PR #24552 完整报告

sgl-project/sglang

[Gemma4] Add test for MTP models

合并时间：2026-05-28 12:36

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/24552

执行摘要

- 一句话：为 Gemma4 MTP 模型添加 GSM8K 端到端测试
- 推荐动作：值得精读测试设计模式，了解 SGLang 中 speculative decoding 端到端测试的编写方法（服务器启动、配置验证、评估执行、断言输出）。可作为新模型 MTP 测试的模板。关注 CI 注册和阶段命名规则，避免后续维护成本。

功能与动机

本 PR 为 PR #24436 添加测试覆盖，使用 GSM8K 数据集验证 Gemma4 26B-A4B、31B 和 Frozen-KV E4B 三种 MTP 模型在 speculative decoding 下的准确率和接受长度。测试注册为 CI extra-stage，确保 PR 触及 Gemma4 模型路径时自动触发验证。

实现拆解

1. 创建测试文件：新建三个 Python 文件置于 test/registered/spec/ 目录，分别对应 26B-A4B、31B 和 Frozen-KV E4B 模型。每个文件定义测试类（继承 CustomTestCase）并通过 register_cuda_ci 注册到 CI 阶段（extra-a 或 base-b）。
2. 服务器启动辅助：每个测试类实现 _server_env（设置环境变量，如禁用 speculative V2 或启用异步断言）、_common_server_args（固定参数：attention-backend=triton、dtype=bfloat16、context-length=2048 等）、_server_args（根据 topk 动态添加 speculative 参数：算法 NEXTN、draft 模型路径、步数、topk、draft token 数）。
3. 核心测试流程 _run_gsm8k_mtp：启动 SGLang 服务器 → 调用 /flush_cache 清空缓存 → 查询 /server_info 验证配置（speculative_eagle_topk、disable_cuda_graph）→ 运行 GSM8K 评估（200 样本，5-shot）→ 获取准确率和平均推测接受长度 → 清理进程。
4. 断言与报告：具体测试方法 test_gsm8k_topk1 和 test_gsm8k_topk3 调用 _run_gsm8k_mtp 并断言准确率超过阈值（26B-A4B: 0.42, 31B: 0.775, E4B: 0.65）且平均接受长度 ≥ 1.5。在 CI 环境下输出 GitHub Step Summary。
5. CI 适配调整：通过多次提交调整注册参数：从 nightly 迁移到 extra-a、修正 stage 命名（stage-b → base-b）、提高接受长度阈值（0.0 → 1.5）、增加异步断言环境变量，确保与最新 CI 框架兼容。

关键文件：

- test/registered/spec/test_gemma4_mtp_26b_a4b_extra.py（模块 集成测试；类别 test；类型 test-coverage；符号 get_server_info, get_avg_spec_accept_length,

TestGemma4MTP26BA4B, _server_env）：Gemma4 26B-A4B MTP 模型的端到端测试，验证 topk=1 和 topk=3 下的 GSM8K 准确率和接受长度。

- test/registered/spec/test_gemma4_mtp_31b_extra.py（模块 集成测试；类别 test；类型 test-coverage；符号 get_server_info, get_avg_spec_accept_length, TestGemma4MTP31B, _server_env）：Gemma4 31B MTP 模型的端到端测试，验证 topk=1 和 topk=3 下的 GSM8K 准确率和接受长度。
- test/registered/spec/test_frozen_kv_mtp.py（模块 集成测试；类别 test；类型 test-coverage；符号 get_server_info, get_avg_spec_accept_length, TestFrozenKVMTP, _server_env）：Frozen-KV E4B MTP 模型的端到端测试，验证 topk=1 和 topk=3 下的 GSM8K 准确率和接受长度，包含更完整的断言和 CI 报告输出。

关键符号：get_server_info, get_avg_spec_accept_length, _run_gsm8k_mtp, test_gsm8k_topk1, test_gsm8k_topk3

