PR #23675 完整报告

sgl-project/sglang

perf: add --prefill-only-disable-kv-cache to skip KV pool allocation

合并时间：2026-05-12 04:10

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/23675

执行摘要

- 一句话：跳过 KV 缓存池分配，节省显存并提升吞吐
- 推荐动作：建议精读。该 PR 的设计模式（no-op pool 子类保持接口兼容）有参考价值。对于 embedding 服务用户，建议启用该标志以获得显存收益。代码结构清晰，测试完善（8 种组合），值得团队内部学习。

功能与动机

在纯 prefill 工作负载中，注意力通过 flash_attn_varlen_func 使用原始 K/V，无需读写物理 KV 缓存。但之前引擎总是分配巨大的 KV 缓冲区，造成显存浪费。该标志消除了这种浪费。

实现拆解

1. 新增 prefill_only_disable_kv_cache 布尔参数到 ServerArgs（位于 python/sglang/srt/server_args.py），并添加 CLI 标志。
2. 实现 NoOpMHATokenToKVPool 子类（位于 python/sglang/srt/mem_cache/memory_pool.py），重写 _create_buffers 分配极小占位符（形状 [page_size, head_num, head_dim]），重写 _finalize_allocation_log 报告零内存使用，重写 get_kv_size_bytes 返回零，重写 set_kv_buffer 抛出 RuntimeError 以防止误用。
3. 在 ServerArgs.__post_init__ 中调用验证方法，检查使用条件：必须设置 --is-embedding、--chunked-prefill-size=-1、--disable-radix-cache，且不兼容 context parallel、HiSparse、FP4 KV 缓存等。
4. 在 ModelRunnerKVCacheMixin._init_pools 中，根据标志选择 NoOpMHATokenToKVPool 而非 MHATokenToKVPool，并调用 _validate_prefill_only_disable_kv_cache_pool_family 拒绝不支持的池族（如 MLA、SWA、Mamba 等）。
5. 添加单元测试（test/registered/unit/server_args/test_server_args.py）验证各种合法与非法配置组合。

关键文件：

- python/sglang/srt/mem_cache/memory_pool.py（模块 内存池；类别 source；类型 core-logic；符号 NoOpMHATokenToKVPool, _create_buffers, _finalize_allocation_log, get_kv_size_bytes）：核心实现：新增 NoOpMHATokenToKVPool 子类，通过极小占位符替代 GB 级 KV 缓冲区，是跳过缓存分配的关键。

- python/sglang/srt/server_args.py（模块 参数配置；类别 source；类型 core-logic；符号 _validate_prefill_only_disable_kv_cache_args, _handle_prefill_only_disable_kv_cache）：参数定义与条件验证：新增 CLI 标志并实现两组前置检查，确保该标志仅在合法配置下使用。
- python/sglang/srt/model_executor/model_runner_kv_cache_mixin.py（模块 运行池；类别 source；类型 data-contract；符号 _validate_prefill_only_disable_kv_cache_pool_family）：池选择与运行时验证：根据标志选择 NoOpMHATokenToKVPool，并拒绝不支持的池族。
- test/registered/unit/server_args/test_server_args.py（模块 单元测试；类别 test；类型 test-coverage；符号 TestPrefillOnlyDisableKvCache, _base_kwargs, test_valid_minimal_config_constructs, test_rejects_when_not_embedding）：测试覆盖：验证 8 种合法 / 非法配置组合，确保参数验证正确。

关键符号：NoOpMHATokenToKVPool, _create_buffers, _finalize_allocation_log, get_kv_size_bytes, set_kv_buffer, _validate_prefill_only_disable_kv_cache_args, _handle_prefill_only_disable_kv_cache, _validate_prefill_only_disable_kv_cache_pool_family, test_valid_minimal_config_constructs, test_rejects_when_not_embedding, test_rejects_when_chunked_prefill_size_not_minus_one, test_rejects_when_radix_cache_enabled, test_rejects_attn_cp_size_greater_than_one, test_rejects_prefill_context_parallel

