PR #35921 完整报告

sgl-project/sglang

[Fix] Read the granite sinks dtype from the exec bag, not the legacy global shim

合并时间：2026-08-22 09:13

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/35921

执行摘要

- 一句话：修复 Granite sinks dtype 读取，改用 *exec bag*
- 推荐动作：值得快速浏览，因为它是清除 legacy 全局 shim 的系列改动之一，且为后续修复 split launch dtype 问题埋下了 TODO。可以关注 granite.py 和 gpt_oss.py 的一致性，以及未来针对 split launch 的修复。

功能与动机

PR #35794 在 build_attention_sinks 中使用了 get_global_server_args() 读取 attention backend，这是已被 runtime context 替代的 legacy 全局 shim。test_legacy_global_ratchet.py 固定该 shim 的调用点数量为 1（即 shim 定义本身），但 PR #35794 使其变为 2，导致 CI 红测。本 PR 将读取方式改为从 *exec bag*（get_exec().kernel.attention_backend）获取，恢复调用点计数，并解除 CI 阻塞，且不引入任何行为变更。

实现拆解

本 PR 的实现步骤：

1. 修改导入语句：在 python/sglang/srt/models/granite.py 中，将 from sglang.srt.runtime_context import get_parallel 改为 from sglang.srt.runtime_context import get_exec, get_parallel，同时移除 from sglang.srt.server_args import get_global_server_args 导入。
2. 修改 build_attention_sinks 函数：将 attn_backend = get_global_server_args().attention_backend 改为 attn_backend = get_exec().kernel.attention_backend。同时添加了 TODO(kpham-sgl) 注释，说明当前实现无法支持 split launch（不同阶段使用不同 backend），并建议在初始化时根据 serving backend 选择 dtype。
3. 测试与验证：虽然本 PR 未新增测试，但通过现有的 test_legacy_global_ratchet.py 和 test_global_config_read_ratchet.py 等测试验证了行为不变性。作者在 PR body 中提供了两个配置下的 sinks dtype 对照表（fa3 对应 torch.bfloat16，trtllm_mha 对应 torch.float32），确认解析结果一致。另外，该改动还使 Granite 与 gpt_oss.py 中的决策逻辑保持一致，后者早已使用 get_exec().kernel.attention_backend。

关键文件：

- python/sglang/srt/models/granite.py（模块 模型层；类别 source；类型 data-contract）：核心修改文件，替换了 legacy 全局 shim 读取，并添加了 TODO 注释。

关键符号：build_attention_sinks

关键源码片段

python/sglang/srt/models/granite.py

核心修改文件，替换了 legacy 全局 shim 读取，并添加了 TODO 注释。

```
# python/sglang/srt/models/granite.py
from sglang.srt.runtime_context import get_exec, get_parallel
# 移除了 from sglang.srt.server_args import get_global_server_args

def build_attention_sinks(num_heads: int) -> nn.Parameter:
    # TODO(kpham-sgl): 单个参数无法同时服务 split launch（不同阶段不同后端）——
    # trtllm_mha 需要 float32, FA4 需要 bfloat16。
    # 应在初始化时根据 serving backend 选择 dtype，并验证 checkpoint 的 dtype。
    attn_backend = get_exec().kernel.attention_backend # 从 exec bag 读取，替代 legacy shim
    sinks_dtype = torch.float32 if attn_backend == "trtllm_mha" else torch.bfloat16
    sinks = nn.Parameter(torch.empty(num_heads, dtype=sinks_dtype), requires_grad=False)
    set_weight_attrs(sinks, {"weight_loader": sharded_weight_loader(0)})
    return sinks
```

评论区精华

本 PR 的 review 评论和讨论较少，没有独立的 review 评论线程。PR body 中作者明确记录了 TODO 和一个需要单独处理的规则问题：当前实现无法支持 split launch（如 --prefill-attention-backend fa4 --decode-attention-backend trtllm_mha），因为一个参数只能有一种 dtype，而不同后端需要不同 dtype（FA4 断言 bfloat16，trtllm_mha 需要 float32）。作者将该问题的修复推迟到后续 PR，避免在本 PR 中引入行为变更。

- 暂无高价值评论线程

风险与影响

- 风险：风险很低，因为该 PR 仅将读取路径从 get_global_server_args().attention_backend 改为 get_exec().kernel.attention_backend，两者解析的值相同。但需注意：
  - 如果 runtime context 初始化顺序不对，get_exec() 可能返回空或未初始化的值，但这种情况在模型加载时已存在，且本 PR 未改变调用时机。
  - 对于 split launch，问题依然存在（一个参数只能有一种 dtype），但这不是本 PR 引入的新风险。
  - 未新增测试，但依赖现有 ratchet 测试覆盖。
  - 影响：影响范围极小，仅涉及 Granite 模型文件 python/sglang/srt/models/granite.py，且行为无变化。主要影响是恢复 CI 绿测，并使得 Granite 与 gpt_oss.py 的 backend 读取方式一致。对用户无直接影响。
  - 风险标记：缺少独立测试，历史遗留 TODO，潜在 split launch 风险

关联脉络

- PR #35794 [Fix] Add granite SWA support: PR #35794 引入了 get_global_server_args() 读取，本 PR 正是为了修复该引入的 CI 红测问题。
- PR #34917 [Fix] Stop deriving the gpt-oss sinks dtype from the configured attention backend: 关联 issue #34917 是 gpt-oss 侧同样问题的修复，与本 PR 处理的问题属于同一类（sinks dtype 与 backend 的耦合）。