PR #24724 完整报告

sgl-project/sglang

[Spec] Disambiguate `verified_id` into `bonus_token(s)` / `accept_tokens`

合并时间：2026-05-09 09:24

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/24724

执行摘要

- 一句话：拆分 verified_id 为 bonus_tokens 和 accept_tokens
- 推荐动作：值得精读的命名重构范例，展示了如何系统地实施命名规约。但对功能无影响，可直接合并。

功能与动机

原有的verified_id字段同时承担两种语义（per-req bonus token和flatten accepted tokens），造成代码阅读和推理困难。根据 #24094 命名规约，需要明确区分 bonus_token(s) 和 accept_tokens，消除歧义。

实现拆解

1. 在 EagleDraftInput 和 EagleVerifyOutput 等 dataclass 中将 verified_id 字段拆分为 bonus_tokens（per-req 标量）和 accept_tokens（扁平 accepted tokens）。
2. 重命名 Triton kernel fill_new_verified_id 为 fill_bonus_tokens，参数从 new_verified_id 变为 bonus_tokens_ptr，语义更清晰。
3. 更新所有 worker 和工具函数（eagle_worker.py、multi_layer_eagle_worker.py、dflash_worker.py、frozen_kv_mtp_worker.py、overlap_utils.py、spec_utils.py、logprob.py）中的字段访问和函数调用，使用新名称。
4. 更新命名规则文档 .claude/rules/speculative-naming.md，删除单数例外，强化规则 2 和规则 7。

关键文件：

- python/sglang/srt/speculative/eagle_info_v2.py（模块 推测解码；类别 source；类型 core-logic；符号 fill_new_verified_id, fill_bonus_tokens）：核心 Triton kernel 重命名，符号 fill_new_verified_id 改为 fill_bonus_tokens，参数也做了语义化调整。
- python/sglang/srt/speculative/eagle_info.py（模块 推测解码；类别 source；类型 core-logic）：EagleDraftInput 和 EagleVerifyOutput 的数据类字段变更，verified_id 拆分为 bonus_tokens 和 accept_tokens，verify 方法重构为返回 accept_tokens。
- python/sglang/srt/speculative/multi_layer_eagle_worker_v2.py（模块 推测解码；类别 source；类型 dependency-wiring）：导入和使用 fill_bonus_tokens 替换 fill_new_verified_id，draft_input.verified_id 改为 bonus_tokens

- python/sglang/srt/speculative/dflash_info.py（模块 推测解码；类别 source；类型 core-logic）：DFlashDraftInput 的 verified_id 改为 bonus_tokens，以及相关 filter_batch 和 merge_batch 方法更新
- python/sglang/srt/speculative/eagle_worker.py（模块 推测解码；类别 source；类型 core-logic）：使用 verify_output.accept_tokens 替换 verified_id，以及条件判断中字段名更新

关键符号：fill_bonus_tokens, EagleDraftInput.bonus_tokens, EagleDraftInput.accept_tokens, EagleVerifyOutput.accept_tokens, DFlashDraftInput.bonus_tokens, FrozenKVMTPDraftInput.bonus_tokens

关键源码片段

python/sglang/srt/speculative/eagle_info_v2.py

核心 Triton kernel 重命名，符号 fill_new_verified_id 改为 fill_bonus_tokens，参数也做了语义化调整。

```
@triton.jit
def fill_bonus_tokens(
    accept_tokens, # flat accepted tokens tensor, shape [sum_accepted]
    accept_lens, # per-req accept lengths ( 包括 bonus 本身 )
    bonus_tokens_ptr, # 输出 per-req bonus tokens, shape [bs]
    num_draft_tokens: tl.constexpr,
):
    # 不能在此 kernel 中融合对 accept_lens 的就地操作，因为此 kernel 读取 accept_lens
    pid = tl.program_id(axis=0)
    # accept_lens 包含 bonus token，最后一个已接受槽位位于 -1
    accept_len = tl.load(accept_lens + pid)

    # 计算 bonus token 在 accept_tokens 中的偏移
    bonus_token_idx = num_draft_tokens * pid + accept_len - 1
    bonus_token = tl.load(accept_tokens + bonus_token_idx)
    tl.store(bonus_tokens_ptr + pid, bonus_token)
```

python/sglang/srt/speculative/eagle_info.py

EagleDraftInput 和 EagleVerifyOutput 的数据类字段变更，verified_id 拆分为 bonus_tokens 和 accept_tokens，verify 方法重构为返回 accept_tokens。

```
@dataclass
class EagleDraftInput(SpecInput, EagleDraftInputV2Mixin):
    # ... 其他字段
    bonus_tokens: torch.Tensor = None
    # Flat accepted-token tensor for draft-extend, shape `[sum_accepted]`.
    # Set right after verify and consumed by `prepare_extend_after_decode` as
    # the extend batch's `input_ids`. Dead after that method returns.
    # TODO: drop this field and pass `accept_tokens` directly to
    # `prepare_extend_after_decode` as a method arg. Its lifetime is bounded
    # by verify -> prepare_extend, no need to live on the dataclass.
```

```
    accept_tokens: torch.Tensor = None
    num_accepted_drafts: torch.Tensor = None
    num_accepted_tokens: torch.Tensor = None
    # ...
```

评论区精华

无 review 讨论，作者直接合并。

- 暂无高价值评论线程

风险与影响

- 风险：重命名风险集中在未完全替换的引用，但 PR 覆盖了 all 18 个文件，包括 import、field access、kernel 调用等。CI 和静态类型检查应能捕获遗漏。由于是纯重命名，无运行时逻辑改变，风险较低。
- 影响：内部接口：修改了所有 speculative 模块的字段名和函数名，但保持行为完全不变。用户：无影响，对外 API 不变。团队：代码可读性提高，降低未来误解风险。
- 风险标记：变更范围广但属机械重命名，无额外测试覆盖

关联脉络

- PR #24094 Add speculative decoding naming convention rule: 本 PR 实施该规则定义的命名规范