

PR #3163 完整报告

radixark/miles

Fix missing Nemotron-H Mamba convolution mappings

合并时间: 2026-09-10 05:43

原文链接: <http://prhub.com.cn/radixark/miles/pull/3163>

执行摘要

- 一句话: 修复 Nemotron-H Mamba 卷积参数映射缺失, 重新启用相关 e2e 测试。
- 推荐动作: 该 PR 是一个重要的兼容性修复, 解决了上游依赖变更带来的关键问题。实现方案精准、测试覆盖充分 (包括新增的单元测试和重新启用的 e2e 测试), 并补充了必要的文档。对于关注模型兼容性或 Nemotron-H 系列的工程师, 建议精读其映射逻辑和测试策略。其解决阻塞性问题并恢复 CI 覆盖率的做法, 是处理此类上游变更的良好范例。

功能与动机

解决 Megatron 上游版本升级后参数命名约定变更导致的兼容性问题。PR body 明确指出: “Recent Megatron versions expose Mamba convolution parameters as `mixer.conv1d_weight` and `mixer.conv1d_bias`, while the Nemotron-H bridge maps the older dotted names. The missing mappings leave those tensors unloaded and omit them from rollout weight updates.”

实现拆解

1. 核心映射添加: 在 `miles_plugins/megatron_bridge/nemotron_h.py` 的 `mapping_registry` 方法中, 新增对 `mixer.conv1d_weight` 和 `mixer.conv1d_bias` 的映射。使用 `MambaConv1dMapping` 类, 它能处理 x/B/C 张量的张量并行转换。实现中检查了上游是否已提供相同映射, 避免重复。
2. 单元测试: 新增 `tests/fast/backends/megatron_utils/test_nemotron_h_conv_mapping.py`, 通过参数化测试验证映射逻辑在各种场景下 (新旧映射共存、仅部分存在等) 都能正确工作, 并确保映射不被覆盖。
3. 端到端测试启用: 移除 `tests/e2e/megatron/model_scripts/test_nemotron_3_ultra_4layer_ci.py` 中的 `disabled` 参数和相关 TODO 注释, 重新启用该测试作为验收。
4. 文档更新: 在 `docs/models/nemotron/nemotron-3-nano-moe.md` 中新增“检查点兼容性”章节, 说明支持的两种参数命名及其影响。

关键文件:

- `miles_plugins/megatron_bridge/nemotron_h.py` (模块 模型桥接; 类别 `source`; 类型 `dependency-wiring`; 符号 `mapping_registry`): 核心修复文件, 在模型桥接插件的映射注册表中添加了缺失的 Mamba 卷积参数映射。

- tests/fast/backends/megatron_utils/test_nemotron_h_conv_mapping.py (模块 映射测试 ; 类别 test; 类型 test-coverage; 符号 Mapping, MambaMapping, load_registry, test_convolution_mapping_coverage) : 新增的单元测试文件, 全面验证卷积映射添加逻辑在各种输入场景下的正确性。
- tests/e2e/megatron/model_scripts/test_nemotron_3_ultra_4layer_ci.py (模块 e2e 测试 ; 类别 test; 类型 test-coverage) : 重新启用的端到端测试, 作为修复的验收测试, 验证 Nemotron-3 Ultra 模型训练脚本的完整流程。
- docs/models/nemotron/nemotron-3-nano-moe.md (模块 模型文档; 类别 docs; 类型 documentation) : 补充文档说明, 解释新旧参数命名约定及其对检查点兼容性的影响。

关键符号: mapping_registry

关键源码片段

miles_plugins/megatron_bridge/nemotron_h.py

核心修复文件, 在模型桥接插件的映射注册表中添加了缺失的 Mamba 卷积参数映射。

```
# miles_plugins/megatron_bridge/nemotron_h.py 中的关键变更片段
from megatron.bridge.models.conversion.param_mapping import MambaConv1dMapping

def mapping_registry(self):
    registry = super().mapping_registry()
    base = list(registry.mappings if hasattr(registry, "mappings") else registry._mappings)
    extras = [AutoMapping(megatron_param=m, hf_param=h) for m, h in _NEMOTRONH_MOE_MAPPINGS.items()]
    # 获取已存在的映射名称, 避免重复添加
    known_names = {mapping.megatron_param for mapping in base}
    # 为 Mamba 卷积参数添加映射, 支持下划线和点号两种命名约定
    mamba = [
        MambaConv1dMapping(
            megatron_param=f"decoder.layers.*.mixer.conv1d_{suffix}",
            hf_param=f"backbone.layers.*.mixer.conv1d.{suffix}",
        )
        for suffix in ("weight", "bias")
        if f"decoder.layers.*.mixer.conv1d_{suffix}" not in known_names
    ]
    # 将新的 mamba 映射添加到注册表中
    return MegatronMappingRegistry(*base, *mamba, *extras)
```

