PR #21694 完整报告

sgl-project/sglang

fix: resolve tensor file overwrite between target and draft models

合并时间：2026-04-28 14:40

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/21694

执行摘要

- 一句话：修复 Eagle 模式下 draft 与 target 张量文件路径冲突
- 推荐动作：值得合并，修复明确、改动小且有测试（虽无自动化测试但手动验证可行）。设计决策上，将角色判断放在模型 runner 层而非 hook 层是合理的，保持了 hook 接口的纯净。文档配套同步值得赞扬。

功能与动机

Issue #21721 报告了 Eagle 模式下 tensor dump 文件因 target 和 draft 模型使用相同输出路径而相互覆盖，导致部分模块输出丢失或无法正确分析。PR body 明确指出 'tensor files generated by the target and draft models share the same output paths, leading to unintended overwriting'，并引用该 issue。

实现拆解

1. 在模型运行器中识别角色并修改输出路径：在 python/sglang/srt/model_executor/model_runner.py 的 load_model 方法中，当 debug_tensor_dump_output_folder 不为 None 时，首先检查是否启用 Eagle 算法（self.spec_algorithm.is_eagle()）。若是，则根据 self.is_draft_worker 决定角色名（'draft' 或 'target'），并将最终路径拼接为 original_folder/role。然后调用 register_forward_hook_for_model 时传入新的 dump_folder，而非原来的统一路径。未启用 Eagle 时行为完全不变。
2. 更新 CLI 参数帮助文档：在 python/sglang/srt/server_args.py 的 add_cli_args 中，修改 --debug-tensor-dump-output-folder 的 help 字符串，增加说明 'In Eagle mode, tensor outputs from draft and target models are stored in separate subdirectories ("draft" and "target")'，告知用户新行为。
3. 配套文档更新：贡献者还在仓库外的文档仓库（sgl-project.github.io#26）提交了相应更新，使行为对外透明。本 PR 不涉及测试变更，因为改动仅影响文件路径，不涉及模型计算逻辑。

关键文件：

- python/sglang/srt/model_executor/model_runner.py（模块 模型执行器；类别 source；类型 core-logic）：核心修改：在 load_model 方法中添加了 Eagle 模式下输出路径分离的逻辑，根据 worker 角色选择 'draft' 或 'target' 子目录。

- python/sglang/srt/server_args.py（模块 配置参数；类别 source；类型 configuration）：更新 --debug-tensor-dump-output-folder 的帮助说明，告知用户 Eagle 模式下目录行为。

关键符号：load_model, add_cli_args

关键源码片段

python/sglang/srt/model_executor/model_runner.py

核心修改：在 load_model 方法中添加了 Eagle 模式下输出路径分离的逻辑，根据 worker 角色选择 'draft' 或 'target' 子目录。

```
# 在 load_model 方法中，当 debug_tensor_dump_output_folder 配置后
if self.server_args.debug_tensor_dump_output_folder is not None:
    dump_folder = self.server_args.debug_tensor_dump_output_folder
    # 如果是 Eagle 模式，按角色分子目录
    if self.spec_algorithm.is_eagle():
        role = "draft" if self.is_draft_worker else "target"
        dump_folder = os.path.join(dump_folder, role)
    # 统一调用 hook 注册，传入正确的路径
    register_forward_hook_for_model(
        self.model,
        dump_folder, # 替换原来的直接引用
        self.server_args.debug_tensor_dump_layers,
        self.tp_size,
        self.tp_rank,
        self.pp_rank,
    )
```

评论区精华

1. 代码简化建议：gemini-code-assist[bot] 建议将条件分支整合为先在外部确定 dump_folder，再单次调用 register_forward_hook_for_model，避免重复代码。作者采纳了该建议，最终实现正是此模式。
2. 文档补充建议：审核者 kpham-sgl 建议在 --debug-tensor-dump-output-folder 的 help 中记录 Eagle 模式下的目录行为。作者立即响应，更新了 help 文本并额外提交了网站文档 PR。两条建议均已解决并获得 approve。

- 代码简化与重复调用消除 (design): 作者采纳建议，实现为单一路径计算 + 单一调用。
- 在 CLI help 中记录行为 (documentation): 作者立即更新了 help 字符串，并额外提交了外部文档仓库的 PR。

风险与影响

- 风险：低风险。变更仅影响 --debug-tensor-dump-output-folder 选项在 Eagle 模式下的行为。主要风险是路径分离可能破坏依赖旧路径（单目录）的自动化脚本，但该选项主要用于调试，影响面小。非 Eagle 模式完全不变。不影响模型推理正确性或性能。
- 影响：影响范围：使用 --debug-tensor-dump-output-folder 且启用 Eagle 模式（如 --speculative-algorithm eagle）的用户。受益于 tensor dump 不再互相覆盖，调试更加可

靠。未使用该功能或未启用 Eagle 的用户无感知。团队可复用该模式处理其他 speculativedecoding 变体（如 EAGLE-3）的类似问题。

- 风险标记：调试路径变更，不影响模型计算

关联脉络

- 暂无明显关联 PR