PR #28665 完整报告

sgl-project/sglang

[Bug] fix(DummyModelLoader): run post_load_weights before process_weights_after_loading

合并时间：2026-06-20 06:57

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/28665

执行摘要

- 一句话：修复 DummyModelLoader 执行顺序导致 BailingMoE 启动崩溃
- 推荐动作：该 PR 变更清晰、修复明确，值得快速合并。虽然改动简单，但揭示了模型加载器中不同 post_load_weights 实现的隐含假设，建议为 DummyModelLoader 添加简单的维度检查测试，以防止未来类似回归。

功能与动机

BailingMoE 模型（Ring-2.5-1T）在 --load-format dummy 下启动崩溃，报错 RuntimeError: The size of tensor a (512) must match the size of tensor b (2048)，原因是 DummyModelLoader 在权重被转置后才调用 post_load_weights，导致通道量化时维度不匹配。

实现拆解

步骤一：分析根因（loader.py L1363-L1400）

DummyModelLoader.load_model 中原有执行顺序是：初始化模型 -> process_weights_after_loading（转置权重）-> initialize_dummy_weights -> _post_load_weights。而 _post_load_weights 内部（如 BailingMoE 的 post_load_weights）期望处理的是 未经转置的原始权重（shape = [2048, 512]），但此时权重已被转置为 [512, 2048]，导致通道量化时 channel_quant_to_tensor_quant 因第一维不匹配（512 vs 2048）而崩溃。

步骤二：调整执行顺序（loader.py L1385-L1400）

将 initialize_dummy_weights(model) 和 _post_load_weights(model) 移到 process_weights_after_loading 之前，使得 _post_load_weights 在权重转置之前执行，与 DefaultModelLoader 的行为一致。核心变更点：

- 先 initialize_dummy_weights 填充随机值，此时权重形状仍保持 [2048, 512]
- 再 _post_load_weights 处理权重（如 BailingMoE 的通道量化），确保维度匹配
- 最后 process_weights_after_loading 执行转置

步骤三：测试和验证

该修复仅涉及 dummy 加载路径，用于性能基准测试，不影响正常权重加载；未添加新单元测试，但 CI 验证已通过。

关键文件：

- python/sglang/srt/model_loader/loader.py（模块 模型加载；类别 source；类型 data-contract）：修复核心所在：调整 DummyModelLoader 中 initialize_dummy_weights、_post_load_weights 和 process_weights_after_loading 的执行顺序。

关键符号：未识别

关键源码片段

python/sglang/srt/model_loader/loader.py

修复核心所在：调整 DummyModelLoader 中 initialize_dummy_weights、_post_load_weights 和 process_weights_after_loading 的执行顺序。

```
# python/sglang/srt/model_loader/loader.py
# DummyModelLoader.load_model 方法中的执行顺序修复

def load_model(self, *, model_config, device_config):
    # ... 初始化模型 ...
    model = _initialize_model(model_config, self.load_config, quant_config)

    # NOTE(woosuk): For accurate performance evaluation, we assign
    # random values to the weights.
    initialize_dummy_weights(model) # 先填充随机值，权重形状仍为原始形状 [2048, 512]

    _post_load_weights(model) # 随后调用各层的 post_load_weights,
                              # 此时权重尚未转置，BailingMoE 的通道量化
                              # 能正确匹配 dim0（2048 == 2048）

    # 最后执行权重转置等后处理
    for _, module in model.named_modules():
        quant_method = getattr(module, "quant_method", None)
        if quant_method is not None:
            # 跳过已量化的层
            if hasattr(module, "is_weights_quantized") and module.is_weights_quantized():
                continue
            quant_method.process_weights_after_loading(module) # 转置发生在最后

    return model.eval()
```

评论区精华

Reviewer b8zhong 建议删除新增的注释 "Could we delete this comment? The fix is straightforward." 并提出简洁的代码替换建议（empty suggestion），表明变更逻辑清晰，无需过多解释。

- 删除冗余注释 (style): 作者接受了建议，最终提交中删除了注释。

风险与影响

- 风险：低风险：变更仅限 DummyModelLoader.load_model 中的 3 行代码重新排序，不改变任何其他逻辑。BailingMoE 之外的模型若依赖 _post_load_weights 的原始布局，同样受益。但由于缺少新增单元测试，若未来其他模型在 _post_load_weights 中假设了转置后的布局，则可能出现回归。不过这类假设本就不应存在，因此风险很低。
- 影响：影响范围小：仅影响使用 --load-format dummy 启动 BailingMoE 模型的路径，用于性能基准测试。正常权重加载（DefaultModelLoader）不受影响。不涉及用户请求处理、推理性能或内存占用。
- 风险标记：缺少测试覆盖，执行顺序敏感

关联脉络

- PR #28386 refactor(runner): add EagerRunner, own the eager path, polymorphic dispatch: 同属 loader 重构系列，关注模型加载路径的一致性问题。