PR #27869 完整报告

sgl-project/sglang

Fix Qwen3.5 deterministic batch-invariant logprobs

合并时间：2026-06-14 14:23

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/27869

执行摘要

- 一句话：修复 Qwen3.5 确定性推理 logprobs 受 batch 大小影响的问题
- 推荐动作：值得精读，尤其是对数值一致性和 kernel 级别回归控制感兴趣的工程师。设计模式：用统一门控函数 is_batch_invariant_mode_enabled() 条件性禁用依赖 batch 大小的性能优化，是控制确定性推断数值一致性的好方法。建议将来类似的 batch size 敏感 kernel 也采用同样的门控。

功能与动机

Qwen3.5 确定性推理存在两个由 batch 大小不一致引起的数值路径：一是 FLA gated layernorm 的 ROWS_PER_BLOCK 根据行数 M 选择，改变浮点归约形状；二是 fused MoE top-k 对 num_tokens ≤ 32 使用 torch.compile 加速，否则用 Triton，导致不同 batch 大小下最后的 reduce 实现不同。详见 PR body。

实现拆解

1. 在 python/sglang/srt/layers/attention/fla/layernorm_gated.py 的 calc_rows_per_block 中增加 is_batch_invariant_mode_enabled() 条件，当启用时直接返回 MAX_ROWS_PER_BLOCK=4，避免行数通过 sm_count 影响归约形状。
2. 在 python/sglang/srt/layers/moe/moe_runner/triton_utils/fused_moe.py 中新增顶层函数 _use_moe_sum_reduce_torch_compile(num_tokens)，返回 num_tokens <= 32 and not is_batch_invariant_mode_enabled()；将原 inplace_fused_experts 中两处 if num_tokens <= 32 替换为调用此函数，确保 batch-invariant 模式下始终使用 Triton sum-reduce 而非 torch.compile 路径。
3. 新增 test/registered/attention/test_qwen35_deterministic.py，基于 TestDeterministicBase 构建 TestQwen35Fa3Deterministic 类，配置 TP=4、FA3 等参数，注册到 CI extra-b 阶段，定期验证 Qwen3.5-35B-A3B 的确定性端到端行为。
4. 根据 review 要求将测试从 nightly 调整到 extras（由 suite= 参数控制）。

关键文件：

- python/sglang/srt/layers/attention/fla/layernorm_gated.py（模块 注意力层；类别 source；类型 core-logic）：修改了 calc_rows_per_block 函数，添加 batch-invariant 模式检查，强制返回 MAX_ROWS_PER_BLOCK 以避免 ROWS_PER_BLOCK 随 batch 大小变化导致的数值漂移。

- python/sglang/srt/layers/moe/moe_runner/triton_utils/fused_moe.py（模块 融合 MoE；类别 source；类型 core-logic；符号 _use_moe_sum_reduce_torch_compile）：新增 _use_moe_sum_reduce_torch_compile 函数门控 torch.compile 路径，在 batch-invariant 模式下强制使用 Triton sum-reduce，确保 MoE top-k 归约的数值一致性。
- test/registered/attention/test_qwen35_deterministic.py（模块 测试；类别 test；类型 test-coverage；符号 TestQwen35Fa3Deterministic, get_model, get_server_args）：新增端到端确定性测试，基于 TestDeterministicBase 验证 Qwen3.5-35B-A3B 在 FA3 后端下的 batch-invariant 行为，注册到 CI extra-b 阶段。

关键符号：calc_rows_per_block, _use_moe_sum_reduce_torch_compile, TestQwen35Fa3Deterministic

关键源码片段

python/sglang/srt/layers/attention/fla/layernorm_gated.py

修改了 calc_rows_per_block 函数，添加 batch-invariant 模式检查，强制返回 MAX_ROWS_PER_BLOCK 以避免 ROWS_PER_BLOCK 随 batch 大小变化导致的数值漂移。

```
def calc_rows_per_block(M: int, device: torch.device) -> int:
    # 当 batch-invariant 或分段 CUDA graph 启用时，使用恒定
    # MAX_ROWS_PER_BLOCK 以避免 batch 大小影响浮点归约数值
    if is_batch_invariant_mode_enabled() or check_cuda_graph_backend(
        Phase.PREFILL, Backend.TC_PIECEWISE
    ):
        return MAX_ROWS_PER_BLOCK
    sm_count = _get_sm_count(device)
    rows_per_block = next_power_of_2(cdiv(M, 2 * sm_count))
    rows_per_block = min(rows_per_block, MAX_ROWS_PER_BLOCK)
    return rows_per_block
```

评论区精华

主要讨论围绕新建测试的 CI 归属：hnyls2002 建议 suite='nightly-4-gpu' 改为 extras，作者 zyzshishui 回复已修改。随后 hnyls2002 批准了 PR。没有其他技术争议。

- 测试 CI 归属 (testing): 作者已修改 register_cuda_ci 的 suite 参数。

风险与影响

- 风险：本 PR 修改仅在 batch-invariant 模式下生效（通过 is_batch_invariant_mode_enabled() 门控），默认行为完全不变，因此对非确定性推理无回归风险。在 batch-invariant 模式下，禁用 torch.compile 小 batch 加速路径可能带来少量性能损失（约 16-33% 吞吐下降已在 PR body 测量），但该模式本身以确定性为优先，性能折中可接受。数值一致性风险极低，因为改动强制使用更保守的恒定参数和 Triton 实现，实测输出差异为 0。建议关注新增测试的稳定性。
- 影响：对 Qwen3.5 用户在启用 --enable-deterministic-inference 后获得稳定、与 batch 无关的 logprobs。其他模型不受影响。团队需注意新增 batch_invariant_ops 模块的依赖

（未在该 PR 新增但被导入），未来可能扩展至其他层。测试注册在 *extras* 阶段，需要额外 4-GPU 资源（*estimated* 360s）。

- 风险标记：核心路径变更，性能折中，测试覆盖依赖硬件

关联脉络

- 暂无明显关联 PR