

PR #31231 完整报告

sgl-project/sglang

Fix gate stride for 4D decode layouts

合并时间: 2026-07-15 11:06

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/31231>

执行摘要

- 一句话: 修复 4D decode 布局下门控张量的步长错误
- 推荐动作: 建议精读此 PR, 理解 4D 张量步长计算的细微差别。该修复虽小, 但涉及 Triton kernel 中张量步长的正确使用, 对类似数据布局问题有参考价值。

功能与动机

修复 4D decode 布局下 gate 张量的 stride 计算错误。在 KDA decode 路径中, gate 张量 `a` 的 shape 为 `[B, T, H, K]`, 属于 4D 布局, 而原有代码 `a.stride()[-2]` 获取的是 head 维度的步长, 导致在 Triton kernel 中索引 token 数据时出错。

实现拆解

在 `fused_sigmoid_gating_delta_rule_update` 函数中, 修改 `stride_a` 的计算逻辑:

1. 检查 `a` 的维度 `a.ndim`。
2. 若为 4D (即 `[B, T, H, K]`), 则使用 `a.stride()[1]`, 对应 token 维度的步长。
3. 否则 (2D 或 3D 布局), 仍然使用 `a.stride()[-2]`, 保持原有行为。该修改仅影响了该函数的第 284 行 (从 `stride_a = a.stride()[-2]` 变为条件赋值)。

关键文件:

- `python/sglang/srt/layers/attention/fla/fused_sigmoid_gating_recurrent.py` (模块 注意力; 类别 source; 类型 core-logic): 核心文件, 修改了 `stride_a` 的计算逻辑, 修复了 4D decode 布局下的 gate 步长错误。

关键符号: `fused_sigmoid_gating_delta_rule_update`

关键源码片段

`python/sglang/srt/layers/attention/fla/fused_sigmoid_gating_recurrent.py`

核心文件, 修改了 `stride_a` 的计算逻辑, 修复了 4D decode 布局下的 gate 步长错误。

```
# Both paths (KDA/GDN) advance p_a once per token, so use the token-axis stride.
# For 2D a ([T, ...]) this is stride(0); for 3D a ([B, T, ...]) this is stride(1).
# Using stride()[-2] covers GDN [T, HV] and KDA layouts ([T, HV*K] / [B, T, HV*K]).
# KDA decode also passes 4-D [B, T, H, K], where [-2] is the head stride, not the
# token stride; take dim 1 explicitly for that layout.
```

```
stride_a = a.stride()[1] if a.ndim == 4 else a.stride()[-2]
```

评论区精华

该 PR 无 review 讨论。

- 暂无高价值评论线程

风险与影响

- 风险：风险很低：改动仅一行核心逻辑，且通过条件分支明确区分了 4D 和非 4D 情况，不影响原有 2D/3D 路径。但缺少针对 4D decode 布局的单元测试，回归风险在于未来若添加新的张量布局（如 5D），当前的条件判断可能不覆盖。
- 影响：直接影响使用 4D decode 布局的 KDA 路径，修复了 gate 步长错误导致的计算结果错误。不影响 2D/3D 布局的现有逻辑。对用户而言，该修复使得 4D decode 模式可以正常工作。
- 风险标记：缺少测试覆盖

关联脉络

- 暂无明显关联 PR