

PR #28577 完整报告

sgl-project/sglang

[Test] Fold EAGLE `return\_hidden\_states` regression into spec triton suite

合并时间: 2026-06-18 07:42

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/28577>

执行摘要

- 一句话: 折叠 EAGLE return_hidden_states 回归测试到 spec triton 套件
- 推荐动作: 建议阅读 SpecHiddenStatesKit 和夹具扩展的模式, 这是一种典型的测试混合类设计, 可用于减少 CI 资源浪费同时保持测试强度。对于需要类似回归清理的测试可参考。

功能与动机

独立测试 test_eagle_hidden_states.py 启动一个专用服务器进行回归验证, 消耗额外的 CI 资源。将其折叠到现有的 spec triton 套件可以减少一次服务器启动 (约 120 秒), 同时利用现有套件的 topk>1 配置能更彻底地测试 hidden states 返回的正确性 (因为 topk>1 会触发树接受路径压缩, 而独立测试只有 topk=4)。

实现拆解

1. 新增 SpecHiddenStatesKit (python/sglang/test/kits/spec_server_kits.py): 定义一个混合类, 包含 test_return_hidden_states 方法, 向 /generate 发送带 return_hidden_states=True 的请求, 断言返回的 hidden_states 列表长度等于 completion_tokens, 并对每个解码行进行形状校验。
2. 扩展 SpecEagleServerBase 夹具 (python/sglang/test/server_fixtures/spec_eagle_fixture.py): 添加类属性 enable_return_hidden_states = False, 并在 _launch_args 方法中条件添加 --enable-return-hidden-states 启动参数。这使得子类可以仅自己启用该选项, 而共享服务器的其他请求不受影响。
3. 在 TestEagleLlama2Triton 中集成 (test/registered/spec/eagle/test_spec_eagle_triton.py): 将 SpecHiddenStatesKit 加入类的基类列表, 并设置 enable_return_hidden_states = True。因为该测试类使用 topk=8 的树模式, 能够覆盖独立测试未触及的接受路径压缩逻辑。
4. 删除独立文件 (test/registered/spec/eagle/test_eagle_hidden_states.py): 移除独立的测试类及其 CI 注册 (register_cuda_ci 标记), 消除专用的服务器启动。
5. 清理过时标签: 修正 test_spec_eagle_triton.py 的模块文档字符串, 将 "spec v1" 更新为准确的 "topk=8 tree"。

关键文件:

- test/registered/spec/eagle/test_eagle_hidden_states.py (模块 回归测试; 类别 test; 类型 deletion; 符号 TestEagleReturnHiddenStates, setUpClass, tearDownClass,

test_hidden_states_length_matches_completion)：要删除的独立测试文件，核心变更对象

- python/sglang/test/kits/spec_server_kits.py (模块 测试工具；类别 test；类型 test-coverage；符号 SpecHiddenStatesKit, test_return_hidden_states)：新增 SpecHiddenStatesKit 混合类，封装 hidden states 回归测试逻辑
- test/registered/spec/eagle/test_spec_eagle_triton.py (模块 EAGLE 测试；类别 test；类型 test-coverage；符号 TestEagleLlama2Triton)：修改测试类以混入 SpecHiddenStatesKit，启用 hidden states 服务器选项
- python/sglang/test/server_fixtures/spec_eagle_fixture.py (模块 夹具；类别 test；类型 test-coverage)：添加 enable_return_hidden_states 配置项和条件启动参数

关键符号：SpecHiddenStatesKit.test_return_hidden_states,
SpecEagleServerBase._launch_args

关键源码片段

python/sglang/test/kits/spec_server_kits.py

新增 SpecHiddenStatesKit 混合类，封装 hidden states 回归测试逻辑

```
class SpecHiddenStatesKit:
    '''return_hidden_states under spec V2 (regression for issue #26163).

    Requires the server launched with --enable-return-hidden-states
    (set ``enable_return_hidden_states = True`` on the fixture class).
    '''

    # 回归测试：验证 hidden_states 返回数量与 completion_tokens 一致
    def test_return_hidden_states(self):
        # 使用两条长度不同的 prompt，触发跨请求 stride 窗口错位（如果实现有 bug）
        prompts = [
            'Repeat: the quick brown fox the quick brown fox the quick brown fox',
            'Count down from ten: ten nine eight',
        ]
        res = requests.post(
            self.base_url + '/generate',
            json={
                'text': prompts,
                'sampling_params': {'temperature': 0, 'max_new_tokens': 32},
                'return_hidden_states': True,
            },
        )
        self.assertEqual(res.status_code, 200)
        outputs = res.json()

        for out in outputs:
            meta = out['meta_info']
            hs = meta['hidden_states']
            ct = meta['completion_tokens']
```

```

# hs[0] 是 prefill 块 (List[List[float]]) , hs[1:] 是每个解码步的行
self.assertEqual(
    len(hs),
    ct,
    f'len(hidden_states)={len(hs)} but completion_tokens={ct}',
)
decode_rows = hs[1:]
self.assertGreater(len(decode_rows), 0)
hidden_dim = len(decode_rows[0])
self.assertGreater(hidden_dim, 0)
for row in decode_rows:
    self.assertIsInstance(row, list)
    self.assertEqual(len(row), hidden_dim)

