PR #32791 完整报告

sgl-project/sglang

【NPU】fix decode MTP + eagle shape error

合并时间：2026-07-30 21:34

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/32791

执行摘要

- 一句话：修复 NPU 上 speculative decoding 的 shape 不匹配问题
- 推荐动作：该 PR 是典型的硬件后端 bugfix，逻辑清晰专注。建议相关 NPU 维护者关注，同时可考虑后续为 actual_seq_lengths_q 添加默认初始化以避免潜在的未定义行为。

功能与动机

在 Eagle 模式下，首先调用 init_forward_metadata，随后 _prepare_eager_forward_batch 执行 padding。该执行顺序导致 q 被 padding，而 forward_metadata 中存储的 KV 相关信息仍为 padding 前的值，造成 shape 不匹配。另外，draft_extend_for_decode 输出的 topk_indices 可在 draft step 复用，但 draft 的 q 可能经历了 padding，导致 q_nope 与 topk_indices 在 npu_sparse_flash_attention 内部 shape 不匹配。

实现拆解

仅修改 python/sglang/srt/hardware_backend/npu/attention/ascend_backend.py 一个文件，包含三处逻辑添加：

1. 在 init_forward_metadata 方法中添加 actual_seq_lengths_q 的赋值（第 500-523 行）：根据 forward_mode 设置该字段的值。对于 target_verify 或 draft_extend_v2 模式，生成等差数列；对于 decode 或 idle 模式，生成从 1 开始递增的序列。这样 DSA indexer 读取到的是与 KV 一致的、未 padding 的 batch size。
2. 新增 _pad_topk_indices 方法（第 746-770 行）：若 topk_indices 的行数少于目标 num_tokens，则在末尾填充 -1 至相同行数，并断言行数不超过目标值。
3. 在 forward_sparse 中调用 _pad_topk_indices（第 1121-1122 行）：在非 DSA 路径下，调用 _pad_topk_indices 确保 topk_indices 的行数与 q_nope 一致，然后才调用 npu_sparse_flash_attention。

关键文件：

- python/sglang/srt/hardware_backend/npu/attention/ascend_backend.py（模块 NPU 后端；类别 source；类型 core-logic；符号 _pad_topk_indices, init_forward_metadata, forward_sparse）：唯一修改的文件，修复了 speculative decoding 下 q padding 导致的 shape 不匹配问题。新增了 actual_seq_lengths_q 字段的赋值、_pad_topk_indices 方法及其在 forward_sparse 中的调用。

关键符号：init_forward_metadata, _pad_topk_indices, forward_sparse

关键源码片段

python/sglang/srt/hardware_backend/npu/attention/ascend_backend.py

唯一修改的文件，修复了 speculative decoding 下 q padding 导致的 shape 不匹配问题。新增了 actual_seq_lengths_q 字段的赋值、_pad_topk_indices 方法及其在 forward_sparse 中的调用。

```
# python/sglang/srt/hardware_backend/npu/attention/ascend_backend.py

# 在 init_forward_metadata 中，根据 forward_mode 设置 actual_seq_lengths_q 字段
# 该字段反映 padding 前的真实 batch size，解决 DSA indexer 读取不一致的问题
if (
    forward_batch.forward_mode.is_target_verify()
    or forward_batch.forward_mode.is_draft_extend_v2()
):
    self.forward_metadata.actual_seq_lengths_q = torch.arange(
        self.speculative_num_draft_tokens,
        self.speculative_num_draft_tokens
        + forward_batch.seq_lens.shape[0] * self.speculative_num_draft_tokens,
        self.speculative_num_draft_tokens,
        dtype=torch.int32,
        device=self.device,
    )
elif forward_batch.forward_mode.is_decode_or_idle():
    self.forward_metadata.actual_seq_lengths_q = torch.tensor(
        [1 + i for i in range(forward_batch.seq_lens.shape[0])],
        dtype=torch.int32,
        device=self.device,
    )

# 新增方法：将 topk_indices 填充到指定行数，确保与 q 的 shape 匹配
def _pad_topk_indices(
    self, topk_indices: torch.Tensor, num_tokens: int
) -> torch.Tensor:
    current_tokens = topk_indices.shape[0]
    if current_tokens == num_tokens:
        return topk_indices
    assert current_tokens <= num_tokens, (
        f"topk_indices rows ({current_tokens}) > num_tokens ({num_tokens}); "
        "this indicates a mismatch between indexer output and q layout."
    )
    pad_size = num_tokens - current_tokens
    padding = torch.full(
        (pad_size, topk_indices.shape[1]),
        -1, # 填充 -1 作为无效索引
        dtype=topk_indices.dtype,
        device=topk_indices.device,
```

```
    )
    return torch.cat([topk_indices, padding], dim=0)

# 在 forward_sparse 中，非 DSA 路径下调用 _pad_topk_indices
if topk_indices is not None:
    topk_indices = self._pad_topk_indices(topk_indices, q_nope.shape[0])
topk_indices = _expand_dsa_sparse_indices(topk_indices)
attn_out, _, _ = torch_npu.npu_sparse_flash_attention(
    query=q_nope,
    ...
)
```

评论区精华

该 PR 的 Review 评论数为 0，由 sglang-npu-bot 自动批准，无人工讨论记录。PR body 中详细描述了背景和问题（注：body 中提到的 'npu_sparse_flash_attention' 与代码中的 'npu_sparse_flash_attention' 拼写略异，但出处一致）。

- 暂无高价值评论线程

风险与影响

- 风险：风险较低，因为改动集中在 NPU 特定的 ascend_backend.py，且仅在 speculative decoding 路径起作用。主要风险：
  - 新增的 actual_seq_lengths_q 字段在非 speculative 路径下不会被赋值，虽目前代码中未使用，但未来若被读取可能未初始化。
  - 填充 -1 到 topk_indices 可能影响 npu_sparse_flash_attention 的行为，需确认后端实现正确忽略 -1 索引。
  - 影响：影响范围有限，仅作用于华为 NPU 硬件上使用 speculative decoding（EAGLE/MTP）的场景。修复了因 padding 导致的 shape 不匹配错误，提升了在该场景下推断的正确性和稳定性。对 CPU、CUDA 等其他后端无影响。
  - 风险标记：新增字段可能未在非 speculative 路径初始化

关联脉络

- PR #32881 [misc] Remove unused multi_layer_draft_forward_cg module: 同为 speculative decoding 相关的清洁和整理，涉及 NPU 后端可能受影响的代码路径。