

PR #3256 完整报告

radixark/miles

e2e: run the DSv4 megatron impl CI on the cudnn DSA backend

合并时间: 2026-09-15 11:28

原文链接: <http://prhub.com.cn/radixark/miles/pull/3256>

执行摘要

- 一句话: 镜像补 FlashMLA/cudnn-frontend, DSv4 e2e 切到 cuDNN DSA 后端
- 推荐动作: 建议精读, 但重点不在测试文件的一行改动, 而在 docker/Dockerfile 中新增的安装块: `--no-deps` 与 `ENABLE_CUDA_13` 门控体现了「镜像依赖决定 CI 能力」的现实约束, 是理解本仓库 GPU CI 体系的一个好样本。阅读时重点关注三点: 一是 `flash_mla` wheel 的来源与 `torch` 版本绑定关系 (升级 `torch` 时需要同步重出 wheel); 二是 5 个 `register_ci_gate` 指标在切换内核后端后是否需要重新标定基线; 三是建议后续给 `flash_mla` 安装补上与其他 wheel 一致的 `ls | grep -q` 存在性守卫, 避免 wheel 缺失时镜像构建直接硬失败。对于只关心业务逻辑的读者, 本 PR 可以略过。

功能与动机

PR body 明确写道: 「The megatron-impl 4-layer e2e has been running `dsa_kernel_backend="none"` (pure-PyTorch fallback) because the CI image shipped neither FlashMLA nor the cuDNN python frontend. This puts the production kernel path under CI」。也就是说, 该 e2e 此前只验证了 megatron-impl 的模型结构路径 (`dsv4_hybrid` attention、native mHC), 生产实际使用的 DSA 内核完全没有 CI 覆盖; 同时镜像缺少 FlashMLA 与 cuDNN Python 前端, 导致默认的 cuDNN 后端根本无法启动。关联 Issue 为空, 动机与验证结论都写在 PR body 和 commit message 里。

实现拆解

1. 镜像侧补齐内核依赖 (docker/Dockerfile): 在 `fast-hadamard-transform` 安装块之后、`Mamba kernel` 块之前插入一段 `RUN if [ "${ENABLE_CUDA_13}" = "1" ],` 先 `pip install /tmp/wheels/flash_mla-*.whl`, 再 `pip install --no-deps "nvidia-cudnn-frontend==1.26.0"`。两个包对应 Megatron cuDNN DSA 后端 (`--dsv4-impl megatron`) 所需的 FlashMLA 与 cuDNN Python 前端; wheel 来自 `wheels release (nv_dev b7643bd`, 针对镜像的 `torch 2.13.0+cu130` 构建, 已进入两个 `cu130-torch213-*` 标签)。
2. 关键取舍: `--no-deps` 与 `cu13` 门控: `cudnn-frontend` 的 `cutedsl extra` 会把 `nvidia-cutlass-dsl` 钉在 4.5.0, 从而降级镜像里被 `sglang / DeepGEMM / flashinfer` 共用的 4.6.2, 而 DSA 是运行时针对镜像自带 `dsl` 做 JIT, 因此跳过依赖声明即可。平台门控则因为 `cu12` 的 wheel 集合里根本没有 `flash_mla`, `cu12` 镜像保持原状。

3. 测试侧切换后端 (tests/e2e/megatron/model_scripts/test_deepseek_v4_flash_4layer_megatron_impl_ci.py) : `_args()` 中 `dsa_kernel_backend` 由 "none" 改为 "cudnn", 并删掉三行解释「镜像没有依赖所以只能退化」的注释; 脚本其余启动字段 (`dsv4_impl="megatron"`、`hardware="H200"`、`--ci-test`、`--num-rollout 2` 等) 保持不动。
4. 测试契约与门控不变: `register_cuda_ci(est_time=1900, suite="stage-c-4-gpu-h200", labels=["megatron", "model-scripts"], hardware=["hopper", "blackwell"])` 与 5 个 `register_ci_gate` (`train/grad_norm`、`train/ppo_kl`、`train/train_rollout_logprob_abs_diff`、`train/train_rollout_kl`、`rollout/raw_reward`) 均未修改——后果是这 5 个 `gate` 从此必须以 cuDNN DSA 的数值为新基线, 是本次改动最需要观察的地方。
5. 提交与验证配套: 两个 `commit` 分别为 81a5019 (docker 依赖) 与 ad1f83e (e2e 切换), 依赖先行、切换在后, 便于单独回滚; PR 带 `run-ci-megatron` 与 `bypass-fastfail` 标签以触发真实 GPU 跑批。PR body 记录了 devbox 验证结论: dsl 保持 4.6.2, `from cudnn import DSA` 与 `import flashinfer` 均正常。

关键文件:

