# PR #50801 完整报告

vllm-project/vllm

[CPU] Refine CPU kernel dispatch

合并时间：2026-08-03 17:42

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/50801

## 执行摘要

- 一句话：CPU 内核调度重构，移除 SGL 实验开关并修复选择条件
- 推荐动作：值得精读，尤其适合 CPU 后端维护者和关注 kernel oracle 机制的人。可学习的点：① 用 vendored 内核源码逐条验证选择条件，而不是靠文档假设；② 把「不支持」的判断前置到 is_supported_config，避免加载期 / 运行期才崩溃；③ 实验性开关在 benchmark 无副作用且范围收窄后做默认化的完整决策流程。注意其遗留的形状门控问题可作为后续 PR 的切入点。

## 功能与动机

作者在 PR body 中给出的核心动机是：benchmark 表明 SGL 派生 AMX 内核在其既有适格窗口（对称 INT8 W8A8、形状对齐、AMX x86）内没有性能副作用，没必要继续藏在实验开关后；同时审计发现 check_cpu_sgl_kernel 的 dtype 门控漏掉 torch.float16，而 vendored 内核全部通过 AT_DISPATCH_REDUCED_FLOATING_TYPES 分发，同时覆盖 BFloat16 与 Half——导致 FP16 权重被路由到更慢的 oneDNN 路径。另外，CPUExpertsFp8/CPUExpertsMxfp4/CPUExpertsInt4 的 _supports_current_device() 只查 is_cpu()，CPUExpertsInt8 只查 x86 不查 AMX，而它们调用的 fused_experts_cpu 与 torch.ops._C.convert_weight_packed 只编译进 AMX 档位扩展，非 AMX x86 或 ARM 上选择这些类会直接 AttributeError 崩溃，而不是由 oracle 在加载前干净拒绝。

## 实现拆解

1. 删除实验开关（vllm/envs.py、docs/getting_started/installation/cpu.md）：从 EnvVars 类和解析字典移除 VLLM_CPU_SGL_KERNEL 声明与 lambda，文档同步删除对应条目。由于 #50133 已先删掉未量化 MoE 的 SGL 分支，flag 唯一剩余使用点是 INT8 W8A8 线性路径，而该路径在本 PR 中已改为无条件，因此可以整体移除。
2. 线性层与未量化 GEMM 调度调整（vllm/model_executor/kernels/linear/scaled_mm/cpu.py、vllm/model_executor/layers/utils.py）：CPUInt8ScaledMMLinearKernel.process_weights_after_loading 删除 envs.VLLM_CPU_SGL_KERNEL 条件，适格时直接 _apply_weights_sgl；dispatch_cpu_unquantized_gemm 删除 SGL 分支（未量化 GEMM 继续仅用 oneDNN），并顺带清理不再需要的 N, K, dtype 提前取值；check_cpu_sgl_kernel 的 dtype 集合加入 torch.float16。
3. 量化 MoE 专家类选择条件修复（vllm/model_executor/layers/fused_moe/experts/cpu_moe.py）：四个量化专家类的 _supports_current_device() 统一为 is_cpu() + x86 + AMX

三重检查；CPUExpertsInt8 新增 is_supported_config 重写，提前拒绝 hidden_dim 或 intermediate_size_per_partition 不是 32 倍数的配置。

4. 依赖升级（cmake/cpu_extension.cmake）：x86 oneDNN FetchContent 从 v3.10 升到 v3.13，ARM 保持独立 pin。
5. 测试与基准（未新增测试文件）：作者在 AMX 主机跑通 8 个 CPU 相关测试文件（651 过 / 289 跳 / 0 失败），并用 vllm bench serve 分别对比 oneDNN 版本与 SGL 开关移除前后的 serving 指标。本 PR 没有为新增的 is_supported_config 添加正式单测，仅靠手工单元检查验证拒绝路径，是本次变更的薄弱点。

