PR #28856 完整报告

sgl-project/sglang

[Spec] Enable FR-Spec in EAGLE draft-extend CUDA graph by sizing logits buffer from the draft head

合并时间：2026-06-22 06:35

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/28856

执行摘要

- 一句话：修复 FR-Spec 下 draft-extend CUDA graph 闪退
- 推荐动作：该 PR 修改精炼（+10/-11），逻辑清晰，是典型的边界条件修复。值得关注的设计决策是：通过 getattr 从 worker 实例获取运行时状态，避免了修改 hf_config 或传递复杂参数，保持了接口简洁。建议阅读 eagle_draft_extend_cuda_graph_runner.py 中的回退逻辑和 eagle_worker_v2.py 中 backend 判断的简化。

功能与动机

FR-Spec 使用缩减词表加速 speculative decoding，但 draft-extend 的 CUDA graph 缓冲区大小固定为完整词表，导致 graph 捕获失败（crash），进而回退到 eager 模式，无法获得 graph 加速。PR body 明确指出："FR-Spec (--speculative-token-map) injects the reduced draft vocab via a late json_model_override_args, so the draft model's hf_config lacks hot_vocab_size and the draft-extend CUDA graph sized next_token_logits_buffer to the full vocab — mismatching the reduced head (graph capture crashed, so FlashInfer fell back to eager)."

实现拆解

1. 在 eagle_draft_extend_cuda_graph_runner.py 中增加 hot_token_id 回退逻辑：在 __init__ 中通过 getattr(self.eagle_worker, "hot_token_id", None) 获取实际词表，当 hf_config 中既无 draft_vocab_size 也无 hot_vocab_size 时，使用 len(hot_token_id) 作为 vocab_size，确保 logits 缓冲区大小与缩减后的输出头匹配。
2. 在 eagle_worker_v2.py 中移除 FlashInfer draft-extend graph 的 FR-Spec 限制：将 FlashInferAttnBackend 加入 graph_supported_backend 的 isinstance 检查元组中，并删除原有的 flashinfer_graph_supported 变量（包含 speculative_token_map is None 检查），使 supports_cuda_draft_extend_graph 直接使用 graph_supported_backend 判断，从而允许 FR-Spec 使用 FlashInfer draft-extend CUDA graph。

关键文件：

- python/sglang/srt/speculative/eagle_worker_v2.py（模块 推测解码；类别 source；类型 core-logic）：移除了 FlashInfer draft-extend graph 的 speculative_token_map is None 限制，将 FlashInferAttnBackend 加入支持的 backend 列表，是 FR-Spec 启用 graph 的

关键开关。

- python/sglang/srt/speculative/eagle_draft_extend_cuda_graph_runner.py（模块 推测解码；类别 source；类型 core-logic）：新增 hot_token_id 回退逻辑，确保 CUDA graph 的 logits 缓冲区大小与 FR-Spec 缩减后的输出头一致，修复 graph 捕获崩溃的根本原因。

关键符号：未识别

关键源码片段

python/sglang/srt/speculative/eagle_worker_v2.py

移除了 FlashInfer draft-extend graph 的 speculative_token_map is None 限制，将 FlashInferAttnBackend 加入支持的 backend 列表，是 FR-Spec 启用 graph 的关键开关。

```
def init_cuda_graphs(self):
    # ... 前面的代码省略 ...
    graph_supported_backend = isinstance(
        self.draft_extend_attn_backend,
        (
            TritonAttnBackend,
            TRTLLMMLABackend,
            TRTLLMHAAttnBackend,
            TokenspeedMLABackend,
            FlashInferAttnBackend, # 新增：FR-Spec 也可以使用 FlashInfer graph
        ),
    )
    # 之前这里有 flashinfer_graph_supported 变量，限制了 speculative_token_map 为 None
    # 现在直接统一判断，不再区分普通和 FR-Spec 场景
    supports_cuda_draft_extend_graph = (
        _is_cuda or _is_musa
    ) and graph_supported_backend
    # 后续捕获 graph 的逻辑不变
```

python/sglang/srt/speculative/eagle_draft_extend_cuda_graph_runner.py

新增 hot_token_id 回退逻辑，确保 CUDA graph 的 logits 缓冲区大小与 FR-Spec 缩减后的输出头一致，修复 graph 捕获崩溃的根本原因。

```
def __init__(self, eagle_worker):
    # ... 前面的代码省略 ...
    # 获取 worker 上的 hot_token_id（FR-Spec 场景下存在）
    hot_token_id = getattr(self.eagle_worker, "hot_token_id", None)

    if hasattr(
        self.model_runner.model_config.hf_config, "draft_vocab_size"
    ): # llama_eagle
        vocab_size = self.model_runner.model_config.hf_config.draft_vocab_size
    elif hasattr(
        self.model_runner.model_config.hf_config, "hot_vocab_size"
    ): # llama_eagle3
        vocab_size = self.model_runner.model_config.hf_config.hot_vocab_size
```

```
elif hot_token_id is not None:
    # FR-Spec: 缩减词表通过 json_model_override_args 注入，hf_config 中缺失此字段
    # 因此从 worker 的 hot_token_id 中获取实际大小
    vocab_size = len(hot_token_id)
else:
    vocab_size = self.model_runner.model_config.vocab_size

next_token_logits_buffer = torch.zeros(
    (
        self.max_bs * self.num_tokens_per_bs,
        vocab_size,
    ),
    dtype=torch.float,
)
```

评论区精华

无实质 review 讨论，仅有一个自动化 bot 的评论总结变更内容，未提出争议或疑虑。

- 暂无高价值评论线程

风险与影响

- 风险：变更影响范围小（2 个文件，共 10 行新增、11 行删除），但涉及 CUDA graph 捕获和 attention backend 选择。若 hot_token_id 意外为 None 或长度错误，可能导致缓冲区越界或 graph 捕获失败，但代码中已通过 getattr 安全获取并回退到 vocab_size。移除 speculative_token_map is None 限制后，所有使用 FlashInfer 的 FR-Spec 场景都会尝试 draft-extend graph，若 attention backend 不支持，graph 捕获可能失败并回退到 eager，不会影响正确性。
- 影响：对使用 FR-Spec（--speculative-token-map）且 attention backend 为 FlashInfer 的用户，draft-extend 将自动使用 CUDA graph 加速，显著提升 speculative decoding 性能。对其他场景无影响。无需用户配置变更。
- 风险标记：核心路径变更，缺少测试覆盖

关联脉络

- PR #28782 [Spec] Support FlashInfer CUDA graph for EAGLE draft-extend: 该 PR 为 EAGLE draft-extend 添加了 FlashInfer CUDA graph 支持，但未处理 FR-Spec 场景；本 PR 在此基础上补全了对缩减词表的支持。