PR #35361 完整报告

sgl-project/sglang

[Fix]: exclude SM120 from attn-res TMA dispatch

合并时间：2026-08-20 14:32

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/35361

执行摘要

- 一句话：修复 SM120 GPU 误用 TMA 快速路径问题
- 推荐动作：该 PR 值得精读，尤其是理解硬件能力门控设计。其亮点在于将复杂的内核选择逻辑拆分为可测试的纯函数，并针对新硬件架构（SM120）给出明确回退策略。可关注跨 PR 演进中 _use_fast 的进一步优化。

功能与动机

Kimi-K3 的 attention-residual dispatcher 原先对所有 compute capability major >= 10 的设备启用 TMA 快速路径，但该路径使用的 warp-specialized TMA 内核依赖 SM100 的 tcgen05/TMEM 特性，而 SM120（如 RTX 5090）不支持，导致在 SM120 GPU 上错误选择内核。PR 描述明确指出：'This incorrectly selected the SM100-oriented TMA kernel on SM120 GPUs such as the NVIDIA GeForce RTX 5090, where the required tcgen05/TMEM path is not supported.'

实现拆解

实现拆解：

1. 提取门控逻辑：将 _use_fast 中原有的能力判断提取为独立的纯函数 _supports_attn_res_tma(capability: tuple[int, int]) -> bool，便于单测和复用。
2. 修改门控条件：将原先的 major >= 10 改为 major >= 10 and major != 12，排除 SM12x 设备。该函数接受 capability 元组，返回布尔值，默认 is_npu() 返回 False。
3. 修改 _use_fast：调用 _supports_attn_res_tma(torch.cuda.get_device_capability()) 并缓存结果，与原先逻辑一致但门控更精细。
4. 更新模块文档字符串：将 'Taken on SM100+ with H=7168.' 改为 'Taken on SM100+ except SM12x with H=7168.'。
5. 新增单元测试：新增 test/registered/unit/layers/test_attn_residual.py，用 unittest 覆盖不同 capability 组合，验证门控逻辑正确性。

关键文件：

- python/sglang/srt/layers/attn_residual.py（模块 注意力模块；类别 source；类型 core-logic；符号 _supports_attn_res_tma, _use_fast）：核心修改文件，修改了 TMA 快速路径的能力门控逻辑，防止在 SM120 设备上错误启用。

- test/registered/unit/layers/test_attn_residual.py（模块 注意力残余；类别 test；类型 test-coverage；符号 TestAttnResidual, test_tma_capability_gate）：新增的单元测试，验证门控函数逻辑的正确性，覆盖了不同 capability 组合。

关键符号：_supports_attn_res_tma, _use_fast

关键源码片段

python/sglang/srt/layers/attn_residual.py

核心修改文件，修改了 TMA 快速路径的能力门控逻辑，防止在 SM120 设备上错误启用。

```
# 提取的设备能力门控函数
# 返回当前设备是否支持 TMA 快速路径
# 条件：compute capability major >= 10 且不等于 12（排除 SM120）
def _supports_attn_res_tma(capability: tuple[int, int]) -> bool:
    """Return whether the device is eligible for the TMA fast path."""
    major, _ = capability
    # SM120 虽然 major=12 >= 10，但不支持 tcgen05/TMEM，因此排除
    return major >= 10 and major != 12


# 原来的逻辑位于 `_use_fast` 内部，现改为调用上述纯函数，便于测试
# 缓存结果到全局变量，避免重复查询
_FAST_SUPPORTED = None


def _use_fast(hidden_size: int) -> bool:
    """The TMA kernel needs SM100+ except SM12x (tcgen05, cp.async.bulk)
    and its H=7168 template; everything else takes the triton pipeline."""
    global _FAST_SUPPORTED
    if is_npu():
        return False
    if _FAST_SUPPORTED is None:
        # 调用门控函数并缓存
        _FAST_SUPPORTED = _supports_attn_res_tma(torch.cuda.get_device_capability())
    return _FAST_SUPPORTED and hidden_size == 7168
```

test/registered/unit/layers/test_attn_residual.py

新增的单元测试，验证门控函数逻辑的正确性，覆盖了不同 capability 组合。

```
# 单元测试：验证门控函数对于不同架构的判断
# 该测试在 CPU CI 上运行，不依赖具体 GPU 硬件
class TestAttnResidual(unittest.TestCase):
    def test_tma_capability_gate(self):
        # SM9x 不支持
        self.assertFalse(_supports_attn_res_tma((9, 0)))
        # SM10x 支持
        self.assertTrue(_supports_attn_res_tma((10, 0)))
        self.assertTrue(_supports_attn_res_tma((10, 3)))
```

```
        # SM11x 支持
        self.assertTrue(_supports_attn_res_tma((11, 0)))
        # SM12x 不支持
        self.assertFalse(_supports_attn_res_tma((12, 0)))
        # SM13x 支持
        self.assertTrue(_supports_attn_res_tma((13, 0)))
```

评论区精华

该 PR 无 code review 评论和 review 评论，只有 CI 触发注释和 GitHub 评论。BBuf 已批准（APPROVED）。

- 暂无高价值评论线程

风险与影响

- 风险：风险分析：
  - 兼容性风险：修改了门控逻辑，可能影响 SM10x、SM11x 等设备上 TMA 路径的启用，但这些架构仍满足条件（major >= 10 且 != 12），因此不受影响。
  - 回归风险：_use_fast 中原有的 is_npu() 检查保留，但 _FAST_SUPPORTED 的缓存逻辑不变，未引入额外回归风险。
  - 测试风险：新增单测仅覆盖纯函数逻辑，未在真实 SM120 设备上验证回退行为。
- 影响：影响分析：
  - 用户影响：修复了 SM120（RTX 5090）用户可能遇到的注意力残差计算错误或性能回退问题，提升这些设备的兼容性和正确性。
  - 系统影响：影响范围仅限于 attn_residual.py 的 TMA 路径门控，不影响其他模块。
  - 团队影响：新增了可单测的门控函数，便于后续维护和扩展。
  - 风险标记：新硬件兼容性修复，测试覆盖有限

关联脉络

- PR #34546 [XPU] Fix/kimi linear xpu: 同为 Kimi 系列模型相关修复，关注硬件兼容性，可能共享相似的回退逻辑。