PR #50268 完整报告

vllm-project/vllm

[Hardware][AMD] Enable fused bf16→fp32 router GEMM on ROCm

合并时间：2026-08-12 10:46

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/50268

执行摘要

- 一句话：ROCm 启用融合 bf16→fp32 路由 GEMM，消除独立拷贝内核
- 推荐动作：值得精读，尤其适合关注 ROCm 推理性能与 MoE 路由优化的工程师。核心看点有三：一是利用 torch.mm 的 out_dtype epilogue 消除跨 kernel 数据移动的思路；二是 dispatch 条件中「无 bias」守卫的严谨性（防止 torch.mm 静默丢 bias）；三是完全设备无关的 eligibility 测试写法——用 monkeypatch 模拟平台谓词，让 ROCm 专属逻辑也能在通用 CI 跑通。若团队在 AMD 上部署 MoE 模型，建议合入后至少在 gfx90a 与最新 ROCm 版本上各做一次回归。

功能与动机

MoE 路由门需要 out_dtype=torch.float32 供 grouped_topk 使用（见 issue #50267），但融合 fp32 输出 GEMM 的实现此前仅对 CUDA 开放。在 ROCm 上 allow_cublas_router_gemm 恒为假，回退为 bf16 GEMM 加独立 cast，导致性能 trace 中每个 MoE 层、每个 decode step 都会出现一个独立的 bf16→fp32 拷贝内核。PR body 明确指出：'This caused a fallback to bf16 GEMM followed by a separate cast operation, creating a standalone bf16→fp32 copy kernel visible in performance traces between the router GEMM and grouped_topk for every MoE layer on each decode step.' hipBLASLt 支持与 cuBLAS 类似的 out_dtype epilogue，因此该平台限制应被移除。

实现拆解

1. 修改核心 dispatch 条件（源码入口）：在 vllm/model_executor/layers/fused_moe/router/gate_linear.py 的 __init__ 中，将 allow_cublas_router_gemm 的判定从 self.allow_specialized_router_gemm 扩展为 self.allow_specialized_router_gemm or (current_platform.is_rocm() and self._router_gemm_no_bias)，并新增 self._router_gemm_no_bias = not bias 属性。原因是 torch.mm 的 epilogue 不含 bias 项，ROCm 分支必须显式排除带 bias 的 gate，避免 bias 被静默丢弃。最终的完整条件仍是 bf16 权重 + fp32 输出。
2. 同步 set_out_dtype 路径：set_out_dtype 用于 gate 构造后延迟设置输出精度（依赖专家量化方式才能确定 out_dtype 的场景）。原实现只对 allow_specialized_router_gemm 重算 eligibility，会导致 ROCm 上延迟设置 fp32 时 fused 路径无法启用；本次同步增加 current_platform.is_rocm() and self._router_gemm_no_bias 分支，保证与 __init__ 行为一致。

3. 新增设备无关单测：新增 tests/kernels/test_gate_linear_rocm_dispatch.py，通过 monkeypatch 平台的 is_cuda、is_rocm、is_device_capability 谓词，在无 GPU 环境下直接断言 allow_cublas_router_gemm 的取值。覆盖 7 个场景：ROCm 无 bias + bf16 + fp32 启用、带 bias 禁用、fp32 权重禁用、非 fp32 输出禁用、非 ROCm 非 CUDA 禁用、set_out_dtype 启用、set_out_dtype 仍尊重 bias 守卫。
4. 性能验证：在 gfx942 上以在线服务基准，3 个 seed 各跑 40 请求（40960 input + 40960 generated tokens，max concurrency 4），确认 trace 中不再出现独立的 bfloat16tofloat32_copy_kernel；TPOT/ITL 平均下降 0.53 ms（约 1.6%），输出吞吐提升约 1.6%，TTFT 基本持平，且种子间方差很小。
5. 配套说明：无配置、schema 或部署改动；CUDA 路径因 current_platform.is_rocm() 为 false 而完全不变。

关键文件：

- vllm/model_executor/layers/fused_moe/router/gate_linear.py（模块 MoE 路由；类别 source；类型 core-logic；符号 GateLinear.init, GateLinear.set_out_dtype, allow_cublas_router_gemm, _router_gemm_no_bias）：核心变更文件：将 allow_cublas_router_gemm 的开启条件从仅 CUDA 扩展为「CUDA 专用 kernel 或 ROCm 无 bias」，并同步 set_out_dtype 的重算逻辑，是本次性能优化的唯一源码改动点。
- tests/kernels/test_gate_linear_rocm_dispatch.py（模块 ROCm 测试；类别 test；类型 test-coverage；符号 _make_gate, test_rocm_no_bias_bf16_fp32_enables_fused_gemm, test_rocm_bias_disables_fused_gemm, test_rocm_fp32_weight_disables_fused_gemm）：新增设备无关单测：通过 monkeypatch 平台谓词，在无 GPU 环境直接断言 ROCm fused GEMM 的 eligibility 条件，覆盖 bias、权重 dtype、输出 dtype、set_out_dtype 等边界，固化本次 dispatch 语义。

