PR #19228 完整报告

sgl-project/sglang

[AMD] optimize Kimi K2.5 fused_moe_triton performance by tuning

合并时间：2026-02-27 03:50

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/19228

执行摘要

本 PR 通过调优 fused_moe_triton 内核和添加 int4_w4a16 量化支持，显著提升了 Kimi K2.5 模型在 AMD 硬件上的性能，prefill 和 decode 阶段速度分别提升约 3 倍和 2 倍，是面向特定硬件的关键优化，直接影响推理效率和用户体验。

功能与动机

动机源于 Kimi K2.5 模型在使用默认配置时 fused_moe_triton 性能较差，如 PR body 所述：'Kimi K2.5 fused_moe_triton use default config so the performance is poor.' 目标是改善该模型在 AMD 平台上的推理速度，通过调优内核参数和扩展量化支持来解决性能瓶颈。

实现拆解

实现分为三个层面：

1. 配置逻辑调整：在 common_utils.py 中，修改 get_model_config 函数，先处理 encoder-decoder 模型的 text_config，再计算 block_shape 和 architecture，并支持 int4 量化的 group_size 提取。
2. 基准测试扩展：在 tuning_fused_moe_triton.py 和 tuning_fused_moe_triton_sep.py 中，添加 use_int4_w4a16 参数，扩展 benchmark_config 函数以支持 int4 权重的初始化和尺度计算，例如：python elif use_int4_w4a16: w1 = torch.randint(0, 255, (num_experts, shard_intermediate_size, hidden_size // 2), dtype=torch.uint8)
3. 优化配置文件新增：添加两个 JSON 配置文件（如 E=384,N=128,device_name=,dtype=int4_w4a16.json），包含调优后的内核参数（如 BLOCK_SIZE_M、GROUP_SIZE_M），针对不同并发场景优化。

评论区精华

Review 过程中没有具体的技术讨论，但 issue 评论中 ZiguanWang 提到：'Notice currently get_device_name will get empty string inside a docker container, so the config device name is empty'，这揭示了配置命名的潜在问题，可能影响跨环境部署。hubertlu-tw 触发了 CI 测试，确保代码质量。

风险与影响

风险：配置变更可能意外影响其他模型（如 DbrxForCausalLM）的量化路径；新增 int4 支持需要更多测试覆盖以避免回归；调优基于 AMD 硬件，可能在其他平台（如 NVIDIA）上性能

不一致。具体风险点包括 common_utils.py 中 block_shape 计算的逻辑调整和 tuning 文件中的尺度初始化复杂性。

影响：用户在使用 Kimi K2.5 模型时将体验到显著的延迟降低和吞吐量提升（如基准测试显示吞吐量从 55.13875 token/s 增至 90.03875 token/s）。系统层面，优化了 fused_moe_triton 内核，为后续模型性能调优提供范例。团队需更新相关文档和集成测试，确保新配置的稳定性和兼容性。

关联脉络

从历史 PR 看，本 PR 与 #21348（修复 MxInt4 MoE 输出变量）相关，都涉及 MoE 和量化模块的改进，显示量化性能优化的持续演进。同时，与 #21103（暴露 get_scheduler_metadata 优化解码）类似，均为性能优化 PR，反映团队在核心路径调优上的集中投入。结合近期 PR 趋势，AMD 硬件优化和量化支持是仓库的重点方向之一。