关键源码片段

test/registered/spec/test_frozen_kv_mtp.py

Frozen-KV E4B MTP 模型的端到端测试，验证 topk=1 和 topk=3 下的 GSM8K 准确率和接受长度，包含更完整的断言和 CI 报告输出。

```
class TestFrozenKVMTP(CustomTestCase):
    # 辅助方法省略 ...

    def _run_gsm8k_mtp(self, topk: int) -> None:
        """启动服务器并运行 GSM8K 评估，验证准确率和接受长度。"""
        process = None
        try:
            # 使用 popen_launch_server 启动目标模型服务器
            process = popen_launch_server(
                "google/gemma-4-E4B-it", # 目标模型
                self.base_url,
                timeout=DEFAULT_TIMEOUT_FOR_SERVER_LAUNCH * 3,
                env=self._server_env(),
                other_args=self._server_args(topk),
            )
            # 清空缓存，确保起始状态
            requests.get(self.base_url + "/flush_cache", timeout=30)

            # 获取服务器信息，验证配置正确
            server_info = get_server_info(self.base_url)
            self.assertEqual(
                server_info.get("speculative_eagle_topk"),
                topk,
                f"E4B: server did not start with topk={topk}",
            )
            self.assertFalse(
                bool(server_info.get("disable_cuda_graph")),
```

```
                f"E4B/topk{topk}: CUDA graph is disabled",
            )

            # 运行 GSM8K 评估
            metrics = run_eval(self._gsm8k_args())
            mtp_score = float(metrics["score"])
            avg_accept = get_avg_spec_accept_length(self.base_url)
        finally:
            if process is not None:
                self._stop_process(process)

        # 输出结果到日志和 CI 摘要
        print(
            f"[Frozen-KV MTP E4B topk={topk}] "
            f"score={mtp_score:.4f} threshold={0.65:.4f} "
            f"avg_spec_accept_length={avg_accept}"
        )
        if is_in_ci():
            write_github_step_summary(
                f"### Frozen-KV MTP E4B topk={topk}\n"
                f"score={mtp_score:.4f}\n"
                f"threshold={0.65:.4f}\n"
                f"avg_spec_accept_length={avg_accept}\n"
            )

        # 断言准确率和接受长度
        self.assertGreaterEqual(mtp_score, 0.65)
        self.assertIsNotNone(avg_accept)
        self.assertGreaterEqual(
            avg_accept,
            1.5,
            f"E4B/topk{topk}: accept length too low",
        )

    def test_gsm8k_topk1(self) -> None:
        """topk=1 时的 MTP 测试。"""
        self._run_gsm8k_mtp(topk=1)

    def test_gsm8k_topk3(self) -> None:
        """topk=3 时的 MTP 测试。"""
        self._run_gsm8k_mtp(topk=3)
```

评论区精华

在 review 中，开发者对平均接受长度阈值提出了修改意见（原 0.0 过于宽松），作者在 commit 3144ab6 中将其提升至 1.5，并移除了无用的 setUpClass 方法。此外，Frozen-KV 测试增加了 SGLANG_ENABLE_ASYNC_ASSERT 环境变量以在 NaN/Inf/OOB 时快速失败。

- 平均接受长度阈值设定 (correctness): 作者将阈值从 0.0 提高到 1.5，并移除了无用的 setUpClass 方法（commit 3144ab6）。
- 启用异步断言 (testing): 在 test_frozen_kv_mtp.py 的环境变量中增加了 SGLANG_ENABLE_ASYNC_ASSERT=1（commit 1c806f2）。

风险与影响

- 风险：
  - 测试依赖 Google 发布的模型权重（google/gemma-4-*）和 assistant 模型，若 HuggingFace 仓库不可用或路径变更，测试将失败。
  - 每个测试耗时长达 720 秒（26B-A4B、31B）或 300 秒（E4B），占用 2 卡或 1 卡 GPU，可能增加 CI 排队时间和资源压力。
  - 准确率阈值基于当前观察值设定，模型微调或数值精度变化可能导致偶发假阳性失败，需要定期校准。
- 影响：
  - 对用户：无直接影响，纯测试变更。
  - 对系统：新增三个 CI 测试，增强对 Gemma4 MTP 功能的回归防护，确保 speculative decoding 质量。
  - 对团队：测试注册在 extra-a 和 base-b 阶段，需通过 run-ci-extra 标签触发，增加 CI 配置复杂度。
  - 风险标记：依赖外部模型权重，长耗时测试，CI 资源消耗

关联脉络

- PR #24436 [Gemma4] Add MTP model support: 本 PR 为此 PR #24436 添加测试覆盖，验证 Gemma4 MTP 功能正确性。
- PR #25197 CI stage registration refactor: 提交消息中提及其 stage/runner_config 注册参数对齐了 #25197 的变更。
- PR #26335 Async assert env consolidation: 提交消息提及将 SGLANG_ENABLE_ASYNC_ASSERT 引入测试，参考了 #26335 的整合。