- `docker/Dockerfile` (模块 容器镜像; 类别 `infra`; 类型 `infrastructure`; 符号 `flash_mla`, `nvidia-cudnn-frontend`, `ENABLE_CUDA_13`) : 新增 `cu13` 分支下的 `FlashMLA + cudnn-frontend` 安装块, 是本次 e2e 能切到 cuDNN DSA 后端的前提, 同时改变所有 `cu130-torch213-*` 镜像的内容; 其中的 `--no-deps` 与 `ENABLE_CUDA_13` 门控是 PR 唯一的实质设计决策。
- `tests/e2e/megatron/model_scripts/test_deepseek_v4_flash_4layer_megatron_impl_ci.py` (模块 端到端测试; 类别 `test`; 类型 `test-coverage`; 符号 `_args`) : 把 `dsa_kernel_backend` 从 `none` 改为 `cudnn`, 并把解释 `fallback` 的注释删除; 这一行决定了该 e2e 从此跑生产内核路径, 同时继续用 5 个 `register_ci_gate` 指标守住 `DSv4 megatron-impl` 的数值回归。

关键符号: `_args`

关键源码片段

docker/Dockerfile

新增 `cu13` 分支下的 `FlashMLA + cudnn-frontend` 安装块, 是本次 e2e 能切到 cuDNN DSA 后端的前提, 同时改变所有 `cu130-torch213-*` 镜像的内容; 其中的 `--no-deps` 与 `ENABLE_CUDA_13` 门控是 PR 唯一的实质设计决策。

```
# Megatron 的 cuDNN DSA 后端 (--dsv4-impl megatron) 需要 FlashMLA 与 cuDNN 的
# Python 前端; 此前 CI 镜像两者都没有, 导致该 e2e 只能退化为纯 PyTorch fallback.
#
# 平台门控: 仅 cu13 安装——cu12 的 wheel 集合里没有 flash_mla, 装了也无从下载。
# 依赖取舍: cudnn-frontend 必须用 --no-deps 安装, 因为它的 cutedsl extra 会把
# nvidia-cutlass-dsl 钉回 4.5.0, 从而降级镜像中 sglang / DeepGEMM / flashinfer
# 共用的 4.6.2; DSA 在运行时针对镜像自带的 dsl 做 JIT, 因此不需要该 extra 的依赖。
# flash_mla wheel 来自 wheels release (nv_dev b7643bd), 按本镜像的 torch 2.13.0+cu130
构建。
RUN if [ "${ENABLE_CUDA_13}" = "1" ]; then \
    pip install /tmp/wheels/flash_mla-*.whl && \
```

```
pip install --no-deps "nvidia-cudnn-frontend==1.26.0"; \
fi
```

tests/e2e/megatron/model_scripts/test_deepseek_v4_flash_4layer_megatron_impl_ci.py

把 dsa_kernel_backend 从 none 改为 cudnn，并把解释 fallback 的注释删除；这一行决定了该 e2e 从此跑生产内核路径，同时继续用 5 个 register_ci_gate 指标守住 DSv4 megatron-impl 的数值回归。

```
# 该用例的调度契约（本 PR 未改动）：suite="stage-c-4-gpu-h200",
# hardware=["hopper", "blackwell"], 并对 5 个训练 /rollout 指标注册 CI gate:
# train/grad_norm、train/ppo_kl、train/train_rollout_logprob_abs_diff、
# train/train_rollout_kl、rollout/raw_reward。
def _args() -> ScriptArgs:
    return ScriptArgs(
        model_name="DeepSeek-V4-Flash-FP8-4layer",
        task="gsm8k",
        enable_eval=False,
        num_nodes=1,
        num_gpus_per_node=4,
        hardware="H200",
        skip_saving=True,
        use_fault_tolerance=False,
        dsv4_impl="megatron",
        # 关键改动：由 "none"（纯 PyTorch fallback）切到 "cudnn"。
        # 镜像补齐 FlashMLA 与 cudnn-frontend 后，e2e 覆盖的是生产内核路径（DSA），
        # 而不再只是 megatron-impl 的模型结构（dsv4_hybrid attention、native mHC）。
        # 副作用：上述 5 个指标 gate 的数值基线随之从 fallback 迁移到 cuDNN DSA。
        dsa_kernel_backend="cudnn",
        extra_args=(
            "--ci-test "
            "--check-weight-update-allow-quant-error "
            "--ci-disable-logprobs-checker "
            "--num-rollout 2 "
        ),
    )
```

评论区精华

本 PR 没有实质性的评审讨论：[claude\[bot\]](#) 只发了一条仓库级提示（该仓库配置为手动 code review，需要 [@claude review](#) 才触发），[guapisolo](#) 与 [Zhichenzzz](#) 均为 APPROVED 且 body 为空，[review_comments_count](#) 为 0。因此所有技术判断都来自 PR body 的自述与作者在 devbox 上的一次性验证，缺少评审侧的交叉验证。最值得关注的两条技术表述是：其一，「This puts the production kernel path under CI」点明了动机；其二，「the cutedsl extra pins nvidia-cutlass-dsl==4.5.0, which would downgrade the image's 4.6.2 shared by sglang/DeepGEMM/flashinfer; DSA JITs against the image's dsl at runtime」，这是 `--no-deps` 的完整论证，也是本 PR 唯一带有设计权衡的决策。

