PR #28642 完整报告

sgl-project/sglang

[FA]Add lost params in fa varlen func

合并时间：2026-06-19 04:30

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/28642

执行摘要

- 一句话：为 flash_attn_varlen_func 补充缺失的 only_qv 参数
- 推荐动作：建议精读该 PR 的讨论，特别是 gemini-code-assist[bot] 关于 v4 下静默忽略的风险提醒。该 PR 本身原子性良好，但后续应跟进在 v4 分支中显式检测 only_qv 并抛出 NotImplementedError，以及补充参数传递的单元测试。

功能与动机

PR body 中明确说明：添加丢失的 only_qv 参数以维持接口一致性。FlashAttention v3 的底层实现已支持 only_qv 选项，但上层封装函数 flash_attn_varlen_func 未暴露该参数，导致调用方无法控制是否仅计算 QV 点积。

实现拆解

1. 在 flash_attn_varlen_func 函数的参数列表中，在 pack_gqa 之后、sm_margin 之前插入 only_qv=False 参数（位于 python/sglang/jit_kernel/flash_attention.py 第 233 行）。
2. 在该函数的 v3 分支中，将 only_qv 参数传递给 fa3_flash_attn_varlen_func。
3. v4 分支保持不变，因 v4 底层接口不支持 only_qv，该参数在 v4 下将被静默忽略。

关键文件：

- python/sglang/jit_kernel/flash_attention.py（模块 JIT 内核；类别 source；类型 core-logic；符号 flash_attn_varlen_func）：核心变更文件，为 flash_attn_varlen_func 添加了缺失的 only_qv 参数，并透传到 v3 分支。

关键符号：flash_attn_varlen_func

关键源码片段

python/sglang/jit_kernel/flash_attention.py

核心变更文件，为 flash_attn_varlen_func 添加了缺失的 only_qv 参数，并透传到 v3 分支。

```
# python/sglang/jit_kernel/flash_attention.py
# 新增 only_qv 参数（第 233 行），并传递至 fa3_flash_attn_varlen_func
def flash_attn_varlen_func(
    q, k, v,
    cu_seqlens_q, cu_seqlens_k,
```

```
    max_seqlen_q=None, max_seqlen_k=None,
    seqused_q=None, seqused_k=None,
    page_table=None, softmax_scale=None,
    causal=False, qv=None,
    q_descale=None, k_descale=None, v_descale=None,
    window_size=(-1, -1),
    attention_chunk=0, softcap=0.0,
    num_splits=1, pack_gqa=None,
    only_qv=False, # <-- 新增：控制是否仅计算 QV 点积（仅 v3 支持）
    sm_margin=0,
    return_softmax_lse=False,
    sinks=None, score_mod=None, aux_tensors=None,
    ver=3, out=None,
):
    if ver == 3:
        return fa3_flash_attn_varlen_func(
            q, k, v, cu_seqlens_q, cu_seqlens_k,
            max_seqlen_q=max_seqlen_q, max_seqlen_k=max_seqlen_k,
            seqused_q=seqused_q, seqused_k=seqused_k,
            page_table=page_table, softmax_scale=softmax_scale,
            causal=causal, qv=qv,
            q_descale=q_descale, k_descale=k_descale, v_descale=v_descale,
            window_size=window_size, attention_chunk=attention_chunk,
            softcap=softcap, num_splits=num_splits, pack_gqa=pack_gqa,
            only_qv=only_qv, # <-- 透传至底层 v3 实现
            sm_margin=sm_margin, return_softmax_lse=return_softmax_lse,
            sinks=sinks, out=out,
        )
    elif ver == 4:
        # v4 不支持 only_qv，但此处未做防御性处理（见风险分析）
        from .flash_attention_v4 import flash_attn_varlen_func as fa4_flash_attn_varlen_func
        return fa4_flash_attn_varlen_func(
            # ... 省略 ...
        )
```

评论区精华

Review 中 gemini-code-assist[bot] 指出：only_qv 仅在 FlashAttention v3 中支持；若用户在 v4 下传入 only_qv=True，该参数会被静默忽略，可能导致意外行为。建议在 v4 分支中显式抛出 NotImplementedError。该建议未被采纳，Fridge003 直接批准了 PR，可能认为现阶段 v3 是主要使用场景，v4 兼容问题可通过后续 PR 解决。

- only_qv 参数在 v4 下被静默忽略的风险 (correctness): 该建议未被采纳，PR 直接合并。团队可能认为 v3 是主体使用场景，v4 兼容问题留待后续处理。

风险与影响

- 风险：主要风险在于接口非对称性：flash_attn_varlen_func 通过对外暴露 only_qv 参数，但该参数仅在 ver==3 时生效。当用户切换到 v4 时，参数会被静默忽略，可能导致难以排

查的语义错误。此外，没有新增测试覆盖该参数的传递和 v3/v4 分支的行为差异。

- 影响：影响范围限定在 JIT kernel 层的 FlashAttention 接口，直接使用者为调用 flash_attn_varlen_func 的注意力后端代码（如 flashinfer 或 triton 后端）。用户现在可以设置 only_qv=True 来仅计算 QV 点积（v3 下），无需绕过该接口。改动极小（+2 行），无破坏性变更。
- 风险标记：接口非对称性，缺少测试覆盖

关联脉络

- 暂无明显关联 PR