

PR #2218 完整报告

radixark/miles

ci: version TITO metrics by session server

合并时间: 2026-08-08 03:09

原文链接: <http://prhub.com.cn/radixark/miles/pull/2218>

执行摘要

- 一句话: TITO 指标按 session server 版本分列, 统一 W&B 与 CI 门禁 key
- 推荐动作: 值得精读。核心看点是三层 key 治理: producer 在源头产出带版本段的 key、collector 保持直接白名单、gate 默认值按已知噪声放宽约束——这种“key 形状单一事实来源 + 显式白名单 + 有依据的宽松阈值”的组合适合迁移到其它多版本并存的指标场景。另一个值得关注的设计是结构约束测试: test_normal_session_e2e_declarates_both_tito_history_gates 用解析器扫描 e2e 目录, 从机制上杜绝新增用例漏注册 gate。review 中关于 drive-by 变更拆分的流程讨论也值得团队借鉴。

功能与动机

此前 rollout/tito_session_mismatch_rate/... 只有无版本 key, 无法区分历史序列来自 session server v1 还是 v2, W&B 与 CI history 的 key 形状也不统一。PR body 明确目标: "Emit v1/v2 in the producer metric namespace so W&B and CI history use the same keys", 即让 producer 直接产出带版本段的 key, 同时为两个版本分别注册历史门禁, 保证双版本并存期间的历史指标互不混淆。

实现拆解

1. producer 端版本化 (miles/ray/rollout/metrics.py 的 _compute_metrics_from_samples) : 读取样本 metadata["tito_session_mismatch"] 后, 先由 args.use_session_server 推导版本段——True 兼容映射为 "v1", 显式字符串 "v1" / "v2" 原样透传, 随后用 assert 拦截其它取值 (fail-fast)。所有 TITO 指标 (总体 mismatch 率与 4 种子类型) 统一挂到 tito_session_mismatch_rate/{v1|v2} 前缀下, ci_test 的严格类型硬断言也改用版本化前缀。
2. collector 白名单扩展 (miles/utils/tracking_utils/ci_history.py) : TARGET_METRIC_KEYS 增加 rollout/tito_session_mismatch_rate/v1/assistant_text 与 v2/assistant_text 两个 key。只有 assistant_text 进历史门禁——严格类型 (special_token_count、special_token_type、non_assistant_text) 由 metrics 侧的 ci_test 硬断言保证为 0, 无需进 history gate。
3. 历史门禁注册 (tests/ci/metric_history/register.py) : GATE_DEFAULTS 为两个版本 key 配置宽松约束 (steps: "last", rel_up: 1.0、rel_down: 1.0、abs_floor_up: 0.1), 因为已知 Nemotron 的 trailing-newline drift 会使 assistant_text 在纯文本 turn 上必然 mismatch, 阈值过严会产生无效告警。

4. e2e 用例声明与结构约束: test_session_server_multi_role 目录下 7 个用例各注册两个 register_ci_gate(...); tests/ci/test/test_metric_selection.py 新增 test_normal_session_e2e_declares_both_tito_history_gates, 用解析器扫描 e2e 目录, 强制每个用例必须同时声明 v1 / v2 两个 gate, 从机制上杜绝新增模型漏注册。
5. CI 调度配套: test_nemotron3.py 从 120B Super (TP2 + EAGLE) 换成 30B Nano (TP1、无 EAGLE, 因 Nano 无 MTP head 且单卡即可容纳); 依据 run 31143566906 的实测耗时重调各文件 est_time; test_r3_router_equivalence.py (review 后拆出) 与 test_minimax_m27.py 因 Miles Router 弃用 / MiniMax-M2.7 不活跃而标记 disabled, 释放 GPU 容量。
6. 测试配套: fast 测试锁定 producer 版本化 key 形状并覆盖 log_rollout_data 端到端 fan-out; tests/ci/test/test_ci_history.py 验证 v1 / v2 序列在 attempt merge 后保持分离 (_merge_attempt_records); metric selection 测试锁定 GATE_DEFAULTS 的宽松初始约束。

关键文件:

