

PR #49372 完整报告

vllm-project/vllm

[Bugfix] Respect declared attention contract for ColQwen3.5 retrievers

合并时间: 2026-07-26 12:08

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/49372>

执行摘要

- 一句话: 让 ColQwen3.5 checkpoint 的注意力契约生效
- 推荐动作: 值得精读。该 PR 展示了如何从硬编码假设转向显式数据契约, 设计优雅且兼容多种训练范式。推荐关注 `verify_and_update_model_config` 的实现以及测试中对 `AttentionType` 的验证。

功能与动机

PR #46108 强制了所有 ColQwen3.5 pooling 模型使用双向注意力, 但 VultronRetriever 检查点是用因果注意力训练和评估的, 导致语义改变。HuggingFace 讨论 (<https://huggingface.co/vultr/VultronRetrieverFlash-Qwen3.5-0.8B/discussions/1>) 报告了该问题。必须尊重检查点的注意力契约。

实现拆解

1. 数据契约声明: ColQwen3.5 检查点的 Hugging Face 配置中新增 `retrieval_attention_contract` 字段, 取值 "causal" 或 "bidirectional"。
2. 配置校验: 在 `ColQwen3_5Config.verify_and_update_model_config()` 中, 从 `hf_config` 和 `hf_text_config` 收集所有声明, 检查是否缺失、类型非法、值超范围或声明冲突, 任一情况抛出 `ValueError`。
3. 注意力模式应用: 根据声明值计算 `is_causal` ("causal" → True, "bidirectional" → False), 并同时设置到 `hf_config` 和 `text_config` 的 `is_causal` 属性。
4. 编码器层保持: 双向声明继续产生 `AttentionType.ENCODER_ONLY` 层, 因果声明恢复解码器注意力, 均无 KV 缓存规格 (保持 #46108 引入的行为)。
5. 测试配套: 新增 5 个测试用例验证契约应用、拒绝非法值、双向构建编码器层以及无 KV 缓存规格。

关键文件:

- `vllm/model_executor/models/config.py` (模块 模型配置; 类别 source; 类型 core-logic; 符号 `ColQwen3_5Config`, `verify_and_update_model_config`): 核心变更: 修改 `ColQwen3_5Config` 根据 checkpoint 声明的 `retrieval_attention_contract` 设置注意力类型, 替代硬编码的双向注意力。
- `tests/models/multimodal/pooling/test_colqwen3_5.py` (模块 模型测试; 类别 test; 类型 test-coverage; 符号 `test_colqwen3_5_config_applies_declared_attention_contract`,

test_colqwen3_5_config_rejects_invalid_attention_contract, test_colqwen3_5_bidirectional_contract_builds_encoder_only_attention, test_colqwen3_5_encoder_only_attention_has_no_kv_cache_spec) : 新增 5 个测试用例覆盖注意力契约配置应用、错误拒绝、构建验证、KV 缓存规格检查。

关键符号: ColQwen3_5Config.verify_and_update_model_config

关键源码片段

vllm/model_executor/models/config.py

核心变更: 修改 ColQwen3_5Config 根据 checkpoint 声明的 retrieval_attention_contract 设置注意力类型, 替代硬编码的双向注意力。

文件: vllm/model_executor/models/config.py

```
class ColQwen3_5Config(Qwen3_5ForConditionalGenerationConfig):
    """Apply the attention contract declared by a ColQwen3.5 checkpoint."""

    @staticmethod
    def verify_and_update_model_config(model_config: "ModelConfig") -> None:
        # 从 hf_config 和 hf_text_config 中收集所有声明的契约值
        configs = {
            id(config): config
            for config in (
                model_config.hf_config,
                model_config.hf_text_config,
            )
        }
        declarations = [
            contract
            for config in configs.values()
            if (contract := getattr(config, "retrieval_attention_contract", None))
            is not None
        ]
        supported = {"causal", "bidirectional"}
        # 拒绝缺失契约
        if not declarations:
            raise ValueError(
                "ColQwen3.5 checkpoints must declare "
                "retrieval_attention_contract as 'causal' or 'bidirectional'"
            )
        # 拒绝类型错误、值不支持或声明冲突
        if (
            any(not isinstance(contract, str) for contract in declarations)
            or any(contract not in supported for contract in declarations)
            or len(set(declarations)) != 1
        ):
            raise ValueError(
                "unsupported or conflicting ColQwen3.5 "
```

```

        f"retrieval_attention_contract declarations: {declarations!r}"
    )
    # 计算 is_causal 并应用到所有 config 对象
    is_causal = declarations[0] == "causal"
    for config in configs.values():
        config.is_causal = is_causal

```

tests/models/multimodal/pooling/test_colqwen3_5.py

新增 5 个测试用例覆盖注意力契约配置应用、错误拒绝、构建验证、KV 缓存规格检查。

文件 : tests/models/multimodal/pooling/test_colqwen3_5.py

```

@pytest.mark.parametrize(
    ("contract", "expected_is_causal"),
    [("causal", True), ("bidirectional", False)],
)
def test_colqwen3_5_config_applies_declared_attention_contract(
    contract: str,
    expected_is_causal: bool,
) -> None:
    """验证契约声明正确设置 is_causal 到 hf_config 和 text_config。"""
    from vllm.model_executor.models.config import (
        MODELS_CONFIG_MAP,
        ColQwen3_5Config,
    )
    assert MODELS_CONFIG_MAP["ColQwen3_5"] is ColQwen3_5Config

    hf_config = SimpleNamespace(retrieval_attention_contract=contract)
    text_config = SimpleNamespace()
    model_config = SimpleNamespace(
        hf_config=hf_config,
        hf_text_config=text_config,
    )
    ColQwen3_5Config.verify_and_update_model_config(model_config)

    # 契约影响两个 config 对象
    assert hf_config.is_causal is expected_is_causal
    assert text_config.is_causal is expected_is_causal

```

评论区精华

作者在 PR Body 和 Issue 评论中提及 HuggingFace 讨论报告的问题，并指出 CI 失败与 PR 无关。维护者 noooop 直接批准，无额外讨论。

- VultronRetriever issue discussion reference (question): PR 通过实现签名契约机制解决了该问题，维护者已批准。

风险与影响

- 风险：低风险。核心逻辑在配置校验阶段，不会影响模型推理性能。风险点包括：
 - 新字段缺失拒绝：未声明 `retrieval_attention_contract` 的老检查点会被拒绝加载，需要更新元数据（作者声称已更新相关检查点）。
 - 仅影响 ColQwen3.5：变更仅作用于一个模型类，其他模型不受影响。
 - 测试覆盖充分：新增 5 个单元测试覆盖正常 / 异常路径。
- 影响：影响范围中等。仅影响 ColQwen3.5 检索模型用户：
 - 对于声明 `causal` 的检查点（如 `VultronRetriever`），修复后行为正确（因果注意力），否则之前是双向导致结果错误。
 - 对于声明 `bidirectional` 的检查点（如 `athrael-soju/colqwen3.5-4.5B-v3`），行为保持不变。
 - 未声明契约的检查点会立即报错，引导用户更新。
 - 部署无需配置变更，但用户可能需要在旧检查点添加字段。
 - 风险标记：数据契约校验，向后兼容性变化

关联脉络

- PR #46108 [Feature] Make ColQwen3.5 pooling models bidirectional: 引入了无条件双向注意力，当前 PR 修复了其导致的因果检查点回归。