PR #24540 完整报告

sgl-project/sglang

[NPU] [Bugfix] Wan quantization fix

合并时间：2026-05-11 13:32

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/24540

执行摘要

- 一句话：修复 Wan 模型在 NPU 上的量化方案识别
- 推荐动作：该 PR 修复了关键 regression，改动虽小但涉及核心量化配置路径，值得精读。设计上通过 reverse_param_names_mapping 解耦内部命名与规范命名的做法值得关注，可在未来支持更多量化模型时复用。

功能与动机

Issue #24518 报告在合并 PR #23625 后，NPU 上运行 Wan2.2-T2V-A14B-Diffusers-w8a8 量化模型出现 No modelslim compatible scheme 错误。根本原因是 ModelSlim 的 _get_scheme_from_parts 使用模型的内部层名（如 blocks.0.self_attn.to_q）去查询量化配置，但量化配置文件使用架构规范名（如 blocks.0.attn1.to_q）。WanVideo 架构配置中已有 reverse_param_names_mapping 映射，但 ModelSlim 初始化路径未使用该映射，导致查询失败。

实现拆解

1. 在 ModelSlimConfig 中接受名称映射：在 __init__和 from_config 方法中新增 reverse_param_names_mapping 参数，并利用 get_param_names_mapping 构建内部名称到规范名称的映射器 _name_mapper。
2. 使用映射器查询量化方案：在 _get_scheme_from_parts 中，先调用映射器将层名转为规范名（若映射器存在），然后基于规范名从 quant_model_description.json 中获取量化类型，并确定正确的 prefix 传递给具体量化方案。
3. 在加载路径中传递映射：在 transformer_load_utils.py 的 _resolve_quant_config 中，从架构配置读取 reverse_param_names_mapping，并作为参数传入 get_quant_config。
4. 在量化工具函数中转发：在 quantization_utils.py 的 get_quant_config 中，新增 reverse_param_names_mapping 参数，并在找到 modelslim 配置时传递给 ModelSlimConfig.from_config。
5. 补充 FSDP 加载支持：在 fsdp_load.py 中添加缺失的量化参数后缀（input_offset、quant_bias、deq_scale）到 sharding 白名单，并设置 requires_grad=False 避免元张量拷贝错误。
6. 更新性能基线和警告信息：更新 perf_baselines_npu.json 中的基线数据（反映修复后的实际延迟变化），并在 npu/utils.py 中改进 CPU 卸载的性能警告提示。

关键文件：

- python/sglang/multimodal_gen/runtime/layers/quantization/modelslim.py（模块 量化配置；类别 source；类型 data-contract；符号 init, from_config, _get_scheme_from_parts）：核心修复文件：接受 reverse_param_names_mapping 并在 _get_scheme_from_parts 中使用映射器转换层名，使量化方案查询能正确匹配架构规范名。
- python/sglang/multimodal_gen/runtime/loader/transformer_load_utils.py（模块 模型加载；类别 source；类型 core-logic）：负责从架构配置读取 reverse_param_names_mapping 并传递给下游，是数据传递的关键环节。
- python/sglang/multimodal_gen/runtime/utils/quantization_utils.py（模块 量化工具；类别 source；类型 core-logic）：作为中转函数，接受并转发 reverse_param_names_mapping 到 ModelSlimConfig.from_config。
- python/sglang/srt/hardware_backend/npu/utils.py（模块 NPU 工具；类别 source；类型 core-logic）：改进性能警告信息，提示用户使用 --dit-cpu-offload false 来避免性能下降。
- python/sglang/multimodal_gen/test/server/ascend/perf_baselines_npu.json（模块 性能基线；类别 test；类型 test-coverage）：更新 NPU 性能基线，反映修复后的延迟变化，确保 CI 性能检测准确。

关键符号：ModelSlimConfig.init, ModelSlimConfig.from_config, ModelSlimConfig._get_scheme_from_parts, get_quant_config, _resolve_quant_config

