

PR #34689 完整报告

sgl-project/sglang

[AMD] CI: drop the spaces from SGL_EVAL_SPEC (fixes ROCm 7.2 stage-a sgl-eval install)

合并时间: 2026-08-14 03:48

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/34689>

执行摘要

- 一句话: 移除 SGL_EVAL_SPEC 空格, 修复 AMD ROCm 7.2 stage-a 安装失败
- 推荐动作: 建议快速浏览本 PR: 改动只有一行, 但讨论区有信息量。值得学习的点包括: 用 stub 报告 argv 的方式可复现地证明拆词问题; 在 review 要求下主动回退大规模重写、保留外科手术式最小修复; 对遗留的 eval 隐患明确划界并拆分为后续 PR。可关注作者声明的 eval 修复后续提交, 以及 stage-b 的 antlr4 冲突修复。

功能与动机

PR body 明确说明 stage-a-test-1-gpu-small-amd-rocm720 在每次通过 label 门禁的 PR Test ROCm 7.2 (AMD) 上必失败: `pip install sgl-eval @ git+...` 报 `Invalid requirement: '@'`, 3 次重试一致失败, 镜像 fallback 无法解决 (语法错误而非网络问题)。stage-a 门禁阻断所有下游 AMD leg, 导致 AMD ROCm 7.2 的 CI 完全不可用, 需要先解除这一阻塞。

实现拆解

实现过程分 4 步:

1. 根因定位: 对比各平台调用方式, CUDA (`$PIP_CMD install "$SGL_EVAL_SPEC"`) 和 NPU (`{UV_PIP_INSTALL} "$SGL_EVAL_SPEC"`) 都带引号直接调用 pip, 只有 AMD 的 `install_with_retry` 用 `local cmd="$@"` 拼接后执行 `eval "$cmd"`, 引号信息丢失, spec 被重新拆成 `sgl-eval / @ / git+https://...` 三段。这是语法错误, 所以镜像 fallback 也无力回天。
2. 初次方案与会话回退: 作者最初在 `scripts/ci/amd/amd_ci_install_dependency.sh` 中重写 `install_with_retry`, 改用 `argv` 数组规避 `eval`; review 建议保持外科手术式最小修复, 作者将 helper 文件逐字节恢复上游, 只保留 `scripts/ci/utis/sgl_eval_ref.sh` 的一行改动 (`+1/-1`)。
3. 最终修复: 在 `sgl_eval_ref.sh` 中把 `SGL_EVAL_SPEC` 从 `"sgl-eval @ git+https://...#{ref}"` 改为 `"sgl-eval@git+https://...#{ref}"`。PEP 508 允许无空格 `name@url` 形式, `packaging.requirements.Requirement` 对两种写法解析结果完全一致, 因此 CUDA/NPU/CPU 行为不变, 同时使 `spec` 对任何不带引号的展开方式都健壮。
4. 验证与配套: 通过未修改的 `eval` 路径用 stub 验证 `argv` 从 6 个 (拆词) 变为 4 个 (完整 spec); 在 ROCm 7.2 CI 镜像 (`rocm/sgl-dev:v0.5.17-rocm720-mi35x-*`, MI355X 节点) 真实安装成功, `import sgl_eval` 与 `sgl-eval --help` 均正常。amd-bot 确认

stage-a-test-1-gpu-small-amd-rocm720 变为成功。无新增测试文件，依托真实 CI log 验证；环境副作用是 sgl-eval 自身 pin 会把 datasets 5.0.1 -> 4.8.5（既有行为，非本 PR 引入）。

关键文件：

- scripts/ci/utils/sgl_eval_ref.sh（模块 CI 脚本；类别 infra；类型 configuration）：唯一改动文件，一行移除 SGL_EVAL_SPEC 中的空格，消除 AMD eval 展开路径下的拆词问题，直接修复 ROCm 7.2 stage-a 门禁。

关键符号：未识别

关键源码片段

scripts/ci/utils/sgl_eval_ref.sh

唯一改动文件，一行移除 SGL_EVAL_SPEC 中的空格，消除 AMD eval 展开路径下的拆词问题，直接修复 ROCm 7.2 stage-a 门禁。

```
# scripts/ci/utils/sgl_eval_ref.sh 核心片段
# 固定 sgl-eval 依赖的 commit ref，供各平台 CI 安装脚本复用
SGL_EVAL_REF="6690895609dcbc5df1e7b00dd57c9502b868ec4d"

# 关键约束：SGL_EVAL_SPEC 不允许包含空格。
# AMD CI 的 install_with_retry 通过 eval 展开命令行，带空格的
# PEP 508 spec 例如 sgl-eval @ git+https://github.com/sgl-project/sgl-eval.git@...,
# 会被拆成 sgl-eval / @ / git+https://... 三段，pip 收到裸 @ 直接报
# Invalid requirement '@'。
# PEP 508 明确允许 name@url 的无空格形式，
# packaging.requirements.Requirement 对两种写法解析结果完全一致。
SGL_EVAL_SPEC="sgl-eval@git+https://github.com/sgl-project/sgl-eval.git@${SGL_EVAL_REF}"
```

