PR #33745 完整报告

sgl-project/sglang

[CI] Fold duplicate-server suites and prune the retract matrix on 1-gpu-5090

合并时间：2026-08-06 03:41

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/33745

执行摘要

- 一句话：折叠重复测试套件，5090 CI 减时约 580 秒
- 推荐动作：值得精读。该 PR 展示了 CI 优化的典型思路：识别重复启动服务器的测试、用 mixin 复用测试方法、矩阵削减保留边界覆盖、基于实测校准预估。对于维护大型 CI 矩阵的团队有借鉴意义。重点关注 kits/ 下的 mixin 设计和 test_retract_decode.py 的对角线削减逻辑。

功能与动机

PR body 明确说明这是 1-gpu-5090 的第二轮削减（Follows #33654），目标是减少 CI 时长：'Second round of 1-gpu-5090 trims, ~580s off base-b per commit (11167s -> ~10580s, one shard fewer).' 通过将启动相同服务器的测试合并，避免重复启动开销，并削减低价值的测试参数组合。

实现拆解

1. 校准 est_time：根据 CI 实际运行结果调整多个测试的预估时间，如 test_weight_cache_daemon.py 从 100 降至 45（TP2 类在 1-GPU 上自跳过）、test_spec_eagle_triton.py 从 480 降至 350、test_retract_decode.py 从 353 降至 215。避免过度预留时间，使调度更紧凑。
2. 修剪 retract decode 矩阵：在 test_retract_decode.py 中删除 TestRetractDecodePaged 和 TestRetractDecodeChunkCache 两个类，只保留基础的 TestRetractDecode（radix + 默认页）和 TestRetractDecodeChunkCachePaged（无 radix + page 16），构成 {radix, page 1} 与 {no-radix, page 16} 的对角线覆盖，每个 flag 的两个值仍都被覆盖，但服务器数量从四个减为两个。
3. 合并 JSON mode 测试：将 test_json_mode.py 中的 JSONModeMixin 提取到新文件 python/sglang/test/kits/json_mode_kit.py，并在 test_constrained_decoding.py 的 TestXGrammarBackend、TestOutlinesBackend、TestLLGuidanceBackend 中混入该 mixin。由于这些类已经启动了三个 grammar 后端服务器，JSON mode 测试无需再额外启动服务器，原 test_json_mode.py 被删除。
4. 折叠 ignore_eos / EBNF / Anthropic 套件：将 test_ignore_eos、test_ebnf、test_ebnf_strict_json 直接移入 test/registered/openai_server/basic/test_openai_server.py 的 TestOpenAIServer 类（xgrammar 已是默认 grammar 后端，无需额外参数）；将 test_anthropic_server.py 改造为 sglang/test/kits/anthropic_messages_kit.py::Anthropi

cMessagesMixin，混入同一个 TestOpenAIServer 类。三个原测试文件被删除。

5. 移除调试打印：在 test_openai_server.py 的 run_completion_stream 中删除每个流式 chunk 的 print(f"{response=}")（PR #17471 引入，日志噪声大），并在 test_skip_tokenizer_init.py 中删除逐 SSE 行打印。这些信息已由每个参数组合的头打印和断言消息携带。
6. 校准 est_time：在最后一次提交中，根据 CI 结果将 test_constrained_decoding.py 的 est_time 从 120 调至 135，test_retract_decode.py 从 353 调至 215。

关键文件：

- test/registered/openai_server/basic/test_openai_server.py（模块 OpenAI 服务；类别 test；类型 test-coverage；符号 TestOpenAIServer, test_ignore_eos, test_ebnf, test_ebnf_strict_json）：作为核心接纳类，吸收了 ignore_eos、EBNF 和 Anthropic messages 三套测试，并移除了调试打印，是本次折叠的主要载体。
- python/sglang/test/kits/anthropic_messages_kit.py（模块 Anthropic 消息；类别 test；类型 rename-or-move；符号 AnthropicMessagesMixin, anthropic_base_url, messages_url, _make_request）：由原 test_anthropic_server.py 重命名并改造为 mixin，使得 Anthropic /v1/messages 测试可在任意共有 base_url/api_key/model 的宿主类上复用。
- python/sglang/test/kits/json_mode_kit.py（模块 JSON 模式；类别 test；类型 test-coverage；符号 JSONModeMixin, test_json_mode_response, test_json_mode_with_streaming）：新增的 JSON mode 测试 mixin，供 constrained decoding 测试类复用，消除了独立的 JSON mode 测试服务器启动。
- test/registered/constrained_decoding/test_constrained_decoding.py（模块 约束解码；类别 test；类型 test-coverage；符号 ServerWithGrammar, TestXGrammarBackend, TestOutlinesBackend, TestLLGuidanceBackend）：混入 JSONModeMixin，使三个 grammar 后端类同时覆盖 JSON mode 测试，并增加 AMD 特定参数，是折叠的关键环节。
- test/registered/scheduler/test_retract_decode.py（模块 解码回退；类别 test；类型 test-coverage；符号 TestRetractDecode, TestRetractDecodeChunkCachePaged）：缩减 retract decode 测试矩阵，从四个组合减为两个对角线组合，并校准 est_time，直接影响 CI 分片数。

关键符号：test_ignore_eos, test_ebnf, test_ebnf_strict_json, JSONModeMixin, AnthropicMessagesMixin, run_completion_stream, anthropic_base_url, messages_url

