

PR #2035 完整报告

radixark/miles

tests: mirror sglang_speculative_algorithm into session-server test namespaces

合并时间: 2026-08-01 03:46

原文链接: <http://prhub.com.cn/radixark/miles/pull/2035>

执行摘要

- 一句话: 补齐测试夹具缺失的 spec 解码参数
- 推荐动作: 建议快速合并。值得关注的点是: 本 PR 暴露了手工构造 session server 命名空间的脆弱性, 后续可以考虑抽一个公共构造器 (例如 `make_session_server_args`) 统一维护字段列表; 另外在新增 server 侧读取的 args 时, 应同步搜索所有 SimpleNamespace 构造点。

功能与动机

PR body 明确指出: Root cause #2028 使 session-side sample assembly 在 `miles/rollout/session/samples/merge.py` 中无条件读取 `args.sglang_speculative_algorithm`, 并更新了 `test_samples.py` 的命名空间, 但漏掉了两个手工构造 session server args 的测试夹具 (`generation_fixtures.py::with_session_server` 与 `rollout_fixtures.py::_with_session_server`)。它们的 SimpleNamespace 缺少该字段, 导致 `POST /sessions/{id}/samples` 抛出 `AttributeError` → HTTP 500; 生产环境因 `start_session_server` 转发完整解析后的 args 而不受影响。

实现拆解

1. 定位全部手工构造 session server args 的测试命名空间, 共 5 处:
`tests/fast/fixtures/generation_fixtures.py`、`tests/fast/fixtures/rollout_fixtures.py`、`tests/fast/router/test_session_samples_op.py`、`tests/fast/router/session_pretokenized_test_utils.py`、`tests/fast/router/test_session_race_conditions.py`。
2. 对两个从 driver args 派生 server 参数的夹具, 把 `sglang_speculative_algorithm` 透传进去: `generation_fixtures.py::with_session_server` 直接写 `args.sglang_speculative_algorithm`; `rollout_fixtures.py::_with_session_server` 用 `getattr(args, "sglang_speculative_algorithm", None)` 保持向后兼容。
3. 对三个 router 级测试命名空间 (不涉及真实 spec 解码路径), 直接写死 `sglang_speculative_algorithm=None`, 避免 `AttributeError` 且不改变测试语义。
4. 验证: 本地按 CI 环境复现, 修复前 `tests/fast/rollout/generate_hub/test_multi_turn.py` 复现相同的 10 个 500 失败、integration 文件 300 秒无输出; 修复后 integration 19 秒 4 项通过, stage-b 全套 85 passed、17 skipped、0 failed。

配套说明：本次无产品代码、无 schema 或部署变更，全部为测试命名空间配置补强。

关键文件：

- tests/fast/fixtures/generation_fixtures.py（模块 测试夹具；类别 test；类型 test-coverage；符号 with_session_server）：generate_hub agentic 测试的核心夹具，手工构造 session server 命名空间；缺少字段导致样本收集 HTTP 500，是正文点明的两个被遗漏命名空间之一。
- tests/fast/fixtures/rollout_fixtures.py（模块 测试夹具；类别 test；类型 test-coverage；符号 _with_session_server）：inference_rollout 集成测试的 session server 启动器，同样缺少字段导致套件停滞，是正文点明的另一个被遗漏命名空间。
- tests/fast/router/test_session_samples_op.py（模块 样本测试；类别 test；类型 test-coverage；符号 _ARGS）：router 级样本组装单测的 _ARGS 命名空间，同样缺少字段会触发 AttributeError，是 stage-b 的失败点之一。
- tests/fast/router/session_pretokenized_test_utils.py（模块 测试工具；类别 test；类型 test-coverage；符号 make_router_env）：pretokenized 测试工具中的 make_router_env 手工构造命名空间，虽然不进入样本合并路径，但同步补充字段保持一致性。
- tests/fast/router/test_session_race_conditions.py（模块 竞态测试；类别 test；类型 test-coverage；符号 _router_env）：会话竞态测试的 _router_env 手工构造命名空间，补齐字段防止样本路径报错。

