

# PR #1995 完整报告

THUDM/slime

fix(multi-agent): preserve rollout logprobs

合并时间: 2026-06-01 16:49

原文链接: <http://prhub.com.cn/THUDM/slime/pull/1995>

## 执行摘要

- 一句话: 保留多智能体 rollout logprobs
- 推荐动作: 可直接合并。修复了 issue 中报告的问题, 代码简洁且与主框架的 rollout 实现一致。

## 功能与动机

Issue #1976 报告多智能体示例因缺失 rollout\_log\_probs 导致 GRPO 优势计算时报错 'NoneType' object is not iterable。根本原因是在 examples/multi\_agent/agent\_system.py 中请求 return\_logprob=True 但只提取 token id, 丢弃了对应的 logprob。

## 实现拆解

1. 提取 logprob: 在 examples/multi\_agent/agent\_system.py 的 generate\_response 函数中, 将 output\_token\_logprobs 中的 logprob (item[0]) 提取到 new\_response\_log\_probs 列表。
2. 写入 Sample: 初始化 sample.rollout\_log\_probs (若为 None 则设为空列表), 然后拼接 new\_response\_log\_probs, 并添加断言确保长度与响应 tokens 一致。
3. 启用训练参数: 在 examples/multi\_agent/run-qwen3-30B-A3B-multi-agent.sh 的 GRPO 参数中添加 --use-rollout-logprobs, 使训练能够使用保存的 rollout logprobs。

关键文件:

- examples/multi\_agent/agent\_system.py (模块 多智能体示例; 类别 source; 类型 core-logic; 符号 generate\_response): 核心修复文件, 在 generate\_response 中添加了 logprob 提取和写入 Sample 的逻辑。
- examples/multi\_agent/run-qwen3-30B-A3B-multi-agent.sh (模块 多智能体示例; 类别 other; 类型 core-logic): 添加 --use-rollout-logprobs 参数, 使 GRPO 训练可以使用保存的 rollout logprobs。

关键符号: generate\_response

## 关键源码片段

[examples/multi\\_agent/agent\\_system.py](#)

核心修复文件, 在 generate\_response 中添加了 logprob 提取和写入 Sample 的逻辑。

```

async def generate_response(args, prompt, key):
    try:
        # ... 前置代码 ...
        payload = {"input_ids": prompt_token_ids, "sampling_params": current_sampling_params,
                  "return_logprob": True}
        output = await post(url, payload)

        # 提取新响应 tokens 和对应的 logprobs
        if "output_token_logprobs" in output["meta_info"]:
            output_token_logprobs = output["meta_info"]["output_token_logprobs"]
            # item[1] 是 token id, item[0] 是 logprob
            new_response_tokens = [item[1] for item in output_token_logprobs]
            new_response_log_probs = [item[0] for item in output_token_logprobs]
        else:
            new_response_tokens = []
            new_response_log_probs = []

        # 更新 sample tokens
        sample.tokens = sample.tokens + new_response_tokens
        sample.response_length += len(new_response_tokens)
        # 初始化 rollout_log_probs 并追加
        if sample.rollout_log_probs is None:
            sample.rollout_log_probs = []
        sample.rollout_log_probs += new_response_log_probs
        # 断言长度一致，避免静默错误
        assert len(sample.rollout_log_probs) == sample.response_length, (
            f"rollout logprob length mismatch: {len(sample.rollout_log_probs)} logprobs "
            f"vs {sample.response_length} response tokens"
        )
        sample.response = output["text"]
        # ... 后续代码 ...
    except Exception as e:
        print(f"Error generating response: {e}")
        return None

```

## 评论区精华

无 review 讨论。

- 暂无高价值评论线程

## 风险与影响

- 风险：风险较低。变更范围小，仅影响多智能体示例代码，不涉及核心框架。断言可确保 logprobs 长度一致，避免静默错误。
- 影响：仅影响多智能体示例的用户。修复后 GRPO 训练可正常使用 rollout logprobs，避免崩溃。对现有其他功能无影响。
- 风险标记：缺少测试覆盖

## 关联脉络

- PR #1991 [ci] Add e2e test for delta weight update: 同为示例相关，但无直接依赖。