

PR #28122 完整报告

sgl-project/sglang

[MLX] Add Metal profiling hooks to server profiler

合并时间: 2026-06-18 04:06

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/28122>

执行摘要

- 一句话: 新增 Apple Silicon Metal 性能分析后端
- 推荐动作: 值得精读。该 PR 展示了如何通过 backend patch 模式在 SGLang 中为新型硬件添加 profiling 支持, 设计思路清晰, 与已有 NPU 后端一致, 可作为跨后端扩展的参考模板。对于 Apple Silicon 开发人员, 还提供了详细的用法和验证结果。

功能与动机

扩展 #22159 首次添加的 Apple Silicon profiling 支持至服务端 profiler, 让 `bench_serving.py` 和 `bench_one_batch_server.py` 也能启用 Metal GPU trace 捕获, 以完成 Apple Device Support 路线图 (issue #19137) 中 Profiling 部分的目标。

实现拆解

1. 新增 `hardware_backend/mlx/profiler.py`: 实现 `MetalCaptureProfiler` 数据类, 管理 MLX (`mx.metal.start_capture`) 和 MPS (`torch.mps.profiler.metal_capture`) 两条捕获路径的启动与停止; `MetalTorchProfiler` 作为 `torch.profiler.profile` 的替代包装, 代理 `start/stop` 和 `trace` 导出; `apply_metal_profiler_patches()` 全局替换 `torch.profiler.profile`, 并保证幂等性。
2. 修改 `profiler_manager.py`: 在模块加载时新增 `_is_mps` 判断, 在 NPU 补丁之后调用 `apply_metal_profiler_patches()`; 将 `ProfileManager` 导入移至文件顶部; 在 `_start_profile` 中将 `self.torch_profiler.start()` 包裹在 `try/except RuntimeError` 中, 捕获失败时返回 `success=False` 而非抛出异常。
3. 新增 `test_metal_profiler.py`: 10 个单元测试覆盖补丁替换、幂等性、MLX/MPS 路径成功与 `RuntimeError` 降级、以及 `SchedulerProfilerManager` 的完整 `start/stop` 周期, 所有测试自动跳过非 Apple Silicon 或缺少 `mlx` 的环境。

关键文件:

- `python/sglang/srt/hardware_backend/mlx/profiler.py` (模块 硬件后端; 类别 `source`; 类型 `dependency-wiring`; 符号 `init`, `MetalCaptureProfiler`, `start_mlx`, `start_mps`): 核心实现, 新增 Metal profiling 后端, 包含 `MetalCaptureProfiler` 和 `MetalTorchProfiler` 两个类以及 `patch` 应用函数。
- `test/registered/unit/hardware_backend/mlx/test_metal_profiler.py` (模块 单元测试; 类别 `test`; 类型 `test-coverage`; 符号 `TestApplyMetalProfilerPatches`, `setUp`, `tearDown`,

test_patch_replaces_profile)：新增 10 个单元测试，覆盖补丁应用、MLX/MPS 路径成功与失败场景。

- python/sglang/srt/managers/scheduler_components/profiler_manager.py (模块 调度器；类别 source；类型 dependency-wiring；符号 is_mps, apply_metal_profiler_patches, _start_profile)：在 profiler_manager 中串联 Metal profiling 补丁，使服务端 profiler 支持 Apple Silicon。

关键符号：MetalCaptureProfiler.start_mlx, MetalCaptureProfiler.start_mps, MetalCaptureProfiler.stop, MetalTorchProfiler.start, MetalTorchProfiler.stop, apply_metal_profiler_patches, SchedulerProfilerManager._start_profile

评论区精华

review 中 yeahdongcn 提出了关键设计建议：

- 使用 is_mps() 条件代替 use_mlx() 以同时支持 MPS 和 MLX 模式，避免遗漏纯 torch MPS 用户。
- 将 Apple-specific profiling 逻辑从 common profiler 中拆出，采用与 NPU 类似的 backend patch 模式，降低主路径耦合。
- 要求处理 start_mps/start_mlx 可能抛出的 RuntimeError，确保意外失败时不传播异常。
- 要求补充单元测试，作者后新增 10 个测试。
- 建议将 ProfileManager 导入移到文件顶部，作者已调整。所有建议均被采纳并实现，最终设计得到 approval。
- MPS/MLX 后端架构选择 (design)：采纳重构意见，采用 backend patch 模式，profiler_manager.py 仅添加 3 行 elif _is_mps 分支。
- 优雅处理 Metal 捕获失败 (correctness)：异常被捕获并优雅降级，不会导致服务崩溃。
- 添加单元测试 (testing)：测试已添加并通过。
- 导入顺序优化 (style)：已调整。

风险与影响

- 风险：风险较低。主要风险包括：
 - 新增 Metal 捕获路径完全条件化 (_is_mps)，不影响现有 CUDA/XPU/NPU 后端。
 - 在非 Apple Silicon 平台上 is_mps() 为 False，补丁不会被应用。
 - Metal 捕获失败（如未设置 MTL_CAPTURE_ENABLED）时通过 try/except 优雅降级返回 success=False，不会导致服务崩溃。
 - MPS 路径依赖 torch.mps.profiler.metal_capture（需 torch >= 2.1），代码中有 hasattr 检查，兼容性较好。
 - 测试覆盖了 MLX 和 MPS 的 mock 路径，但未在真实硬件上集成测试，建议在 CI 中添加 Apple Silicon runner 以验证。
 - 影响：对 Apple Silicon 用户是纯增益：可通过 SGLANG_TORCH_PROFILER_DIR 环境变量和 --profile 参数在服务端 profiling 时获得 Metal GPU trace (.gputrace 文件)。两种模式均可：MLX 模式 (SGLANG_USE_MLX=1) 使用 mx.metal API，MPS 模式（

默认) 使用 `torch.mps.profiler` API。生成 trace 需设置 `MTL_CAPTURE_ENABLED=1` 环境变量。不影响非 Apple Silicon 平台, 性能影响仅限于 profiling 开启时的额外开销。代码设计易于后续新增其他硬件后端。

- 风险标记: 新硬件后端, 依赖 MLX 包, 异常处理优雅降级

关联脉络

- PR #22159 [MLX] Add Apple Silicon profiling support to `bench_one_batch.py`: 本 PR 扩展了 #22159 首次添加的 Apple Silicon profiling 支持到服务端 profiler。
- PR #19137 Apple Device Support Roadmap: 本 PR 是 Apple Device Support 路线图中 Profiling 部分的实现。