

PR #29311 完整报告

sgl-project/sglang

[MLX] Fix FutureMap relay unit test to use RelayPayload

合并时间: 2026-06-26 07:53

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/29311>

执行摘要

- 一句话: 修复 MLX 单元测试使用过时的 `stash()` API
- 推荐动作: 该 PR 是典型的 API 重构后未同步测试的反例, 值得注意: 当核心函数签名变更时, 应系统地检查所有调用点 (包括测试)。建议精读以了解 `RelayPayload` 的用法。

功能与动机

PR 的 Motivation 中指出, `test_mlx_scheduler_init_overlap_keeps_future_map_relay` 在 `main` 分支上因 `FutureMap.stash()` 已重构为接受 `RelayPayload` 数据类, 但测试仍传入原始 `torch.Tensor`, 导致 `AttributeError: 'Tensor' object has no attribute 'draft_probs'`。该问题由 @changminbark 报告, 本次作为单独 PR 修复。

实现拆解

1. 在文件 `test/registered/unit/hardware_backend/mlx/test_attention_patching.py` 中, 新增导入 `from sglang.srt.managers.overlap_utils import RelayPayload`。
2. 将第 420 行 `scheduler.future_map.stash(indices, torch.tensor([7], dtype=torch.int64))` 改为 `scheduler.future_map.stash(indices, RelayPayload(bonus_tokens=torch.tensor([7], dtype=torch.int64)))`, 确保传入的第二个参数是 `RelayPayload` 实例, 以匹配 `stash()` 当前的预期签名。
3. 断言语句未改动, 仍验证 `output_tokens_buf[1].item()` 等于 7。
4. 该变更为纯测试修复, 不涉及生产逻辑。

关键文件:

- `test/registered/unit/hardware_backend/mlx/test_attention_patching.py` (模块 MLX 后端; 类别 test; 类型 test-coverage): 唯一的变更文件, 修复了单元测试中对 `FutureMap.stash()` 的调用, 使其适配新的 `RelayPayload` 数据类 API。

关键符号: 未识别

关键源码片段

`test/registered/unit/hardware_backend/mlx/test_attention_patching.py`

唯一的变更文件, 修复了单元测试中对 `FutureMap.stash()` 的调用, 使其适配新的 `RelayPayload` 数据类 API。

```
# test/registered/unit/hardware_backend/mlx/test_attention_patching.py
# 修复: FutureMap.stash() 现在需要 RelayPayload 类型, 而非原始 Tensor

from sglang.srt.managers.overlap_utils import RelayPayload # 新增导入

# ... (省略已存在的 setup 代码)

self.assertIsNotNone(scheduler.future_map)
indices = torch.tensor([1], dtype=torch.int64)
scheduler.future_map.stash(
    indices,
    RelayPayload(bonus_tokens=torch.tensor([7], dtype=torch.int64)) # 修正: 包装为
    RelayPayload
)
self.assertEqual(int(scheduler.future_map.output_tokens_buf[1].item()), 7)
```

评论区精华

无 reviewer 评论或其他讨论。PR 已获 @yeahdongcn 批准。

- 暂无高价值评论线程

风险与影响

- 风险: 风险极低: 变更仅涉及单行测试代码和导入语句, 不影响生产逻辑。唯一风险是如果 RelayPayload 构造函数签名在未来变化, 该测试可能再次失效, 但属于正常维护范畴。
- 影响: 影响范围仅限于 MLX 硬件后端的一个单元测试。修复后该测试可正常通过, 避免 CI 失败阻塞其他变更。对其他用户或系统无影响。
- 风险标记: 测试修复

关联脉络

- PR #29286 [sgl-kernel/cpu]: exclude amx gemm source from arm build: 同为硬件后端 (CPU/MLX) 的测试修复, 但无直接代码依赖。