PR #29316 完整报告

sgl-project/sglang

[PD] Early-send cached-prefix KV overlapping uncached prefill forward

合并时间：2026-06-26 04:31

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/29316

执行摘要

- 一句话：PD 预填提前发送缓存 KV，降低首 Token 延迟
- 推荐动作：建议细致阅读 maybe_send_cached_prefix_chunk 的实现逻辑和 run_batch 中的调用时机。设计上采用非最终 KV 块提前发送以重叠传输与计算，是一种典型的延迟隐藏优化，值得学习。对于使用 PD 分离部署的团队，建议在长前缀场景下验证 TTFT 改善。

功能与动机

在长共享前缀场景中，传统方式在完整预填前向完成后再发送 KV，传输与计算串行导致 TTFT 偏高。通过将已缓存的设备驻留前缀 KV 提前发送，使传输与后续未缓存后缀计算重叠，可显著降低首 Token 延迟。PR body 指出：'overlapping the transfer with compute to cut TTFT for long shared prefixes'。

实现拆解

1. 新增环境变量（python/sglang/srt/environ.py）：增加 SGLANG_DISAGG_PREFILL_EARLY_SEND_CACHED_PREFIX = EnvBool(True)，默认开启，提供关闭开关。
2. 移动 enable_staging 初始化（python/sglang/srt/managers/scheduler.py）：在 init_disaggregation 中早期赋值 self.enable_staging，确保流水线并行预填事件循环在 run_batch 调用前即可获取该值，修复第二提交中的 AttributeError。
3. 在 run_batch 前插入提前发送逻辑（python/sglang/srt/managers/scheduler.py）：在 run_batch 方法中，判断 disaggregation_mode == PREFILL 后，遍历当前批次所有请求并调用 maybe_send_cached_prefix_chunk。
4. 实现核心方法 maybe_send_cached_prefix_chunk（python/sglang/srt/disaggregation/prefill.py）：检查条件（环境变量启用、非 staging 模式、请求已完成 bootstrap），计算设备驻留缓存前缀的结束位置 cached_end，若 cached_end > req.start_send_idx 则调用 self.send_kv_chunk(req, last_chunk=False, end_idx=cached_end) 发送非最终 KV 块。
5. 移除 mooncake 限制（python/sglang/srt/disaggregation/prefill.py 之 process_prefill_chunk）：去掉 envs.SGLANG_DISAGG_STAGING_BUFFER.get() 的重复设置和 mooncake 相关 gate，因 KV 池在启动时已注册一次。

6. 移除事件循环中的 staging 赋值（python/sglang/srt/disaggregation/prefill.py）：*event_loop_normal_disagg_prefill* 和 *event_loop_overlap_disagg_prefill* 中删除 *self.enable_staging = ...*，避免冗余。

关键文件：

- python/sglang/srt/disaggregation/prefill.py（模块 分离调度；类别 source；类型 *core-logic*；符号 *maybe_send_cached_prefix_chunk*）：实现核心方法 *maybe_send_cached_prefix_chunk*，控制提前发送逻辑；同时移除事件循环中的 staging 赋值和修复流水线并行兼容性。
- python/sglang/srt/managers/scheduler.py（模块 调度器；类别 source；类型 *core-logic*）：在 run_batch 中插入提前发送调用点，并在 init_disaggregation 中移动 *enable_staging* 初始化以修复流水线并行问题。
- python/sglang/srt/*environ*.py（模块 环境配置；类别 source；类型 *core-logic*）：新增环境变量 SGLANG_DISAGG_PREFILL_EARLY_SEND_CACHED_PREFIX，默认开启，提供关闭开关。

关键符号：*maybe_send_cached_prefix_chunk*

关键源码片段

python/sglang/srt/disaggregation/prefill.py

实现核心方法 *maybe_send_cached_prefix_chunk*，控制提前发送逻辑；同时移除事件循环中的 staging 赋值和修复流水线并行兼容性。

```
def maybe_send_cached_prefix_chunk(self: Scheduler, req: Req) -> None:
    # 仅当环境变量启用、非 staging 模式且请求已完成 bootstrap 时才执行
    if (
        not envs.SGLANG_DISAGG_PREFILL_EARLY_SEND_CACHED_PREFIX.get()
        or self.enable_staging
        or req.pending_bootstrap
    ):
        return

    # 计算设备驻留缓存前缀的结束位置（页对齐）
    cached_end = len(req.prefix_indices) - req.host_hit_length
    # 如果没有可提前发送的 KV 块（cached_end <= start_send_idx），则跳过
    if cached_end <= req.start_send_idx:
        return
    # 断言页对齐，确保发送不越界
    assert cached_end % self.token_to_kv_pool_allocator.page_size == 0
    # 发送非最终 KV 块（last_chunk=False），后续还有未缓存后缀的 KV 需要发送
    self.send_kv_chunk(req, last_chunk=False, end_idx=cached_end)
```

评论区精华

未发现 Review 评论。PR 由作者自行合并，提交历史显示第二提交修复了流水线并行下的 AttributeError 和 mooncake 兼容问题。评论仅包含 CI 触发命令和自动化结果，无人工讨论。

- 暂无高价值评论线程

风险与影响

- 风险：
  - 回归风险：新增 maybe_send_cached_prefix_chunk 在 run_batch 前无条件触发，若条件判断遗漏（如 staging 模式或 bootstrap 未完成），可能导致异常。但实现有守卫条件（not envs.SGLANG_DISAGG_PREFILL_EARLY_SEND_CACHED_PREFIX.get() or self.enable_staging or req.pending_bootstrap），风险较低。
  - 性能风险：提前发送增加了一次 KV 传输，若共享前缀不长，可能反而增加开销。但默认仅在 cached_end > start_send_idx 时发送，且可通过环境变量关闭。
  - 兼容性：移除了 mooncake 传输后端的 gate，可能影响特定后端的稳定性。但 PR 说明 KV 池启动时已注册，移除后不影响。
  - 流水线并行：第二提交修复了 event_loop_pp_disagg_prefill 中的 AttributeError，但若测试覆盖不足可能遗留问题。未见直接相关测试文件变更。
- 影响：
  - 用户可见：对使用 PD 分离部署且存在长共享前缀的场景，TTFT 预计降低。默认启用，用户无需额外配置；可通过 SGLANG_DISAGG_PREFILL_EARLY_SEND_CACHED_PREFIX=false 关闭。
  - 系统内部：改动集中在预填调度路径，不涉及 decode 端。maybe_send_cached_prefix_chunk 仅在预填节点调用，影响范围有限。
  - 团队协作：变更设计清晰，注释充分，易于理解。缺少测试文件可能增加后续维护风险。
  - 风险标记：核心路径变更，缺少测试覆盖

关联脉络

- 暂无明显关联 PR