- miles/ray/rollout/metrics.py (模块 指标计算; 类别 source; 类型 core-logic; 符号 _compute_metrics_from_samples): 核心源码改动: 所有 TITO mismatch 指标改为版本化 key 前缀, 并新增 use_session_server 的 fail-fast 断言, 是本次 key 治理的源头。
- tests/fast/ray/rollout/test_metrics.py (模块 指标测试; 类别 test; 类型 test-coverage; 符号 test_clean_tito_metadata_yields_zero_rates_per_mismatch_type, test_tito_metadata_requires_session_server_version, test_rollout_log_fans_out_versioned_tito_keys): 最重要的测试文件: 参数化锁定 v1/v2 版本化 key 形状, 新增 fail-fast 断言测试与 log_rollout_data 端到端 fan-out 测试。
- miles/utils/tracking_utils/ci_history.py (模块 指标采集; 类别 source; 类型 core-logic; 符号 TARGET_METRIC_KEYS, CiHistoryBackend): collector 白名单 TARGET_METRIC_KEYS 扩展两个版本化 TITO key, 是 CI history 门禁的捕获边界。
- tests/ci/metric_history/register.py (模块 门禁注册; 类别 test; 类型 test-coverage; 符号 GATE_DEFAULTS, register_ci_gate): GATE_DEFAULTS 新增两个宽松门禁默认值, 直接决定 TITO assistant_text 序列在 CI history 上的告警灵敏度。
- tests/ci/test/test_metric_selection.py (模块 门禁校验; 类别 test; 类型 test-coverage; 符号 test_session_tito_gate_default_is_loose, test_normal_session_e2e_declares_both_tito_history_gates): 新增结构约束测试, 强制每个 session e2e 用例同时声明 v1/v2 两个 gate, 防止未来新增模型漏注册。
- tests/ci/test/test_ci_history.py (模块 历史测试; 类别 test; 类型 test-coverage; 符号 test_v1_v2_tito_metrics_remain_separate_after_attempt_merge): 验证 v1/v2 两个 TITO 序列在 attempt merge 后保持分离, 确保白名单 key 在 CI history 汇聚链路不被合并。
- tests/e2e/sglang/test_session_server_multi_role/test_nemotron3.py (模块 会话测试; 类别 test; 类型 test-coverage): 除注册双版本 gate 外, 还顺带把 CI 用例从 120B Super (TP2+EAGLE) 换为 30B Nano (TP1、无 EAGLE), 避免 H200 双卡 lane 加载超大模型。

- tests/e2e/sglang/test_session_server_multi_role/test_deepseekv4.py (模块 会话测试; 类别 test; 类型 test-coverage) : 代表其余 6 个 session e2e 用例: 统一注册双版本 gate 并依据实测重调 est_time (1000→1400) , 是 gate 覆盖面的主要载体。

关键符号: _compute_metrics_from_samples, log_rollout_data, CiHistoryBackend.log, register_ci_gate, _merge_attempt_records