评论区精华

核心讨论围绕 PR 描述准确性和修复范围展开：

- bingxche 复查后质疑“pip 从未被调用”的断言，kangwangamd 用 stderr stub 复验后承认错误并更正：> 被当作重定向，pip 实际被调用，但 >=0.25.0 下限被剥离、输出落入垃圾文件 =0.25.0。描述改为“版本约束被静默丢弃且 pip 输出从 CI 日志消失”，并特别说明 --upgrade 安装最新版已满足下限，当前不算功能故障。
- review 要求保持最小修复，作者回退了 install_with_retry 的 argv 重写，只保留一行 spec 改动；同时指出 eval 仍是遗留隐患（未来任何带空格或重定向字符的参数都会被静默破坏），决定另开独立 PR 处理。
- amd-bot 的 CI 状态确认 stage-a 已转绿，但 ROCm 7.2 整条流水线仍红：stage-b 出现与本次改动无关的 antlr4-python3-runtime 版本冲突（ImportError: Unsupported ANTLR version 4.7.2），需单独修复。
- httpx 断言勘误：pip 是否被调用 (correctness)：PR 描述改为“版本约束被静默丢弃且 pip 输出从 CI 日志消失”，并非 silent no-op；--upgrade 装最新版已满足下限，当前无功能故障。

- 是否重写 `install_with_retry` 为 `argv` 数组 (design): 接受最小修复方案; `eval` 遗留隐患 (`httpx` 下限被吞、参数拆分) 留作独立 PR。
- ROCm 7.2 CI 是否全绿 (testing): 本 PR 目标达成, 但 ROCm 7.2 流水线仍需单独修复 `antlr4` 冲突才能全绿; NVIDIA base-b 失败与本 PR 无关。

风险与影响

- 风险: 整体风险很低, 但仍有几个具体关注点:
 1. 解析一致性: 无空格 `name@url` 形式的 PEP 508 解析等价性已在描述中用 `packaging.requirements.Requirement` 验证, CUDA/NPU/CPU 路径带引号调用不受影响, 回归风险低。
 2. `eval` 遗留隐患: `scripts/ci/amd/amd_ci_install_dependency.sh` 中的 `eval` 仍会吞掉 `--upgrade 'httpx>=0.25.0'` 的下限并重定向 `pip` 输出; 本次修复相当于给 `spec` 打了补丁, 但同一土地雷仍在, 未来添加带空格或重定向字符的参数时可能再次触发静默故障。
 3. 环境副作用: `sgl-eval` 的依赖 `pin` 会把 CI 镜像中的 `datasets 5.0.1` 降级到 `4.8.5`, 可能影响同一镜像上其他端到端评测用例的可复现性 (作者标注为既有行为)。
 4. CI 未全绿: `stage-b` 的 `antlr4` 冲突意味着 ROCm 7.2 仍需后续修复才能作为可信门禁, 本 PR 只解决了第一层阻塞。
 - 影响: 对最终用户和推理服务无影响, 改动仅在 CI 脚本层。对团队的影响是直接解除 AMD ROCm 7.2 门禁阻塞: `stage-a` 是所有下游 AMD leg 的前置条件, 修复后 AMD PR 的 ROCm 7.2 测试可以继续跑完并暴露后续问题 (如 `stage-b` 的 `antlr4` 冲突)。对 CI 基础设施维护者而言, 本 PR 提供了 `eval` 拆词问题的完整根因分析和可复现的最小修复范例, 并明确划定了后续独立工作的边界 (`eval` 重构、`antlr4` 修复)。
 - 风险标记: CI helper 仍依赖 `eval` 展开 (遗留隐患), 仅覆盖 AMD 路径验证, `stage-b` 存在无关的 `antlr4` 阻塞

关联脉络

- PR #34477 (上下文未提供标题): 该 PR 新增 `scripts/ci/utils/sgl_eval_ref.sh`, 其中导出的带空格 PEP 508 `spec` 正是本 PR 所修复问题的根因来源。