- CI 长期只跑 PyTorch fallback，生产内核路径无覆盖 (testing): 补齐镜像依赖后把测试切到 cudnn，使内核路径进入 CI 回归范围；未讨论是否需要额外保留一条 fallback 用例作为对照。
- cudnn-frontend 用 --no-deps 安装，避免降级 nvidia-cutlass-dsl (design): 采用 --no-deps，并以 devbox 验证兜底 (dsl 保持 4.6.2, from cudnn import DSA 与 import flashinfer 均可正常 import)；验证范围仅一次 devbox 运行。
- flash_mla 仅安装在 cu13 镜像 (design): cu12 镜像行为不变，但该 e2e 在 cu12 环境无法运行；测试注册的 hopper + blackwell 跨代行为本次只在 H200 suite 验证。
- 评审讨论缺失，靠作者自证与 CI 跑批兜底 (other): 关键设计 (--no-deps、cu13 门控) 未经过第三方技术质询，可信度依赖 PR body 的 devbox 结论与带 run-ci-megatron 标签的 GPU 跑批。

风险与影响

- 风险:
 1. wheel 安装缺少存在性守卫 (docker/Dockerfile)：同文件中 fast_hadamard_transform、causal_conv1d、mamba_ssm 等步骤都先用 ls /tmp/wheels/... | grep -q . 保护再安装，而新增的 pip install /tmp/wheels/flash_mla-*.whl 直接执行；一旦 wheels release 中没有匹配文件 (glob 不展开)，cu13 镜像构建会直接失败，影响面覆盖两个 cu130-torch213-* 标签的所有使用方。
 2. 依赖版本耦合 (ABI 敏感)：flash_mla wheel 是针对 torch 2.13.0+cu130 构建的，属于 ABI 敏感依赖；镜像后续升级 torch 或 CUDA 时需要同步重出 wheel，否则表现为 import 失败或运行期异常，而这类问题只会在 CI/ 训练启动后才暴露。
 3. --no-deps 的回退风险：跳过 cudnn-frontend 声明的依赖后，若 DSA 前端运行期还需要其他未被镜像预装的包，只会在执行到 kernel 时才报错；当前证据仅限 devbox 上一次验证，缺少 CI 之外的环境矩阵。
 4. 指标基线漂移：register_ci_gate 的 5 个指标此前由 PyTorch fallback 产生，切到 cuDNN DSA 后数值可能发生变化，可能触发 gate 误报，也可能因为阈值被放宽而掩盖真实回归；这是本 PR 最需要在后续 nightly 中盯住的一项。
 5. cu12 与跨硬件覆盖缺口：ENABLE_CUDA_13=1 之外的镜像不会安装 flash_mla，该 e2e 在 cu12 环境下无法运行；测试注册的硬件是 hopper + blackwell，但 PR 只在 H200 (hopper) suite 上跑过，blackwell 侧行为未验证。
 6. 镜像体积与构建时间：多装两个包（含一个预编译 wheel 与一个 frontend）会摊到所有复用该镜像的 CI job 上，属于低成本但持续的成本项。- 影响：对系统：CI 首次覆盖 DSv4 megatron-impl 的 cuDNN DSA 生产内核路径，回归网从「模型结构」扩展到「内核实现」，同时 5 个训练 /rollout 指标 gate 的语义随之后移。对用户与团队：所有使用 cu130-torch213-* 镜像做训练或 CI 的成员都会拿到新依赖 (FlashMLA + cudnn-frontend)，镜像构建失败的回归面从本用例扩大到全仓库 cu13 job；后续 torch / cutlass-dsl / CUDA 升级需要同时考虑 flash_mla wheel 与 cudnn-frontend 的兼容性，形成了新的版本维护负担。对流程：本 PR 无 Issue、无内联 review、由作者自己合并，说明这类「CI 能力补齐」改动在团队内属于低争议、由少数人把关的常规操作

，但也意味着风险主要靠 CI 跑批而非评审来兜底。 - 风险标记：镜像构建硬依赖外部 wheel, 缺少 wheel 存在性守卫，指标基线随内核切换漂移，cu13 专有分支，cu12 无覆盖，ABI 敏感的 torch 版本耦合，评讨论论缺失

关联脉络

- PR #2701 feat(ci): declare supported GPU generations on every CUDA test: 本 PR 修改的 tests/e2e/megatron/model_scripts/test_deepseek_v4_flash_4layer_megatron_impl_ci.py 正是 #2701 硬件声明改造的对象之一，其 register_cuda_ci(hardware=["hopper", "blackwell"], suite="stage-c-4-gpu-h200") 沿用该基础设施；本 PR 是同一主线上「把生产内核路径也纳入 CI」的下一步。
- PR #3184 ci: declare hardware on the two CUDA tests that landed after #2701: 同属「CUDA 测试注册 / 硬件感知调度」这条 CI 治理主线，说明测试注册表的补齐工作仍在持续，本 PR 则在同一批 e2e 上叠加后端覆盖要求。
- PR #3192 fix(docker): bump torch_memory_saver to pick up the free-while-paused fix: 同样通过改动 docker/Dockerfile 中的第三方依赖来改变 CI 与运行时行为，与本 PR 「镜像依赖决定 CI 能测什么」的处理模式一致，可对照看镜像依赖升级的评审惯例。