

PR #6551 完整报告

verl-project/verl

[docker, misc] chore: upgrade vllm, nccl and numpy > 2.0.0, to resolve packages dependency error

合并时间: 2026-06-18 10:44

原文链接: <http://prhub.com.cn/verl-project/verl/pull/6551>

执行摘要

- 一句话: 升级 numpy、vLLM、NCCL, 修复依赖错误
- 推荐动作: 建议精读此 PR 以了解 verl 依赖管理策略, 尤其是 vLLM 补丁应用模式和 NCCL 覆盖方法。对于涉及 NumPy 2.0 迁移的贡献者, 需关注 protocol.py 中的修改模式。

功能与动机

PR 标题和 body 明确指出升级 numpy>2.0.0 是为了解决包依赖错误, 具体由 uv 迁移触发。同时需要升级 vLLM 以支持新特性并修复已知问题, NCCL 升级用于后续完全异步训练所需的 suspend/resume 能力。

实现拆解

1. 放宽 numpy 版本限制: 在 setup.py、requirements.txt、scripts/install_vllm_sglang_mcore.sh 中将 numpy<2.0.0 改为 numpy>=2.0.0; 在 verl/protocol.py 中移除 np.reshape 的 newshape 参数以适配 NumPy 2.0 废弃的 API。
2. 升级 vLLM 并应用补丁: 在 docker/Dockerfile.stable.vllm 中将 VLLM_VERSION 从 0.20.2 提升至 0.23.0, 并从 GitHub 拉取两个修复补丁 (#44483 非法内存访问、#45589 FlashInfer-TRTLLM MoE OOM) 通过 git apply 打入; 同时更新 CI 镜像标签为 vllm023.dev1。
3. 升级 NCCL 版本: 在两个 Dockerfile 末尾添加 pip install --no-deps --upgrade nvidia-nccl-cu13>=2.29.7, 覆盖 CUDA 13 捆绑的 NCCL 2.28.9, 以支持 ncclCommSuspend/Resume, 该功能在 torch>=2.12.0 前需要手动覆盖。
4. 排除 transformers 5.6.0: 在 setup.py 中将 transformers 约束改为 !=5.6.0, 附注释说明该版本包含损坏的 flash-attention 路径 (huggingface/transformers#45588)。
5. 更新 CI 工作流: 修改多个 .github/workflows/*.yml 中的镜像标签为新版本, 并在模型测试和 E2E 测试步骤中添加 pip3 list 调试步骤, 方便排查依赖问题; 同时为 e2e_fully_async_policy.yml 添加了 requirements.txt 安装。

关键文件:

- setup.py (模块 安装配置; 类别 source; 类型 core-logic): 核心安装文件, 修改了 numpy 和 transformers 的版本约束, 直接影响所有用户依赖解析。

- verl/protocol.py (模块 数据协议; 类别 source; 类型 core-logic; 符号 fold_batch_dim, unfold_batch_dim) : NumPy 2.0 废弃了 np.reshape 的 newshape 参数, 本次修改在 fold_batch_dim 和 unfold_batch_dim 中移除了该命名参数, 是兼容升级的关键适配。
- docker/Dockerfile.stable.vllm (模块 vLLM 镜像; 类别 infra; 类型 infrastructure) : vLLM 升级至 0.23.0 并应用了两个修复补丁, 同时覆盖 NCCL 版本, 是本次升级中改动最复杂的文件。
- docker/Dockerfile.stable.sglang (模块 SGLang 镜像; 类别 infra; 类型 infrastructure) : 同步 NCCL 覆盖, 并调整了 flash-attn-4 的安装逻辑。
- requirements.txt (模块 依赖清单; 类别 config; 类型 configuration) : 基础依赖清单, 同步 numpy 版本升级。
- scripts/install_vllm_sglang_mcore.sh (模块 安装脚本; 类别 infra; 类型 configuration) : 安装脚本, 同步 numpy 版本以保持一致性。
- .github/workflows/model.yml (模块 CI 工作流; 类别 infra; 类型 infrastructure) : CI 工作流, 更新镜像标签为 vllm023.dev1, 并添加调试步骤。

关键符号: fold_batch_dim, unfold_batch_dim

关键源码片段

setup.py

核心安装文件, 修改了 numpy 和 transformers 的版本约束, 直接影响所有用户依赖解析。

```
# setup.py — 安装依赖核心配置
install_requires = [
    "accelerate",
    "codetiming",
    "datasets",
    "dill",
    "hydra-core",
    "numpy>=2.0.0", # 升级至 NumPy 2.0+, 解决 uv 依赖冲突
    "pandas",
    "peft",
    "pyarrow>=19.0.0",
    "pybind11",
    "pylatexenc",
    "ray[default]>=2.41.0",
    "torchdata",
    "tensordict>=0.8.0,<=0.10.0,!0.9.0",
    # 5.6.0 ships a broken flash-attention path (crashes on s_aux=None for
    # sink-less models); fixed in 5.6.1. See huggingface/transformers#45588.
    "transformers!=5.6.0", # 排除损坏版本
    "wandb",
    "packaging>=20.0",
    "tensorboard",
]
```

verl/protocol.py

NumPy 2.0 废弃了 `np.reshape` 的 `newshape` 参数，本次修改在 `fold_batch_dim` 和 `unfold_batch_dim` 中移除了该命名参数，是兼容升级的关键适配。

