PR #26238 完整报告

sgl-project/sglang

refactor(dsv4): route MHC prenorm through DeepGEMM wrapper

合并时间：2026-05-28 08:45

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/26238

执行摘要

- 一句话：将 MHC prenorm 内核接入通用 DeepGEMM wrapper/warmup 路径
- 推荐动作：对于 DeepSeek V4 和 DeepGEMM wrapper 的维护者值得精读，了解如何将新内核类型接入统一预热框架。建议后续追踪吞吐下降原因，并考虑是否调整为与主基线一致的预热策略。

功能与动机

PR body: 'Route the DeepSeek V4 MHC prenorm DeepGEMM kernel through the common DeepGEMM wrapper entrypoint so it can share the same wrapper/warmup infrastructure as other DeepGEMM kernels.' 以及 Fridge003 在评论中要求 revert #25810 中不需要的预热变更。

实现拆解

1. 添加内核类型枚举与工厂映射：在 compile_utils.py 的 DeepGemmKernelType 枚举中增加 TF32_HC_PRENORM_GEMM 值，并在 _BaseWarmupExecutor.create 的字典中添加到 _TF32HcPrenormWarmupExecutor 的映射。
2. 实现预热执行器：新增 _TF32HcPrenormWarmupExecutor 类，__init__ 中预分配 bf16 输入张量 x 和 fp32 权重 fn，并根据 num_splits 决定是否分配分块输出；execute 方法调用 deep_gemm.tf32_hc_prenorm_gemm 并传递正确维度的张量。同时更新 get_memory_requirement 以正确估计该内核的内存需求。
3. 新增包装函数：在 entrypoint.py 中添加 tf32_hc_prenorm_gemm 函数，通过 deep_gemm_execution_hook 包装对底层 deep_gemm.tf32_hc_prenorm_gemm 的调用，并处理 m=0 的空输入情况。
4. 路由调用点：在 deepseek_v4.py 的 hc_pre 方法和 mhc.py 的 mhc_pre 函数中，将原来的 import deep_gemm; deep_gemm.tf32_hc_prenorm_gemm(...) 替换为从 entrypoint 导入 tf32_hc_prenorm_gemm 并调用，同时修正参数命名（从 num_splits= 改为 n_splits）。
5. 清理模型级预热代码：删除 deepseek_v4.py 中的 prewarm_mhc_token_counts 和 prewarm_mhc_token_count_buckets 方法，以及 deepseek_v4_nextn.py 中的对应委托方法；在 model_runner.py 的 kernel_warmup 中移除对模型 kernel_warmup 钩子的调用，因为预热现在统一由 DeepGEMM wrapper 管理。

关键文件：

- python/sglang/srt/models/deepseek_v4.py（模块 主模型；类别 source；类型 data-contract；符号 prewarm_mhc_token_counts, prewarm_mhc_token_count_buckets）：核心模型文件，移除了模型级预热方法（prewarm_mhc_token_counts/prewarm_mhc_token_count_buckets），并在 hc_pre 方法中将直接调用 deep_gemm 改为调用 wrapper 入口。
- python/sglang/srt/layers/deep_gemm_wrapper/compile_utils.py（模块 预热工具；类别 source；类型 core-logic；符号 _TF32HcPrenormWarmupExecutor, init, execute, DeepGemmKernelType.TF32_HC_PRENORM_GEMM）：新增 TF32_HC_PRENORM_GEMM 内核类型和专门的 _TF32HcPrenormWarmupExecutor，扩展了 DeepGEMM 的预热基础设施。
- python/sglang/srt/layers/mhc.py（模块 MHC 内核；类别 source；类型 core-logic；符号 get_mhc_pre_token_count_representatives）：MHC 内核实现，修改了 deep_gemm 调用点为 wrapper 包装函数，并移除 get_mhc_pre_token_count_representatives 函数。
- python/sglang/srt/layers/deep_gemm_wrapper/entrypoint.py（模块 入口包装；类别 source；类型 core-logic；符号 tf32_hc_prenorm_gemm）：DeepGEMM wrapper 的入口点，新增 tf32_hc_prenorm_gemm 包装函数，将新内核类型集成到统一的编译 / 预热钩子中。
- python/sglang/srt/model_executor/model_runner.py（模块 运行器；类别 source；类型 data-contract；符号 kernel_warmup）：修改 kernel_warmup 方法，移除对模型特定 warmup 钩子的调用，简化预热流程。
- python/sglang/srt/models/deepseek_v4_nextn.py（模块 推测模型；类别 source；类型 data-contract；符号 prewarm_mhc_token_count_buckets）：推测模型（NextN）中的委托预热方法被移除，因为不再需要。