关键源码片段

test/registered/openai_server/basic/test_openai_server.py

作为核心接纳类，吸收了 ignore_eos、EBNF 和 Anthropic messages 三套测试，并移除了调试打印，是本次折叠的主要载体。

```
# test_ignore_eos 验证 ignore_eos=True 时生成会超过 EOS 并达到 max_tokens 上限。
def test_ignore_eos(self):
    client = openai.Client(api_key=self.api_key, base_url=self.base_url)

    max_tokens = 200 # 目标长度：正常模型会在 EOS 处提前结束
```

```
    # 对照组：ignore_eos=False，遇到 EOS 即停
    response_default = client.chat.completions.create(
        model=self.model,
        messages=[
            {"role": "system", "content": "You are a helpful assistant."},
            {"role": "user", "content": "Count from 1 to 20."},
        ],
        temperature=0,
        max_tokens=max_tokens,
        extra_body={"ignore_eos": False},
    )

    # 实验组：ignore_eos=True，应持续生成直到 max_tokens
    response_ignore_eos = client.chat.completions.create(
        model=self.model,
        messages=[
            {"role": "system", "content": "You are a helpful assistant."},
            {"role": "user", "content": "Count from 1 to 20."},
        ],
        temperature=0,
        max_tokens=max_tokens,
        extra_body={"ignore_eos": True},
    )

    # 比较两组 token 数：ignore_eos 要么多于默认，要么达到 max_tokens 上限
    default_tokens = len(self.tokenizer.encode(response_default.choices[0].message.content))
    ignore_eos_tokens = len(self.tokenizer.encode(response_ignore_eos.choices[0].message.
    content))
    self.assertTrue(
        ignore_eos_tokens > default_tokens or ignore_eos_tokens >= max_tokens,
        f"ignore_eos did not generate more tokens: {ignore_eos_tokens} vs {default_tokens}",
    )

    # 断言事件原因：ignore_eos 模式应因 length 终止
    self.assertEqual(
        response_ignore_eos.choices[0].finish_reason,
        "length",
        f"Expected finish_reason='length' for ignore_eos=True, got {response_ignore_eos.
        choices[0].finish_reason}",
    )
```

python/sglang/test/kits/anthropic_messages_kit.py

由原 test_anthropic_server.py 重命名并改造为 mixin，使得 Anthropic /v1/messages 测试可在任意共有 base_url/api_key/model 的宿主类上复用。

```
# AnthropicMessagesMixin 提供 /v1/messages 接口的测试方法。
# 宿主测试类只需提供 self.base_url、self.api_key 和 self.model 即可复用。
class AnthropicMessagesMixin:
    @property
```

```
    def anthropic_base_url(self):
        # 去掉可能已追加的 /v1 后缀，保证与 SDK 拼接时不重复
        base = self.base_url
        return base[: -len("/v1")] if base.endswith("/v1") else base

    @property
    def messages_url(self):
        # 统一构造 /v1/messages 端点
        return self.anthropic_base_url + "/v1/messages"

    def _make_request(self, payload, stream=False):
        headers = {
            "Content-Type": "application/json",
            "Authorization": f"Bearer {self.api_key}",
        }
        return requests.post(
            self.messages_url,
            headers=headers,
            json=payload,
            stream=stream,
        )
```

评论区精华

该 PR 没有 review 评论，仅作者在 Issue 中执行了 /rerun-test 命令并确认 3 个测试（test_openai_server.py、test_constrained_decoding.py、test_retract_decode.py）在 1-gpu-5090 上通过。没有实质性的技术讨论。

- 暂无高价值评论线程

风险与影响

- 风险：
  1. 测试覆盖调整风险：retract decode 从 2x2 矩阵缩减为对角线，虽然每个 flag 的两个值均被覆盖，但缺失了 {radix, page 16} 和 {no-radix, page 1} 组合，可能遗漏这两个组合下的交互问题，如 radix cache 与 page size 的兼容性。
  2. est_time 校准风险：基于单次 CI 运行校准，若硬件或模型波动，实际耗时可能超出预估，导致超时或排队不合理。尤其 test_weight_cache_daemon.py 从 100 降至 45，但该文件在 1-GPU runner 上仅运行 TP1 smoke，实际耗时可能接近 45s，余量较小。
  3. mixin 契约风险：AnthropicMessagesMixin 要求宿主类提供 base_url、api_key、model，并通过 anthropic_base_url 属性去除 /v1 后缀。若未来宿主类的 base_url 格式变化（如已无 /v1 或带其他路径），可能导致 URL 拼接错误。
  4. 日志打印移除：run_completion_stream 中的 print(f"{response=}") 被移除，可能降低流式响应异常的定位能力，但断言消息已包含关键上下文，风险可控。 - 影响：CI 效率：base-b 阶段耗时减少约 580 秒（约 5.2%），分片数减少一个，显著提升 CI 反馈速度，对高频测试迭代的开发流程收益明显。测试结构：引入 sglang/test/kits/ 下的

mixin 复用模式（JSONModeMixin、AnthropicMessagesMixin），未来可推广到更多共享服务器的测试，减少重复代码和启动开销。开发者体验：相关测试类路径和运行命令发生变化（如 TestAnthropicServer 不再独立存在），开发者在本地运行或调试时需使用新的类名（如 TestOpenAIServer），文档中的示例命令可能需要同步更新。跨平台影响：多个文件同时调整了 AMD CI 的 est_time（如 test_openai_server.py 从 200 升至 280），会影响 AMD 测试调度，需关注 AMD 侧是否有超时或排队问题。

- 风险标记：测试组合削减，est_time 校准依赖实测，日志打印移除可能影响调试

关联脉络

- PR #33654 [CI] Move CPU-only unit tests to the CPU suite and trim dead 5090 registrations: 这是本 PR 的前一轮 5090 CI 削减，本 PR 明确标注 Follows #33654，延续其合并与修剪策略。
- PR #17471 Add debug print to completion stream test: 本 PR 移除了该 PR 中为调试 flake 加入的逐 token print，属于对该变更的回收。