PR #34485 完整报告

sgl-project/sglang

[AMD] Let the diffusion AITer backend take grouped-query K/V (fix Cosmos3-Nano startup)

合并时间：2026-08-19 15:17

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/34485

执行摘要

- 一句话：放开 AITer 后端 GQA 头部检查，修复 Cosmos3 ROCm 启动失败
- 推荐动作：值得精读。改动仅 11 行源码却完整示范了三个工程要点：一是 " 守卫必须对应当前后端真实能力而非历史限制 "——先查被包装库的文档与 ASM 路径代码再决定守卫条件；二是 " 错误链首帧定位 "——surface 报错（diffusers 缺属性）是红鲱鱼，真实阻塞在上一帧的 NotImplementedError，排查跨层故障时值得借鉴；三是 "CI 自动化误报的甄别 "——cooldown gate 导致零代码被执行却报 9 个失败，需要像 amd-bot 那样明确区分 " gate 失败 " 与 " 代码失败 "。

功能与动机

Cosmos3-Nano T2I 在 ROCm 上启动失败，CI 运行 multimodal-gen-test-1-gpu-amd*（run 31519319449）挂掉。PR body 指出 surface 报错是误导性的：错误链实际是 NotImplementedError: AITer backend does not support Grouped Query Attention yet. ← Cosmos3CrossAttention → USPAttention(num_heads=32, num_kv_heads=8)，随后 loader 回退到 AutoModel.from_pretrained，再因为 diffusers 没有 Cosmos3OmniTransformer 属性而报错，最终 Rank 0 scheduler 死亡。真实阻塞是构造期守卫：aiter 的 flash_attn_func/flash_attn_varlen_func 文档明确支持 MQA/GQA（"Supports multi-query and grouped-query attention (MQA/GQA) by passing in KV with fewer heads than Q"），gfx942/gfx950 ASM 快速路径也显式接受 nhead_q % nhead_k == 0，守卫在拒绝后端本就支持的形状。

实现拆解

1. 放宽头部比例校验（python/sglang/multimodal_gen/runtime/layers/attention/backends/aiter.py）：将 AITerImpl.__init__ 中拒绝一切 num_kv_heads != num_heads 的硬性检查，改为仅校验 num_heads % num_kv_heads != 0 的整除关系；异常类型从 NotImplementedError（暗示功能未实现）改为 ValueError（暗示配置非法）。依据是 aiter 的 flash_attn_func/flash_attn_varlen_func 文档声明支持 GQA/MQA，且 gfx942/gfx950 ASM 快速路径显式接受 nhead_q % nhead_k == 0 的分组形状。
2. 保持 FP8 路径契约不变：_can_use_fmha_fp8_prefill 仍会拒绝 grouped shape 并日志提示回退 BF16，因此 FP8 ASM 路径的 MHA-only 约束不受影响，无需额外改动。
3. 新增单元测试（python/sglang/multimodal_gen/test/unit/test_aiter_attention_impl.py，+41 行）：通过 pytest.importorskip("aiter") 在非 ROCm 环境整体跳过；参数化覆盖

MHA (32/32)、GQA (32/8)、MQA (32/1)、默认 None 四种合法形状，以及 32/5 这种非法比例的 ValueError 分支，并保留 __main__ 入口便于单独执行。该测试纳入 multimodal-gen-unit-test-amd* 工作流。

4. 验证策略：作者无本地 AMD GPU，仅 stub aiter 与平台模块验证构造逻辑（32/32、32/8、32/1、None 通过，32/5 拒绝，_can_use_fmha_fp8_prefill(128, 128, 128, 32, 8) 仍返回 False）；真实 ROCm 验证依赖 multimodal-gen-test-1-gpu-amd-rocm720 与 multimodal-gen-unit-test-amd。cosmos3_nano_t2i 的 run_consistency_check=True 本身即新启用的 GQA cross-attention 路径的精度检查。

