PR #26085 完整报告

sgl-project/sglang

drop `FutureIndices` wrapper class

合并时间：2026-05-22 17:44

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/26085

执行摘要

- 一句话：移除 FutureIndices 包装类，直接使用 torch.Tensor
- 推荐动作：该 PR 是简单的清理重构，评审风险低，建议合并。可快速回顾代码变更，确认没有遗漏引用。

功能与动机

PR body 明确指出 FutureIndices 是一个单字段的数据类，没有方法或额外语义，直接传递 req_pool_indices tensor 更简洁，减少不必要的包装。

实现拆解

1. 移除 FutureIndices 类定义(overlap_utils.py)：删除 @dataclass class FutureIndices，并移除 dataclass 导入。
2. 更新 FutureMap 方法签名(overlap_utils.py)：set_input_ids_sentinel、publish、stash 方法中的 future_indices: FutureIndices 改为 future_indices: torch.Tensor，内部访问 .indices 的操作改为直接使用 tensor。
3. 更新 FutureMap 内部使用(overlap_utils.py)：_resolve_spec_extras 和 resolve_seq_lens_cpu 中 draft_input.future_indices.indices 改为 draft_input.future_indices。
4. 更新消费者代码：在 scheduler.py、decode_schedule_batch_mixin.py、eagle_info.py、utils.py 中，所有创建 FutureIndices 实例的地方改为直接传递 req_pool_indices tensor；EagleDraftInput.future_indices 字段类型从 Optional[FutureIndices] 改为 Optional[torch.Tensor]；filter_batch 和 merge_batch 方法中的相应操作也改为直接操作 tensor。
5. 清理导入：在 scheduler.py、eagle_info.py、utils.py 中移除了 from sglang.srt.managers.overlap_utils import FutureIndices 的导入。

关键文件：

- python/sglang/srt/managers/overlap_utils.py（模块 调度器；类别 source；类型 core-logic；符号 FutureIndices, publish）：核心变更文件：移除了 FutureIndices 类定义，更新了 FutureMap 的所有相关方法签名。
- python/sglang/srt/speculative/eagle_info.py（模块 推测解码；类别 source；类型 dependency-wiring）：更新了 EagleDraftInput 的 future_indices 字段类型，以及

filter_batch/merge_batch 中的操作。

- python/sglang/srt/managers/scheduler.py（模块 调度器；类别 source；类型 dependency-wiring）：移除 FutureIndices 导入，并直接使用 batch.req_pool_indices。
- python/sglang/srt/disaggregation/decode_schedule_batch_mixin.py（模块 调度器；类别 source；类型 dependency-wiring）：移除 local import 的 FutureIndices，直接传递 tensor。
- python/sglang/srt/managers/utils.py（模块 调度器；类别 source；类型 dependency-wiring）：更新 GenerationBatchResult 中的字段类型。

关键符号：FutureMap.set_input_ids_sentinel, FutureMap.publish, FutureMap.stash, FutureMap._resolve_spec_extras, FutureMap.resolve_seq_lens_cpu, EagleDraftInput.filter_batch, EagleDraftInput.merge_batch

关键源码片段

python/sglang/srt/managers/overlap_utils.py

核心变更文件：移除了 FutureIndices 类定义，更新了 FutureMap 的所有相关方法签名。

```
# 原 FutureIndices 包装类被移除，现在直接使用 torch.Tensor

class FutureMap:
    # ... 其他方法保持不变 ...

    def _resolve_spec_extras(self, batch: ScheduleBatch) -> None:
        draft_input: EagleDraftInput = batch.spec_info
        if draft_input is None:
            # FIXME(lsyin): only prefill; not compatible with mixed mode
            return
        # 之前是 indices = draft_input.future_indices.indices
        indices = draft_input.future_indices # 现在直接是 tensor
        # FIXME: indices = batch.req_pool_indices, pinned 2 iters via
        # record_batch_in_overlap; record_stream here is redundant.
        indices.record_stream(torch.get_device_module(self.device).current_stream())
        draft_input.topk_p = self.topk_p_buf[indices]
        draft_input.topk_index = self.topk_index_buf[indices]
        draft_input.bonus_tokens = self.output_tokens_buf[indices]
        if spec_need_hidden_states():
            draft_input.hidden_states = self.hidden_states_buf[indices]

    def set_input_ids_sentinel(
        self, batch: ScheduleBatch, future_indices: torch.Tensor # 类型从 FutureIndices 改为
        torch.Tensor
    ) -> None:
        # Sentinel for the decode portion so mixed batches can cat extend
        # (positive real tokens) + decode (negative sentinels) into one
        # input_ids; resolve_future translates negatives via output_tokens_buf.
        batch.input_ids = -future_indices # 之前是 -future_indices.indices
```

```
    def publish(self, future_indices: torch.Tensor, new_seq_lens: torch.Tensor) -> None:
        indices = future_indices # 之前是 future_indices.indices
        if indices.shape[0] == 0:
            return # DP idle
        self.new_seq_lens_buf[indices] = new_seq_lens.to(self.new_seq_lens_buf.dtype)
        # ...

    def stash(
        self,
        future_indices: torch.Tensor, # 类型从 FutureIndices 改为 torch.Tensor
        payload: Union[torch.Tensor, EagleDraftInput],
    ) -> None:
        indices = future_indices # 之前是 future_indices.indices
        if indices.shape[0] == 0:
            return # DP idle: payload is empty stub; lazy-init shape peek would IndexError.
        # ...
```

python/sglang/srt/speculative/eagle_info.py

更新了 EagleDraftInput 的 future_indices 字段类型，以及 filter_batch/merge_batch 中的操作。

```
@dataclass
class EagleDraftInput(SpecInput, EagleDraftInputV2Mixin):
    # ...
    # V2 overlap worker only: req_pool_indices used as buf slot keys.
    future_indices: Optional[torch.Tensor] = None # 原为 Optional[FutureIndices]
    # ...

    def filter_batch(self, new_indices: torch.Tensor, has_been_filtered: bool = True):
        if self.future_indices is not None:
            self.future_indices = self.future_indices[new_indices] # 原为 self.future_indices.
            indices[new_indices]
        # ...

    def merge_batch(self, spec_info: "EagleDraftInput"):
        if self.future_indices is not None:
            assert spec_info.future_indices is not None
            self.future_indices = torch.cat(
                [self.future_indices, spec_info.future_indices]
            ) # 原为 FutureIndices(indices=torch.cat([...]))
        # ...
```

评论区精华

仅有一条来自 gemini-code-assist[bot] 的自动化 review，指出该重构简化了数据结构，无进一步反馈。

- 暂无高价值评论线程

风险与影响

- 风险：改动较小且语义等价，风险低。主要风险在于：如果未来有外部代码依赖 FutureIndices 类，可能出现兼容性问题。但该 PR 移除了类定义和导入，外部代码将无法编译。鉴于该 PR 是合并到 main 分支，项目内部已全部适配。
- 影响：无功能影响，代码更简洁。降低了数据结构认知负荷，有利于后续维护。影响范围限于 overlap 调度和推测解码相关的 5 个源文件。
- 风险标记：低风险重构

关联脉络

- PR #25944 [core] step 1: route non-spec seq_lens via FutureMap with per-mode bootstrap fixes: 引入了 FutureMap 和 FutureIndices 的相关逻辑，本 PR 是后续清理。
- PR #26020 [core] step 2: drop seq_lens sentinel; SB maintains GPU as seq_lens_cpu mirror: 进一步的重构，与 FutureMap 和 overlap 调度相关。