PR #22054 完整报告

sgl-project/sglang

[Fix] XGrammarGrammarBackend reset to clear inherited cache

合并时间：2026-04-04 05:18

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/22054

执行摘要

修复了 XGrammarGrammarBackend.reset() 方法未调用父类 BaseGrammarBackend.reset() 的问题，确保语法缓存能完全清理，影响范围限于使用 xgrammar 后端的缓存管理，变更极小且风险低。

功能与动机

问题：BaseGrammarBackend.reset() 会清理 Python 侧的语法缓存，但 XGrammarGrammarBackend.reset() 仅清理 xgrammar 编译器缓存，导致 Scheduler.flush_cache() 在 xgrammar 路径上无法完全清除语法缓存状态。

动机：根据PRbody描述，需要确保缓存清理逻辑的完整性，避免残留缓存状态影响系统行为。

实现拆解

仅修改一个文件：python/sglang/srt/constrained/xgrammar_backend.py。

关键变更：在 reset() 方法中添加 super().reset() 调用。 defreset(self): super().reset() # 新增：调用父类缓存清理 self.grammar_compiler.clear_cache() 模块：constrained（语法约束后端）。

评论区精华

review 讨论较少，主要确认变更正确性：

- gemini-code-assist[bot]："This pull request updates the reset method in XGrammarGrammarBackend to call super().reset(), ensuring that the base class's reset logic is properly executed."
- DarkSharpness：直接批准。

Issue 评论中尝试触发 CI 测试时遇到问题：

- test_base_grammar_backend.py 未注册为 CUDA CI 测试，导致失败。
- 最终通过 /tag-and-rerun-ci 重新运行 CI。

风险与影响

风险：

- 变更极小，仅添加一行代码，逻辑清晰。
- 需确保 super().reset() 调用顺序无副作用，当前先父类后编译器缓存的顺序合理。
- 低风险，主要修正逻辑缺陷。

影响：

- 仅影响使用 XGrammarGrammarBackend 的语法缓存清理，修复后 Scheduler.flush_cache() 能完全清除缓存。
- 对用户透明，可能改善长时间运行服务的缓存管理。
- 无性能或兼容性影响。

关联脉络

- 未发现直接关联的历史 PR，但涉及语法后端缓存管理，可能与近期 bugfix PR（如 #21906、#22065）在代码健壮性方面有相似目标。
- 属于底层框架修正，确保缓存一致性，避免潜在的内存泄漏或状态错误。