PR #32118 完整报告

sgl-project/sglang

Fix nightly CI: NVFP4 cuda-graph crash, NVILA batching, CuTe paged-KV zero-size, Kimi-VL OOM

合并时间：2026-07-29 16:39

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/32118

执行摘要

- 一句话：修复 4 个夜间 CI 崩溃，涵盖内核、注意力、加载路径
- 推荐动作：建议精读。此 PR 展示了高效的多 bugfix 模式：每个修复都有独立复现脚本，使用窄范围条件来最小化风险，值得借鉴。特别是 CuTe 内核的微调条件和 Kimi-VL 的注意力封装迁移，对深入理解 sglang 推理栈有参考价值。对于测试团队，可以关注如何编写独立复现脚本（如 PR body 中描述的 standalone repro）。

功能与动机

Nightly CI (Nvidia) 运行中多次出现崩溃，影响 Kimi-VL、NVILA、MiMo-V2.5 等模型测试。PR body 明确指出每个失败都经过复现和根因分析后再修复。

实现拆解

本 PR 按四个独立 fix 组织：

1. CuTe 分页 KV 内核零大小修复(python/sglang/kernels/ops/attention/flash_attn/cute/interface.py): 在 SM100 (B300) 上，当 head_dim != head_dim_v 且 split-KV 参数 num_splits > 1 时，之前会收缩 n_block_size 到 64 以节省 SMEM。但对于 q_stage==1 (decode) 且启用分页 KV 的非 TMA 加载路径，使用 128 线程，page_entry_per_thread = n_block_size // num_threads 可能为 0。修复在收缩条件中额外排除 page_table is not None and q_stage == 1 的组合。
2. Kimi-VL 视觉注意力 OOM 修复(python/sglang/srt/models/kimi_vl_moonvit.py, kimi_vl.py): 原代码中 MoonVitEncoderLayer 使用自定义的 multihead_attention 和 sdpa_attention 函数，这些函数直接调用 flash_attn 或 torch SDPA，但未受 sglang 调度器管理。此 PR 将其替换为 sglang 的 VisionAttention 封装，该封装适配 sglang 的 flash-attn 后端并正确管理显存。因此删除了老函数，重构 MoonVitEncoderLayer 使用 VisionAttention。同时移除 QKVParallelLinear 导入，因为 VisionAttention 内部已有。
3. Kimi-VL 权重加载 KeyError 修复(python/sglang/srt/models/kimi_vl.py): 由于第 2 步中将注意力模块名称从 wqkv/wo 改为 attn.qkv_proj/attn.proj，load_weights 中需要对视觉权重进行名称 remap。在 vision 分支里加上 name = name.replace("wqkv.", "attn.qkv_proj.").replace("wo.", "attn.proj.")。

4. NVILA batching 修复(python/sglang/srt/models/nvila.py): get_image_feature 中，不同分辨率图片经过棋盘分割后 patch 数不同，使用 torch.stack 要求所有张量 shape 完全相同，导致崩溃。改为 torch.cat，并移除后续多余的 rearrange（因为输出已是平铺序列）。

配套测试(test/registered/unit/models/test_kimi_vl.py): 更新测试以反映 MoonVitEncoderLayer 的新接口：不再检查 wqkv/wo，改为检查 attn.qkv_proj/attn.proj；移除已删除 multihead_attention 函数的独立测试。

