PR #22726 完整报告

sgl-project/sglang

feat(metrics): expose raw KV cache pool token counts as prometheus gauges

合并时间：2026-04-14 09:30

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/22726

执行摘要

本 PR 新增了三个 Prometheus Gauge 指标（sglang:kv_available_tokens、sglang:kv_evictable_tokens、sglang:kv_used_tokens），暴露 KV 缓存池的原始令牌计数，解决了现有 token_usage 指标无法反映可回收 radix 缓存节点导致内存监控失真的问题。变更集中在可观测性模块，对核心推理路径无影响，为运维人员提供了更灵活的内存使用分析能力，尤其适用于代理式工作负载场景。

功能与动机

现有 sglang:token_usage 指标仅报告非可回收令牌（活跃请求 + 固定会话），而可回收的 radix 缓存节点被排除在外。这导致缓存池看起来比实际更空，在代理式工作负载中可能产生误导——例如，token_usage 显示约 2% 使用率时，物理 GPU 内存实际消耗了 72%。PR 作者指出，暴露原始计数能让运维人员在 PromQL/Grafana 中推导所需比率，如物理使用率或可回收比例，从而准确诊断内存问题。

实现拆解

实现涉及两个文件，改动简洁：

1. metrics_collector.py：
   - 在 SchedulerStats 数据类中添加三个整数字段：python kv_available_tokens: int = 0 kv_evictable_tokens: int = 0 kv_used_tokens: int = 0
   - 在 SchedulerMetricsCollector 类中创建对应的 Prometheus Gauge 对象，并设置名称、文档和标签。
   - 在 log_stats() 方法中调用 _log_gauge() 更新指标值。
2. scheduler_runtime_checker_mixin.py：
   - 在 update_scheduler_stats() 方法中，将内部属性值赋给 stats 对象：python stats.kv_available_tokens = self.full_available_size stats.kv_evictable_tokens = self.full_evictable_size stats.kv_used_tokens = self.full_num_used

评论区精华

由于 review 评论为空，未发现具体的技术讨论或争议。PR 由作者自行合并，表明变更可能较为直接或已通过其他渠道达成共识。

风险与影响

风险：

- 指标数据准确性依赖 full_available_size 等内部属性的正确计算，若这些属性有误将导致指标失真。
- 新增指标可能引入微小性能开销，但通常可忽略。
- 无兼容性风险，因为纯新增字段和指标。

影响：

- 运维团队获得更细粒度的 KV 缓存监控，可灵活计算衍生指标，提升内存问题诊断能力。
- 系统增加三个 Prometheus 时间序列，存储开销轻微。
- 代码变更集中在可观测性模块，易于维护。

关联脉络

从近期历史 PR 看，本 PR 与以下变更相关：

- PR #22331（澄清 HiSparse 解码令牌使用日志）：同属可观测性改进，但聚焦日志而非指标。
- PR #22506（网关毫秒级日志支持）：扩展了可观测性能力，但针对日志精度。
- PR #21971（跳过嵌入模式 KV 缓存）：涉及 KV 缓存性能优化，而本 PR 提供更细监控。

整体上，本 PR 是 sglang 项目持续增强系统可观测性的一部分，特别是在 KV 缓存管理领域，为复杂工作负载下的内存分析提供了必要工具。