

PR #1939 完整报告

THUDM/slime

add more cpu ci

合并时间: 2026-05-25 14:40

原文链接: <http://prhub.com.cn/THUDM/slime/pull/1939>

执行摘要

- 一句话: 新增 Sample 序列化和规则打分器 CPU 单元测试
- 推荐动作: 建议阅读 `tests/test_sample.py` 和 `tests/test_rm_math.py`, 其测试组织结构 (清晰分离 round-trip 合约、边界值参数化、手算期望值) 可作为编写同类单元测试的参考。其余测试文件按需精读。

功能与动机

本仓库的 Sample 序列化、boxed-answer 提取、数学归一化、GPQA 字母提取和 F1 计算等模块此前缺乏单元测试, 存在静默回归风险 (如句柄不一致时奖励信号从 1 变为 0 而不会触发任何 CI 告警)。该 PR 旨在通过新增精准的 CPU 单元测试锁定这些合约, 防止未来迭代时引入难以察觉的错误。

实现拆解

1. 新增 Sample 序列化 round-trip 测试 (`tests/test_sample.py`): 构建涵盖所有字段非默认值的 Sample 实例, 测试 `to_dict` 将枚举状态序列化为字符串、展平嵌套的 `SpecInfo/PrefixCacheInfo`, 并验证 `from_dict` 能完整恢复原始对象。同时测试 `update_from_meta_info` 中 `finish_reason` 到 `Status` 的映射正确性。
2. 新增数学类评分器单元测试 (`tests/test_rm_math.py`): 锁定 `last_boxed_only_string` 和 `remove_boxed` 的 brace 计数逻辑及 `\fbox` 回退; 验证 `extract_answer` 和 `extract_boxed_answer` 的便利组合; 通过参数化测试覆盖 `_strip_string` 中 LaTeX 归一化的多个关键替换规则。
3. 新增 DAPO 数学评分器测试 (`tests/test_rm_math_dapo.py`): 区别于 `math_utils` 的行为, 测试 `remove_boxed` 在异常输入时抛出 `AssertionError`; 验证 `normalize_final_answer` 的替换列表、删除列表、正则管道; 覆盖 `is_correct_strict_box` 的精确匹配与截断 (最后 300 字符) 逻辑。
4. 新增 GPQA 评分器测试 (`tests/test_rm_gpqa.py`): 测试 `_extract_letter_from_response` 的三种命名模式、独立字母回退、`</think>` 分离; 验证 `compute_gpqa_reward` 接受标签为单字母、整数索引、富文本标签, 以及 `choices` 传入格式为列表或有序字典。
5. 新增 F1 评分器测试 (`tests/test_rm_f1.py`): 锁定 `normalize_answer` 的定冠词、标点、空白处理顺序; 验证精确匹配为 (1,1,1)、部分重叠的 `precision/recall` 手工计算、无重叠

时的零元组、预测 / 标注为 None 的容错，以及 yes/no/noanswer 特殊分支。

6. 新增 DeepScaler 评分器测试 (`tests/test_rm_deepscaler.py`)：测试基于 `</think>` 和 `###Response` 的分段逻辑，确保仅对尾部内容评分；覆盖无分隔符、无 boxed 答案、空标签、标签类型强制转换 (int/float) 等边界情况。
7. 调整 CI 工作流 (`pr-test.yml.j2`、`pr-test.yml`)：在 CPU 作业中添加 `pytest tests/test_sample.py tests/test_rm_math.py ...` 执行命令，确保新增测试在 PR 预提交阶段自动运行。

