PR #29131 完整报告

sgl-project/sglang

[NPU] Adapt MiMo-V2.5-W8A8

合并时间：2026-07-21 09:16

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/29131

执行摘要

- 一句话：为 NPU 适配 MiMo-V2.5-W8A8，添加量化配置回退和测试覆盖
- 推荐动作：该 PR 变更虽小，但展示了在量化配置兼容性处理上一种简洁的 fallback 模式，值得类似场景参考。新增的集成测试配置完整，可作为 NPU 模型新适配的模板。建议在类似量化路径重构时考虑更统一的错误处理。

功能与动机

MiMo-V2.5-W8A8 模型在 NPU 部署时，由于量化描述（quant_description）的 key 使用了非标准命名（如 fused name 未解包），导致 packed_modules_mapping 重映射后的前缀在描述中找不到，进而触发 KeyError。PR 通过回退机制解决此兼容性问题，并新增集成测试确保部署正确性。同时，针对音频编码器加载失败做了优雅降级（后因范围问题回退）。

实现拆解

1. 量化配置键回退（python/sglang/srt/layers/quantization/modelslim/modelslim.py get_quant_method）：在 packed_modules_mapping 重映射后，检查 prefix_in_quant_config + '.weight' 是否存在于 self.quant_description 中；不存在则回退为原始 prefix，避免模型加载时异常。
2. 模型权重常量添加（python/sglang/test/ascend/test_ascend_utils.py）：新增 MIMO_V2_5_W8A8_WEIGHTS_PATH 常量指向模型路径 solinliu/MiMo-V2.5-W8A8，供测试复用。
3. 集成测试类（test/manual/ascend/llm_models/test_npu_mimo_v2_5_w8a8.py）：新增 TestMiMoV25W8A8GraphWithMTP 类，继承 GSM8KAscendMixin 和 CustomTestCase，配置 8 卡 TP、EAGLE 推测解码、modelslim 量化、DataParallel 等参数，验证 GSM8K 准确率不低于 0.9。

关键文件：

- python/sglang/srt/layers/quantization/modelslim/modelslim.py（模块 量化层；类别 source；类型 data-contract；符号 get_quant_method）：核心变更文件。在 get_quant_method 中增加前缀重映射后验证与回退逻辑，解决 MiMo-V2.5 量化描述 key 不匹配问题。
- test/manual/ascend/llm_models/test_npu_mimo_v2_5_w8a8.py（模块 模型测试；类别 test；类型 test-coverage；符号 TestMiMoV25W8A8GraphWithMTP）：新增的 NPU 集

成测试，验证 MiMo-V2.5-W8A8 在 GSM8K 上的推理精度，覆盖 EAGLE 推测解码、DataParallel、modelslim 量化等特性。

- python/sglang/test/ascend/test_ascend_utils.py（模块 测试工具；类别 test；类型 test-coverage）：添加 MIMO_V2_5_W8A8_WEIGHTS_PATH 常量，供测试引用模型权重路径。

关键符号：get_quant_method (modelslim.py), TestMiMoV25W8A8GraphWithMTP (test_npu_mimo_v2_5_w8a8.py)

关键源码片段

python/sglang/srt/layers/quantization/modelslim/modelslim.py

核心变更文件。在 get_quant_method 中增加前缀重映射后验证与回退逻辑，解决 MiMo-V2.5 量化描述 key 不匹配问题。

```
# python/sglang/srt/layers/quantization/modelslim/modelslim.py (get_quant_method)
class ModelSlimConfig:
    ...
    def get_quant_method(self, layer, prefix):
        ...
        if isinstance(layer, LinearBase):
            ... # 省略无关部分
            packed_modules_mapping_subset = self.packed_modules_mapping.get(key, {})
            prefix_in_quant_config = prefix
            proj_name = prefix.split('.')[-1]
            if proj_name in packed_modules_mapping_subset:
                prefix_in_quant_config = prefix.replace(
                    proj_name, packed_modules_mapping_subset[proj_name][0]
                )
                # 验证重映射后的前缀是否存在于 quant_description 中。
                # 如果不存在（例如量化 JSON 使用了 fused name 未解包），
                # 则回退到原始前缀，避免后续查询 scheme 时 KeyError。
                if prefix_in_quant_config + '.weight' not in self.quant_description:
                    prefix_in_quant_config = prefix
            if self.is_layer_skipped(prefix, packed_modules_mapping_subset) \
               or self.is_layer_skipped(prefix, self.packed_modules_mapping):
                return UnquantizedLinearMethod()
            layer.scheme = self.get_linear_scheme(layer, prefix_in_quant_config)
            ...
```