```

test/registered/spec/eagle/test_spec_eagle_triton.py

修改测试类以混入 SpecHiddenStatesKit, 启用 hidden states 服务器选项

```
'''triton attention backend (EAGLE3 topk=1 chain + EAGLE/Llama-2 topk=8 tree).
```

```
triton runs everywhere, so this stays on the cheap (5090) runner.
```

```
'''
```

```
import unittest
```

```
from sglang.srt.environ import envs
```

```
from sglang.test.ci.ci_register import register_cuda_ci
```

```
from sglang.test.kits.matched_stop_kit import MatchedStopMixin
```

```
from sglang.test.kits.spec_server_kits import (
```

```
    SpecAccuracyKit,
```

```
    SpecFeatureKit,
```

```
    SpecHiddenStatesKit, # 新增: 折叠后的 hidden states 测试套件
```

```
    SpecLogprobKit,
```

```
    SpecPenaltyKit,
```

```
)
```

```
from sglang.test.server_fixtures.spec_eagle_fixture import Eagle3Base, EagleLlama2Base
```

```
register_cuda_ci(est_time=480, stage='base-b', runner_config='1-gpu-small')
```

```
class TestEagle3Triton(
```

```
    Eagle3Base,
```

```
    MatchedStopMixin,
```

```
    SpecAccuracyKit,
```

```
    SpecLogprobKit,
```

```
    SpecPenaltyKit,
```

```
    SpecFeatureKit,
```

```
):
```

```
    '''EAGLE3 spec v2 on triton (kits listed in bases).'''
```

```
    attention_backend = 'triton'
```

```

max_running_requests = 64
cuda_graph_max_bs = 64
gsm8k_num_examples = 1000
gsm8k_check_accept_len = False
env_overrides = ((envs.SGLANG_ENABLE_STRICT_MEM_CHECK_DURING_BUSY, 1),)

class TestEagleLlama2Triton(
    EagleLlama2Base, SpecAccuracyKit, SpecFeatureKit, SpecHiddenStatesKit
):
    '''EAGLE/Llama-2 topk=8 tree on triton.

    Hosts SpecHiddenStatesKit: topk>1 exercises the tree accept-path
    compaction that the per-req hidden-state stride slicing depends on.
    '''

    attention_backend = 'triton'
    enable_return_hidden_states = True # 为此类打开 --enable-return-hidden-states
    env_overrides = ((envs.SGLANG_ENABLE_STRICT_MEM_CHECK_DURING_BUSY, 1),)

if __name__ == '__main__':
    unittest.main()

```

评论区精华

无实质 review 讨论。仅有一条自动的 `/rerun-test` 命令执行了测试，结果显示通过。

- 暂无高价值评论线程

风险与影响

- 风险：风险极低。变更仅涉及测试组织，不改变生产代码。潜在风险是独立测试删除后，如果新套件未在特定配置下运行则可能遗漏回归，但新套件在相同的 triton 后端和更高 topk 下运行，覆盖更广。另外 enable_return_hidden_states 开关使用不当可能导致测试服务器未开启 hidden_states，但 SpecHiddenStatesKit 仅在被混入且开关置 True 的类中使用，不会影响其他测试。
- 影响：对用户无影响。对测试系统减少了 CI 任务数量（移除一个 1-gpu-large 任务，节省约 120 秒执行时间）。对团队，测试代码更集中，维护成本降低。SpecHiddenStatesKit 可被其他 spec 测试套件复用。
- 风险标记：暂无

关联脉络

- PR #28496 Add standalone test_eagle_hidden_states.py: 前置 PR，添加了独立的 test_eagle_hidden_states.py 回归测试；此 PR 将其折叠进现有套件。