关键文件：

- `tests/test_sample.py` (模块 核心类型；类别 test；类型 test-coverage；符号 `_make_sample`, `test_to_dict_serializes_status_as_string_value`, `test_to_dict_flattens_spec_info_and_prefix_cache_info`, `test_round_trip_preserves_every_field`)：覆盖核心数据类型 Sample 的序列化 round-trip 和 status 映射，是训练 / rollout 边界的关键合约测试。
- `tests/test_rm_math.py` (模块 数学规则；类别 test；类型 test-coverage；符号 `test_last_boxed_returns_last_when_multiple`, `test_last_boxed_handles_nested_braces`, `test_last_boxed_falls_back_to_fbox`, `test_last_boxed_returns_none_when_missing`)：锁定 math 风格评分器的核心字符串处理函数 (brace 计数、LaTeX 归一化)，这些函数在被所有 math 类 `rm_type` 共享，且无其他测试覆盖。
- `tests/test_rm_math_dapo.py` (模块 DAPO 数学；类别 test；类型 test-coverage；符号 `test_last_boxed_picks_rightmost`, `test_last_boxed_balances_nested_braces`, `test_last_boxed_returns_none_when_missing`, `test_last_boxed_returns_none_on_terminated`)：验证 DAPO 数学评分器的独特行为 (`remove_boxed` 抛出异常而非静默返回 None)，这些差异容易在统一重构时被忽略。
- `tests/test_rm_gpqa.py` (模块 GPQA 规则；类别 test；类型 test-coverage；符号 `test_extract_letter_named_patterns_and_fallback`, `test_extract_letter_strips_chain_of_thought_before_matching`, `test_extract_letter_returns_none_on_no_match`, `test_extract_letter_returns_none_on_empty`)：覆盖 GPQA 规则打分器的字母提取分支 (多种模式) 和不同的 label/choices 格式，防止字符串解析回归导致奖励信号错误。
- `tests/test_rm_f1.py` (模块 F1 评分；类别 test；类型 test-coverage；符号 `test_normalize_answer`, `test_f1_exact_match_is_perfect`, `test_f1_partial_overlap_hand_derived`, `test_f1_no_token_overlap_returns_zero_metric`)：锁定 F1 评分器的 `normalize_answer` 顺序和特殊标记分支 (yes/no/noanswer)，避免因正则调整导致静默得分变化。
- `tests/test_rm_deepscaler.py` (模块 DeepScaler；类别 test；类型 test-coverage；符号 `test_response_split_on_think_marker_grades_tail`, `test_response_split_on_response_marker_grades_tail`, `test_response_without_any_marker_returns_zero`, `test_response_with_no_boxed_answer_returns_zero`)：覆盖 DeepScaler 风格响应分段规则 (和 `###Response`)，防止因聊天模板变更导致所有样本得零分。

- `slime/utils/misc.py` (模块 工具模块; 类别 `source`; 类型 `dependency-wiring`) : 小幅调整导入关系, 适应新增测试的依赖。改动虽小, 但涉及公共工具模块, 需确保向后兼容。
- `.github/workflows/pr-test.yml.j2` (模块 CI 模板; 类别 `infra`; 类型 `infrastructure`) : CI 模板文件, 新增 CPU 测试的执行步骤, 确保新测试在 PR 自动运行。
- `.github/workflows/pr-test.yml` (模块 CI 配置; 类别 `infra`; 类型 `infrastructure`) : 更新 CI 配置文件以包含新测试 job。

关键符号: `_make_sample`, `_make_args`, `test_to_dict_serializes_status_as_string_value`,
`test_to_dict_flattens_spec_info_and_prefix_cache_info`,
`test_round_trip_preserves_every_field`, `test_from_dict_preserves_unknown_fields_as_attributes`,
`test_round_trip_through_default_constructed_sample`,
`test_status_mapping_for_each_finish_reason`, `test_last_boxed_returns_last_when_multiple`,
`test_last_boxed_handles_nested_braces`, `test_last_boxed_falls_back_to_fbox`,
`test_last_boxed_returns_none_when_missing`,
`test_last_boxed_returns_none_on_terminated_box`,
`test_remove_boxed_strips_wrapper`, `test_remove_boxed_preserves_inner_braces`,
`test_remove_boxed_returns_none_on_malformed_input`,
`test_last_boxed_picks_rightmost`, `test_last_boxed_balances_nested_braces`,
`test_last_boxed_returns_none_when_missing`,
`test_last_boxed_returns_none_on_terminated`, `test_remove_boxed_strips_wrapper`,
`test_remove_boxed_raises_on_malformed`, `test_normalize_final_answer_canonical_substitutions`,
`test_normalize_final_answer_strips_end_tokens`,
`test_extract_letter_named_patterns_and_fallback`,
`test_extract_letter_strips_chain_of_thought_before_matching`,
`test_extract_letter_returns_none_on_no_match`,
`test_extract_letter_returns_none_on_empty`, `test_extract_letter_respects_valid_letters_restriction`,
`test_reward_letter_label_match`, `test_reward_letter_label_mismatch_returns_zero`,
`test_reward_int_label_maps_via_choices_length`, `test_normalize_answer`,
`test_f1_exact_match_is_perfect`, `test_f1_partial_overlap_hand_derived`,
`test_f1_no_token_overlap_returns_zero_metric`,
`test_f1_none_prediction_returns_zero_metric`,
`test_f1_special_token_pred_mismatch_returns_zero`,
`test_f1_special_token_gt_mismatch_returns_zero`,
`test_f1_special_token_exact_match_uses_token_path`,
`test_response_split_on_think_marker_grades_tail`,
`test_response_split_on_response_marker_grades_tail`,
`test_response_without_any_marker_returns_zero`,
`test_response_with_no_boxed_answer_returns_zero`, `test_empty_label_returns_zero`,
`test_label_as_int_is_coerced_to_string`, `test_label_as_float_is_coerced_to_string`,
`test_label_with_boxed_marker_is_extracted_too`

