

PR #32353 完整报告

sgl-project/sglang

[Spec] Consolidate the grammar sync decision into ScheduleBatch.grammar_needs_sync

合并时间: 2026-07-25 11:42

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/32353>

执行摘要

- 一句话: 将 grammar 同步判断集中到 ScheduleBatch.grammar_needs_sync
- 推荐动作: 这是一个小型但值得关注的重构, 展示了如何逐步将分散的条件逻辑抽象为命名的语义方法, 提高可读性和一致性。对于需要理解语法约束与推测解码交互的开发者, 建议精读 supports_grammar_overlap 和 grammar_needs_sync 的文档注释。

功能与动机

PR body 指出本次变更为纯注释和重构, 用于明确 supports_grammar_overlap 的实际判定条件——它要求一个 GPU 草稿阶段, 其目标验证前向可以隐藏 grammar CPU 工作——并记录主机端推测算法 (如 NGRAM) 有意停留在同步 grammar 路径上, 而非待迁移状态。

实现拆解

1. 新增 `grammar_needs_sync` 方法: 在 `schedule_batch.py` 的 `ScheduleBatch` 类中添加 `grammar_needs_sync()` 方法, 封装 `has_grammar` and `not spec_algorithm.supports_grammar_overlap()` 判断。
2. 更新调度器条件: 在 `scheduler.py` 的 `is_disable_overlap_for_batch` 中将原本内联的 `not supports_grammar_overlap` and `has_grammar` 替换为调用 `batch.grammar_needs_sync()`, 并更新注释说明同步路径对 host-draft 算法是永久设计。
3. 对齐 ngram 工作节点: 在 `ngram_worker.py` 的 `_prepare_draft_tokens` 和 `_update_ngram_corpus` 中, 将 `not batch.has_grammar` 替换为 `not batch.grammar_needs_sync()`, 保持行为一致。
4. 完善算法文档: 在 `spec_info.py` 的 `supports_grammar_overlap` 方法中添加注释, 阐明 NGRAM 因依赖主机端语料库查询而无法参与 overlap。

关键文件:

- `python/sglang/srt/managers/schedule_batch.py` (模块 批次管理; 类别 source; 类型 core-logic; 符号 `grammar_needs_sync`): 新增 `grammar_needs_sync` 方法, 集中了 grammar 同步判断逻辑, 是本次重构的核心。
- `python/sglang/srt/managers/scheduler.py` (模块 调度器; 类别 source; 类型 core-logic): 在 `is_disable_overlap_for_batch` 中使用新方法替代内联判断, 并更新注释。
- `python/sglang/srt/speculative/ngram_worker.py` (模块 ngram 推测; 类别 source; 类型 core-logic): 将两个条件引用从 `batch.has_grammar` 替换为

batch.grammar_needs_sync() 以统一判断。

- python/sglang/srt/speculative/spec_info.py (模块 推测算法定义; 类别 source; 类型 core-logic) : 补充注释说明 NGRAM 不支持 grammar overlap 的原因。

关键符号: grammar_needs_sync, is_disable_overlap_for_batch, _prepare_draft_tokens, _update_ngram_corpus

关键源码片段

python/sglang/srt/managers/schedule_batch.py

新增 grammar_needs_sync 方法, 集中了 grammar 同步判断逻辑, 是本次重构的核心。

```
# python/sglang/srt/managers/schedule_batch.py (class ScheduleBatch)

def is_empty(self):
    return len(self.reqs) == 0

def is_dllm(self):
    return self.dllm_config is not None

def grammar_needs_sync(self) -> bool:
    """Whether grammar forces this batch onto the synchronous path, i.e. the
    previous batch's result is resolved before this forward."""
    return self.has_grammar and not self.spec_algorithm.supports_grammar_overlap()

def prepare_encoder_info_extend(
    self, input_ids: List[array[int]], seq_lens: List[int]
):
    # ... (subsequent method body unchanged)
```

python/sglang/srt/managers/scheduler.py

在 is_disable_overlap_for_batch 中使用新方法替代内联判断, 并更新注释。

```
# python/sglang/srt/managers/scheduler.py (method is_disable_overlap_for_batch)

# Sync so the FSM advance lands before the next batch's bitmask. Permanent
# path for host-draft algorithms, not a pending migration.
need_grammar_sync = (
    batch
    and not batch.spec_algorithm.is_none()
    and batch.grammar_needs_sync() # ← consolidated check
    and batch.forward_mode.is_decode()
    and len(self.result_queue) > 0
)

# Algorithms that support grammar overlap advance the FSM inside verify()
# via the grammar barrier (overlapping the target forward), which resolves
# whatever result is still pending in the queue — including the
# extend->decode boundary — so no grammar-specific overlap disable is needed.
```

return disable_overlap_for_batch or need_grammar_sync

评论区精华

本 PR 无 review 讨论，由作者直接合并。仅包含作者发起的 CI 重跑命令。

- 暂无高价值评论线程

风险与影响

- 风险：由于仅涉及谓词提取和注释调整，未改变任何执行逻辑，且通过现有 CI 测试验证，回归风险极低。但需注意：如果未来新增推测算法时忘记在 `supports_grammar_overlap` 中更新，`grammar_needs_sync` 可能返回错误值。建议在算法注册时添加文档提醒。
- 影响：对用户无直接影响；对开发人员改进了代码可读性，集中了 grammar 同步策略的判定点，降低了未来误解和维护成本。
- 风险标记：无行为变更，集中决策逻辑

关联脉络

- 暂无明显关联 PR