PR #45219 完整报告

vllm-project/vllm

[ROCm][CI] Fix nixl tests

合并时间：2026-06-24 02:11

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/45219

执行摘要

- 一句话：修复 ROCm NIXL 测试的 head_size 和 Mamba 同步问题
- 推荐动作：值得精读，特别是 head_dim 契约的防御性处理和 ROCm 同步的添加理由。变更设计清晰，测试覆盖充分。对于维护 ROCm 或 KV 传输的工程师有参考价值。

功能与动机

PR body 指出 deepseek-ai/deepseek-vl2-tiny 的 HF 配置在加载后 head_dim 被具体化为 0，导致 ModelArchitectureConfig.head_size 与实际注意力 head size（128）不一致，破坏了 ROCm NIXL 非对称 TP 路径上的 KV 传输。另外，在 Hybrid SSM 测试中 ROCm 上 Mamba 直接 GPU 接收后存在数据可见性竞争，需要同步。

实现拆解

1. 修复 head_size 契约：在 vllm/transformers_utils/model_arch_config_convertor.py 的 get_head_size 中，将 head_dim 检查从简单的 is not None 增强为 is not None and head_dim > 0，跳过被具体化为 0 的字段，从而 fallback 到 hidden_size // num_attention_heads 计算。
2. 添加 ROCm 设备同步：在 vllm/distributed/kv_transfer/kv_connector/v1/nixl/base_worker.py 的 get_finished 返回前，调用新增的 _sync_device_after_mamba_recv，仅当运行在 ROCm、存在 Mamba 层、不使用 host buffer 并且有成功接收时执行 torch.accelerator.synchronize()。
3. 更新 NIXL EP 导入测试：在 tests/v1/kv_connector/nixl_integration/test_nixl_imports.py 中，将硬编码的 .so 扫描替换为 _import_nixl_ep_cpp 智能模块发现，支持 CUDA 版本特定 wheel 布局。
4. 补充回归测试：在 tests/v1/kv_connector/unit/test_nixl_connector_hma.py 中添加 test_sync_device_after_mamba_recv_gates 参数化单元测试；在 tests/config/test_model_arch_config.py 中添加 test_head_size_falls_back_when_head_dim_is_zero。
5. 调整 CI 配置：在 .buildkite/test_areas/disaggregated.yaml 中为现有 nixl 测试添加 AMD (mi300_4) mirror；在 .buildkite/test-amd.yaml 中移除 MI250 上过时的测试，添加 Spec Decode 测试，并调整超时。

关键文件：

- vllm/distributed/kv_transfer/kv_connector/v1/nixl/base_worker.py（模块 KV 传输；类别 source；类型 core-logic；符号 _sync_device_after_mamba_recv）：修复核心：在 get_finished 中添加 ROCm Mamba 同步屏障，保证数据可见性。
- vllm/transformers_utils/model_arch_config_convertor.py（模块 模型配置；类别 source；类型 data-contract）：修复 head_dim=0 契约，避免 KV 传输使用错误 head size。
- tests/v1/kv_connector/unit/test_nixl_connector_hma.py（模块 单元测试；类别 test；类型 test-coverage；符号 test_sync_device_after_mamba_recv_gates）：新增参数化单元测试覆盖同步屏障的触发条件。
- tests/v1/kv_connector/nixl_integration/test_nixl_imports.py（模块 集成测试；类别 test；类型 test-coverage；符号 _import_nixl_ep_cpp）：重构 NIXL EP 导入测试以支持 CUDA 版本特定 wheel。
- tests/config/test_model_arch_config.py（模块 配置测试；类别 test；类型 test-coverage；符号 test_head_size_falls_back_when_head_dim_is_zero）：添加 head_dim=0 fallback 回归测试。
- .buildkite/test_areas/disaggregated.yaml（模块 CI 配置；类别 config；类型 configuration）：为 nixl 测试添加 AMD mirror，提高 CI 覆盖。
- tests/v1/kv_connector/nixl_integration/config_sweep_accuracy_test.sh（模块 测试脚本；类别 test；类型 test-coverage）：修复脚本中变量获取方式。
- .buildkite/test-amd.yaml（模块 CI 配置；类别 config；类型 configuration）：移除过时测试，新增 Spec Decode 测试，调整超时。

关键符号：_sync_device_after_mamba_recv, _import_nixl_ep_cpp, get_head_size, test_sync_device_after_mamba_recv_gates, test_head_size_falls_back_when_head_dim_is_zero, get_finished