关键源码片段

`tests/test_sample.py`

覆盖核心数据类型 Sample 的序列化 round-trip 和 status 映射，是训练 /rollout 边界的关键合约测试。

```
# -----
# 辅助函数：构建覆盖所有字段非默认值的 Sample 实例，确保 round-trip 测试
# 能触发 to_dict/from_dict 的每一条分支，而非仅验证默认值路径。
# -----
def _make_sample(**overrides) -> Sample:
    """
    构建包含各种非默认字段的 Sample，使得 round-trip 测试可以覆盖
    ``to_dict``/``from_dict`` 的所有代码路径，而不仅仅是默认值情况。
    """
    base = dict(
        group_index=0,
        index=42,
        rollout_id=7,
        prompt="hello",
        tokens=[1, 2, 3],
        multimodal_inputs={"images": ["fake_url"]},
        response="world",
        response_length=5,
        label="42",
        reward=0.75,
        loss_mask=[1, 1, 0, 1, 1],
        weight_versions=["v1"],
        rollout_log_probs=[-0.1, -0.2],
        rollout_routed_experts=[[0, 1], [2, 3]],
        remove_sample=False,
        teacher_log_probs=[-0.3, -0.4],
        status=Sample.Status.COMPLETED,
        metadata={"rm_type": "math"},
        generate_function_path="some.module.fn",
        train_metadata={"loss_type": "policy_loss"},
        session_id="uuid-1234",
        non_generation_time=1.5,
    )
    base.update(overrides)
    return Sample(**base)

@pytest.mark.unit
def test_to_dict_serializes_status_as_string_value():
    """
    验证 ``status`` 字段（枚举）在序列化时被展平为字符串值，
    以便能安全地穿越 Ray actor 边界（JSON / pickle）。
    """
    sample = _make_sample()
    d = sample.to_dict()
    assert d["status"] == "completed" # 必须是字符串，而非枚举成员
```

```
assert isinstance(d["status"], str)
```

tests/test_rm_math.py

锁定 math 风格评分器的核心字符串处理函数（brace 计数、LaTeX 归一化），这些函数在被所有 math 类 rm_type 共享，且无其他测试覆盖。

```
# -----  
# 测试 last_boxed_only_string —— 手工维护的 brace 计数器  
# -----  
  
@pytest.mark.unit  
def test_last_boxed_returns_last_when_multiple():  
    """存在多个 ``\boxed{}`` 时，``rfind`` 应返回最后一个。"""  
    s = r"first attempt \boxed{wrong}, final \boxed{42}"  
    assert last_boxed_only_string(s) == r"\boxed{42}"  
  
@pytest.mark.unit  
def test_last_boxed_handles_nested_braces():  
    """Brace 计数器必须能平衡嵌套花括号，例如 ``\boxed{\frac{1}{2}}``  
    不应因内部 ``{}`` 而过早终止。"""  
    s = r"answer: \boxed{\frac{1}{2}}"  
    assert last_boxed_only_string(s) == r"\boxed{\frac{1}{2}}"  
  
@pytest.mark.unit  
def test_last_boxed_falls_back_to_fbox():  
    """若 ``\boxed`` 不存在，函数也应接受 ``\fbox``。"""  
    s = r"answer: \fbox{7}"  
    assert last_boxed_only_string(s) == r"\fbox{7}"  
  
@pytest.mark.unit  
def test_last_boxed_returns_none_when_missing():  
    assert last_boxed_only_string("plain text, no box") is None  
  
@pytest.mark.unit  
def test_last_boxed_returns_none_on_terminated_box():  
    """花括号未闭合 → 括号计数无法平衡 → 返回 None（而非抛出 IndexError）。"""  
    assert last_boxed_only_string(r"start \boxed{never closes") is None
```

评论区精华

该 PR 无 review 评论，由作者自行合并，未产生公开讨论。

- 暂无高价值评论线程

风险与影响

- 风险：主要风险来自测试代码本身可能存在的缺陷（如 mock 不真实或断言不充分）。但新增测试均为纯逻辑验证，不涉及外部依赖，风险较低。CI workflow 调整若拼写错误可能导致测试不被执行，需确认 YAML 格式正确。
- 影响：对用户无直接影响。对系统，显著提升核心数据类型和规则评分器的测试覆盖率，降低未来重构引入静默 bug 的风险。对团队，提供了标准测试范本，便于后续扩展更多测试。
- 风险标记：测试接口易变，CI 配置正确性

关联脉络

- 暂无明显关联 PR