PR #7136 完整报告

PaddlePaddle/FastDeploy

[Optimization]【Hackathon 10th Spring No.49】GPU ngram_match: BlockScan Phase 2 -optimized

合并时间：2026-04-07 16:36

原文链接：http://prhub.com.cn/PaddlePaddle/FastDeploy/pull/7136

执行摘要

本 PR 将 speculative decoding 中的 ngram_match GPU kernel 从串行 Phase 2 升级为并行 CUB BlockScan 实现，实现单次调用延迟从 270 µs 降至 19 µs，极端场景加速比达 1885 倍，并消除所有 GPU-CPU 同步点。这是一个高性能优化变更，显著提升推理效率，但需注意 batch size 限制和边界条件处理。

功能与动机

该变更是 Hackathon 10th Spring No.49（Issue #74773）的任务，旨在优化 ngram_match GPU kernel 以满足生产级基准（Issue #7200）。动机源于消除 CPU 路径的 D2H/H2D 拷贝开销，实现完全在设备端执行的并行加速。PR body 明确指出："Experimental variant of PR #6960 — adds CUB BlockScan parallel Phase 2, ... Addresses Hackathon 10th Spring No.49。"

实现拆解

模块	关键文件	改动描述
CUDA kernel	ngram_match.cu / ngram_match_mixed.cu	引入两阶段并行架构：Phase 1（<<<>>>）并行搜索，Phase 2（<<<1, 1024>>>）CUB BlockScan 阈值裁剪；提取公共逻辑到 ngram_match_common.cuh，包含 atomicMin64 CAS 和模板特化搜索。
Python 调用层	ngram.py / mtp.py	移除 .cpu() 和 .cuda() 回拷，直接传递 GPU tensor 给 kernel，消除额外拷贝。例如：share_inputs["input_ids_cpu"].cuda() 替代原有 CPU 路径。
测试	test_ngram_gpu_kernel.py	新增 12 个正确性测试用例和性能基准，覆盖极端规模（bsz=256, seq=131K）和多种命中模式。

关键代码逻辑示例（从 ngram_match_common.cuh）：

```
__device__ __forceinline__ void atomicMin64(int64_t *addr, int64_t val) {
    unsigned long long *addr_ull = reinterpret_cast<unsigned long long *>(addr);
    unsigned long long val_ull = static_cast<unsigned long long>(val);
    unsigned long long old = *addr_ull;
```

```
    while (val_ull < old) {
        unsigned long long assumed = old;
        old = atomicCAS(addr_ull, assumed, val_ull);
        if (old == assumed) break;
    }
}
```

评论区精华

- Copilot 关于性能开销：" 这里对 share_inputs[\"input_ids_cpu\"] 每次调用都执行 .cuda()，会把 CPU 上预分配的大张量整块拷到 GPU，产生显著 H2D 带宽和临时显存开销。" 作者回应："Acknowledged — upstream pattern, predates this PR。"
- fastdeploy-bot 关于设计限制：" 当 max_batch_size > NGRAM_GATHER_THREADS (1024) 时，Phase 2 kernel 无法处理所有 batch items。" 作者修复："Fixed in d37b581a9 — PD_CHECK(max_batch_size <= NGRAM_GATHER_THREADS) added。"
- 预算计算一致性：Copilot 指出 mixed 版本 budget 计算未预留后续项，作者修正为双 BlockScan 扫描，确保阈值语义对齐 CPU。

风险与影响

- 技术风险：Phase 2 的 batch size 限制为 1024，超出将导致未初始化输出；静态 scratch buffer 在多 GPU 环境可能引发设备不匹配；GPU-CPU 语义在 encoder-active 项存在细微差异。
- 影响范围：用户推理延迟大幅降低，系统吞吐提升；但变更涉及核心路径，需确保向后兼容；团队需维护更复杂的 CUDA 代码。

关联脉络

从近期历史 PR 看，speculative decoding 模块持续优化：

- PR #6960 是本 PR 的前身，展示了初步 GPU 优化。
- PR #7172 修复 MTP 在 TP 并行中的 bug，反映该模块的活跃开发。
- Issue #7200 定义了本 PR 的基准目标，推动性能标准提升。 整体趋势显示 FastDeploy 在 speculative decoding 上投入大量 GPU 优化，以应对大规模推理场景。