关键源码片段

python/sglang/multimodal_gen/runtime/layers/quantization/modelslim.py

核心修复文件：接受 reverse_param_names_mapping 并在 _get_scheme_from_parts 中使用映射器转换层名，使量化方案查询能正确匹配架构规范名。

```
# modelslim.py - 核心变更：接受 reverse_param_names_mapping 用于名称映射
from sglang.multimodal_gen.runtime.loader.utils import get_param_names_mapping

class ModelSlimConfig(QuantizationConfig):
    def __init__(
        self,
        quant_config: Dict[str, Any] = {},
        reverse_param_names_mapping: dict = None, # 新增参数，默认 None 保持兼容
    ):
        super().__init__()
        self.quant_description = quant_config
        # ... 其他初始化 ...
        # 构建映射器：如果提供了映射字典，则生成从内部名到规范名的映射函数
        self._name_mapper = (
            get_param_names_mapping(reverse_param_names_mapping)
            if reverse_param_names_mapping is not None
            else None
        )

    @classmethod
```

```
    def from_config(
        cls, config: Dict[str, Any], reverse_param_names_mapping: dict = None
    ) -> ModelSlimConfig:
        # 现在将映射参数传递到构造函数
        return cls(config, reverse_param_names_mapping)

    # ... 在查询量化方案的方法中:
    def _get_scheme_from_parts(self, layer_name: str) -> ModelSlimLinearScheme:
        full_weight_name = layer_name + ".weight"
        if self._name_mapper is not None:
            # 使用映射器将内部层名转为规范名（例如 blocks.0.self_attn.to_q -> blocks.0.attn1.to_
            q）
            mapped_name, _, _ = self._name_mapper(full_weight_name)
        else:
            mapped_name = full_weight_name

        quant_type = self.quant_description.get(mapped_name, "")
        prefix = mapped_name.removesuffix(".weight")

        if quant_type == "W8A8_DYNAMIC" or quant_type == "W8A8":
            return ModelSlimW8A8Int8(quant_config=self.quant_description, prefix=prefix)
        elif quant_type == "W4A4_DYNAMIC":
            return ModelSlimW4A4Int4(quant_config=self.quant_description, prefix=prefix)
        # ... 其他方案
```

评论区精华

- gemini-code-assist指出两个关键问题：
  1. from_config 新增参数后，其他调用点未传递该参数可能导致 TypeError（已通过添加默认值 None 解决）。
  2. 在 _get_scheme_from_parts 的 fallback 路径中，mapped_name 未包含 .weight 后缀导致配置查询失败（已通过统一使用 full_weight_name 修复）。 - ping1jing2建议将 import get_param_names_mapping 移到 TYPE_CHECKING 块之上，但因循环导入问题未能实现，作者回复 “Tried, not worked”。 所有讨论已在最终提交中得到解决或说明。
- from_config 新增参数可能破坏其他调用点 (design): 作者为 reverse_param_names_mapping 添加默认值 None，保持向后兼容。已解决。
- 查询量化配置时缺少 .weight 后缀 (correctness): 作者通过统一使用 full_weight_name = layer_name + '.weight' 修复。
- get_param_names_mapping 导入位置 (style): 保留当前位置，因循环导入限制。

风险与影响

- 风险：
  - 回归风险：修改 ModelSlimConfig 初始化签名，但通过默认参数保持向后兼容；其他量化模型加载不受影响。
  - 性能风险：名称映射仅在模型加载时执行一次，对运行时性能无影响。

  - 兼容性：仅影响 NPU 后端的 ModelSlim 量化加载流程，GPU 端不受影响。
  - 测试覆盖：缺乏新增单元测试，仅更新了性能基线（该基线会随模型输出变化，但未验证错误场景）。
- 影响：
  - 用户影响：修复了 Wan2.2-W8A8 量化模型在 NPU 上的加载问题，受影响的用户可直接使用该模型。
  - 系统影响：无。
  - 团队影响：明确了量化配置中名称映射的重要性，为后续类似模型的量化支持提供参考模式。
  - 风险标记：量化配置路径变更，NPU 专用修复，未添加新测试

关联脉络

- PR #23625 Flux2 nvfp4 quantization correctness on Blackwell (B200): 该 PR 改变了量化前缀格式，直接导致了此 PR 修复的 regression（内部层名与规范名不匹配）。