关键源码片段

vllm/distributed/kv_transfer/kv_connector/v1/nixl/base_worker.py

修复核心：在 get_finished 中添加 ROCm Mamba 同步屏障，保证数据可见性。

```
# 在 get_finished 中，所有后处理完成后、超时处理前插入同步
self._sync_device_after_mamba_recv(done_recving, failed_recv_reqs)

# Handle timeout to avoid stranding blocks on remote.
# ... 后续保持不变

def _sync_device_after_mamba_recv(
    self,
    done_recving: set[str],
    failed_recv_reqs: set[str],
) -> None:
    """Synchronize ROCm direct-GPU Mamba receives before model execution."""
    # 仅在以下所有条件满足时执行同步：
    # 1. 当前平台是 ROCm；2. 模型包含 Mamba 层；
    # 3. 不使用 host buffer（直接 GPU 传输）；
```

```
    # 4. 存在成功接收的请求（排除失败请求）。
    if (
        not current_platform.is_rocm()
        or not self._has_mamba
        or self.use_host_buffer
        or not (done_recving - failed_recv_reqs)
    ):
        return
    # 显式同步设备，确保所有 GPU 写入完成
    torch.accelerator.synchronize()
```

vllm/transformers_utils/model_arch_config_convertor.py

修复 head_dim=0 契约，避免 KV 传输使用错误 head size。

```
def get_head_size(self) -> int:
    # 处理 MLA 分支 ...
    # 注意：某些配置类（如 DeepseekVLV2TextConfig）可能将缺失的 head_dim 具体化为 0
    # 先前仅判断 is not None，现在增加 >0 检查以跳过被清零的字段
    if (
        head_dim := getattr(self.hf_text_config, "head_dim", None)
    ) is not None and head_dim > 0:
        return head_dim
    # 尝试 hidden_size_per_head 字段
    if getattr(self.hf_text_config, "hidden_size_per_head", None) is not None:
        return self.hf_text_config.hidden_size_per_head
    # 最终 fallback 到 hidden_size // num_attention_heads
    if (total_num_attention_heads := self.get_total_num_attention_heads()) == 0:
        return 0
    return self.get_hidden_size() // total_num_attention_heads
```

评论区精华

模型替换决策：NickLucche 建议将 deepseek-vl2-tiny 替换为实际的 MLA 模型 DeepSeek-V2-Lite-Chat，以避免非标准配置的维护负担。AndreasKaratzas同意并执行迁移。

同步屏障必要性：NickLucche 质疑 base_worker 中新增同步屏障的意图，AndreasKaratzas 解释这是为 ROCm 上 Hybrid SSM 路径中 Mamba 直接 GPU 传输后添加的 fence，防止下级 kernel 读到过期数据。NickLucche 询问 #45357 是否已修复该竞争，回复确认未修复，因此保留同步。

性能权衡：NickLucche 最终 Approved 但指出同步可能引入 ROCm 性能下降，建议后续更精确地处理竞争。

- 测试模型替换为 DeepSeek-V2-Lite-Chat (design): 用 deepseek-ai/DeepSeek-V2-Lite-Chat 替换 deepseek-ai/deepseek-vl2-tiny。
- ROCm Mamba 设备同步的必要性 (correctness): 保留同步，但后续应考虑更精准的同步策略以避免性能开销。
- CI 测试超时设置 (performance): 接受当前超时设置。

风险与影响

- 风险：
  1. 性能风险：_sync_device_after_mamba_recv 在每轮 get_finished 中可能引入额外同步开销，尤其是当 Mamba 模型启用时。但当前仅当成功接收且未使用 host buffer 时触发，影响面有限。
  2. head_dim 契约变更风险：对 head_dim=0 的跳过可能会影响未来类似配置的模型，但 fallback 逻辑已存在（计算 hidden_size // num_attention_heads），且添加了回归测试覆盖。
  3. CI 超时风险：AMD mirror 的超时设置较长（110 分钟），可能掩盖真正的性能退步，但讨论中解释为适应 CI 环境。 - 影响：用户影响：修复了 DeepSeek-VL2 在 ROCm 上的 NIXL 传输错误，可能使该模型在 ROCm 上可用。同步屏障可能小幅降低 Mamba 模型在 ROCm 上的吞吐，但保证了正确性。系统影响：CI 配置变更使更多 nixl 测试在 AMD 硬件上运行，提高覆盖。团队影响：需要关注 ROCm 性能回归，若出现显著下降可考虑更细粒度的同步策略。
- 风险标记：新增设备同步屏障可能导致性能退化，HEAD_DIM 契约变更可能影响其他模型，CI 超时设置冗长可能掩盖回归

关联脉络

- PR #45357 Fix async scheduler race condition: NickLucche 询问该 PR 是否已修复 ROCm 上的 Mamba 同步竞争，但 AndreasKaratzas 确认未修复，因此本 PR 的同步屏障仍为必要。