PR #22046 完整报告

sgl-project/sglang

Revert "[Kernel] Fuse temperature + softmax in sampling for decode speedup"

合并时间：2026-04-03 21:32

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/22046

执行摘要

本 PR 回滚了先前引入的融合 temperature+softmax Triton 内核，恢复了 PyTorch 原生操作。此变更可能影响解码性能，特别是对大词汇量模型，但简化了代码结构并可能解决原内核的问题，背后原因未在材料中明确说明。

功能与动机

原 PR #20501 旨在通过内核融合减少解码延迟，将 temperature scaling 和 softmax 合并为单个 Triton 内核以降低 kernel 启动开销和内存访问。本 PR 将其回滚，动机仅由 PR body 中的“Reverts sgl-project/sglang#20501”暗示，可能源于融合内核在实际运行中的正确性、性能或兼容性问题，但具体原因未在提供的材料中详述。

实现拆解

主要变更涉及以下文件：

- python/sglang/srt/layers/fused_sampling.py：完全移除，删除了融合内核的 Triton 实现。
- python/sglang/srt/layers/sampler.py：恢复标准采样路径，代码如下：`logits.div_(sampling_info.temperatures) logits[:] = torch.softmax(logits, dim=-1) probs = logits` 移除了原条件调用融合内核的逻辑。
- python/sglang/srt/model_executor/model_runner.py：移除 `_warmup_fused_sampling` 函数及其调用，简化预热流程。
- benchmark/kernels/bench_fused_temperature_softmax.py和 test/registered/sampling/test_fused_temperature_softmax.py：移除相关基准测试和单元测试。

评论区精华

review 中仅有 gemini-code-assist[bot] 的一条评论：

> “The comment `# In-place op to save memory` is misleading because `torch.softmax` is not an in-place operation and still requires a temporary allocation of the same size as the input. Additionally, `logits[:] = ...` performs a redundant data copy into the existing buffer. To improve efficiency, you can avoid the copy by assigning the result of `softmax` directly to `probs` and updating the reference in `logits_output.next_token_logits`.”

该评论指出了代码风格和性能优化点，但未涉及回滚原因，且状态为未解决。

风险与影响

- 性能风险：回滚后，解码步骤恢复为两个独立的 CUDA 内核（div_ 和 softmax），可能增加延迟，尤其对于 vocab size 大的模型（如 128K 词汇）。
- 正确性风险：sampler.py 的变更需确保与融合内核输出一致，避免采样偏差。
- 测试风险：移除单元测试减少了采样路径的覆盖，可能隐藏回归问题。
- 影响范围：用户可能体验到推理速度下降；系统代码更简洁，但失去性能优化；团队需评估后续优化策略。

关联脉络

本 PR 直接回滚 PR #20501，该 PR 是近期性能优化线的一部分。结合历史 PR 分析（如 #21511 为 AMD 启用 FP8 优化、#19246 为 NPU 优化 GLM4.7），可见团队持续进行硬件特定和通用性能调优。此次回滚可能表明融合内核在特定场景下未达预期，或引入了不稳定因素，建议关注后续是否有替代优化方案出现。