PR #41341 完整报告

vllm-project/vllm

[ROCm][CI] Add ROCm score absolute tolerance floor

合并时间：2026-05-01 09:59

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/41341

执行摘要

- 一句话：ROCm 测试绝对值容差优化
- 推荐动作：该 PR 值得快速合入，因为它是针对特定硬件平台测试稳定性的低风险调整。设计决策（绝对值与相对值组合）值得参考，但无需深入研读。

功能与动机

修复 ROCm 7.2/gfx950 上交叉编码器视觉测试中低概率文本对比文本分值的微小数值漂移问题（漂移约 0.005-0.006），同时保持对较大分值（如图像和多模态）的严格容差。

实现拆解

1. 新增绝对值容差配置：在 tests/entrypoints/pooling/scoring/test_cross_encoder_online_vision.py 中，添加 BACKEND_ABS_TOL 字典，为 default、ROCM_AITER_FA 和 FLEX_ATTENTION 后端指定绝对值容差（分别为 0.0、0.005、0.006）。
2. 新增辅助函数：实现 get_abs_tol(backend: str) -> float 函数，从字典中获取绝对值容差，若 backend 未定义则返回默认值。
3. 更新断言逻辑：修改 assert_score 函数，获取绝对值容差并在 pytest.approx 中添加参数 abs=abs_tol。同时更新调试打印和断言消息以包含绝对值容差值。

关键文件：

- tests/entrypoints/pooling/scoring/test_cross_encoder_online_vision.py（模块 测试；类别 test；类型 test-coverage；符号 get_abs_tol）：唯一的变更文件，添加了绝对值容差配置和相关函数，修改了断言逻辑。

关键符号：get_abs_tol

关键源码片段

tests/entrypoints/pooling/scoring/test_cross_encoder_online_vision.py

唯一的变更文件，添加了绝对值容差配置和相关函数，修改了断言逻辑。

```
# ... (before assert_score)

# ROCm 7.2/gfx950 shows small absolute drift on the low text-vs-text
# probability even though larger scores remain well inside the relative
```

```
# tolerance. Keep the relative tolerances tight and add only a small floor.
BACKEND_ABS_TOL: dict[str, float] = {
    "default": 0.0,
    "ROCM_AITER_FA": 0.005,
    "FLEX_ATTENTION": 0.006,
}

def get_abs_tol(backend: str) -> float:
    return BACKEND_ABS_TOL.get(backend, BACKEND_ABS_TOL["default"])

def assert_score(actual: float, expected: float, backend: str, label: str):
    tol = get_tol(backend)
    abs_tol = get_abs_tol(backend)
    diff = abs(actual - expected)
    rel_diff = diff / abs(expected) if expected != 0 else diff
    print(
        f"[{backend}] {label}: actual={actual:.6f} expected={expected:.6f} "
        f"diff={diff:.6f} rel_diff={rel_diff:.4f} tol={tol} abs_tol={abs_tol}"
    )
    assert actual == pytest.approx(expected, rel=tol, abs=abs_tol), (
        f"[{backend}] {label}: score mismatch — "
        f"actual={actual:.6f}, expected={expected:.6f}, "
        f"rel_diff={rel_diff:.4f}, tol={tol}, abs_tol={abs_tol}"
    )
```

评论区精华

Reviewer Bortlesboat 确认方法正确，指出 pytest.approx(expected, rel=tol, abs=abs_tol) 在底层使用 max(rel * |expected|, abs)，因此绝对值下限仅在相对容差低于它时才生效。通过示例验证了阈值设置合理：对于期望值约 0.1 的文本对比文本分值，相对容差与绝对值下限相当；对于较大期望值，相对容差仍主导。

- 暂无高价值评论线程

风险与影响

- 风险：该变更仅影响测试文件，不涉及生产代码。风险极低，主要风险为万一硬件行为变化导致绝对值容差设置过快或过慢，但范围非常窄（仅 ROCM_AITER_FA 和 FLEX_ATTENTION 后端且仅对低分值生效）。
- 影响：影响范围限于 ROCm 平台上的交叉编码器在线视觉测试，特别是使用 ROCM_AITER_FA 和 FLEX_ATTENTION 注意力后端的场景。变更后测试将更稳定，减少因数值漂移导致的假阳性失败，同时保持对较大分值的敏感性。
- 风险标记：暂无

关联脉络

- PR #35569 Issue about ROCm attention tolerance: PR body 引用了该 issue 作为先前放宽相对容差的依据。

- PR #33493 Disable skinny GEMM for ROCm: 先前提交中已经关联了禁用 ROCm skinny GEMM 的更改，本 PR 部分与之有关。