关键符号：tf32_hc_prenorm_gemm, _TF32HcPrenormWarmupExecutor.init, _TF32HcPrenormWarmupExecutor.execute, hc_pre, mhc_pre, kernel_warmup

关键源码片段

python/sglang/srt/layers/deep_gemm_wrapper/compile_utils.py

新增 TF32_HC_PRENORM_GEMM 内核类型和专门的 _TF32HcPrenormWarmupExecutor，扩展了 DeepGEMM 的预热基础设施。

```
# compile_utils.py: 新内核类型注册与预热执行器
class DeepGemmKernelType(IntEnum):
    TF32_HC_PRENORM_GEMM = auto() # 新增：MHC prenorm GEMM 内核

class _TF32HcPrenormWarmupExecutor(_BaseWarmupExecutor):
    def __init__(self, max_m, n, k, num_groups):
        self.x = torch.empty((max_m, k), device='cuda', dtype=torch.bfloat16)
        self.fn = torch.empty((n, k), device='cuda', dtype=torch.float32)
        self.n = n
        self.num_splits = num_groups if num_groups > 0 else None
```

```
def execute(self, m):
    if self.num_splits is None:
        out = torch.empty((m, self.n), device='cuda', dtype=torch.float32)
        sqrsum = torch.empty((m,), device='cuda', dtype=torch.float32)
    else:
        out = torch.empty((self.num_splits, m, self.n), device='cuda', dtype=torch.float32)
        sqrsum = torch.empty((self.num_splits, m), device='cuda', dtype=torch.float32)
    deep_gemm.tf32_hc_prenorm_gemm(self.x[:m], self.fn, out, sqrsum, num_splits=self.num_
    splits)
```

评论区精华

Fridge003 在 Issue 评论中要求回退 #25810 的模型级预热变更：'Please revert the changes in #25810, since they are not needed'。作者确认已 revert，并重新运行 GSM8K 测试，结果在 PR body 中展示。无其他审查评论。

- 回退 #25810 的预热变更 (design): 作者已回退，并将 PR 聚焦在路由内核到通用 wrapper。

风险与影响

- 风险：
  1. 精度与性能退化：GSM8K 分数下降 0.5%，吞吐下降约 11%，虽然可能是由于预热路径变化，但需要确认是否影响生产部署。
  2. 首次推理延迟风险：移除了模型级明确预热后，若 wrapper 的 warmup 未能覆盖所有 token count 范围，可能在首次遇到新分桶时触发即时编译，增加延迟。
  3. 代码回归：重构涉及多层调用关系，错误的路由可能导致 num_splits 参数传递错误（例如命名不匹配），但从 patch 看已修正。 - 影响：影响范围限于 DeepSeek V4 模型及相关组件。用户无感知，API 未变化。内部架构统一有利于维护和扩展。性能略有下降需关注，但功能正确性保持。 - 风险标记：精度下降待确认，移除模型级预热风险，首次运行延迟可能增加

关联脉络

- PR #25810 Pre-warm MHC kernel for DeepSeek V4: 此 PR 是 #25810 的 follow-up，并回退了其中模型级预热变更，将内核调用统一接入 DeepGEMM wrapper。