关键符号：with_session_server, _with_session_server, make_router_env, _router_env

关键源码片段

tests/fast/fixtures/generation_fixtures.py

generate_hub agentic 测试的核心夹具，手工构造 session server 命名空间；缺少字段导致样本收集 HTTP 500，是正文点明的两个被遗漏命名空间之一。

```
@contextmanager
def with_session_server(
    backend_url: str,
    args: Namespace,
    *,
    port: int,
):
    # 镜像 start_session_server (router_manager.py) 的行为：
    # 每个端口生成一个实例 ID，OpenAIEndpointTracer.create 需要从 per-port map 读取。
    instance_id = uuid.uuid4().hex
    args.session_server_instance_ids = {port: instance_id}

    # 手工构造 session server 命名空间。
    # 样本组装在服务器内部执行，因此 decode 形状参数必须在服务器启动前
    # 传入 server_args，之后再修改 driver args 不会生效。
    server_args = SimpleNamespace(
        miles_router_timeout=30,
        hf_checkpoint=args.hf_checkpoint,
```

```

chat_template_path=args.chat_template_path,
tito_model=args.tito_model,
use_rollout_routing_replay=args.use_rollout_routing_replay,
# 修复点: merge.py 会无条件读取该字段,
# 缺失会导致样本收集路径抛 AttributeError 并返回 HTTP 500。
sglang_speculative_algorithm=args.sglang_speculative_algorithm,
num_layers=getattr(args, "num_layers", None),
moe_router_topk=getattr(args, "moe_router_topk", None),
save_debug_trajectory_data=getattr(args, "save_debug_trajectory_data", None),
session_server_instance_id=instance_id,
)
session_server = SessionServer(server_args, backend_url=backend_url)

server = UvicornThreadServer(session_server.app, host="127.0.0.1", port=port)
server.start()

try:
    yield port
finally:
    server.stop()

```

tests/fast/fixtures/rollout_fixtures.py

inference_rollout 集成测试的 session server 启动器，同样缺少字段导致套件停滞，是正文点明的另一个被遗漏命名空间。

```

@contextmanager
def _with_session_server(args: Namespace, backend_url: str) -> Iterator[UvicornThreadServer]:
    """为 agentic 变体启动 SessionServer（需要 TITO 会话跟踪）。"""
    from types import SimpleNamespace

    # 修复点: 与 generation fixtures 相同的字段透传,
    # 使用 getattr 加默认值, 保证不传该参数的老调用方仍可工作。
    session_args = SimpleNamespace(
        miles_router_timeout=30,
        hf_checkpoint=args.hf_checkpoint,
        chat_template_path=getattr(args, "chat_template_path", None),
        tito_model=getattr(args, "tito_model", "default"),
        use_rollout_routing_replay=getattr(args, "use_rollout_routing_replay", False),
        save_debug_trajectory_data=getattr(args, "save_debug_trajectory_data", None),
        sglang_speculative_algorithm=getattr(args, "sglang_speculative_algorithm", None),
    )
    session_server = SessionServer(session_args, backend_url=backend_url)
    port = find_available_port(31000)
    server = UvicornThreadServer(session_server.app, host="127.0.0.1", port=port)
    try:
        server.start()
        args.session_server_ip = "127.0.0.1"
        args.session_server_ports = [port]
        yield server

```

```
finally:  
    server.stop()
```

评论区精华

PR 没有实质 review 评论，Zhichenzzz 直接批准。有价值的讨论集中在 PR body 中：作者用本地复现证明根因（与 CI 相同的 10 个 `agentic_tool_call` 500）并验证修复效果（stage-b 85 passed, integration 19s 完成），同时解释为何 `session_pretokenized_test_utils.py::make_router_env` 不需要修改（它不镜像 `save_debug_trajectory_data`，也不会进入样本合并路径）。

- 暂无高价值评论线程

风险与影响

- 风险：
 1. 这 5 处命名空间均为手写，未来新增 session server 参数时仍可能遗漏同类字段，属于反复出现的维护风险。
 2. `rollout_fixtures.py` 使用 `getattr` 默认 `None`，可能掩盖“参数未定义”的问题，让测试在不经意间偏离生产行为。
 3. `router` 级命名空间写死 `None`，如果未来这些测试要覆盖 `spec` 解码路径，需要显式改写。
 4. 修复范围限于测试，生产路径不受影响。 - 影响：影响面集中在 CI 与本地测试：修复 stage-b CPU 的 10 个 `[agentic_tool_call]` 失败，以及 stage-a 两个分区因套件停滞导致的约 56 分钟静默与任务取消；开发者本地运行 `generate_hub` 与 `inference_rollout` 的 `agentic` 测试不再遇到 500。对生产系统无影响，因为生产使用完整解析后的 `args`。
- 风险标记：测试夹具字段同步风险，参数默认值掩盖缺参

关联脉络

- PR #2028 session: collect speculative-decoding counters: 直接根因：commit 1849b5596 让 `merge.py` 无条件读取 `sglang_speculative_algorithm`，并只更新了 `test_samples.py` 的命名空间，漏掉本 PR 修复的 5 处测试命名空间。
- PR #1759 (2/2) refactor(session): assemble training samples on the session server; records never leave it: session 侧样本组装重构的语境，`merge.py` 的读取逻辑来自该演进方向，手工命名空间承载字段增多的脆弱性由此放大。