PR #22614 完整报告

sgl-project/sglang

Fix: Add token heuristic increment in total_tokens load balancing

合并时间：2026-04-21 16:29

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/22614

执行摘要

- 一句话：修复 total_tokens 负载均衡中因缺少启发式 token 增量导致请求堆积的问题。
- 推荐动作：该 PR 值得精读，尤其是 DPBudget.dispatch 方法的修改，展示了如何在负载均衡中平衡启发式增量和快照校正的设计。关注 gemini-code-assist[bot] 提出的多模态 token 低估问题，这可能影响未来扩展。

功能与动机

PR body 指出，在生产日志中观察到使用 total_tokens 负载均衡方法时，预分配请求（prealloc requests）会堆积在单个 DP worker 上（如 DP0 有 21 个预分配请求），而其他 DP worker 几乎空闲。根本原因是 DPBudget.dispatch() 在调度后只增加 total_requests 的启发式增量，未增加 total_tokens，导致在两次调度快照之间，total_tokens 值过时，所有新请求持续路由到同一 worker，造成 KV 内存不足和队列阻塞。

实现拆解

1. 修改 DPBudget.dispatch 方法签名和逻辑：
   - 文件：python/sglang/srt/managers/data_parallel_controller.py
   - 关键符号：dispatch 方法
   - 变更：为 dispatch 方法添加 estimated_tokens: int = 0 参数，并在选择目标 rank 后，增加 self.total_tokens[target_rank] += estimated_tokens 行。
   - 原因：使 total_tokens 负载均衡方法在调度时能基于请求的输入 token 数进行启发式增量，避免请求堆积。
   - 影响：total_tokens 值在调度间隙得到临时更新，但后续 update_budget() 会用真实快照覆盖，确保自校正。
2. 更新 total_tokens_scheduler 调用：
   - 文件：python/sglang/srt/managers/data_parallel_controller.py
   - 关键符号：total_tokens_scheduler 方法
   - 变更：在调用 dp_budget.dispatch() 前计算 estimated_tokens = len(req.input_ids)，并将其作为参数传入。
   - 原因：为 TOTAL_TOKENS 方法提供 token 估计值，匹配 get_load() 报告的队列请求度量。

- 影响：仅影响 TOTAL_TOKENS 负载均衡路径；TOTAL_REQUESTS 和 ROUND_ROBIN 方法不受影响，保持向后兼容。

3. 测试与配置配套：
   - 本次变更未包含直接对应的测试文件修改，但 PR 已通过 CI 标签（run-ci）运行测试。

关键文件：

- python/sglang/srt/managers/data_parallel_controller.py（模块 调度器；类别 source；类型 core-logic；符号 dispatch, total_tokens_scheduler）：这是唯一变更的文件，包含负载均衡的核心逻辑修改，直接影响 DP worker 的请求调度。

关键符号：dispatch, total_tokens_scheduler

评论区精华

review 中，gemini-code-assist[bot] 指出一个潜在问题：使用 len(req.input_ids) 作为 estimated_tokens 对于多模态请求可能显著低估实际负载，因为图像等会扩展为更多 token（例如 576 个以上）。建议考虑从 TokenizerManager 传递更准确的估计值，或至少记录此限制。此评论未得到直接回复或解决，但 PR 仍被合并，表明团队可能认为当前启发式增量已足够，或计划后续处理。其他 reviewer（Ratish1 和 hnyls2002）简单批准。

- 多模态请求的 token 估计准确性 (correctness): 未直接解决，PR 仍被合并，可能视为可接受的启发式或待后续优化。

风险与影响

- 风险：
  - 回归风险：低。变更仅影响 TOTAL_TOKENS 负载均衡路径，且 estimated_tokens 默认值为 0，确保其他方法不受影响。update_budget() 使用赋值覆盖，防止启发式增量永久漂移。
  - 性能风险：中。对于多模态请求，len(req.input_ids) 可能低估 token 数，导致负载均衡不准确，可能仍引发轻微堆积，但比修复前大幅改善。
  - 兼容性风险：低。API 向后兼容，因添加了默认参数。
  - 安全风险：无显著安全影响。
- 影响：
  - 用户影响：使用解耦部署和 total_tokens 负载均衡的用户将观察到请求分布更均匀，减少因单 worker 过载导致的 KV 内存耗尽和延迟。
  - 系统影响：提升 DP worker 间的资源利用率，避免预分配队列阻塞，增强系统稳定性和吞吐量。
  - 团队影响：修复了生产环境中观察到的关键负载均衡缺陷，需关注多模态请求的潜在低估问题。
  - 风险标记：核心路径变更，多模态负载估计不足

关联脉络

- PR #22493 Add MambaPool kvcache offloading during retraction: 同样涉及调度和 KV 缓存管理，关注负载均衡和内存优化。
- PR #22911 [perf] support return_routed_experts with overlap scheduling: 涉及调度性能优化，与本 PR 的负载均衡改进相关。
- PR #23173 fix: pass v_head_dim to MHA KV pools and validate MiMo HiCache geometry: 涉及 KV 缓存和 HiCache，与本 PR 中因 KV 内存耗尽导致的预分配队列阻塞问题相关。