关键文件：

- python/sglang/multimodal_gen/runtime/layers/attention/backends/aiter.py（模块 AITer 后端；类别 source；类型 core-logic）：核心修复文件：AITerImpl.__init__ 的头部比例守卫从严格等值改为整除校验，异常类型从 NotImplementedError 调整为 ValueError，解除了 GQA/MQA 形状的构造期封锁，是 Cosmos3 启动失败的真正修复点。
- python/sglang/multimodal_gen/test/unit/test_aiter_attention_impl.py（模块 AITer 测试；类别 test；类型 test-coverage；符号 _impl_cls, _build, test_accepts_grouped_and_multi_query_kv_heads, test_rejects_kv_heads_that_do_not_divide_the_query_heads）：新增单元测试，覆盖 MHA (32/32)、GQA (32/8)、MQA (32/1)、默认 None 四种合法构造形状与 32/5 非法比例的 ValueError 分支，并通过 importorskip 保证非 ROCm 环境自动跳过；是本次修复行为契约的回归护栏。

关键符号：AITerImpl.init, _impl_cls, _build, test_accepts_grouped_and_multi_query_kv_heads, test_rejects_kv_heads_that_do_not_divide_the_query_heads

关键源码片段

python/sglang/multimodal_gen/runtime/layers/attention/backends/aiter.py

核心修复文件：AITerImpl.__init__ 的头部比例守卫从严格等值改为整除校验，异常类型从 NotImplementedError 调整为 ValueError，解除了 GQA/MQA 形状的构造期封锁，是 Cosmos3 启动失败的真正修复点。

```
class AITerImpl(AttentionImpl):
    """
    Implementation of attention using AITemplate.
    """

    def __init__(
        self,
        num_heads: int,
        head_size: int,
        softmax_scale: float,
        causal: bool = False,
        num_kv_heads: int | None = None,
        prefix: str = "",
        dropout_p: float = 0.0,
```

```
        **extra_impl_args,
    ) -> None:
        # aiter 的 mha 入口直接接受 GQA/MQA 的 K/V（它会将每个 KV head 广播到
        # 对应组的 query head 上），因此唯一硬性要求是整除关系。FP8 ASM 路径
        # 仅支持 MHA，但 grouped shape 已在 _can_use_fmha_fp8_prefill 中被拒
        # 并回退到 BF16，无需在此构造期拦截。
        if num_kv_heads is not None and num_heads % num_kv_heads != 0:
            raise ValueError(
                f"AITer backend requires num_heads ({num_heads}) to be a "
                f"multiple of num_kv_heads ({num_kv_heads})."
            )
        self.causal = causal
        self.dropout_p = dropout_p
        self.softmax_scale = softmax_scale
```

python/sglang/multimodal_gen/test/unit/test_aiter_attention_impl.py

新增单元测试，覆盖 MHA (32/32)、GQA (32/8)、MQA (32/1)、默认 None 四种合法构造形状与 32/5 非法比例的 ValueError 分支，并通过 importorskip 保证非 ROCm 环境自动跳过；是本次修复行为契约的回归护栏。

```
# SPDX-License-Identifier: Apache-2.0
"""AITer attention impl construction guards (ROCm-only; skipped elsewhere)."""

import pytest

HEAD_SIZE = 128


def _impl_cls():
    # aiter 仅随 ROCm 发行，导入后端模块依赖它，非 ROCm 平台直接跳过。
    pytest.importorskip("aiter", reason="AITer is a ROCm-only dependency")
    from sglang.multimodal_gen.runtime.layers.attention.backends.aiter import AITerImpl

    return AITerImpl


def _build(num_heads: int, num_kv_heads: int | None):
    return _impl_cls()(
        num_heads=num_heads,
        head_size=HEAD_SIZE,
        softmax_scale=HEAD_SIZE**-0.5,
        num_kv_heads=num_kv_heads,
    )


@pytest.mark.parametrize("num_kv_heads", [32, 8, 1, None])
def test_accepts_grouped_and_multi_query_kv_heads(num_kv_heads):
    # 覆盖 MHA (32/32)、GQA (32/8)、MQA (32/1) 与默认 None 四种合法形状。
    assert _build(32, num_kv_heads).softmax_scale == pytest.approx(HEAD_SIZE**-0.5)
```

```
def test_rejects_kv_heads_that_do_not_divide_the_query_heads():
    # 32 个 query head 无法被 5 个 KV head 整除，构造期应直接失败。
    with pytest.raises(ValueError, match="multiple of num_kv_heads"):
        _build(32, 5)
```

