

# PR #41572 完整报告

vllm-project/vllm

[ROCm][CI] Skip ROCm batch invalid-input test pending torch fix

合并时间: 2026-05-13 18:50

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/41572>

## 执行摘要

- 一句话: 拒绝 bad\_words 空字符串, 避免 500 错误
- 推荐动作: 该 PR 适合快速了解 vllm 的输入验证模式和 review 流程, 但无重大技术决策。可以精读以理解异常选择权衡。

## 功能与动机

根据 PR body, 原因为: Reject empty bad\_words in SamplingParams so stateless generate returns 400 instead of 500。关联 Issue 指出修复了 mi355\_1 测试失败。

## 实现拆解

1. 在 vllm/sampling\_params.py 的 \_verify\_args 方法中, 在 stop 验证之后添加了对 bad\_words 字段的验证: 先使用 assert isinstance(self.bad\_words, list) 确保类型, 然后检查是否存在空字符串, 若存在则抛出 ValueError。
2. 移除了之前尝试将其他 VLLMValidationError 替换为 ValueError 的无关改动 (在 review 中讨论后保留原有用法)。
3. 原本计划在 tests/entrypoints/openai/test\_run\_batch.py 中添加的 ROCm 跳过标记 (@pytest.mark.skipif) 在后续提交中被 revert, 因为 ROCm 基础镜像已包含 PyTorch 修复。

关键文件:

- vllm/sampling\_params.py (模块 采样参数; 类别 source; 类型 core-logic) : 核心变更文件, 在 SamplingParams 验证方法中添加了对 bad\_words 参数的检查, 避免空字符串导致内部错误。

关键符号: 未识别

## 关键源码片段

### vllm/sampling\_params.py

核心变更文件, 在 SamplingParams 验证方法中添加了对 bad\_words 参数的检查, 避免空字符串导致内部错误。

```
# vllm/sampling_params.py _verify_args 方法中新增的验证
```

```
# 原有 stop 验证 ...
```

```

assert isinstance(self.stop, list)
if any(not stop_str for stop_str in self.stop):
    raise ValueError("stop cannot contain an empty string.")
if self.stop and not self.detokenize:
    raise ValueError(
        "stop strings are only supported when detokenize is True. "
        "Set detokenize=True to use stop."
    )

# 新增: 确保 bad_words 是 list 且不含空字符串
assert isinstance(self.bad_words, list)
if any(not bad_word for bad_word in self.bad_words):
    raise ValueError(
        f"bad_words cannot contain an empty string. "
        f"Got bad_words={self.bad_words}"
    )

```

## 评论区精华

1. 异常类型选择: njhill 建议避免使用 `assert` 而改用 `ValueError`; DarkLight1337 建议继续使用 `VLLMValidationError`, 因为它是 `ValueError` 的子类且能与 FastAPI 错误处理集成。最终提交只保留了对 `bad_words` 的简单验证, 使用了 `ValueError`。
  2. TODO 注释: DarkLight1337 询问是否应在新增验证处添加 TODO 以便未来移除。AndreasKaratzas 认为验证本身是合理的改进, 不需要移除, 因此没有添加注释。
  3. ROCm skip 决定: tjanaa 询问 ROCm 基础镜像是否已包含修复。AndreasKaratzas 确认后移除了 `skip`, 使测试重新启用。
- 异常类型选择: 使用 `assert` vs `ValueError` vs `VLLMValidationError` (design): 最终采用 `ValueError`, 未使用 `VLLMValidationError`, 但保留了 `assert` 用于类型检查。
  - 是否应添加 TODO 注释以便未来移除验证 (other): 不添加 TODO, 保留验证作为永久性改进。
  - ROCm 测试 `skip` 是否仍然需要 (testing): Revert `skip`, 重新启用测试。

## 风险与影响

- 风险: 风险极低。变更仅增加输入验证, 不会影响正常请求。若 `bad_words` 是 `None` 或非 `list`, `assert` 会抛出 `AssertionError`, 导致 500 错误, 但之前代码未做任何检查, 行为一致。引入的风险主要是如果外部调用者之前传入了空字符串但期望成功, 现在会失败返回 400。但这更符合规范。
- 影响: 影响范围小: 只影响使用 `SamplingParams` 传入空 `bad_words` 的请求。对于所有其他请求无影响。团队收益: 维护性提升, 错误响应更精确。
- 风险标记: 极小变更

## 关联脉络

- 暂无明显关联 PR