关键源码片段

python/sglang/srt/mem_cache/memory_pool.py

核心实现：新增 NoOpMHATokenToKVPool 子类，通过极小占位符替代 GB 级 KV 缓冲区，是跳过缓存分配的关键。

```
class NoOpMHATokenToKVPool(MHATokenToKVPool):
    """KV cache pool that skips physical K/V buffer allocation.

    在 embedding 模式 prefill-only 工作负载中使用 FA backend 的
    fa_skip_kv_cache 路径时，attention 通过 flash_attn_varlen_func
    使用原始 K/V，无需读写池。该类保持调度器的容量视图，
    但仅分配 (page_size, head_num, head_dim) 占位符。
    """

    def _create_buffers(self):
        # 分配极小占位符。形状为 [page_size, head_num, head_dim] 每层，
        # 使得 FA backend 顶部的 view 操作无条件成功。
        with self.memory_saver_adapter.region(GPU_MEMORY_TYPE_KV_CACHE):
            self.k_buffer = [
                torch.zeros(
                    (self.page_size, self.head_num, self.head_dim),
                    dtype=self.store_dtype,
                    device=self.device,
                )
                for _ in range(self.layer_num)
```

```
            ]
            self.v_buffer = [
                torch.zeros(
                    (self.page_size, self.head_num, self.v_head_dim),
                    dtype=self.store_dtype,
                    device=self.device,
                )
                for _ in range(self.layer_num)
            ]
        self.k_data_ptrs = torch.tensor(
            [x.data_ptr() for x in self.k_buffer],
            dtype=torch.uint64,
            device=self.device,
        )
        self.v_data_ptrs = torch.tensor(
            [x.data_ptr() for x in self.v_buffer],
            dtype=torch.uint64,
            device=self.device,
        )
        self.data_ptrs = torch.cat([self.k_data_ptrs, self.v_data_ptrs], dim=0)
        self.data_strides = torch.tensor(
            [np.prod(x.shape[1:]) * x.dtype.itemsize for x in self.k_buffer + self.v_buffer],
            device=self.device,
        )

    def _finalize_allocation_log(self, num_tokens: int):
        self.mem_usage = 0.0
        placeholder_bytes = (2 * self.layer_num * self.page_size * self.head_num
                             * max(self.head_dim, self.v_head_dim) * self.store_dtype.itemsize)
        logger.info(
            f"KV Cache skipped (no-op pool). Logical #tokens: {num_tokens}, "
            f"physical K/V size: ~{placeholder_bytes / 1024:.1f} KB placeholder"
        )

    def get_kv_size_bytes(self):
        # 报告零，使下游内存核算反映真实情况。
        return (0, 0)

    def set_kv_buffer(self, *args, **kwargs):
        raise RuntimeError(
            "NoOpMHATokenToKVPool.set_kv_buffer was called. This pool is only "
            "valid in prefill-only modes (e.g. --is-embedding, scoring) with "
            "the FA backend's fa_skip_kv_cache path active; the attention "
            "backend must never write to it."
        )
```

python/sglang/srt/server_args.py

参数定义与条件验证：新增 CLI 标志并实现两组前置检查，确保该标志仅在合法配置下使用。

```
def _validate_prefill_only_disable_kv_cache_args(self):
    """为 --prefill-only-disable-kv-cache 执行标志/前置条件约束验证。
    在 dummy-model 短路之前运行，以便及早拒绝错误配置。
    """
    if not self.prefill_only_disable_kv_cache:
        return

    # 目前限定为 embedding 模式，后续可扩展到其他 prefill-only 工作负载。
    if not self.is_embedding:
        raise ValueError(
            "--prefill-only-disable-kv-cache currently requires --is-embedding. "
            "Other prefill-only workloads may be supported in a future change."
        )
    if self.kv_cache_dtype == "fp4_e2m1":
        raise ValueError(
            "--prefill-only-disable-kv-cache does not support --kv-cache-dtype=fp4_e2m1."
        )
    # 结构前提：chunked_prefill_size == -1 且 disable_radix_cache。
    if self.chunked_prefill_size != -1:
        raise ValueError("--prefill-only-disable-kv-cache requires --chunked-prefill-size=-1.")
    if not self.disable_radix_cache:
        raise ValueError("--prefill-only-disable-kv-cache requires --disable-radix-cache.")

    # 不兼容 context parallel prefill（CP 路径会调用 set_kv_buffer）。
    if self.attn_cp_size is not None and self.attn_cp_size > 1:
        raise ValueError("--prefill-only-disable-kv-cache is incompatible with --attn-cp-size>1.")
    if self.enable_prefill_context_parallel:
        raise ValueError("--prefill-only-disable-kv-cache is incompatible with --enable-prefill-
        context-parallel.")
```

评论区精华

reviewer hzh0425 指出该标志初期不兼容 context-parallel prefill（因为 CP 路径会调用 set_kv_buffer），且仅支持 MHA 池。作者积极响应，添加了多项检查：在 ServerArgs 中拒绝 attn_cp_size > 1 和 enable_prefill_context_parallel，在 _validate_prefill_only_disable_kv_cache_pool_family 中列出不支持的池族并给出清晰错误。经过多轮迭代后，hzh0425 给予了批准。

- 不兼容 context-parallel prefill 和缺少池族保护 (design): 作者接受反馈，在 ServerArgs 中拒绝 attn_cp_size > 1 和 enable_prefill_context_parallel，并在 _init_pools 中添加 _validate_prefill_only_disable_kv_cache_pool_family 明确列出不支持的池族。
- 函数拆分与代码组织 (style): 作者将 __post_init__中的内联验证拆分为 _validate_prefill_only_disable_kv_cache_args 和 _handle_prefill_only_disable_kv_cache 两个方法。

风险与影响

- 风险：

1. 如果用户在非 prefill-only 模式下错误启用标志，set_kv_buffer 会抛出 RuntimeError，导致请求失败——这比静默数据损坏更安全，但仍可能影响服务可用性。
2. 标志目前仅支持 CUDA FA 后端（fa3/fa4），其他后端如 NPU、AMD 等未经过测试，可能在启用时启动失败。
3. 代码修改集中在核心内存池和调度逻辑，可能影响未来其他工作负载的兼容性。
4. 占位符形状依赖 page_size，若 FA 后端未来修改 view 操作，可能引发不匹配。 - 影响：对 embedding/scoring 用户是显著改进：显存释放数十 GB，吞吐提升约 6%。默认未启用，对现有用户无影响。新增标志文档化后，有利于 embedding 服务的部署成本优化。团队需要确保该标志与其他新增特性（如 NSA、MLA）的兼容性。 - 风险标记：错误启用导致请求失败，仅支持 CUDA FA 后端，依赖特定参数组合，核心内存池修改影响面广

关联脉络

- 暂无明显关联 PR