

PR #5627 完整报告

verl-project/verl

[perf, trainer, training_utils, ray, worker] fix: Add set_numa_affinity() for engine workers: TrainingWorker.

合并时间: 2026-03-23 13:55

原文链接: <http://prhub.com.cn/verl-project/verl/pull/5627>

执行摘要

- 一句话: 为 TrainingWorker 添加 NUMA 亲和性设置, 提升 GPU 训练性能 6.1%。
- 推荐动作: 建议技术管理者关注此 PR 以了解 NUMA 优化在分布式训练中的应用和硬件兼容性权衡; 工程师可精读代码学习如何集成性能调优函数, 但需注意 NPU 限制。

功能与动机

PR body 中说明: 'We find a consistent performance gap of `sft_trainer.py` vs `sft_trainer_ray.py` on SFT jobs on AWS p5 instances, and this PR reduce the gap though still not fully resolved'. 目标是通过设置 NUMA 亲和性优化 GPU 性能, 基于测试数据 MFU 提升 6.1%。

实现拆解

实现集中在 `verl/workers/engine_workers.py` 文件。关键改动: 在 `TrainingWorker.__init__` 方法中, 在初始化全局进程组后添加 `set_numa_affinity()` 调用, 从 `verl.utils.distributed` 导入该函数。这是一个小范围变更, 类似 PR #3471 对 Megatron 的处理。

关键文件:

- `verl/workers/engine_workers.py` (模块 `workers`): 唯一修改的文件, 在 `TrainingWorker` 初始化中添加 `set_numa_affinity()` 调用, 直接影响引擎工作进程性能和 NUMA 优化。

关键符号: `TrainingWorker.init`, `set_numa_affinity`

评论区精华

review 中 `gemini-code-assist[bot]` 指出: '`set_numa_affinity()` 当前在 NPU 系统上是 no-op, 性能提升仅限于 GPU 环境。' 作者回复: 'Ack, this PR is to fix GPU issues and fully tested E2E on AWS p5 instances.' 讨论聚焦于硬件兼容性, 结论是 PR 针对 GPU 优化, NPU 限制未被解决。

- NUMA 亲和性在 NPU 上的限制 (performance): 作者确认 PR 针对 GPU 问题, 测试在 AWS p5 实例, NPU 限制未被解决但作为已知问题。

风险与影响

- 风险：风险包括：1) NPU 系统上 `set_numa_affinity()` 是 no-op，导致这些硬件错过性能优化；2) 依赖 `verl.utils.distributed.set_numa_affinity()` 函数的正确性，如果函数实现有误可能影响所有 `TrainingWorker`；3) 缺少对 NPU 场景的测试覆盖，可能引入未检测的兼容性问题。
- 影响：影响：1) 对 GPU 训练：显著提升 MFU (+6.1%) 并降低方差 (-64.6%)，改善分布式训练性能稳定性；2) 对 NPU 训练：无影响，因函数为 no-op；3) 影响所有使用 `TrainingWorker` 的 Ray 训练作业，但对其他模块无直接影响。
- 风险标记：NPU 兼容性限制，依赖外部函数

关联脉络

- PR #3471 类似 PR 添加 `set_numa_affinity()` 用于 Megatron: PR body 中提及，同属性能优化，添加相同 util 函数到不同模块，显示跨模块 NUMA 优化趋势。