关键源码片段

miles/ray/rollout/metrics.py

核心源码改动: 所有 TITO mismatch 指标改为版本化 key 前缀, 并新增 use_session_server 的 fail-fast 断言, 是本次 key 治理的源头。

```
def _compute_metrics_from_samples(args, samples):
    """汇总一批样本的 rollout 指标; TITO mismatch 按 session server 版本分列。

    key 设计: 版本段 (v1 / v2) 在 producer 侧生成, 使 W&B 与 CI history
    共用同一 key 形状; collector 侧只需维护直接白名单。
    """
    response_lengths = [sample.effective_response_length for sample in samples]

    log_dict = {}
    log_dict |= dict_add_prefix(compute_statistics(response_lengths), "response_len/")
    log_dict |= _compute_zero_std_metrics(args, samples)
    log_dict |= _compute_spec_metrics(args, samples)
    log_dict |= _compute_prefix_cache_metrics(args, samples)
    log_dict |= _compute_reward_cat_metrics(args, samples)
    log_dict["repetition_frac"] = np.mean([int(has_repetition(s.response)) for s in samples]).item()
    log_dict["truncated_ratio"] = np.mean([int(s.status == Sample.Status.TRUNCATED) for s in
    samples]).item()

    oldest_versions = [s.oldest_weight_version for s in samples if s.oldest_weight_version is not
    None]
    if oldest_versions:
        log_dict |= dict_add_prefix(compute_statistics(oldest_versions), "weight_version/")
        mixed = sum(1 for s in samples if len(set(s.weight_versions)) > 1)
        log_dict["weight_version/mixed_version_ratio"] = mixed / len(samples)

    # TITO session mismatch 指标: 本 PR 核心改动, 从无版本 key 改为版本化 key。
    tito_vals = [s.metadata.get("tito_session_mismatch") for s in samples]
    tito_vals = [v for v in tito_vals if v is not None]
    if tito_vals:
        # `use_session_server=True` 兼容映射为 `v1`; 显式字符串 `"v1"` / `"v2"`
        # 原样透传。未版本化配置下携带 TITO 元数据会触发 assert, 快速失败。
        session_server_version = "v1" if args.use_session_server is True else args.use_session_
        server
        assert session_server_version in ("v1", "v2"), "TITO metrics require session server v1 or
        v2"
        metric_prefix = f"tito_session_mismatch_rate/{session_server_version}"
```

```

# 总体 mismatch 率与 4 种子类型全部挂到版本化前缀下。
log_dict[metric_prefix] = np.mean([len(v) > 0 for v in tito_vals]).item()
for mtype in ("special_token_count", "special_token_type", "non_assistant_text", "assistant_text"):
    log_dict[f"{metric_prefix}/{mtype}"] = np.mean(
        [any(m.get("type") == mtype for m in v) for v in tito_vals]
    ).item()

if args.ci_test:
    # CI 模式下，三种严格类型必须为 0——出现即代表 TITO 算法或
    # chat template 缺陷；assistant_text 非关键（token 继承自
    # pretokenized prefix），仅记录不拦截。
    for strict_type in ("special_token_count", "special_token_type", "non_assistant_text"):
        rate = log_dict.get(f"{metric_prefix}/{strict_type}", 0)
        assert rate == 0, (
            f"{metric_prefix}/{strict_type}={rate:.4f} must be 0 — "
            "this indicates a bug in the TITO algorithm or chat template. "
            "Please check your tito model and chat template."
        )

return log_dict

```

miles/utils/tracking_utils/ci_history.py

collector 白名单 **TARGET_METRIC_KEYS** 扩展两个版本化 TITO key，是 CI history 门禁的捕获边界。

```

# CI history collector 捕获的指标 key 直接白名单（不做前缀通配）。
# 设计约束：producer（miles/ray/rollout/metrics.py）在源头写入版本段，
# collector 这里保持直接枚举，v1 / v2 两个序列各自成线，互不合并。
TARGET_METRIC_KEYS: tuple[str, ...] = (
    "train/grad_norm",
    "train/ppo_kl",
    "train/train_rollout_logprob_abs_diff",
    "train/train_rollout_kl",
    "rollout/raw_reward",
    # 只有 assistant_text 进入历史门禁：严格类型（special_token_count 等）
    # 在 ci_test 下由 metrics 侧硬断言为 0，无需再进 history gate。
    "rollout/tito_session_mismatch_rate/v1/assistant_text",
    "rollout/tito_session_mismatch_rate/v2/assistant_text",
)

```

tests/ci/metric_history/register.py

GATE_DEFAULTS 新增两个宽松门禁默认值，直接决定 TITO assistant_text 序列在 CI history 上的告警灵敏度。

```

# GATE_DEFAULTS：未显式声明约束时各指标的历史门禁默认值。
# TITO assistant_text 刻意宽松：rel_up / rel_down 均为 1.0（允许序列相对
# 上一步翻倍或归零），abs_floor_up = 0.1 提供绝对值告警下限——已知
# Nemotron 的 trailing-newline drift 会在每个纯文本 turn 上产生 mismatch，

```

```
# 阈值过严会导致常态性误报。
"rollout/tito_session_mismatch_rate/v1/assistant_text": {
  "steps": "last",
  "constraint": {"rel_up": 1.0, "abs_floor_up": 0.1, "rel_down": 1.0},
},
"rollout/tito_session_mismatch_rate/v2/assistant_text": {
  "steps": "last",
  "constraint": {"rel_up": 1.0, "abs_floor_up": 0.1, "rel_down": 1.0},
},
```

