

PR #2215 完整报告

radixark/miles

fix(mtp): double-shift GPT-path MTP labels

合并时间: 2026-08-06 05:02

原文链接: <http://prhub.com.cn/radixark/miles/pull/2215>

执行摘要

- 一句话: 修复 GPT 路径 MTP 标签双重偏移, 还原训练正确性
- 推荐动作: 值得精读。虽然改动仅 6 行, 但它体现了跨路径数据约定的设计问题: 共享的预处理不应与路径特有逻辑重复。建议关注 `mtp_kwargs["mtp_labels"]` 的契约定义, 以及把偏移收敛到路径边界的思路。同时建议后续补充 nemotron MTP 的 CI 覆盖和针对该契约的单元测试。

功能与动机

PR body 中指出 #1284 添加的 `torch.roll(-1)` 预偏移与 `process_mtp_loss` 的约定一致 (next-token-shifted, 每层再滚一次), 但 GPT inline 路径在 `gpt_model._postprocess` 中已经对原始 tokens 做了预滚, 导致每个 GPT MTP 模型都训练在偏移一位的目标上: `test_mtp1_spec_v2_r3` 失败, MTP loss $14.57 > 1.0$ (首次在 #1572 的 CI 暴露, 与 #1572 本身无关)。

实现拆解

实现分为三步:

1. 取消共享路径的预偏移: 在 `miles/backends/megatron_utils/model.py` 的 `forward_step` 中, 将 `--enable-mtp-training` 分支下的 `torch.roll(batch["tokens"], shifts=-1, dims=-1)` 改回直接传 `batch["tokens"]`, 即 `mtp_kwargs["mtp_labels"]` 现在恒为原始 token。这一改动使数据契约更清晰, 也避免双重偏移。
2. 在 NemotronH shim 中做偏移: 在 `miles_plugins/megatron_bridge/nemotron_h.py` 的 `_install_mamba_model_loss_mask_shim` 的 `forward` 中, 从 `mtp_kwargs` 取出 `mtp_labels` 后, 若非 `None` 则先执行 `torch.roll(shifts=-1, dims=-1)`, 再传给 `MambaModel.forward`。同时补上 `import torch`。
3. 约定统一: 将偏移逻辑放在路径专有边界。GPT inline 路径在 `_postprocess` 中自行逐层 roll; mamba 路径在 shim 中先整体 shift 一次, 再由 `process_mtp_loss` 每深度 roll。两者语义一致, 且 `mtp_kwargs` 的输入输出契约不再包含隐式偏移。

测试方面未新增文件, 依赖已有 e2e 测试 `test_mtp1_spec_v2_r3` 和 nemotron 4-layer smoke 验证。

关键文件:

- `miles/backends/megatron_utils/model.py` (模块 训练循环; 类别 `source`; 类型 `data-contract`; 符号 `forward_step`) : MTP 标签数据契约的源头: 将 `mtp_kwargs["mtp_labels"]` 从预偏移改为原始 token, 消除 GPT 路径双重偏移。
- `miles_plugins/megatron_bridge/nemotron_h.py` (模块 模型桥接; 类别 `source`; 类型 `dependency-wiring`; 符号 `_install_mamba_model_loss_mask_shim`, `MambaModel.forward`) : Mamba 路径的入口 shim: 在这里对 `mtp_labels` 做一次 `torch.roll(-1)`, 使 `nemotron` 语义与之前一致。

关键符号: `forward_step`, `_install_mamba_model_loss_mask_shim`

关键源码片段

`miles/backends/megatron_utils/model.py`

MTP 标签数据契约的源头: 将 `mtp_kwargs["mtp_labels"]` 从预偏移改为原始 token, 消除 GPT 路径双重偏移。

```
# miles/backends/megatron_utils/model.py 的 forward_step (启用 MTP 训练时)
if args.enable_mtp_training:
    # 约定: mtp_kwargs["mtp_labels"] 始终携带原始 token。
    # GPT inline 路径会在 gpt_model._postprocess 中自行逐层 roll;
    # Mamba 路径在 nemotron_h shim 中先整体 shift 一次, 再交给 process_mtp_loss。
    forward_kwargs["mtp_kwargs"] = {"mtp_labels": batch["tokens"]}
```