评论区精华

核心讨论集中在两点：一是 CI 验证可信度问题——amd-bot 指出此前 9 个 CI 失败全部是 per-user cooldown gate（pr-gate/call-gate）误伤，下游测试被整体跳过，PR 变更代码实际上没有被执行过，结论是 " 零失败由本 PR 代码造成，但零代码被验证过 "，需要干净重跑；作者随后以有写权限账号重跑，绕过 gate 后 multimodal-gen-unit-test-amd-rocm720 全绿，cosmos3_nano_t2v 在 1-GPU 分区 PASSED，修复在 MI300 上确认生效。二是方案取舍——reviewer yichiche 在 APPROVE 时指出 PR#35456 试图解决同一问题，但 "your change is cleaner"，本 PR 获采纳。

- CI cooldown gate 导致 PR 代码未被实际验证 (testing): 以有写权限账号重跑后 CI 全绿，修复在真实 ROCm 硬件上得到验证。
- 与 PR#35456 的同类修复方案取舍 (design): 本 PR 方案获采纳并替代 PR#35456 的竞争性改动。

风险与影响

- 风险：
  1. 回归风险（低 - 中）：校验放宽后，此前构造期直接失败的 GQA 形状会首次进入 aiter 真实 kernel 路径（aiter.py 的 forward 分支）。虽然 PR 依据 aiter 文档与 ASM 路径代码推断安全，但对整除但异常的分组（如极端 head 数）若 aiter 存在隐藏约束，将从构造期失败变为运行时 kernel 错误，错误定位成本更高。
  2. 契约漂移风险（中）：新校验依赖 aiter 的内部 GQA 能力契约（文档化而非 sglang 自有 API）。aiter 后续版本若收紧该能力，此守卫无法感知，可能静默引入数值错误。
  3. 验证缺口（中）：作者环境无 AMD GPU，真实 kernel 路径仅靠一轮 CI（cosmos3_nano_t2v PASSED）背书，cosmos3_nano_t2i 的交叉注意力尚未有独立的数值断言。
  4. 错误信息变更（低）：异常消息从 "does not support Grouped Query Attention yet" 变为 "requires num_heads ... multiple of num_kv_heads"，依赖旧文本的日志解析或监控规则需同步更新。
- 影响：
  1. 模型 / 用户影响：Cosmos3-Nano T2I/T2V 在 ROCm 上可正常启动，multimodal-gen-test-1-gpu-amd* CI 从持续失败恢复为通过；CUDA 侧不受影响（is_cuda() 路径不经过此守卫，且该后端由 ROCm 默认启用）。
  2. 系统能力边界：AITer 后端的构造契约从 MHA-only 扩展为 MHA/GQA/MQA，异常语义从 " 未实现功能 " 修正为 " 非法配置 "，后续扩散模型若引入 GQA cross-attention 不再被误杀。

3. 团队协作：与 PR#34481（FLUX warmup crash 修复）共同清掉 2026-08-12 每日报告中 R327/R338 两个 AMD diffusion CI 回归，恢复该平台信号可信度。 - 风险标记：ROCm 专属路径，依赖 aiter 内部能力契约，缺少本地 AMD GPU 验证，新启用 GQA 路径首轮验证

关联脉络

- PR #34481 [AMD] Keep the PTX-inline-asm diffusion norm fusions off on ROCm (fix FLUX warmup crash): 同一份 multimodal-gen-test-1-gpu-amd* CI 运行中另一个独立的 FLUX 回归修复，PR body 明确互引（"that fix is #34481" / "that fix is #34485"），两者共同恢复 AMD diffusion CI 信号。
- PR #35456 （同题修复）AITer GQA 支持：Reviewer 在审核中提及：PR#35456 试图解决同一 Cosmos3 GQA 启动问题，但本 PR 改动更干净，最终本 PR 被合并。
- PR #34351 （同根因系列）AMD 扩散数值 / 内核修复：PR body 提及 "Same root cause as #34351 / #34352"：该系列同样源于 ROCm 后端能力守卫与真实数值 / 内核行为不一致导致的启动或崩溃故障，本 PR 是同一根因方向的又一环。