PR #40786 完整报告

vllm-project/vllm

Fix PP in Gemma4

合并时间：2026-04-29 18:17

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/40786

执行摘要

- 一句话：修复 Gemma4 PP 中 residual 和 per_layer_inputs 同步
- 推荐动作：建议阅读此 PR 以了解 Gemma4 在 PP 下的张量同步设计，特别是 IntermediateTensors 如何按需传递。对于有类似 PP + PLE 实现的模型开发者，这是一个值得关注的决策案例——如何平衡泛化与模型特定优化。

功能与动机

PR body 指出："Fixes PP in Gemma4 for Gemma4ForCausalLM. per_layers_inputs needs to be passed only for smaller models and residual is None from decoder layers so need not to be synchronized among different ranks." 原始实现中不必要的张量同步可能导致错误或性能开销。

实现拆解

1. 移除 _make_empty_intermediate_tensors 中的 residual 张量（文件 vllm/model_executor/models/gemma4.py）：在创建空的 IntermediateTensors 时不再包含 "residual" 项，因为 Gemma4 的 decoder layers 不输出独立 residual。
2. 调整 forward 中非首 rank 分支：不再无条件从 intermediate_tensors 读取 residual 和 per_layer_inputs，改为仅当 per_layer_inputs 参数非 None（即首 rank 传递了该值）时才从 intermediate_tensors 中读取；residual 在两者中都统一设为 None。
3. 调整最终返回的 IntermediateTensors：在非末 rank 返回时，只打包 hidden_states 和（非 None 的）per_layer_inputs，不再包含 residual。 以上改动确保 PP 各 rank 间只传递必要张量，符合 Gemma4 的设计预期。

关键文件：

- vllm/model_executor/models/gemma4.py（模块 模型执行器；类别 source；类型 data-contract；符号 forward, _make_empty_intermediate_tensors）：修复 PP 在 Gemma4 中的核心模型文件，所有 9 行新增和 16 行删除均在此文件，涉及 IntermediateTensors 构造和 forward 路径。

关键符号：forward, _make_empty_intermediate_tensors

关键源码片段

vllm/model_executor/models/gemma4.py

修复 PP 在 Gemma4 中的核心模型文件，所有 9 行新增和 16 行删除均在此文件，涉及 IntermediateTensors 构造和 forward 路径。

```
# vllm/model_executor/models/gemma4.py

# 改动 1: _make_empty_intermediate_tensors 中移除 residual 张量
# Gemma4 decoder layers 不产生独立 residual，无需同步
def _make_empty_intermediate_tensors(
    batch_size: int, dtype: torch.dtype, device: torch.device
) -> IntermediateTensors:
    tensors: dict[str, torch.Tensor] = {
        "hidden_states": torch.zeros(
            (batch_size, hidden_size), dtype=dtype, device=device
        ),
        # 之前存在 "residual" 项，已移除
    }
    if ple_dim and ple_dim > 0:
        tensors["per_layer_inputs"] = torch.zeros(
            (batch_size, num_layers, ple_dim), dtype=dtype, device=device
        )
    return IntermediateTensors(tensors)

# 改动 2: forward 中非首 rank 分支调整
# 之前无条件读取 intermediate_tensors["residual"] 和 intermediate_tensors["per_layer_inputs"]
# 现在 per_layer_inputs 仅在显式传递时（非 None）才从 intermediate_tensors 读取
else:
    assert intermediate_tensors is not None
    hidden_states = intermediate_tensors["hidden_states"]
    # per_layer_inputs 参数在非首 rank 默认为 None，因此不读取，符合 Gemma4 设计
    if per_layer_inputs is not None:
        per_layer_inputs = intermediate_tensors["per_layer_inputs"]
residual = None # residual 总是 None

# 改动 3: 返回 IntermediateTensors 时也不再包含 residual
# 只包含 hidden_states 和非 None 的 per_layer_inputs
if not get_pp_group().is_last_rank:
    tensors: dict[str, torch.Tensor] = {
        "hidden_states": hidden_states,
    }
    if per_layer_inputs is not None:
        tensors["per_layer_inputs"] = per_layer_inputs
    return IntermediateTensors(tensors)
```

评论区精华

- gemini-code-assist[bot] 评论：指出 forward 中 if per_layer_inputs is not None: 条件在非首 rank 始终为 False（因为参数默认 None），认为这会禁用 PLE 功能。但 PR 作者和

reviewer 未采纳此建议，理由是 per_layer_inputs 仅对较小模型需要。

- 作者提供 GSM8k 验证：PP=4 时准确率 0.954，证明改动不影响正确性。
- DarkLight1337 批准：要求运行 lm-eval 验证后确认无误。
    - per_layer_inputs 条件错误可能禁用 PLE (correctness): PR 作者和 reviewer 认为此行为符合 Gemma4 设计（per_layer_inputs 仅针对较小模型），并提供了 GSM8k 验证结果，未修改代码即合并。

风险与影响

- 风险：风险点：改动较局部，但 PP 路径是核心逻辑。
- 缺失测试覆盖：未增加 PP 特定测试，若未来对其他模型或 Gemma4 变种启用 PLE，则当前条件可能导致 per_layer_inputs 无法正确传递。
- review 中未解决的疑虑：gemini-code-assist[bot] 指出的条件问题虽未被采纳，但若后续有模型需要跨 rank 传递 per_layer_inputs，当前代码存在隐患。
- 兼容性风险：改动仅影响 Gemma4 模型，不影响其他模型。
- 影响：影响范围：仅限 Gemma4 模型使用 Pipeline Parallelism 的场景。修复后 PP 能正确运行，消除可能因 residual 同步导致的错误。用户无需任何配置更改。 影响程度：中等，修复了正确性问题，但改动量小，且经过正确性验证。
- 风险标记：缺少 PP 测试覆盖，per_layer_inputs 条件可能隐藏 bug

关联脉络

- 暂无明显关联 PR