关键符号：GateLinear.set_out_dtype, _make_gate, test_rocm_no_bias_bf16_fp32_enables_fused_gemm, test_rocm_bias_disables_fused_gemm, test_rocm_set_out_dtype_enables_fused_gemm

关键源码片段

vllm/model_executor/layers/fused_moe/router/gate_linear.py

核心变更文件：将 allow_cublas_router_gemm 的开启条件从仅 CUDA 扩展为「CUDA 专用 kernel 或 ROCm 无 bias」，并同步 set_out_dtype 的重算逻辑，是本次性能优化的唯一源码改动点。

```
# vllm/model_executor/layers/fused_moe/router/gate_linear.py
# 关键变更：让 allow_cublas_router_gemm 在 ROCm 上也成立。
# torch.mm 的 out_dtype epilogue 可以将 bf16 乘法的 fp32 转换折入 GEMM，
# 避免独立 bf16 -> fp32 拷贝 kernel；但 epilogue 没有 bias 项，
# 所以带 bias 的 gate 必须回退，否则 bias 会被静默丢弃。

# __init__ 中：
self._router_gemm_no_bias = not bias
self.allow_cublas_router_gemm = (
```

```
        (
            # CUDA 侧走 SM90+ 专用 kernel 路径（原逻辑不变）
            self.allow_specialized_router_gemm
            # ROCm 侧不要求专用 kernel，只要无 bias 即可走 hipBLASLt epilogue
            or (current_platform.is_rocm() and self._router_gemm_no_bias)
        )
        and self.weight.dtype == torch.bfloat16
        and self.out_dtype == torch.float32
    )


    def set_out_dtype(self, out_dtype: torch.dtype) -> None:
        """设置路由 logits 的输出精度（可能晚于 __init__ 才确定）。"""
        if self.out_dtype is not None:
            raise ValueError("out_dtype has already been set")
        self.out_dtype = out_dtype

        # out_dtype 由 None 变为 fp32 时，需要重算 fused GEMM 的 eligibility；
        # 这里必须与 __init__ 保持同一份 ROCm/CUDA 条件，避免两处逻辑漂移。
        if (
            not self.allow_cublas_router_gemm
            and (
                self.allow_specialized_router_gemm
                or (current_platform.is_rocm() and self._router_gemm_no_bias)
            )
            and out_dtype == torch.float32
        ):
            self.allow_cublas_router_gemm = self.weight.dtype == torch.bfloat16

        # cuteDSL ll_bf16_gemm 仍是 SM90+ 专属路径（CUDA），ROCm 不进入分支
        if self.allow_specialized_router_gemm:
            from vllm.model_executor.kernels.linear.cute_dsl.ll_bf16 import is_available

            self.allow_ll_bf16_gemm = (
                self.weight.dtype == torch.bfloat16
                and out_dtype == torch.float32
                and is_available()
            )
```

tests/kernels/test_gate_linear_rocm_dispatch.py

新增设备无关单测：通过 monkeypatch 平台谓词，在无 GPU 环境直接断言 ROCm fused GEMM 的 eligibility 条件，覆盖 bias、权重 dtype、输出 dtype、set_out_dtype 等边界，固化本次 dispatch 语义。

```
# tests/kernels/test_gate_linear_rocm_dispatch.py
# 设备无关的 eligibility 测试：直接 mock 平台谓词，不需要 GPU，
# 因此 ROCm 专属分支也能在通用 CI 上得到验证。

def _make_gate(
```

```
    monkeypatch,
    *,
    is_rocm: bool,
    is_cuda: bool = False,
    bias: bool = False,
    params_dtype: torch.dtype = torch.bfloat16,
    out_dtype: torch.dtype | None = torch.float32,
) -> GateLinear:
    """构造 GateLinear 并 mock 平台与张量并行环境。"""
    # 单卡环境：rank = 0，world_size = 1
    for target in ("vllm.model_executor.layers.linear",
                   "vllm.model_executor.parameter"):
        monkeypatch.setattr(f"{target}.get_tensor_model_parallel_rank",
                            lambda: 0)
        monkeypatch.setattr(f"{target}.get_tensor_model_parallel_world_size",
                            lambda: 1)

    platform = gate_linear_mod.current_platform
    monkeypatch.setattr(platform, "is_cuda", lambda: is_cuda)
    monkeypatch.setattr(platform, "is_rocm", lambda: is_rocm)
    # 强制关掉 CUDA 专用 kernel 的能力检测（SM90/SM100），
    # 让测试只关注 ROCm 分支本身
    monkeypatch.setattr(platform, "is_device_capability",
                        lambda *a, **k: False)
    monkeypatch.setattr(platform, "is_device_capability_family",
                        lambda *a, **k: False)

    return GateLinear(input_size=2048, output_size=64, bias=bias,
                      out_dtype=out_dtype, params_dtype=params_dtype)


def test_rocm_no_bias_bf16_fp32_enables_fused_gemm(monkeypatch):
    # 核心正向用例：ROCm + 无 bias + bf16 权重 + fp32 输出，应走融合路径
    gate = _make_gate(monkeypatch, is_rocm=True, bias=False)
    assert not gate.allow_specialized_router_gemm
    assert gate.allow_cublas_router_gemm


def test_rocm_bias_disables_fused_gemm(monkeypatch):
    # torch.mm 的 epilogue 没有 bias 项，带 bias 的 gate 必须回退，
    # 否则 bias 会被静默丢弃（这是 ROCm 分支最重要的守卫）
    gate = _make_gate(monkeypatch, is_rocm=True, bias=True)
    assert not gate.allow_cublas_router_gemm


def test_rocm_set_out_dtype_enables_fused_gemm(monkeypatch):
    # out_dtype 延迟设置（init 时为 None）后在 ROCm 上应能重新启用融合路径
    gate = _make_gate(monkeypatch, is_rocm=True, bias=False, out_dtype=None)
    assert not gate.allow_cublas_router_gemm
```

```
gate.set_out_dtype(torch.float32)
assert gate.allow_cublas_router_gemm
```

