PR #49974 完整报告

vllm-project/vllm

[Test] dynamic_shapes_compilation

合并时间：2026-07-29 12:54

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/49974

执行摘要

- 一句话：测试改用 VllmRunner 确保引擎正确关闭
- 推荐动作：值得合并。该变更是对测试基础设施的改进，模式清晰，可推广至其他类似测试中。建议在后续 PR 中继续迁移剩余使用 LLM 手动管理的测试代码。

功能与动机

原测试在每次使用 LLM 后需要手动调用 del、gc.collect、empty_cache、synchronize 等步骤，且不同模型之间可能存在 GPU 内存竞争，导致测试不稳定。使用 VllmRunner 上下文管理器可在退出时自动清理引擎状态，确保测试隔离。PR body 明确说明 'replace LLM() with VllmRunner() for a proper shutdown along the testing'。

实现拆解

1. 修改测试函数使用 vllm_runner fixture：在 test_dynamic_shapes_compilation 和 test_piecewise_backend_empty_sym_shape_indices 中，将原有的 LLM(model=...) 直接构造替换为 with vllm_runner(...) as vllm_model: 上下文管理器。生成逻辑移入 with 块，退出上下文时自动触发引擎关闭和资源释放。
2. 移除手动内存清理代码：删除 del model、gc.collect()、torch.accelerator.empty_cache()、synchronize() 以及 wait_for_rocm_memory_to_settle() 等不再需要的清理步骤，这些现在由 VllmRunner.__exit__ 统一处理。
3. 更新导入语句：移除 gc 和 wait_for_rocm_memory_to_settle 的导入；将 from vllm import LLM, SamplingParams 简化为 from vllm import SamplingParams，因为 LLM 不再需要。
4. 添加兼容性配置：为 vllm_runner 添加 enable_chunked_prefill=None 参数，以符合 VllmRunner 的默认配置要求，确保测试行为一致。

关键文件：

- tests/compile/test_dynamic_shapes_compilation.py（模块 编译测试；类别 test；类型 test-coverage；符号 test_dynamic_shapes_compilation, test_piecewise_backend_empty_sym_shape_indices）：本次变更的唯一文件，将测试中直接的 LLM 创建替换为 VllmRunner 上下文管理器，确保引擎正确关闭并移除手动清理代码。

关键符号：test_dynamic_shapes_compilation, test_piecewise_backend_empty_sym_shape_indices

关键源码片段

tests/compile/test_dynamic_shapes_compilation.py

本次变更的唯一文件，将测试中直接的 LLM 创建替换为 VllmRunner 上下文管理器，确保引擎正确关闭并移除手动清理代码。

```
# tests/compile/test_dynamic_shapes_compilation.py

@pytest.mark.parametrize('model_name', get_test_models())
@pytest.mark.parametrize(
    'shapes_type',
    [
        DynamicShapesType.BACKED,
        DynamicShapesType.UNBACKED,
        DynamicShapesType.BACKED_SIZE_OBLIVIOUS,
    ],
)
@pytest.mark.parametrize('use_aot_compile', ['0', '1'])
@pytest.mark.parametrize('use_bytecode_hook', [True, False])
@pytest.mark.parametrize('evaluate_guards', [False, True])
@pytest.mark.skipif(not is_torch_equal_or_newer('2.10.0'), reason='requires torch 2.10')
def test_dynamic_shapes_compilation(
    monkeypatch,
    vllm_runner, # VllmRunner fixture 确保上下文退出时 shutdown
    model_name,
    shapes_type,
    use_aot_compile,
    use_bytecode_hook,
    evaluate_guards,
):
    '''Test that all dynamic shapes types compile successfully'''
    if shapes_type == DynamicShapesType.UNBACKED and not is_torch_equal_or_newer('2.11.0')
    :
        pytest.skip('unbacked dynamic shapes with shape_id require torch>=2.11')
    if evaluate_guards and shapes_type == DynamicShapesType.UNBACKED:
        pytest.skip('unbacked dynamic shapes do not add guards')
    if evaluate_guards and use_aot_compile:
        pytest.skip('evaluate_guards requires use_aot_compile=0')

    monkeypatch.setenv('VLLM_USE_AOT_COMPILE', use_aot_compile)
    monkeypatch.setenv('VLLM_USE_BYTECODE_HOOK', '1' if use_bytecode_hook else '0')

    prompt = 'Hello, my name is'
    sampling_params = SamplingParams(max_tokens=5, temperature=0, logprobs=10)
    test_prompts = [prompt, 'The capital of France is']
```

```
# 使用 VllmRunner 管理编译模型生命周期
with vllm_runner(
    model_name,
    compilation_config={
        'mode': CompilationMode.VLLM_COMPILE,
        'dynamic_shapes_config': {
            'type': shapes_type.value,
            'evaluate_guards': evaluate_guards,
        },
    },
    max_model_len=1024,
    enable_chunked_prefill=None, # 显式设置与 VllmRunner 默认一致
) as vllm_model:
    compiled_outputs = []
    for p in test_prompts:
        output = vllm_model.llm.generate(p, sampling_params)[0].outputs[0]
        assert len(output.text.strip()) > 0, 'Compiled model produced empty output'
        compiled_outputs.append((output.token_ids, output.text, output.logprobs))

# 第二个 VllmRunner 以 eager 模式运行对照
with vllm_runner(
    model_name,
    enforce_eager=True,
    max_model_len=1024,
    enable_chunked_prefill=None,
) as vllm_model:
    eager_outputs = []
    for p in test_prompts:
        output = vllm_model.llm.generate(p, sampling_params)[0].outputs[0]
        assert len(output.text.strip()) > 0, 'Eager model produced empty output'
        eager_outputs.append((output.token_ids, output.text, output.logprobs))

# 验证 eager 与 compiled 输出的 logprobs 一致性
check_logprobs_close(
    outputs_0_lst=eager_outputs,
    outputs_1_lst=compiled_outputs,
    name_0='eager',
    name_1='compiled',
)
```

评论区精华

无实质性讨论。Copilot 代码审查机器人自动总结了变更内容，维护者 Isotr0py 直接批准。

- 暂无高价值评论线程

风险与影响

- 风险：极低风险。变更仅涉及测试代码，不改变任何生产逻辑。新代码在 CI 中已通过全部测试用例（包括 GPT2 等模型），验证了行为一致性。唯一需关注的是 *enable_chunked_prefill=None* 的显式设置，但该值已是 VllmRunner 的默认行为，不会引入功能差异。
- 影响：仅影响一个测试文件，消除了因引擎未正确关闭导致的 GPU 内存残留和测试不稳定问题。提升了编译测试套件的隔离性和可靠性。对用户无影响，对开发者在本地运行测试时更加稳定。
- 风险标记：测试稳定性，资源管理

关联脉络

- 暂无明显关联 PR