

PR #49904 完整报告

vllm-project/vllm

[Build] Fix CUDA arch detection producing kernel-less builds on SM121

合并时间: 2026-07-28 13:01

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/49904>

执行摘要

- 一句话: 修复 SM121 下无内核的构建错误
- 推荐动作: 值得精读, 尤其是 CMake 构建系统的开发者。该 PR 展示了如何处理上游工具链的 bug、如何通过优先读取更可靠的元数据 (code=sm_* 而非 arch=compute_*) 来规避问题, 以及在构建时添加防御性检查避免静默失败。

功能与动机

在 GB10 / DGX Spark (计算能力 12.1, CUDA 13) 上, 按照文档中的默认源码构建路径 (`uv pip install -e .` 且 `TORCH_CUDA_ARCH_LIST` 未设置) 时, 构建看似正常但内核 `.so` 仅包含 `sm_75` cubins。使用任何 FP8 模型 (如 `RedHatAI/gemma-4-31B-it-FP8-block`) 提供服务时均会在 `profile_run` 中崩溃, 报错 `RuntimeError: cutlass_scaled_mm, ./scaled_mm_entry.cu:265, NotImplementedError: No compiled cutlass_scaled_mm for a compute capability less than CUDA device capability: 121`。该问题的根本原因是 PyTorch 旧版 `select_compute_arch.cmake` 中的 `string(REPLACE "2.1" "2.1(2.0)")` 错误地将 SM121 的版本号 12.1 中的 2.1 替换为 2.1(2.0), 导致 `-gencode arch=compute_20,code=sm_121`, 而 vLLM 仅读取 `arch=compute_*` 部分得到 2.0, 不匹配任何 `CUDA_SUPPORTED_ARCHS`, 所有架构相关的内核均被跳过。

实现拆解

1. 核心逻辑修复 (`cmake/utils.cmake`): 修改 `extract_unique_cuda_archs_ascending()` 函数, 优先从 `-gencode` 标志的 `code=sm_*` (实际 SASS 目标) 部分提取架构版本, 仅在 PTX-only 标志 (`code=compute_*`) 时回退到 `arch=compute_*`。这样即使 `arch=compute_*` 被错误改写, 也能正确提取真实的 SASS 目标 (如 12.1)。
2. 构建时错误拦截 (`CMakeLists.txt`): 在 `extract_unique_cuda_archs_ascending()` 和 `cuda_archs_loose_intersection()` 处理后, 如果 `CUDA_ARCHS` 为空, 则直接 `message(FATAL_ERROR ...)` 停止构建, 并提示用户设置 `TORCH_CUDA_ARCH_LIST`。避免构建出无任何可用内核的二进制文件。
3. 测试补充 (`tests/test_cmake_utils.py`): 新增 `test_extract_archs_prefers_sass_target_over_corrupted_virtual_arch` 测试函数, 模拟 Torch 发出的带有被错误改写的 `arch=compute_20,code=sm_121` 的 `-gencode` 标志, 验证 `extract_unique_cuda_archs_ascending()` 正确返回 `8.0;12.1` (即优先使用 `code=sm_121` 对应的真实架构, 同时正确处理正常的 PTX-only 标志)

arch=compute_80,code=compute_80) 。

关键文件：

- cmake/utils.cmake (模块 构建脚本；类别 other；类型 core-logic；符号 extract_unique_cuda_archs_ascending)：核心修复所在：修改 extract_unique_cuda_archs_ascending() 函数，优先从 code=sm_* 提取架构，回退到 arch=compute_*。
- CMakeLists.txt (模块 构建脚本；类别 other；类型 configuration)：增加构建时错误拦截：如果最终架构列表为空，则停止构建并给出明确错误信息。
- tests/test_cmake_utils.py (模块 构建脚本；类别 test；类型 test-coverage；符号 test_extract_archs_prefers_sass_target_over_corrupted_virtual_arch)：新增测试验证修复的正确性：模拟被错误改写的 gencode 标志，确保提取函数返回正确架构。