评论区精华

Shi-Dong 对 `test_r3_router_equivalence.py` 的 disabled 标记提出流程建议: "Seems a drive-by? Next time it'd be better to send a separate PR for this kind of changes.", 作者回复 "yes. I will sent a standalone pr", 最终合并版本中该文件确实已移出本 PR (对应 "tests split" commit)。对 `test_minimax_m27.py` 的同类建议 ("Same here. Probably deserve a standalone PR."), 该文件仍保留在本 PR 中。Shi-Dong 还质疑 `test_session_v1_v2_parity.py` 的 `est_time` 从 480 降到 190 的依据, 作者回答 "From ci-e2e-time-tune skills in miles repo", 即数据来自仓库内 skill 基于实际 CI 运行统计的调优结果。

- `test_r3_router_equivalence.py` 的 drive-by 改动 (style): guapisolo 回复 "yes. I will sent a standalone pr", 最终合并版本中该文件已不在改动列表, 说明已拆出。
- `test_minimax_m27.py` 的 disabled 与 `est_time` 调整 (style): 作者未直接回复; 该文件的禁用与 `est_time` 调整仍保留在本 PR 中 (未拆分)。
- `test_session_v1_v2_parity.py` 的 `est_time` 下调依据 (question): guapisolo 回答 "From ci-e2e-time-tune skills in miles repo", 即数据来自仓库内 skill 基于实际 CI 运行统计的调优。

风险与影响

- 风险:
 - 指标 key 结构变更 (兼容性): 旧的无版本 key `rollout/tito_session_mismatch_rate/assistant_text` 将不再产出, 依赖旧 key 的 W&B 面板、告警或外部 consumers 会断线; 这是有意统一, 但需确认下游面板已完成迁移。
 - fail-fast 断言 (可用性): `_compute_metrics_from_samples` 中 `assert session_server_version in ("v1", "v2")` 会在任何未配置 `use_session_server` (如 `None`) 但样本携带 TITO 元数据的路径上直接崩溃; 若存在未走参数默认值的调试或本地工作流会受影响。
 - 门禁约束宽松 (漏检风险): `GATE_DEFAULTS` 的 `rel_up / rel_down = 1.0`、`abs_floor_up = 0.1` 使 `assistant_text` 序列只有绝对值超过 0.1 才告警, 真实退化若长期低于该阈值会被漏掉——这是针对 Nemotron 已知漂移的有意权衡。
 - 布尔 / 字符串双态语义: `True` 隐式映射 `v1`, 未来引入 `v3` 时需同步更新此映射、断言、白名单和门禁三处。

- CI 覆盖调整: Miles Router 与 MiniMax-M2.7 用例被禁用, 相关回归覆盖移出常驻 CI; est_time 重调基于单次 run 实测, 不同负载下可能再次漂移。
- 影响:
 - 用户与团队: 所有启用 session server 的 rollout 训练产生的 TITO 指标 key 变化; CI history gate 新增 2 个序列, 每个 session e2e 用例必须显式声明双版本 gate (结构约束测试强制)。
 - 系统: producer 命名空间、TARGET_METRIC_KEYS 白名单、GATE_DEFAULTS 三层 key 治理职责明确, 未来新增版本只需同步三处。
 - CI: est_time 重调与禁用弃用用例释放 H200 GPU 容量; Nemotron 覆盖从 120B 缩到 30B, CI 上验证的模型规格变小。
 - 团队流程: 本 PR 顺带暴露了“CI 调度与指标功能混提”的流程问题, reviewer 明确要求后续 drive-by 变更拆分独立 PR。
 - 风险标记: 指标 key 结构变更 (旧 key 断线), fail-fast 断言依赖参数双态语义, CI 门禁约束刻意宽松, 依赖父 PR #2217

关联脉络

- PR #2217 test(ci): right-size session model GPU coverage: PR body 明确 "Depends on: #2217", 本 PR 建立在父分支之上, 且 Nemotron 用例换模与 est_time 调整与 #2217 的 GPU 覆盖右尺寸化方向一致。
- PR #2202 fix(tito): prevent DeepSeek V4 system-tail mismatch: 同属 TITO 会话一致性治理主线: 本 PR 为 TITO mismatch 提供版本化可观测性与 CI 门禁, 是这类修复的度量配套。
- PR #2123 refactor(session): extract helpers shared by v1 and v2: session server v1/v2 双版本共存是版本化指标的前提, 该重构提取的共享原语与本 PR 的 key 版本化策略同属双版本治理。
- PR #2231 fix(ci): cancel PR tests after closure: 同为 CI 基础设施演进 (gate/metrics 与 workflow 调度), 反映仓库近期对 CI 容量和门禁体系的密集治理。