PR #38083 完整报告

vllm-project/vllm

[Bugfix] Fix DeepGemm E8M0 accuracy degradation for Qwen3.5 FP8 on Blackwell

合并时间：2026-03-26 16:21

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/38083

执行摘要

该 PR 自动禁用 DeepGemm for Qwen3.5 模型在 Blackwell GPU 上，以修复 FP8 量化导致的精度下降约 12 个百分点。通过配置检查和 FP8 栈传播禁用标志，恢复模型性能，并扩展 CI 测试覆盖。关键变更涉及核心配置和量化层，但 review 中揭示的未解决 MoE 问题暗示后续优化空间。

功能与动机

根据 Issue #37804，DeepGemm 在 Blackwell GPU 上使用 E8M0 scale 格式，导致 Qwen3.5 FP8 模型精度显著下降（GSM8K 准确率从 0.8122 降至 0.6917）。PR body 说明，此格式在 FP8 块量化层中损失精度，累积影响约 80 个线性层。修复动机是自动规避此问题，提升模型可靠性。

实现拆解

实现分为三个模块：

1. 配置层：在 vllm/config/vllm.py 的 VllmConfig.__post_init__ 中添加检查，当模型类型为 qwen3_5_text 或 qwen3_5_moe_text 且 GPU 为 Blackwell 时，设置 quant_config.use_deep_gemm = False 并输出警告。
2. 量化层：修改 vllm/model_executor/layers/quantization/fp8.py 中的 Fp8Config、Fp8LinearMethod 和 W8A8BlockFp8LinearOp，传播 use_deep_gemm 标志以跳过 UE8M0 权重转换。
3. 测试层：更新 GSM8K 测试文件，添加 Qwen3.5 模型配置，将全局容忍度 TOL = 0.08 改为每配置 tolerance 字段，并调整准确率阈值（例如 Qwen3.5-35B-A3B-FP8 从 0.86 降至 0.79）。

评论区精华

Review 讨论亮点：

- 条件冗余：gemini-code-assist[bot] 指出 input_quant_fp8.py 中 self.use_ue8m0 可能冗余，建议简化。
- 修复不完整：claude[bot] 认为 Fp8MoEMethod 未处理 use_deep_gemm，导致 MoE 层仍用 DeepGemm，部分解释精度残留差距。vadiklyutiy 回复："I want to disable gemm only, don't attempt to disable deephemm's MoE"。

- 覆盖性问题：claude[bot] 指出 VLLM_USE_DEEP_GEMM=1 不能覆盖 auto-disable，与 PR 描述矛盾。
- 虚假警告：claude[bot] 提到 should_auto_disable_deep_gemm 在 DeepGemm 未激活时可能触发误导警告。

风险与影响

技术风险：

- MoE 层未完全禁用 DeepGemm，可能导致精度残留问题（实测 0.8029 vs 基线 0.8122）。
- CI 阈值调整（如降低准确率阈值）可能掩盖其他 bug 或引入误判。
- 条件冗余增加代码复杂性，潜在维护负担。
- 警告信息不准确，用户可能混淆 auto-disable 行为。

影响评估：

- 用户端：Qwen3.5 模型精度恢复，提升黑盒使用体验。
- 系统端：FP8 量化栈修改局限于特定模型和硬件，但需监控兼容性。
- 团队端：CI 扩展提高测试覆盖，但阈值调整需谨慎验证。

关联脉络

与历史 PR 关联显示团队对 Qwen 和 FP8 问题的持续关注：

- PR #37348 修复 Qwen3.5-FP8 在 TPU 上的权重加载错误。
- PR #38152 优化 Qwen3 性能，禁用双流执行。
- PR #38092 修复 Marlin FP8 内核，显示 FP8 量化活跃维护。 这些 PR 共同指向仓库在模型特定优化和量化技术上的演进趋势，尤其是针对新兴硬件（如 Blackwell）和流行模型（如 Qwen）的深度调优。