PR #20697 完整报告

sgl-project/sglang

Fix VRAM leak in overlap scheduling with structured output (#20640)

合并时间：2026-03-23 08:07

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/20697

执行摘要

该 PR 修复了在启用重叠调度（extra_buffer）和结构化 JSON 输出时，因闭包持有 GPU 张量引用而导致的 VRAM 泄漏。通过及时释放关键张量，解决了内存持续增长直至 OOM 的问题，提升了系统稳定性，建议关注其内存管理策略。

功能与动机

根据 Issue #20640，在生产环境中使用 Qwen3.5 模型时，启用 extra_buffer 并请求结构化输出会导致 VRAM 缓慢但持续增长，几小时后引发 OOM 错误。此问题仅在两个条件同时满足时触发：重叠调度激活并创建闭包，以及基于语法的结构化输出分配 vocab_mask。修复旨在消除这一内存泄漏，确保资源高效利用。

实现拆解

- scheduler.py：在 launch_batch_sample_if_needed 方法中，添加代码将 batch_result.delay_sample_func 设为 None，并清理 logits_output.next_token_logits，以断开闭包对张量的引用。python batch_result.delay_sample_func = None if batch_result.logits_output is not None: batch_result.logits_output.next_token_logits = None
- model_runner.py：在 _preprocess_logits 方法中，设置 sampling_info.vocab_mask = None，在应用掩码后立即释放 GPU 张量。python sampling_info.vocab_mask = None 这些改动共同确保张量在不再需要时被及时回收，防止内存累积。

评论区精华

审核讨论中，hnyls2002 提出疑问：" 为什么 VRAM 会持续增长而不是仅延迟一轮释放？" Cishoon 解释：" 在 tp_worker.py 中，重叠调度创建的 delay_sample_func 闭包捕获了 forward_batch 和 logits_output，导致张量保持引用。" 双方最终确认修复方案合理，无进一步争议。

风险与影响

- 风险：变更仅涉及引用置空，核心逻辑未变，但若下游代码错误依赖已释放张量（如 vocab_mask），可能引发异常；缺少单元测试覆盖具体泄漏场景。
- 影响：直接解决了生产环境 OOM 问题，提高了系统可靠性；影响范围限于重叠调度和结构化输出场景，对普通用户无影响，但优化了整体内存管理。

关联脉络

此 PR 独立修复了特定内存泄漏，未在近期历史 PR 中发现直接关联（如 PR 20625 涉及中间件 bug，PR 20862 为扩散模型功能）。它反映了在异步调度中管理 GPU 张量生命周期的重要性，可能为类似内存优化提供借鉴。