`miles_plugins/megatron_bridge/nemotron_h.py`

Mamba 路径的入口 shim: 在这里对 `mtp_labels` 做一次 `torch.roll(-1)`, 使 `nemotron` 语义与之前一致。

```
# miles_plugins/megatron_bridge/nemotron_h.py
# Mamba 路径的 MTP 标签约定: process_mtp_loss 期望 next-token-shifted 标签,
# 而 miles 通用训练循环传入的是原始 tokens, 因此在这里统一做一次 shift。
```

```
import torch
```

```
def _install_mamba_model_loss_mask_shim() -> None:
```

```
    """让 MambaModel.forward 静默接受并丢弃 loss_mask, 同时完成 MTP 标签偏移。"""
```

```
    from megatron.core.models.mamba import MambaModel
```

```
    if getattr(MambaModel, "_miles_loss_mask_shim_installed", False):
        return
```

```
    _orig_forward = MambaModel.forward
```

```
    def forward(self, *args, loss_mask=None, mtp_kwargs=None, **kwargs):
```

```
        # process_mtp_loss 期望 next-token-shifted 标签; miles 传入原始 tokens。
```

```
        mtp_labels = (mtp_kwargs or {}).get("mtp_labels")
```

```
        if mtp_labels is not None:
```

```
            # 在每个 MTP 深度再 roll 之前, 先整体左移一位, 保证目标为下一个 token。
```

```
mtp_labels = torch.roll(mtp_labels, shifts=-1, dims=-1)
return _orig_forward(self, *args, loss_mask=loss_mask, mtp_labels=mtp_labels, **kwargs)
```

```
MambaModel.forward = forward
```

```
MambaModel._miles_loss_mask_shim_installed = True
```

评论区精华

该 PR 无 review 评论，只有 yushengsu-thu 的 APPROVE。PR 作者在描述末尾提醒：

qwen35 标签的测试不会运行在 model-scripts 标签的 PR 上，这正是 #1284 漏掉此问题的原因，MTP 测试可能需要更广的 suite 或 label。该提醒指出了 CI 标签策略的盲区，但未形成正式讨论。

- 暂无高价值评论线程

风险与影响

- 风险：

1. 依赖 GPT inline 行为：修复后 GPT 路径完全依赖 gpt_model._postprocess 对原始 token 的预滚逻辑，若该逻辑变更或新增其他 GPT 变体，可能出现再次 off-by-one。
2. nemotron 路径 CI 覆盖不足：PR 说明 nemotron 4-layer CI 配置未启用 --enable-mtp-training，shim 路径未被 CI 直接验证，回归依赖约定和人工验证。
3. 缺少直接测试：本次未新增测试文件，回归保护依赖已有 test_mtp1_spec_v2_r3，无法覆盖 nemotron 的 MTP 路径。
4. 数据契约变化：mtp_kwargs["mtp_labels"] 语义从“已偏移”变为“原始 token”，若存在其他消费方（如自定义 model forward 或插件）会有兼容性风险。
 - 影响：影响范围集中在启用 --enable-mtp-training 的 GPT 系列模型训练（如 qwen3.5 mtp1+spec-v2+r3），修复了训练目标偏移导致的 MTP loss 异常（从 14.57 降至 0.39-0.49）。对 nemotron 路径无行为变化（同样的一次 shift 只是移动了位置）。团队影响：消除 #1572 等后续 PR 的 CI 噪音，恢复测试可信度。影响面中等，核心训练路径有小改动。
 - 风险标记：核心训练路径变更，缺少直接测试覆盖，跨路径约定依赖

关联脉络

- PR #1284 Nemotron RL support: 本 PR 修复了 #1284 引入的 MTP 标签预偏移在 GPT 路径上的双重偏移问题，是直接的回归源。