

PR #2279 完整报告

radixark/miles

Run the launch script snapshot tests by hand instead of in CI

合并时间: 2026-08-09 18:59

原文链接: <http://prhub.com.cn/radixark/miles/pull/2279>

执行摘要

- 一句话: 启动脚本快照测试移出 CI, 改为手动执行
- 推荐动作: 值得快速阅读, 重点理解“用目录位置控制 CI 发现范围”的模式与 tests/snapshots/README.md 中固化下来的手动执行约定; 若团队后续想把更多重测试移出 CI, tests/manual 目录可复用。本 PR 无源码逻辑改动, 不建议作为精读对象, 但建议在未来修改 tests/ci/ci_register.py 或移动 harness 时回头核对本 PR 建立的路径耦合。

功能与动机

PR body 说明: CI runner 通过 tests/ci/ci_register.py 中的 _DISCOVERY_ROOTS 发现 tests/fast、tests/fast-gpu、tests/e2e、tests/ci 下的测试, 且 tests/fast 下一切按位置隐式注册。这 4 个快照测试只在改动 launcher 或 model 定义时有验证价值 (例如把 62 个 shell 版 model 配置改写成 python 时需要快照锚定行为), 每次 CI 都跑收益低、成本高; 把它们移到 tests/manual 可以“不删除但退出 CI 默认发现范围”, 既保留回归网又避免每次 CI 重复付出约 495 个用例的成本。

实现拆解

1. 迁移测试文件: 4 个快照测试文件 (test_model_args.py、test_py_launch_scripts.py、test_sh_launch_scripts.py、test_self_executing_launchers.py) 整体 rename 到 tests/manual/launch_scripts/, 内容零改动 (+0/-0, 纯文件移动), 并新增 tests/manual/launch_scripts/init.py 使其成为可导入的测试包。
2. 保留 harness 与 CI 覆盖: sh_harness.py、py_harness.py、model_args_harness.py 留在 tests/fast/launch_scripts, 因此 harness 自测 (test_sh_harness.py) 与 shell 脚本卫生检查 (test_shell_script_hygiene.py) 继续在 CI 中运行; 迁移后的测试继续按绝对路径 tests.fast.launch_scripts.* 导入这些 harness, 测试内容无需任何改动。
3. 更新快照文档: tests/snapshots/README.md 把各目录的生成方路径从 tests/fast/launch_scripts/test_.py 改为 tests/manual/launch_scripts/test_.py, 补齐原来缺失的 launch_scripts/self_executing/ 与 model_args/ 两行说明, 并新增手动运行命令 (pytest tests/manual/launch_scripts) 与重新生成快照命令 (MILES_UPDATE_LAUNCH_SCRIPT_SNAPSHOTS=1 pytest tests/manual/launch_scripts)。
4. 验证: discover_ci_files() 在 launch_scripts 下现在只返回 test_sh_harness.py 与 test_shell_script_hygiene.py; 本地执行 pytest tests/manual/launch_scripts 495

passed、pytest tests/fast/launch_scripts 39 passed、pytest tests/ci 319 passed
1 skipped。

关键文件：

- tests/manual/launch_scripts/test_model_args.py（模块 模型参数；类别 test；类型 rename-or-move；符号 TestEveryModelType, TestDiscovery, test_model_args_match_snapshot, test_model_args_are_flags_and_values）：覆盖 60+ 模型 golden argv 的核心快照测试，是 4 个迁移对象中覆盖面最广的一个；内容零改动，迁移后仅手动运行。
- tests/manual/launch_scripts/test_py_launch_scripts.py（模块 启动脚本；类别 test；类型 rename-or-move；符号 glm_checkpoint, _nemotron_checkpoint, _SCRIPTS_WHOSE_DEFAULTS_ARE_UNSUPPORTED）：对所有 scripts/run*.py entrypoint 的 shell 命令与生成配置做快照，含大量模型特例（如 _glm_checkpoint、_nemotron_checkpoint 沙箱构造），迁移不影响其绝对路径导入。
- tests/manual/launch_scripts/test_sh_launch_scripts.py（模块 启动脚本；类别 test；类型 rename-or-move；符号 LaunchScriptCase, recorded）：对全部 sh launcher 外部命令（含完整 ray job submit argv）做快照，是参数化用例数最多的文件，也是本次移出 CI 的主要受益者。
- tests/manual/launch_scripts/test_self_executing_launchers.py（模块 启动脚本；类别 test；类型 rename-or-move；符号 LauncherCase, recorded, _ENTRYPOINTS_THE_HARNESS_CANNOT_SANDBOX）：对自构建命令行的 launcher（p2p 权重传输示例、formal-math 示例）做命令快照，与 sh/py 快照共同构成 launch 体系回归网。
- tests/manual/launch_scripts/__init__.py（模块 测试包；类别 test；类型 test-coverage）：新增空包初始化文件，使 tests/manual/launch_scripts 成为可导入的测试包，是 pytest 正常收集该目录测试的前提。
- tests/snapshots/README.md（模块 快照文档；类别 docs；类型 documentation）：唯一有实际内容变化的文件（+13/-3）：更新生成方路径、补齐缺失目录行、新增手动运行与重新生成快照说明，是团队执行手动测试的入口文档。

关键符号：test_model_args_match_snapshot, test_model_args_are_flags_and_values, test_every_model_is_discovered_and_snapshotted, recorded (fixture in test_sh_launch_scripts.py), recorded (fixture in test_self_executing_launchers.py), _glm_checkpoint / _nemotron_checkpoint (test_py_launch_scripts.py)

