch.zeros((4, 2, 3))
    key = torch.zeros_like(query)
    value = torch.zeros_like(query)

    op_names = {
        (False, False): "unified_attention_with_output",
        (False, True): "unified_attention_with_output_and_lse",
        (True, False): "breakable_unified_attention_with_output",
        (True, True): "breakable_unified_attention_with_output_and_lse",
    }

    for breakable in (False, True):
        for return_lse in (False, True):
            with self.subTest(breakable=breakable, return_lse=return_lse):
                forward_batch = SimpleNamespace(
                    forward_mode=ForwardMode.EXTEND,
                    mha_return_lse=return_lse,
                )
                # 使用 ExitStack 进行 mock，验证正确操作被调用
```

评论区精华

Oasis-Git 在 review 中指出需要解决冲突并运行 CI。作者解决冲突后请求合并，Oasis-Git 最终批准。讨论简洁，无技术争议。

- 冲突解决并运行 CI (other): 作者解决冲突后，Oasis-Git 批准并合并。

风险与影响

- 风险：核心注意力路径变更可能引入回归，特别是自定义 op 边界逻辑。新增 LSE 路径需要额外计算和内存，可能影响 prefill 性能。DeepSeek MHA chunked prefix 路径修改需确认与 FlashInfer、TRT-LLM 等后端兼容。测试覆盖全面但缺少端到端性能基准。
- 影响：对使用 Chunked-prefix MHA 的用户直接影响，提供更准确的注意力合并。对普通用户无影响（保留原有路径）。团队需关注 FullCG 场景下的性能回归。
- 风险标记：核心路径变更，接口扩展，需要性能验证

关联脉络

- 暂无明显关联 PR