关键符号：extract_unique_cuda_archs_ascending,

test_extract_archs_prefers_sass_target_over_corrupted_virtual_arch

关键源码片段

cmake/utils.cmake

核心修复所在：修改 `extract_unique_cuda_archs_ascending()` 函数，优先从 `code=sm_*` 提取架构，回退到 `arch=compute_*`。

```
# cmake/utils.cmake
# 修改 extract_unique_cuda_archs_ascending 函数，优先从 code=sm_* 提取架构
function(extract_unique_cuda_archs_ascending OUT_ARCHES CUDA_ARCH_FLAGS)
    set(_CUDA_ARCHES)
    foreach(_ARCH ${CUDA_ARCH_FLAGS})
        # 优先匹配 code=sm_* (实际 SASS 目标)，例如 code=sm_121
        string(REGEX MATCH "code=sm_([0-9]+[af]*)" _COMPUTE ${_ARCH})
        if (NOT _COMPUTE)
            # 如果没有 code=sm_*, 则回退到 arch=compute_* (PTX-only 标志)
            string(REGEX MATCH "arch=compute_([0-9]+[af]*)" _COMPUTE ${_ARCH})
        endif()
        if (_COMPUTE)
            set(_COMPUTE ${CMAKE_MATCH_1})
        endif()
        # ... 后续去重、排序逻辑不变
    endforeach()
    # ...
endfunction()
```

CMakeLists.txt

增加构建时错误拦截：如果最终架构列表为空，则停止构建并给出明确错误信息。

```
# CMakeLists.txt ( 相关片段 )
# 在 extract 和 intersection 之后，检查架构列表是否为空
cuda_archs_loose_intersection(CUDA_ARCHS "${CUDA_SUPPORTED_ARCHS}" "${CUDA_ARCHS}"
")
```

```
message(STATUS "CUDA supported target architectures: ${CUDA_ARCHS}")

if(NOT CUDA_ARCHS)
  message(FATAL_ERROR
    "No supported CUDA architectures; the build would produce a binary "
    "with no usable kernels. Detected gencode flags: ${CUDA_ARCH_FLAGS}; "
    "supported: ${CUDA_SUPPORTED_ARCHS}. "
    "Set TORCH_CUDA_ARCH_LIST for your GPU (e.g. 12.0).")
endif()
```

评论区精华

审核人 Harry-Chen 快速批准了该 PR，并评论 "Good catch! This reminds me of the story of why Windows 9 was not released"，强调这是一个类似因版本号子串匹配错误而引发的经典问题。此外，审核过程中要求完成 DCO (Signed-off-by)，作者已按要求更新了提交信息。

- DCO 签署要求 (other): 作者在提交信息中添加了 Signed-off-by，满足要求。
- 类似 Windows 9 的版本号匹配问题 (other): 无进一步讨论，已批准合并。

风险与影响

- 风险：变更仅涉及 CMake 构建系统和测试文件，不修改任何运行时逻辑，回归风险极低。
主要风险是：如果未来有其他架构的 `arch=compute_*` 和 `code=sm_*` 数字部分不一致，新的提取逻辑可能仍会错误处理。但考虑到目前 `code=sm_*` 与 `arch=compute_*` 通常是同步的（PTX-only 标志除外），且测试已覆盖了正常和异常两种场景，风险可控。
- 影响：直接影响所有使用 Blackwell SM121 GPU（如 DGX Spark、DGX B200）进行源码构建的用户，使其能够成功构建并运行 FP8 模型。对已有正常工作流（非 SM121）的用户无影响。构建系统改进了健壮性——在架构不匹配时提前报错而非静默生成有缺陷的二进制文件。
- 风险标记：构建系统变更，影响 SM121 GPU，测试覆盖新增

关联脉络

- PR #47149 Kernel feature macro scoping and arch warning: 作者明确指出该 PR 非 #47149 的重复——#47149 不处理架构自动检测，也不在构建时捕获空架构列表情况。
- PR #43003 Add 12.1 to supported arch list: 作者指出 #43003（已关闭）曾添加 12.1 到支持列表，但 review 显示 12.0f 家族后备已可处理 SM121，本 PR 修复的是该逻辑运行前的架构提取错误。
- PR #38484 Change published wheel arch targets: 作者明确本 PR 非 #38484 的重复——#38484 针对的是发布的 wheel 目标架构，而非源码构建。