PR #27919 完整报告

sgl-project/sglang

Revert "[AMD] Fix DeepSeek V4 Pro c128 state tensor dtype mismatch error and c4_sparse_raw_indices attribute error in cuda graph phase"

合并时间：2026-06-12 05:25

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/27919

执行摘要

- 一句话：撤销 #27529 导致的 DeepSeek V4 性能回归
- 推荐动作：建议仔细阅读 #27380 的变更，确认其确实解决了原生 dtype 不匹配问题。同时建议在 AMD 硬件上运行 DeepSeek V4 的回归测试，验证性能回归已被修复且原始错误不重现。该 PR 本身是安全的回滚，但应关注后续可能仍需修复 #27529 提出的 dtype 统一问题。

功能与动机

PR #27529 修复了 AMD 上 DeepSeek V4 的 dtype 不匹配和属性错误，但引入了 kernel 性能回归（见 #27529 review 评论）。同时，CUDA graph 启动错误已在 #27380 中得到修复，且 #27529 的逻辑与 #27380 冲突会导致崩溃。因此需要回滚 #27529。

实现拆解

该 PR 回滚了 #27529 的所有变更，涉及四个文件：

1. python/sglang/jit_kernel/dsv4/compress.py：恢复 _jit_compress_module 函数签名，移除 dtype_buf 参数；compress_forward 不再传递 kv_score_buffer.dtype。
2. python/sglang/srt/layers/attention/dsv4/compressor.py：移除 apply_ape_hotfix 中将 ape 和 norm.weight 转换为 bf16 的代码（if _use_aiter 分支）。
3. python/sglang/jit_kernel/csrc/deepseek_v4/c4_v2.cuh 和 c128_v2.cuh：移除 BufFloat 模板参数，统一使用 InFloat 类型加载和存储，将 float 转换推迟到计算阶段。

关键文件：

- python/sglang/jit_kernel/dsv4/compress.py（模块 JIT 编译；类别 source；类型 core-logic；符号 _jit_compress_module, compress_forward）：核心 JIT 模块构建，修改了 _jit_compress_module 签名并移除了 dtype_buf 参数，compress_forward 简化调用。
- python/sglang/srt/layers/attention/dsv4/compressor.py（模块 压缩器；类别 source；类型 core-logic；符号 apply_ape_hotfix）：移除了 apply_ape_hotfix 中的 bf16 转换逻辑，该逻辑曾是 AMD 修复的一部分。
- python/sglang/jit_kernel/csrc/deepseek_v4/c4_v2.cuh（模块 CUDA 内核；类别 other；类型 core-logic；符号 c4_forward）：移除了 BufFloat 模板参数，统一使用 InFloat 加载，简化内核逻辑。

- python/sglang/jit_kernel/csrc/deepseek_v4/c128_v2.cuh（模块 CUDA 内核；类别 other；类型 core-logic；符号 c128_forward）：与 c4_v2.cuh 类似，移除 BufFloat 模板参数，简化内核。

关键符号：_jit_compress_module, compress_forward, apply_ape_hotfix, c4_forward, c128_forward

关键源码片段

python/sglang/jit_kernel/dsv4/compress.py

核心 JIT 模块构建，修改了 _jit_compress_module 签名并移除了 dtype_buf 参数，compress_forward 简化调用。

```
# python/sglang/jit_kernel/dsv4/compress.py (revert 后 )
@cache_once
def _jit_compress_module(
    head_dim: int,
    dtype_in: torch.dtype, # 输入 dtype
    dtype_out: torch.dtype, # 输出 dtype
    ratio: Literal[4, 128],
) -> Module:
    # 只使用 dtype_in/dtype_out，不再需要 dtype_buf
    args = make_cpp_args(head_dim, dtype_in, dtype_out, is_arch_support_pdl())
    kernel_class = f"FlashCompress{ratio}Kernel<{args}>"
    return load_jit(
        make_name(f"compress_{ratio}_v2"),
        *args,
        cuda_files=[f"deepseek_v4/c{ratio}_v2.cuh"],
        cuda_wrappers=[
            ("decode", f"{kernel_class}::run_decode"),
            ("prefill", f"{kernel_class}::run_prefill"),
        ],
        extra_cuda_cflags=["-use_fast_math"],
    )


def compress_forward(
    kv_score_buffer: torch.Tensor,
    kv_score_input: torch.Tensor,
    ape: torch.Tensor,
    plan: Union[CompressorDecodePlan, CompressorPrefillPlan],
    *,
    head_dim: int,
    compress_ratio: Literal[4, 128],
    out: Optional[torch.Tensor] = None,
    is_online: bool = False,
) -> torch.Tensor:
    # ... 省略前面部分 ...
    else:
```

```
        # 只从 input/out 获取 dtype，忽略 kv_score_buffer 的 dtype
        dtype_in, dtype_out = kv_score_input.dtype, out.dtype
        module = _jit_compress_module(head_dim, dtype_in, dtype_out, compress_ratio)
    fn = module.decode if plan.is_decode else module.prefill
    fn(kv_score_buffer, kv_score_input, out, ape, *plan[1:3])
    return out
```

python/sglang/srt/layers/attention/dsv4/compressor.py

移除了 apply_ape_hotfix 中的 bf16 转换逻辑，该逻辑曾是 AMD 修复的一部分。

```
# python/sglang/srt/layers/attention/dsv4/compressor.py (revert 后 )
def apply_ape_hotfix(self):
    assert not self.ape_converted
    self.ape_converted = True

    if self.overlap:
        ape = torch.chunk(self.ape.data, 2, dim=-1)
        ape = torch.cat([ape[0], ape[1]], dim=0)
        self.ape.data.copy_(ape.view(self.ratio, -1))
    # 注意：移除了 if _use_aiter 分支下的 dtype 转换
```

评论区精华

PR body 说明了回滚的两个原因：性能回归（引用 #27529 的 review 评论）和逻辑冲突（与 #27380）。审核者 kkHuang-amd approve 了该 PR。Gemini Code Assist 自动生成了代码总结，指出简化了模板参数和运行时转换。未发现其他讨论。

- 性能回归与冲突原因 (performance): 回滚 #27529，等待 #27380 解决原始问题。

风险与影响

- 风险：回滚可能重新暴露 AMD 上 DeepSeek V4 的 dtype 不匹配和 c4_sparse_raw_indices 属性错误，但 PR 声称这些已被 #27380 解决。需要确认 #27380 在 AMD 上的覆盖范围。此外，回滚后代码更简洁，之前引入的性能回归应被修复，但需回归测试验证。
- 影响：影响范围：主要影响使用 AMD GPU 运行 DeepSeek V4 模型的用户。回滚后，这些用户可能不再遇到回归导致的性能下降，但需确认原始 bug 是否已由 #27380 彻底修复。对 CUDA 用户无影响，因为相关逻辑仅针对 AMD 后端。
- 风险标记：性能回归修复，回滚可能重提已修复 bug，核心路径变更，AMD 特定

关联脉络

- PR #27529 [AMD] Fix DeepSeek V4 Pro c128 state tensor dtype mismatch error and c4_sparse_raw_indices attribute error in cuda graph phase: 被本 PR 回滚的原始修改。
- PR #27380 Unspecified (related to fixing cuda graph launch error): PR body 提到 #27380 解决了 cuda graph 启动错误，且与本 PR 逻辑冲突。

- PR #27525 Unspecified (mentioned in issue body of #27529): 在 #27529 的 issue 中被提及为协作 PR，可能相关。