veril/protocol.py — 批次维度折叠与展开，适配 NumPy 2.0

```
def fold_batch_dim(data: "DataProto", new_batch_size):
    """
    Fold a batch dim from [bsz, xxx] into [new_bsz, bsz // new_bsz, xxx]
    """
    batch_size = data.batch.batch_size[0]
    assert batch_size % new_batch_size == 0

    tensor: TensorDict = data.batch
    non_tensor = data.non_tensor_batch

    tensor = tensor.view(new_batch_size, -1)
    tensor.auto_batch_size_(batch_dims=1)

    for key, val in non_tensor.items():
        # NumPy >=2.0 废弃 newshape 参数，改为仅使用位置参数
        non_tensor[key] = np.reshape(val, (new_batch_size, -1, *val.shape[1:]))

    return type(data)(batch=tensor, non_tensor_batch=non_tensor, meta_info=data.meta_info)


def unfold_batch_dim(data: "DataProto", batch_dims=2):
    """
    Unfold the first n dims as new batch dim
    """
    tensor: TensorDict = data.batch
    non_tensor = data.non_tensor_batch
    tensor.auto_batch_size_(batch_dims=batch_dims)
    tensor = tensor.view(-1)

    batch_size = tensor.batch_size[0]

    non_tensor_new = {}
    for key, val in non_tensor.items():
        # 同上，移除 newshape 参数
        non_tensor_new[key] = np.reshape(val, (batch_size, *val.shape[batch_dims:]))

    return type(data)(batch=tensor, non_tensor_batch=non_tensor_new, meta_info=data.meta_info)
```

`docker/Dockerfile.stable.vllm`

vLLM 升级至 0.23.0 并应用了两个修复补丁，同时覆盖 NCCL 版本，是本次升级中改动最复杂的文件。

docker/Dockerfile.stable.vllm — vLLM 升级与补丁

ARG VLLM_VERSION=0.23.0 # 从 0.20.2 升级

克隆 vLLM 源码并 checkout 对应 tag, 然后应用两个修复补丁

```
RUN git clone https://github.com/vllm-project/vllm.git && cd vllm && \
    git checkout v${VLLM_VERSION} && \
    # #44483: illegal memory access during partial wake_up (sleep mode)
    curl -fsSL https://github.com/vllm-project/vllm/pull/44483.diff -o /tmp/vllm-pr-44483.diff && \
    git apply --3way --whitespace=nowarn /tmp/vllm-pr-44483.diff && \
    # #45589: FlashInfer-TRTLLM MoE load OOM
    curl -fsSL https://github.com/vllm-project/vllm/pull/45589.diff -o /tmp/vllm-pr-45589.diff && \
    git apply --3way --whitespace=nowarn /tmp/vllm-pr-45589.diff && \
    MAX_JOBS=256 pip install -e .
```

... 后续步骤 ...

升级 NCCL 以支持 suspend/resume (RFC: <https://github.com/verl-project/verl/issues/6266>)

```
RUN pip install --no-deps --upgrade "nvidia-nccl-cu13>=2.29.7,<3.0"
```

评论区精华

- gemini-code-assist[bot]: 建议将 `numpy>2.0.0` 改为 `numpy>=2.0.0`, 以避免严格排除 2.0.0 自身导致依赖解析问题。该建议已被采纳 (最终 diff 中为 `>=`) 。
- ji-huazhong: 询问最新 torch-npu 是否与 NumPy 2.0+ 兼容, 请求 @wucong25 确认。ETOGaosion 回复表示 NPU 相关 CI 将先迁移至 uv, 届时可在 uv 中手动固定 `numpy<2.0`, 因此该 PR 暂不影响 NPU 路径。
- numpy 版本约束: `> vs >= (correctness)`: 建议被采纳, 最终 diff 中全部改为 `>=2.0.0`。
- NumPy 2.0 与 torch-npu 兼容性 (question): ETOGaosion 回复 NPU CI 将先行迁移至 uv, 可在 uv 中手动固定 `numpy<2.0`, 此 PR 暂时不影响 NPU 路径。

风险与影响

- 风险: 兼容性风险: NumPy 2.0 引入了破坏性更改, 虽然 `protocol.py` 已修复 `newshape` 参数, 但其他隐式用法 (如 `np.object` 移除) 可能导致运行时错误, 需监控。vLLM 升级风险: 从 0.20.2 直接升至 0.23.0, 跳跃较大, 可能引入新 bug 或行为变化; 所打补丁只覆盖了两个已知问题, 未涵盖全部。NCCL 覆盖风险: 通过 `--no-deps --upgrade` 替换 NCCL, 可能与其他依赖 (如 PyTorch 预期版本) 不一致, 若未来 torch 升级则需移除覆盖。Transformers 排除: 排除 5.6.0 可能影响需要该版本的用户, 但已知 bug 严重, 排除是合理的。回归范围: CI 镜像标签变更影响所有使用旧镜像的 CI 工作流, 需确保新镜像已通过测试。
- 影响: 用户: 使用 `pip install verl` 或基于 Docker 的用户将自动升级 numpy 至 2.0+, 可能需适配 API 变化。系统: Docker 镜像体积因 vLLM 重新构建而增加; NCCL 替换可能影响多卡通信稳定性。团队: vLLM 和 NCCL 的新特性 (如 `suspend/resume`) 可为后续完全异步训练铺路。
- 风险标记: 核心依赖升级, NumPy 2.0 破坏性变更, vLLM 大版本跳跃, NCCL 覆盖可能冲突, Transformers 版本排除

关联脉络

- PR #6730 [docker] feat: bump stable trtllm image to 1.3.0rc15: 同为 Docker 镜像依赖升级，展示了类似的依赖管理实践（覆盖 NCCL 等）。
- PR #6728 [rollout, vllm] fix: Adjust cuda graph capture sizes config logic for vLLM $\geq 0.11.1$ compatibility: 同一时期针对 vLLM 版本的兼容性修复，与本 PR 的 vLLM 升级相关。