关键文件：

- vllm/model_executor/layers/fused_moe/experts/cpu_moe.py（模块 专家内核；类别 source；类型 data-contract；符号 _supports_current_device, is_supported_config）：核心修复所在：四个量化 MoE 专家类补 x86/AMX 设备检查，CPUExpertsInt8 新增 oracle 阶段形状对齐门控，防止非 AMX 硬件运行时 AttributeError 崩溃。
- vllm/model_executor/layers/utils.py（模块 内核调度；类别 source；类型 data-contract；符号 check_cpu_sgl_kernel, dispatch_cpu_unquantized_gemm）：删除未量化 GEMM dispatch 中的 SGL 分支（未量化 GEMM 本就只用 oneDNN），并在 check_cpu_sgl_kernel 的 dtype 门控中加入 torch.float16。
- vllm/model_executor/kernels/linear/scaled_mm/cpu.py（模块 量化线性；类别 source；类型 data-contract；符号 CPUInt8ScaledMMLinearKernel.process_weights_after_loading）：INT8 W8A8 线性层 process_weights_after_loading 移除 flag 条件，使 SGL 路径在适格时无条件启用；同时移除 envs 导入。
- vllm/envs.py（模块 环境配置；类别 source；类型 core-logic）：移除 VLLM_CPU_SGL_KERNEL 的声明与解析，完成该实验开关的终结。
- cmake/cpu_extension.cmake（模块 构建配置；类别 other；类型 core-logic）：x86 oneDNN FetchContent pin 从 v3.10 升到 v3.13，ARM 保持独立 pin 不受影响。
- docs/getting_started/installation/cpu.md（模块 文档说明；类别 docs；类型 documentation）：删除 VLLM_CPU_SGL_KERNEL 环境变量文档条目。

关键符号：CPUExpertsInt8.is_supported_config, CPUExpertsInt8._supports_current_device, CPUExpertsFp8._supports_current_device, CPUExpertsMxfp4._supports_current_device, CPUExpertsInt4._supports_current_device, check_cpu_sgl_kernel, dispatch_cpu_unquantized_gemm, CPUInt8ScaledMMLinearKernel.process_weights_after_loading

## 关键源码片段

### vllm/model_executor/layers/fused_moe/experts/cpu_moe.py

核心修复所在：四个量化 MoE 专家类补 x86/AMX 设备检查，CPUExpertsInt8 新增 oracle 阶段形状对齐门控，防止非 AMX 硬件运行时 AttributeError 崩溃。

```
@staticmethod
def _supports_current_device() -> bool:
    # SGL 派生的 fused_experts_cpu / convert_weight_packed 只编译进
```

```
        # AMX 档位的 _C 扩展，非 AMX x86 与 ARM 上直接调用会 AttributeError，
        # 因此必须在 oracle 选择阶段就拒绝。
        return (
            current_platform.is_cpu()
            and current_platform.get_cpu_architecture() == CpuArchEnum.X86
            and torch.cpu._is_amx_tile_supported()
        )

    @staticmethod
    def is_supported_config(
        cls: type[mk.FusedMoEExperts],
        moe_config: FusedMoEConfig,
        weight_key: QuantKey | None,
        activation_key: QuantKey | None,
        activation_format: mk.FusedMoEActivationFormat,
    ) -> tuple[bool, str | None]:
        supported, reason = mk.FusedMoEExperts.is_supported_config(
            cls, moe_config, weight_key, activation_key, activation_format
        )
        if not supported:
            return supported, reason
        # convert_weight_packed（四个量化专家共用的 VNNI 预打包）要求
        # OC % TILE_N(16) == 0 且 IC % TILE_K(32) == 0；moe_int8.cpp 的
        # w1 门控核还要求中间维本身（非 2x）是 32 的倍数，综合起来
        # hidden_dim 与 intermediate_size_per_partition 都必须是 32 的倍数。
        if moe_config.hidden_dim % 32 != 0:
            return False, "kernel requires hidden dim divisible by 32"
        if moe_config.intermediate_size_per_partition % 32 != 0:
            return False, "kernel requires intermediate dim divisible by 32"
        return True, None
```

