PR #26791 完整报告

sgl-project/sglang

Fix Gemma4 NVFP4 MoE default attention backend

合并时间：2026-06-09 14:33

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/26791

执行摘要

- 一句话：修复 Gemma4 NVFP4 MoE 默认 attention 后端为 triton
- 推荐动作：建议精读此 PR，特别是条件化默认后端的设计模式。值得关注的决策：使用 getattr 安全检测配置属性，以及缓存 model_config 对象减少重复调用。建议后续添加一个简单的回归测试，验证 NVFP4 MoE 模型默认使用 triton。

功能与动机

用户 mmangkad 发现 nvidia/Gemma-4-26B-A4B-NVFP4 模型在默认 trtllm_mha attention 后端下 MMLU 准确率从 0.622 骤降至 0.037，输出无意义。该模型使用 NVFP4 量化和 MoE，此组合在 SM100+ 硬件上存在 trtllm_mha 精度 bug。

实现拆解

1. 缓存 model_config 对象（server_args.py:1784-1785）：将 self.get_model_config() 结果赋值给局部变量 model_config，避免后续多次调用，并为后续派生属性检测做准备。
2. 检测 NVFP4 MoE 条件（server_args.py:2335-2339）：通过 model_config.quantization == "modelopt_fp4" 和 getattr(model_config.hf_text_config, "enable_moe_block", False) 判断是否为 modelopt_fp4 量化且启用了 MoE 的 Gemma4 模型。
3. 条件化默认 attention 后端（server_args.py:2342-2345）：将原来的 "trtllm_mha" if is_sm100_supported() else "triton" 改为仅当 SM100 支持且不是 NVFP4 MoE 时才使用 trtllm_mha，否则回退到 triton。
4. 复用已检测的量化变量（server_args.py:2369-2370）：在后续 MoE runner 后端选择中，使用已缓存的 is_gemma4_modelopt_fp4 替代 self.get_model_config().quantization，减少重复调用。

关键文件：

- python/sglang/srt/server_args.py（模块 服务器参数；类别 source；类型 core-logic；符号 _handle_model_specific_adjustments）：唯一修改文件，新增 NVFP4 MoE 模型检测与默认 attention 后端回退逻辑

关键符号：_handle_model_specific_adjustments

关键源码片段

python/sglang/srt/server_args.py

唯一修改文件，新增 NVFP4 MoE 模型检测与默认 attention 后端回退逻辑

```
# python/sglang/srt/server_args.py (head version, simplified)
elif model_arch in ("Gemma4ForConditionalGeneration", "Gemma4ForCausalLM"):
    # 检测是否为 modelopt_fp4 量化且启用 MoE 的 Gemma4 模型
    is_gemma4_modelopt_fp4 = model_config.quantization == "modelopt_fp4"
    is_gemma4_moe = getattr(model_config.hf_text_config, "enable_moe_block", False)
    is_gemma4_modelopt_fp4_moe = is_gemma4_modelopt_fp4 and is_gemma4_moe

    # TODO: switch back to trtllm_mha after SM10X accuracy issue is fixed
    default_attention_backend = (
        "trtllm_mha"
        if is_sm100_supported() and not is_gemma4_modelopt_fp4_moe
        else "triton"
    )
    if self.is_attention_backend_not_set():
        self.attention_backend = default_attention_backend
        logger.info(f"Use {self.attention_backend} as default attention backend for Gemma4")
```

评论区精华

1. gemini-code-assist[bot]建议将 is_gemma4_moe 通过 bool(...) 进行显式转换，防止 enable_moe_block 为 None 时引发意外逻辑。该建议未被接受为强制修改。
2. pyc96（原 trtllm_mha 默认设置的作者）确认了精度问题，并同意回退到 triton 是合理的临时方案。
3. nvpohanh要求创建 issue 跟踪 trtllm_mha 精度 bug，pyc96 表示复用 #26518。
4. 讨论确认该问题在 SM103（GB300）上复现，不仅是 SM100。

- is_gemma4_moe 的布尔安全转换 (correctness): 未采纳显式 bool 转换，现有 getattr(..., False) 在语义上已可处理 None（None 为 falsy），但若值是非布尔对象可能仍有风险。
- 确认 trtllm_mha 精度问题及后续跟踪 (design): 同意回退到 triton 作为临时方案，等到 trtllm_mha 问题修复后再切回。

风险与影响

- 风险：
  1. 回归风险（低）：仅修改 NVFP4 MoE 模型的默认 attention 后端，对其他模型无影响。
  2. 性能影响（中）：triton 后端在 SM100+ 上的推理速度可能略低于 trtllm_mha，但精度更重要。
  3. 兼容性（无）：用户仍可通过 --attention-backend trtllm_mha 显式覆盖。
  4. 缺少测试覆盖（中）：没有对应的回归测试，手动 benchmark 是唯一验证。
- 影响：
  1. 用户影响：修复了 NVFP4 MoE Gemma4 模型的精度崩溃问题。
  2. 系统影响：仅影响配置分支，无运行时逻辑变更。

3. 团队影响：暴露了 SM10X trtllm_mha 后端的深层精度问题，需要后续追查。 - 风险标记：缺少测试覆盖，临时性修复

关联脉络

- PR #25054 Set default attention backend for SM100 to TRTLLM: 引入了 Gemma4 默认使用 trtllm_mha 的变更， PR 作者 mmangkad 在 issue 评论中直接引用了该 PR。
- PR #26518 Track trtllm_mha accuracy issue on SM100: 被指定为跟踪 trtllm_mha 精度问题的 issue，本 PR 的讨论中确认复用该 issue。