PR #46177 完整报告

vllm-project/vllm

[Bugfix][Model] Support tensor parallelism for DiffusionGemma (#45719)

合并时间：2026-06-27 04:05

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/46177

执行摘要

- 一句话：修复 DiffusionGemma TP>1 自条件化 matmul 崩溃
- 推荐动作：值得精读。展示了在 torch.compile 编译区域中引入分布式通信的设计模式：通过虚设注册 custom op 使 all-reduce 可追踪，同时保持分片权重避免显存浪费。对于其他需要向量并行 embedding 的模型（如 Mixture of Experts）有借鉴意义。

功能与动机

Issue #45719 报告 DiffusionGemma 在 TP>1 时代码崩溃，导致多 GPU 部署不可用。自条件化 soft-embedding 中 probs @ embed_tokens.weight 的 embed_tokens 是 VocabParallelEmbedding，其权重被切分，而 probs 针对全词汇，导致 dynamo 跟踪时 reduction 维度不匹配。

实现拆解

1. 获取分片信息：在 custom_sampler 中，通过 embed_tokens.shard_indices 获取当前 rank 的 org_vocab_start_index 和 org_vocab_end_index，并通过 get_tp_group() 获取 world_size 和 unique_name。
2. 扩展 _compiled_sample_step 接口：新增 sc_vocab_start、sc_vocab_end、tp_size、tp_group_name 四个参数，将分片边界和通信组传递到编译区域。
3. 修改 self-conditioning 计算：将原先的 probs @ embed_weight（全词汇 matmul）替换为 local_probs = probs[..., sc_vocab_start:sc_vocab_end] 与本地切片 embed_weight[:sc_vocab_end-sc_vocab_start] 的 matmul，若 tp_size > 1 则执行 torch.ops.vllm.all_reduce 求和局部结果，再乘以归一化因子。
4. 删除无效单元测试：移除仅测试 vocab_parallel_embedding 公共方法的 CPU 单元测试 test_diffusion_gemma_parallel.py。
5. 添加端到端评估：新增 tests/evals/gsm8k/configs/DiffusionGemma-26B-A4B-it-FP8-dynamic.yaml，使用 FP8 动态量化模型在 TP=2 下运行 GSM8K 评估，并将该配置加入 models-small-tp.txt 以被 CI 选中。
6. 调整 CI 流水线：在 .buildkite/test_areas/lm_eval.yaml 中新增 LM Eval Small Models (2xL4) 步骤（num_devices: 2，optional: true），用于自动运行 TP=2 评估。

关键文件：

- vllm/model_executor/models/diffusion_gemma.py（模块 模型执行器；类别 source；类型 core-logic；符号 _compiled_sample_step, custom_sampler, get_tp_group）：核心修复文件：修改 self-conditioning soft-embedding 计算以支持 TP 分片与 all-reduce。
- tests/evals/gsm8k/configs/DiffusionGemma-26B-A4B-it-FP8-dynamic.yaml（模块 评估测试；类别 test；类型 test-coverage）：新增 TP=2 端到端评估配置，验证修复在真实分布式环境下的正确性。
- .buildkite/test_areas/lm_eval.yaml（模块 CI/ 构建；类别 config；类型 configuration）：新增 2xL4 CI 流水线，用于自动执行 TP 模型评估。
- tests/evals/gsm8k/configs/models-small-tp.txt（模块 评估配置；类别 docs；类型 documentation）：引用 DiffusionGemma 评估配置，使其被 CI 流水线选中执行。

关键符号：_compiled_sample_step, custom_sampler

评论区精华

- LucasWilkinson建议将 vocab_start/vocab_end 重命名为 sc_vocab_start/sc_vocab_end 以明确作用域，作者已采纳。
- LucasWilkinson指出原 CPU 单元测试未实际导入修复代码，建议替换为真实端到端测试，作者删除该测试并添加了 GSM8K TP=2 评估配置。
- mgoin建议使用量化 checkpoint（FP8）以加快评估速度并同时验证量化路径，作者将模型从 bf16 切换到 RedHatAI/diffusiongemma-26B-A4B-it-FP8-dynamic。
- mgoin提醒评估配置需加入 .txt 运行列表才会被 CI 执行，作者先加入 models-blackwell.txt，后因 GPU 数量不匹配移至 models-small-tp.txt。
- 作者备注 accuracy_threshold: 0.84 为未验证占位符，需后续实际运行后校准。
- CPU 单元测试不覆盖修复代码 (testing): 作者删除该单元测试，添加 GSM8K TP=2 评估配置。
- 使用 FP8 量化 checkpoint 加速评估 (performance): 作者将测试目标从 bf16 切换到 RedHatAI/diffusiongemma-26B-A4B-it-FP8-dynamic。
- 评估配置需加入运行列表 (test): 作者先将配置加入 models-blackwell.txt，后因 GPU 数量不匹配（B200 单卡）移至 models-small-tp.txt，最终 CI 在 2xL4 上运行。
- 准确率阈值占位符未验证 (test): 当前保留为占位符，需在后续 PR 或 CI 结果中调整。

风险与影响

- 风险：
  - 回归风险（低）：TP=1 时分片范围为整个词汇，all_reduce 被跳过，计算逻辑与原路径完全一致（字节级等价）。
  - 性能风险（中）：TP>1 时每个 decode step 新增一次对 [num_decode, canvas, hidden] 张量的 all-reduce，该张量较小（canvas <= 4096，hidden=2816），但跨 GPU 通信仍可能引入微秒级延迟，对吞吐有轻微影响（已在 TP=4 实测正常）。
  - 准确率风险（中）：评估配置的 accuracy_threshold: 0.84 未在实际 CI 中跑出基线，可能因模型或环境波动导致阈值偏低或偏高，合并后需实测校准。

    - 编译兼容性：修改发生在 @torch.compile 区域，torch.ops.vllm.all_reduce 通过 fake 注册保证可追踪，但仍依赖 PyTorch 版本对 custom op 的 tracing 支持。
- 影响：
    - 用户：修复后 DiffusionGemma 可在多 GPU（TP>1）上运行，尤其对使用 16GB 等消费级显卡的用户意义重大，模型从“不可用”变为“可用”。
    - 系统：新增 TP=2 评估配置和 L4 CI 流水线，每次 CI 自动检查 DiffusionGemma 在 tensor parallelism 下的正确性。
    - 团队：维护新增的评估 YAML 和 CI 步骤；accuracy_threshold 需在首次运行后调整。
    - 风险标记：核心路径变更，通信开销，准确率未验证，编译兼容性

关联脉络

- PR #45719 [Bug]: DiffusionGemma crashes under tensor-parallel (TP>1) and pipeline-parallel (PP>1) — multi-GPU is unusable: 本 PR 修复该 issue 报告的 TP 部分。
- PR #45774 （另一修复方案：all-gather embedding 权重）：PR body 中提到的另一种方案，本 PR 采用更省内存的 all-reduce 分片方法。
- PR #46212 （作者 shubhamprshr27 的重复修复）：独立验证的相同 TP 修复，已关闭并合入本 PR。
- PR #45828 （PP 相关，建议阻塞式失败）：PR body 提到 PP 问题故意留给另一 PR，本 PR 不处理。