关键文件：

- python/sglang/srt/models/kimi_vl_moonvit.py（模块 视觉模型；类别 source；类型 core-logic；符号 multihead_attention, sdpa_attention, MoonVitEncoderLayer, apply_rope）：核心重构：移除自定义 multihead_attention/sdpa_attention，替换为 sglang 的 VisionAttention 封装，重构 MoonVitEncoderLayer 以使用 qkv_proj/proj。删除了约 200 行冗余代码。
- python/sglang/kernels/ops/attention/flash_attn/cute/interface.py（模块 注意力内核；类别 infra；类型 core-logic；符号 _flash_attn_fwd）：零大小崩溃修复：在 B300 上特定分页 KV + decode + diff-headdim 组合下，n_block_size 收缩导致 page_entry_per_thread=0。添加条件避免该情况。
- python/sglang/srt/models/kimi_vl.py（模块 视觉模型；类别 source；类型 data-contract；符号 KimiVLForConditionalGeneration.init, KimiVLForConditionalGeneration.load_weights）：权重加载 KeyError 修复：在 load_weights 中添加视觉权重的名称 remap，映射检查点中的 wqkv/wo 到 VisionAttention 的 qkv_proj/proj。同时移除 __init__中传递给 MoonVitPretrainedModel 的 use_tensor_parallel 参数。
- python/sglang/srt/models/nvila.py（模块 视觉模型；类别 source；类型 core-logic；符号 NVILAForConditionalGeneration.get_image_feature）：Batching 修复：将 torch.stack 改为 torch.cat，不同分辨率图片 patch 数不同的情况下不再崩溃。
- test/registered/unit/models/test_kimi_vl.py（模块 测试；类别 test；类型 test-coverage；符号 test_moonvit_uses_tensor_parallel_layers, test_moonvit_attention_accepts_precomputed_max_seqlen）：测试配套：更新断言以匹配新的 VisionAttention 接口，删除已移除函数 multihead_attention 的测试。

关键符号：_flash_attn_fwd, MoonVitEncoderLayer.init, KimiVLForConditionalGeneration.load_weights, KimiVLForConditionalGeneration.init, NVILAForConditionalGeneration.get_image_feature

关键源码片段

python/sglang/srt/models/kimi_vl_moonvit.py

核心重构：移除自定义 multihead_attention/sdpa_attention，替换为 sglang 的 VisionAttention 封装，重构 MoonVitEncoderLayer 以使用 qkv_proj/proj。删除了约 200 行冗余代码。

```
# python/sglang/srt/models/kimi_vl_moonvit.py（关键部分）
fromsglang.srt.layers.attention.visionimport (    VisionAttention,
VisionAttentionMetadata,    prepare_vision_attention_metadata, ) # 移除了
```

QKVParallelLinear 导入，因为 VisionAttention 内部管理
classMoonVitEncoderLayer(nn.Module): def__init__(self, num_heads: int, hidden_dim: int, mlp_dim: int, *, activation=F.gelu, attn_bias: bool = False, attn_drop: float = 0.0, window_size: Optional[Tuple[int, int]] = None, rope: Optional[Learnable2DInterpPosEmb] = None, ce_keep_order: bool = False, mlp_layer: int = 2, norm_layer: str = "rms_norm", use_scale: bool = False, prefix: str = "", quant_config: Optional[QuantizationConfig] = None,): super().__init__()
# 使用 sglang 的 VisionAttention 替换原生的 multihead_attention/sdpa_attention
self.attn = VisionAttention(hidden_dim=hidden_dim, num_heads=num_heads, bias=attn_bias, dropout=attn_drop, window_size=window_size, rope=rope, ce_keep_order=ce_keep_order, quant_config=quant_config, prefix=add_prefix("attn", prefix),)
# ... 其余初始化 (norm, mlp) 不变 （注意：实际片段应为完整类定义，但此处仅展示核心变更）

python/sglang/kernels/ops/attention/flash_attn/cute/interface.py

零大小崩溃修复：在 B300 上特定分页 KV + decode + diff-headdim 组合下，n_block_size 收缩导致 page_entry_per_thread=0。添加条件避免该情况。

```
# python/sglang/kernels/ops/attention/flash_attn/cute/interface.py ( 关键 diff)

if (
    arch // 10 in [10, 11]
    and head_dim != head_dim_v
    and num_splits > 1
    and not (page_table is not None and q_stage == 1) # 新增：当使用分页 KV 且为 decode 阶段时，跳过收缩避免零大小
):
    if num_n_blocks >= 64 and head_dim_v != 512:
        tile_n = 64
        num_n_blocks = (seqlen_k_loaded + tile_n - 1) // tile_n
```