tests/fast/backends/megatron_utils/test_nemotron_h_conv_mapping.py

新增的单元测试文件, 全面验证卷积映射添加逻辑在各种输入场景下的正确性。

```
# tests/fast/backends/megatron_utils/test_nemotron_h_conv_mapping.py 中的核心测试函数
@pytest.mark.parametrize("attribute", ["mappings", "_mappings"])
@pytest.mark.parametrize("existing_suffixes", [(), ("weight",), ("weight", "bias")])
def test_convolution_mapping_coverage(
    monkeypatch: pytest.MonkeyPatch, attribute: str, existing_suffixes: tuple[str, ...]
) -> None:
```

```

# 模拟已有的旧格式映射（使用点号）
legacy = [
    MambaMapping(f"decoder.layers.*.mixer.conv1d.{suffix}", f"backbone.layers.*.mixer.
conv1d.{suffix}")
    for suffix in ("weight", "bias")
]
# 模拟可能已存在的新格式映射（使用下划线）
existing = [
    MambaMapping(f"decoder.layers.*.mixer.conv1d_{suffix}", f"backbone.layers.*.mixer.
conv1d.{suffix}")
    for suffix in existing_suffixes
]
base = legacy + existing
# 加载被测试的映射注册模块
mappings = load_registry(monkeypatch, base, attribute)
by_name = {mapping.megatron_param: mapping for mapping in mappings}
# 验证新旧两种格式的映射都存在且类型正确
for separator in (".", "_"):
    for suffix in ("weight", "bias"):
        mapping = by_name[f"decoder.layers.*.mixer.conv1d{separator}{suffix}"]
        assert isinstance(mapping, MambaMapping)
        assert mapping.hf_param == f"backbone.layers.*.mixer.conv1d.{suffix}"
# 验证原始映射未被覆盖
for mapping in base:
    assert by_name[mapping.megatron_param] is mapping

```

评论区精华

PR review 过程中无实质性技术讨论。Claude[bot] 的自动审查未发现问题，最终由 Zhichenzzz 批准合并。讨论焦点集中在实现正确性验证上，PR body 提供了详细的验证证据，包括回归测试、模型审计结果和训练烟雾测试计划。

- 暂无高价值评论线程

风险与影响

- 风险：风险较低。变更集中在模型桥接插件的映射逻辑，影响范围明确限定在 Nemotron-H 模型。核心逻辑是添加映射而非修改现有路径，且增加了条件检查（if not in known_names），降低了引入重复或冲突的风险。主要风险在于映射错误可能导致特定模型参数加载失败，但已通过全面的单元测试和端到端测试覆盖。
- 影响：直接影响使用 Nemotron-H 模型（如 Nano-30B-A3B, Super-120B-A12B）进行训练和推理的用户。确保这些模型在 Megatron 新版本下能够正确加载权重并进行分布式训练，修复了之前因参数缺失导致的静默错误。对团队而言，解决了模型启用的阻塞问题，并提升了持续集成的覆盖率。
- 风险标记：上游依赖变更兼容性

关联脉络

- PR #3169 fix(ci): keep the Nemotron-H plugin importable without megatron.bridge: 该 PR 修复了同一 Nemotron-H 插件在 CPU CI 环境下因模块导入失败导致的问题，与本 PR 修改的是同一个文件 `miles_plugins/megatron_bridge/nemotron_h.py`，属于同一功能线的连续维护。
- PR #3160 [CI] disable the two FT determinism tests #3148 missed: 同为近期为保持 CI 绿灯而进行的测试调整（禁用测试），与本 PR 的重新启用测试形成对比，反映了团队在维护 CI 稳定性方面的持续工作。