评论区精华

该 PR 的 review 讨论很少：claude[bot] 指出这是来自 fork 的 PR，自动 review 被禁用，维护者可评论 @claude review 触发一次性检查；维护者 tjtanaa 直接 approve 并合入，未留下实质性设计质疑。唯一值得注意的约束来自 PR body 自身：torch.mm 没有 bias 项，因此 ROCm 融合分支必须以无 bias 为前提，否则 bias 会被静默丢弃——这一设计决定已被新增测试（test_rocm_bias_disables_fused_gemm、test_rocm_set_out_dtype_respects_bias_guard）固化。

- fork PR 自动 review 被禁用 (other): 未触发额外自动 review；维护者 tjtanaa 直接 approve 并合入，未留设计讨论。

风险与影响

- 风险：
  1. 平台覆盖有限：仅 gfx942 单卡实测，其他 ROCm 架构（gfx90a、RDNA 等）与 hipBLASLt 版本未验证；代码没有探测 hipBLASLt 是否真正支持 out_dtype epilogue，若旧版本不支持可能静默回退或异常。
  2. 数值语义变化：fused epilogue 的 fp32 累积与「bf16 GEMM + 单独 cast」的数值结果不同，PR 声明与 fp32 参考误差约 6e-5，但 grouped_topk 的选路对 logits 微小差异是否敏感未深入测试。
  3. 双处逻辑同步维护：__init__ 与 set_out_dtype 各自维护一份 ROCm eligibility 逻辑，虽有 _router_gemm_no_bias 抽取，未来 CUDA/ROCm 条件再扩展时仍存在两处漂移的维护风险；新增测试可部分缓解。
  4. 回归面：改动集中在单一 dispatch 标志，只影响 ROCm 上「无 bias + bf16 权重 + fp32 输出」的 router gate 路径；带 bias 的 gate、fp32 权重以及所有 CUDA 模型行为均不变。 - 影响：对 AMD/ROCm 用户：MoE 模型推理的 decode 阶段每个 MoE 层每步少一个 kernel launch 与一次全量数据搬移，实测 TPOT/ITL 下降约 1.6%、输出吞吐提升约 1.6%，无正确性回归（误差 ~6e-5）。对 CUDA 及其他平台：零影响。对团队：这是一次小范围、低风险的平台能力补齐，配合设备无关单测可在任意 CI 环境验证 dispatch 逻辑；测试设计（mock 平台谓词）也为后续其他平台的同类判定提供了可复用范式。 - 风险标记：ROCm 平台专属路径，仅 gfx942 实际验证，hipBLASLt 版本差异未探测，init 与 set_out_dtype 双处逻辑需同步

关联脉络

- PR #51980 [Bugfix][ROCm][MoE] Update AITER MXFP4 W4A16 tests to the renamed expert_mask: 同为 ROCm MoE 路径的测试适配，反映 ROCm MoE 链路正在持续补齐与加固。
- PR #51860 [ROCm][K3] Dequantize the fp8 decode query for MLA backends without quant-query support - TRITON_MLA: 同为 ROCm 上 MoE/ 解码路径的 kernel 行为修正

，印证 ROCm 平台是近期活跃维护方向。

- PR #51464 [ROCm] update triton in base docker for gluon compatibility: ROCm 基础镜像与内核兼容性基础设施调整，与本 PR 同属 AMD 平台支撑工作。