python/sglang/srt/models/kimi_vl.py

权重加载 KeyError 修复：在 load_weights 中添加视觉权重的名称 remap，映射检查点中的 wqkv/wo 到 VisionAttention 的 qkv_proj/proj。同时移除 __init__中传递给 MoonVitPretrainedModel 的 use_tensor_parallel 参数。

```
# python/sglang/srt/models/kimi_vl.py load_weights 中
if "vision" in name:
    if self.vision_tower is not None:
        # MoonViT's attention is wrapped in sglang's VisionAttention,
        # whose sub-modules are named qkv_proj/proj instead of the
        # checkpoint's wqkv/wo.
        name = name.replace("wqkv.", "attn.qkv_proj.").replace(
            "wo.", "attn.proj."
        )
```

use_default_weight_loading = True

评论区精华

该 PR 的 review 过程简洁，Fridge003 直接批准无评论。但 PR body 作者详细描述了每个 bug 的复现和根因，可视为隐性讨论。关键决策点：

- 对于 CuTe 内核，选择在条件中跳过特定组合而不是全局禁用 SMEM 优化，避免了性能回退。
- 对于 Kimi-VL，选择使用 sglang 的 VisionAttention 封装而非尝试配置 transformers 的 _attn_implementation，因为后者的 dispatch 检查与 sglang 自定义模型不兼容。
- 对于权重加载，作者最初尝试在 kimi_vl_moonvit.py 内部处理，后来发现更好的方式是在 kimi_vl.py 的 load_weights 中统一 remap，这与已存在的 _KEYS_TO_MODIFY_MAPPING 模式一致。
- CuTe paged-KV 零大小根因讨论 (correctness): 在收缩条件中增加 not (page_table is not None and q_stage == 1) 以避免零大小。
- Kimi-VL OOM 修复设计权衡 (design): 使用 VisionAttention 封装，避免了 HF 的 dispatch 机制，同时也统一了视觉注意力路径。
- NVILA batching 修复 (correctness): 改为 torch.cat 直接连接可变长度的 patch 序列。

风险与影响

- 风险：每个修复都有针对性验证，但存在回归风险：
  1. CuTe 内核：跳过 SMEM 优化仅在特定组合下避免，但可能在其他配置下引入性能损失？不过修复只会让该特定组合回退到非优化的 n_block_size，应该是安全的。
  2. VisionAttention 替换：MoonVitEncoderLayer 的 forward 逻辑已重构，依赖新封装的 VisionAttention 接口，需要确认所有调用路径（包括 data-parallel 训练？但这里是推理）都适配。
  3. 权重加载 remap：名称替换可能影响其他视觉模块（如 MoonVitPretrainedModel 中可能有其他 wqkv 模式），但视觉编码器中应该只有注意力层。
  4. NVILA batching：从 stack 改为 cat 后，输出的形状序列结构与原不同（原 stack 后 rearrange 相当于 cat 后的展平），但等价；需确保下游投影层接收兼容的张量。 整体风险较低，因为每个修复都经单独复现测试。
- 影响：影响范围：4 个不同模型（Kimi-VL, NVILA, MiMo-V2.5 (CuTe 内核)）和对应的测试。 影响程度：中等——修复了 CI 通道阻塞，但对用户可见功能无新增加。性能无退化（CuTe 内核仅在特定条件回退）。 团队影响：减少夜间 CI 失败噪音，提升发布信心。 兼容性：权重加载 remap 完全向后兼容，因为检查点名称不变。
- 风险标记：kernel 边界条件，模型加载兼容，接口重构影响面

关联脉络

- PR #32612 Support DCP for Kimi Linear model: 与 Kimi-VL 相关，kimi_linear.py 与 kimi_vl.py、kimi_vl_moonvit.py 同属 Kimi 系列模型，共用一些 pattern。

- PR #31538 [diffusion] support resident layers for DiT: 不同模块但都是视觉模型修复，说明 visual 后端持续改进。
- PR #32701 [Perf] Free KV pages by segment in the paged allocator without a device sync: 涉及分页 KV 缓存，与 CuTe 内核修复中的 paged-KV 路径可能有交集。