### vllm/model_executor/layers/utils.py

删除未量化 GEMM dispatch 中的 SGL 分支（未量化 GEMM 本就只用 oneDNN），并在 check_cpu_sgl_kernel 的 dtype 门控中加入 torch.float16。

```
def check_cpu_sgl_kernel(n: int, k: int, dtype: torch.dtype) -> bool:
    # 底层 SGL 内核（weight_packed_linear、int8_scaled_mm_with_quant、
    # fused_experts_cpu）统一走 AT_DISPATCH_REDUCED_FLOATING_TYPES，
    # 同时覆盖 BFloat16 与 Half，因此 FP16 也具备适格性，
    # 此前漏掉 torch.float16 会把 FP16 模型错误地路由到 oneDNN。
    return (
        torch.cpu._is_amx_tile_supported()
        and (dtype in (torch.bfloat16, torch.float16, torch.int8))
        and k % 32 == 0
        and n % 16 == 0
    )
```

## 评论区精华

该 PR 来自 fork 分支，claude[bot] 自动 review 被禁用；维护者 jikunshang 直接 APPROVED，未留下技术评论。实际的技术讨论沉淀在 PR body 中：作者主动标注了遗留问题——CPUExpertsFp8/CPUExpertsMxfp4/CPUExpertsInt4 仍缺形状对齐门控，原因是 convert_weight_packed 对 mxfp4/int4 字节打包的 IC 计账会翻倍，且 moe_fp8.cpp/moe_int4.cpp 没有运行时 TORCH_CHECK 可交叉验证，贸然加模数判断可能误拒合法配置，因此留作后续而非猜测。

- 暂无高价值评论线程

## 风险与影响

- 风险：
  1. 配置静默失效（breaking change）：设置过 VLLM_CPU_SGL_KERNEL=0 的用户升级后开关被忽略，适格条件下会自动启用 SGL 路径，可能带来精度 / 性能的意外变化。
  2. FP16 行为变化：check_cpu_sgl_kernel 新增 FP16 适格性后，FP16 模型在对称 + 对齐 + AMX 下会从 oneDNN 切到 SGL 内核；作者未提供 FP16 的专项基准对比。
  3. oneDNN v3.13 数值差异：body 报告 gemma-7b 有一个 prompt 输出与 v3.10 发散（BF16 舍入顺序敏感）；对依赖字节级稳定输出的用户需要关注。
  4. 遗留形状门控：FP8/MXFP4/INT4 专家类仍可能误选 / 误拒非常规形状配置，未来模型若出现非 32 倍数的中间维度可能重新触发崩溃。
  5. 依赖 vendored 内核细节：选择条件与 csrc/cpu/sgl-kernels/ 内部约束（TILE_N=16、TILE_K=32、moe_int8 的 32 倍数）耦合，上游 sglang 更新或 vendored 变更时需同步审计。 - 影响：对用户：AMX x86 上的 INT8 W8A8 在线服务自动获得 SGL 内核提速（body 基准：mean TTFT 4136.7→3721.5 ms，total tok/s 359.8→385.1）；非 AMX/ARM 用户从潜在运行时崩溃变为 oracle 干净拒绝；BF16 模型在 oneDNN v3.13 下 TPOT/ 吞吐提升约 2-3%（6-prompt 样本）。对系统：删除一个实验开关，减少 envs 与文档维护成本；CPU MoE 内核选择逻辑更接近自解释。对团队：与 #50133 形成 CPU MoE 调度从「全局 flag + 特判」到「oracle 按设备 / 形状 / 量化自述能力」的连续演进，后续可继续补齐 FP8/MXFP4/INT4 的形状门控。 - 风险标记：实验开关静默移除，oneDNN 升级数值差异，FP16 新增 SGL 路径，FP8/MXFP4/INT4 形状门控遗留，依赖 vendored 内核约束

## 关联脉络

- PR #50133 [CPU] Migrate unquantized MoE to the modular-kernel experts structure: 该 PR 提前删除了未量化 MoE 的 SGL dispatch 分支与 flag 检查，使 VLLM_CPU_SGL_KERNEL 仅剩线性层一个使用点；本 PR 在此基础上 rebase 并完成 flag 的最终移除，两者共同推进 CPU MoE 调度的模块化。
- PR #50547 cpu_model_runner.py: skip the warm up if CompilationMode.NONE: 同为 CPU 后端近期改动，聚焦启动 / 执行路径的清理与提速，与本 PR 同属 CPU 后端演进脉络。