test/manual/ascend/llm_models/test_npu_mimo_v2_5_w8a8.py

新增的 NPU 集成测试，验证 MiMo-V2.5-W8A8 在 GSM8K 上的推理精度，覆盖 EAGLE 推测解码、DataParallel、modelslim 量化等特性。

```
# test/manual/ascend/llm_models/test_npu_mimo_v2_5_w8a8.py
import unittest

from sglang.test.ascend.gsm8k_ascend_mixin import GSM8KAscendMixin
```

```
from sglang.test.ascend.test_ascend_utils import MIMO_V2_5_W8A8_WEIGHTS_PATH
from sglang.test.test_utils import CustomTestCase

class TestMiMoV25W8A8GraphWithMTP(GSM8KAscendMixin, CustomTestCase):
    """Testcase: 验证 MiMo-V2.5-W8A8 在 GSM8K 上使用 cuda graph 和
    MTP（推测解码）的推理精度。

    覆盖特性：Prefill+Decode, cuda graph, EAGLE, modelslim 量化。
    """

    # 指定模型权重路径（由 test_ascend_utils 提供常量）
    model = MIMO_V2_5_W8A8_WEIGHTS_PATH
    # 期望的最低准确率
    accuracy = 0.9
    # 与 --reasoning-parser mimo 配套的启动参数
    other_args = [
        '--trust-remote-code',
        '--mem-fraction-static', '0.75',
        '--attention-backend', 'ascend',
        '--tp-size', '8',
        '--reasoning-parser', 'mimo',
        '--speculative-algorithm', 'EAGLE',
        '--speculative-num-steps', '3',
        '--speculative-eagle-topk', '1',
        '--speculative-num-draft-tokens', '4',
        '--enable-multi-layer-eagle',
        '--quantization', 'modelslim',
        '--speculative-draft-model-quantization', 'unquant',
        '--dp-size', '2',
        '--enable-dp-attention',
        '--enable-dp-lm-head',
    ]

if __name__ == '__main__':
    unittest.main()
```

评论区精华

主要交互为 CI 流程自动评论：sglang-npu-bot 要求“Only modify the code related to the NPU. After the NPU test cases are completed, the changes will be merged.” 作者通过 /rerun-failed-ci 触发重跑。未发现技术设计层面的深入讨论。

- NPU 代码修改范围限制 (other): 作者遵循此原则，最终所有修改均限于 NPU 相关代码（modelslim 回退逻辑和 NPU 集成测试）。音频编码器异常处理被回退（commit 'revert changes for audio'）。

风险与影响

- 风险：影响范围有限：仅修改 modelslim 量化路径中的前缀查找逻辑及新增测试。风险在于回退逻辑可能导致错误地跳过量化的层，但回退后仍然会进入 is_layer_skipped 判断，若实际层需要量化但回退后找不到 scheme，最终会 fallback 到 UnquantizedLinearMethod，可能导致推理精度偏离预期。不过由于该回退仅在 prefix 重映射后找不到时触发，且原模型已测试精度达标，风险可控。
- 影响：对用户：使 MiMo-V2.5-W8A8 模型能在 NPU 上正常加载和推理，扩展了支持的模型列表。对系统：新增约 50 行测试代码，不影响已有功能。对团队：提供了 NPU + 量化 + EAGLE 的组合测试范本。
- 风险标记：核心量化路径变更，新增测试覆盖

关联脉络

- PR #31456 [NPU] fix modelslim quant tensor name: 同一文件 python/sglang/srt/layers/quantization/modelslim/modelslim.py，同样修复 NPU 上 modelslim 量化兼容性问题。本 PR 的 fallback 逻辑可能与张量命名问题的修复相关联。