PR #26193 完整报告

sgl-project/sglang

Add a little env var for disabling Flashinfer autotune cache

合并时间：2026-05-28 14:59

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/26193

执行摘要

- 一句话：新增 FlashInfer 自调优缓存开关环境变量
- 推荐动作：推荐合并。代码简洁、逻辑清晰，无回归风险。该功能为开发者工具，默认不影响生产。可进一步考虑增加单元测试验证环境变量的行为。

功能与动机

PR 作者在开发 FlashInfer 内核时，希望看到端到端的自调优结果，需要一个简单的开关来禁用缓存，同时将结果保存到磁盘以供调试。

实现拆解

1. 在 python/sglang/srt/environ.py 中注册环境变量：在 Envs 类中添加 SGLANG_FLASHINFER_AUTOTUNE_CACHE = EnvBool(True)，默认启用。
2. 在 python/sglang/srt/model_executor/model_runner.py 中修改 _flashinfer_autotune 方法：根据环境变量判断是否复用缓存。若启用，使用标准缓存路径；若禁用，创建一个带时间戳的新路径（runs/<name>.<timestamp>.json），从而不影响原始缓存文件。
3. 在 docs_new/docs/references/environment_variables.mdx 中更新文档：添加新环境变量的说明，包括默认值、行为描述和禁用效果。

关键文件：

- python/sglang/srt/environ.py（模块 配置层；类别 source；类型 configuration）：新增环境变量 SGLANG_FLASHINFER_AUTOTUNE_CACHE 的定义，是整个变更的配置入口。
- python/sglang/srt/model_executor/model_runner.py（模块 调度器；类别 source；类型 core-logic；符号 _flashinfer_autotune）：实际使用该环境变量，控制自调优缓存的路径选择，是核心逻辑变更所在。
- docs_new/docs/references/environment_variables.mdx（模块 文档；类别 other；类型 documentation）：文档更新，为新环境变量提供说明，帮助用户理解其用途。

关键符号：_flashinfer_autotune

关键源码片段

python/sglang/srt/environ.py

新增环境变量 SGLANG_FLASHINFER_AUTOTUNE_CACHE 的定义，是整个变更的配置入口。

```
# python/sglang/srt/environ.py
# 在 Envs 类中新增一行，位于缓存目录相关变量之后
class Envs:
    # 省略其他变量 ...
    SGLANG_CACHE_DIR = EnvStr(os.path.expanduser("~/.cache/sglang"))
    # 新增：控制 FlashInfer 自调优缓存是否复用，默认 True
    SGLANG_FLASHINFER_AUTOTUNE_CACHE = EnvBool(True)
```

python/sglang/srt/model_executor/model_runner.py

实际使用该环境变量，控制自调优缓存的路径选择，是核心逻辑变更所在。

```
# python/sglang/srt/model_executor/model_runner.py
# 方法 _flashinfer_autotune 的修改片段
from flashinfer.autotuner import autotune

cache_path = self._flashinfer_autotune_cache_path()
# 根据环境变量决定是否复用已有缓存
if envs.SGLANG_FLASHINFER_AUTOTUNE_CACHE.get():
    autotune_cache = cache_path # 使用标准缓存路径
    logger.info("Running FlashInfer autotune with cache: %s", autotune_cache)
else:
    # 禁用缓存时，将新结果写入 runs/ 目录下带时间戳的文件，不覆盖原缓存
    timestamp = datetime.datetime.now().strftime("%Y%m%d_%H%M%S")
    runs_dir = cache_path.parent / "runs"
    runs_dir.mkdir(parents=True, exist_ok=True)
    autotune_cache = (
        runs_dir / f"{cache_path.stem}.{timestamp}{cache_path.suffix}"
    )
    logger.info(
        "Running FlashInfer autotune (cache reuse DISABLED via "
        "SGLANG_FLASHINFER_AUTOTUNE_CACHE=0); writing fresh result to: %s",
        autotune_cache,
    )

# 后续使用 autotune_cache 调用 autotune
self.forward_stream.wait_stream(torch.cuda.current_stream())
with torch.get_device_module(self.device).stream(self.forward_stream):
    with torch.inference_mode(), autotune(True, cache=str(autotune_cache)):
        self._dummy_run(batch_size=self.req_to_token_pool.size)
torch.cuda.current_stream().wait_stream(self.forward_stream)
logger.info("FlashInfer autotune completed.")
```

评论区精华

无实质讨论。自动化代码审查 bot 未提出具体意见，随后由合并者 Fridge003 直接批准。

- 暂无高价值评论线程

风险与影响

- 风险：风险极低。该变更仅影响 FlashInfer 自调优的缓存行为，不修改核心推理路径或数据结构。新增环境变量由 EnvBool 管理，默认值为 True，保持原有行为。唯一可能的影响是：若用户误将 SGLANG_FLASHINFER_AUTOTUNE_CACHE=0 用于生产环境，每次启动都会重新运行自调优，增加启动时间。但该变量主要用于开发调试，生产环境建议保持默认。
- 影响：影响范围：仅限于 FlashInfer 自调优流程，仅影响使用 MoE runner 后端且计算能力 >= 9.0 的 GPU 用户。影响程度：低。默认行为不变；显式禁用后，启动时自调优结果不会被缓存重用，并写入独立文件，便于调试。用户受益：FlashInfer 内核开发者可以在不污染缓存的情况下反复调优，同时保留调试记录。
- 风险标记：暂无

关联脉络

- 暂无明显关联 PR