关键源码片段

tests/manual/launch_scripts/test_model_args.py

覆盖 60+ 模型 golden argv 的核心快照测试，是 4 个迁移对象中覆盖面最广的一个；内容零改动，迁移后仅手动运行。

```
import json
```

```
import pytest
```

```
# 迁移后仍按绝对路径导入原位置 harness，保证 CI 中的 harness 自测与被迁移测试共用同一份实现
from tests.fast.launch_scripts.model_args_harness import expand_model_args, iter_model_types
from tests.fast.launch_scripts.sh_harness import REPO_ROOT, assert_matches_snapshot
```

```
# 快照目录与模型枚举在模块加载时确定，手动执行与 CI 执行行为一致
_SNAPSHOT_DIR = REPO_ROOT / "tests" / "snapshots" / "model_args"
```

```
_MODEL_TYPES = iter_model_types()
```

```
class TestEveryModelType:
```

```
    @pytest.mark.parametrize("model_type", _MODEL_TYPES)
```

```
    def test_model_args_match_snapshot(self, model_type: str) -> None:
```

```
        """每个模型的 golden argv，防止后续重写模型定义时悄悄漂移"""
```

```
        # 逐 token 做 json.dumps 后按行拼接，使快照可读且可精确比对
```

```
        actual = "\n".join(json.dumps(token) for token in expand_model_args(model_type)) + "\n"
```

```
        assert_matches_snapshot(_SNAPSHOT_DIR / f"{model_type}.txt", actual, model_type)
```

```
    @pytest.mark.parametrize("model_type", _MODEL_TYPES)
```

```
    def test_model_args_are_flags_and_values(self, model_type: str) -> None:
```

```
        """消费端按空白切分 argv，含空白 token 会静默变成两个参数"""
```

```
        tokens = expand_model_args(model_type)
```

```
        assert tokens
```

```
        assert tokens[0].startswith("--")
```

```
        assert all(token == token.strip() and " " not in token for token in tokens)
```

```
class TestDiscovery:
```

```
    def test_every_model_is_discovered_and_snapshotted(self) -> None:
```

```
        """模型若不再匹配发现 glob，其 golden 文件会静默丢失，此测试兜底"""
```

```
        snapshotted = {path.stem for path in _SNAPSHOT_DIR.glob("*.txt")}
```

```
        assert set(_MODEL_TYPES) == snapshotted
```

```
        assert len(_MODEL_TYPES) > 60
```

评论区精华

本 PR 没有任何 review 评论（comments_count 与 review_comments_count 均为 0），无评审交锋可提炼。设计取舍体现在 PR body 与提交信息中：作者强调“移出发现范围而非禁用”，用目录位置替代 skip 标记，保证测试随时可手动运行；提交历史中反复出现 Keep the generated snapshots under one obvious tests/snapshots tree、Group the launch script tests by subject 等工程化约束，说明该快照体系是逐步收敛的结果，而非一次性设计。

- 暂无高价值评论线程

风险与影响

- 风险:

1. CI 覆盖缩减 (主要风险) : 约 495 个快照用例不再自动运行, launcher 或 model 定义回归只能靠手动执行 `pytest tests/manual/launch_scripts` 发现; 若开发者改动后忘记补跑, 回归可能直接合入 main。
2. 绝对路径耦合: 迁移后的测试仍 `import tests.fast.launch_scripts.*` 的 harness (如 `sh_harness`、`py_harness`、`model_args_harness`) , 未来 harness 一旦移动即破坏这些导入; `_SNAPSHOT_DIR` 也依赖 `tests/snapshots` 的相对目录布局。
3. 文档与发现机制耦合: README 直接引用 `_DISCOVERY_ROOTS` 常量与 `tests/ci/ci_register.py`, 若 CI 注册机制重构 (改名或改为显式注册) , 该文档会失真。
4. 无安全、性能或兼容性风险: 纯测试文件搬移, 运行时代码零改动。 - 影响: 对用户与系统无功能影响, 运行时产物不变。对团队与 CI 的影响是双向的: 每次 PR CI 减少约 495 个参数化快照用例, 缓解 runner 压力, 与近期多个 CI 减负 PR 方向一致; 但代价是要求团队建立“改动 launcher 或 model 定义后手动补跑快照”的执行纪律, 否则快照回归网形同虚设。影响范围仅限测试目录布局与 CI 测试发现结果。 - 风险标记: CI 覆盖缩减, 依赖人工执行纪律, 绝对路径耦合, 文档与发现机制耦合

关联脉络

- PR #2231 fix(ci): cancel PR tests after closure: 同属 CI 减负方向: 取消已关闭 PR 的排队 / 运行中测试释放 GPU runner, 与本 PR 把高开销快照测试移出默认发现范围的做法互补。
- PR #2230 fix(ci): disable MI300X runner jobs: 同样通过调整测试注册与调度来释放 CI 资源, 与本 PR“移出而非禁用”的策略形成对照。
- PR #2290 fix(ci): persist every step metric for historical gate: 都在 `tests/ci` 注册与统计机制上做文章; 未来若修改 `_DISCOVERY_ROOTS` 或 `ci_register.py`, 需与本 PR 建立的 `tests/manual` 约定协同。
- PR #2221 fix(ci): split CPU and GPU reusable workflows: CI 工作流与测试编排重构, 与本 PR 同处 CI 基础设施演进线, 共同减少冗余 job 与 runner 占用。