

PR #2759 完整报告

radixark/miles

feat(tito): support Qwen3.5 and Qwen3.6 templates

合并时间: 2026-08-27 09:00

原文链接: <http://prhub.com.cn/radixark/miles/pull/2759>

执行摘要

- 一句话: TITO 新增 Qwen3.6 模板, 拆分 Qwen3.5/3.6 序列化合约
- 推荐动作: 值得精读。核心看点在于 "按家族拆分模板而非共享模板" 的设计决策: 当两个模型家族的渲染合约存在细微序列化差异时, 共享模板会静默改变旧家族行为, 拆分并配以参数化测试锁定差异是稳健做法。建议关注 `tito_tokenizer.py` 中 `FIXED_TEMPLATE` 注册模式、`preserve_thinking` 与 `clear_thinking` 的语义区别, 以及 e2e 测试如何用 `tito_session_mismatch_rate` 指标做 CI gate。合并时注意 reviewer 提到的与 PR 2711 的兼容性提醒。

功能与动机

Qwen3.5 和 Qwen3.6 使用不同的原生工具参数序列化, 而 Miles 此前只有 Qwen3.5 注册, 且其 `thinking-preservation` 行为与新版 Qwen 合约不一致。PR body 明确指出: 共享一个模板会静默改变 Qwen3.5 的 `boolean` 和 `null` 工具调用 `token`; 分开模板才能让每个家族保留自己的模型合约, 同时都通过 `preserve_thinking` 保留先前的 `reasoning`。

实现拆解

1. 注册家族类与枚举: 在 `miles/utils/chat_template_utils/tito_tokenizer.py` 中新增 `Qwen36TITOTokenizer`, 继承 `Qwen3TITOTokenizer` 以复用 `<lim_endl>` 换行边界合并逻辑; `Qwen35TITOTokenizer` 的 `extra_kwargs` 从 `clear_thinking=False` 调整为 `preserve_thinking=True`; `TITOTokenizerType` 增加 `QWEN36` 枚举, `get_tokenizer_class` 工厂方法增加 `case cls.QWEN36` 分支。
2. 模板拆分: 新增 `miles/utils/chat_template_utils/templates/qwen3.6_fixed.jinja` (151 行), 基于 HF 上 Qwen3.6-35B-A3B revision 995ad96 的 `tokenizer_config`, 并移除 Miles 合成增量渲染前缀所需的 `no-user-query guard`; `qwen3.5_fixed.jinja` 仅做单行调整, 保持其标量风格序列化 (`boolean` 为 `False`、`null` 为 `None`)。
3. 参数与渲染联动: `--tito-model qwen35|qwen36` 通过 `resolve_fixed_chat_template` 自动解析到各自模板与 `preserve_thinking=True`; `miles_validate_args` 的既有校验 (拒绝自定义模板覆盖、拒绝冲突 `kwargs`) 无需改动即可覆盖新家族。
4. 测试配套: `tests/fast/utils/chat_template_utils/test_fixed_templates.py` 新增参数化测试锁定两种序列化差异 (Qwen3.5 输出 `False/None`, Qwen3.6 输出 `false/null`); `tests/fast/utils/test_arguments.py` 验证家族解析; 新增 e2e 测试 `tests/e2e/sglang/test_session_server_multi_role/test_qwen36.py` 及

test_session_pretokenized_e2e.py 的 smoke 配置；其余 fast 测试同步适配新枚举。

5. 文档：docs/user-guide/agent-ic-rollout.md 补充 --tito-model qwen35/qwen36 用法说明。

关键文件：

- miles/utils/chat_template_utils/tito_tokenizer.py (模块 模板引擎；类别 source；类型 core-logic；符号 Qwen36TITOTokenizer, Qwen35TITOTokenizer, TITOTokenizerType, get_tokenizer_class)：核心源码文件：新增 Qwen36TITOTokenizer 类，拆分 Qwen35/Qwen36 的 FIXED_TEMPLATE 注册，并在 TITOTokenizerType 枚举与 get_tokenizer_class 工厂中新增 qwen36 分支，是本次功能的注册入口。
- miles/utils/chat_template_utils/templates/qwen3.6_fixed.jinja (模块 模板文件；类别 other；类型 core-logic)：新增的 Qwen3.6 固定聊天模板，按 HF revision 995ad96 同步，是 Qwen3.6 家族工具参数 JSON 风格序列化的实际承载者。
- tests/fast/utils/chat_template_utils/test_fixed_templates.py (模块 模板测试；类别 test；类型 test-coverage；符号 test_qwen35_and_qwen36_preserve_family_tool_argument_serialization)：核心测试文件：新增参数化测试 test_qwen35_and_qwen36_preserve_family_tool_argument_serialization，直接锁定两个家族的序列化差异 (False/None vs false/null)，防止未来误合并模板。
- tests/e2e/sglang/test_session_server_multi_role/test_qwen36.py (模块 e2e 测试；类别 test；类型 test-coverage；符号 test_qwen36)：新增端到端测试，用 Qwen/Qwen3.6-35B-A3B 验证 session server 多角色 TITO 渲染，并注册 CUDA/ROCm CI 与 mismatch-rate 指标 gate，是 PR 2711 兼容性提醒的位置。
- tests/fast/utils/test_arguments.py (模块 参数测试；类别 test；类型 test-coverage；符号 test_qwen35_and_qwen36_resolve_family_template)：验证 --tito-model qwen35/qwen36 参数解析到对应模板与 preserve_thinking=True，确保参数校验链路对新家族生效。
- tests/fast/router/test_session_pretokenized_e2e.py (模块 会话测试；类别 test；类型 test-coverage)：补充 qwen3.6-fixed 的 smoke 配置，把新家族纳入 pretokenized 会话的快速验证矩阵。
- docs/user-guide/agent-ic-rollout.md (模块 用户文档；类别 docs；类型 documentation)：文档补充 --tito-model qwen35/qwen36 用法，方便用户发现新能力。

关键符号：Qwen36TITOTokenizer, Qwen35TITOTokenizer, TITOTokenizerType.get_tokenizer_class, resolve_fixed_chat_template, test_qwen35_and_qwen36_preserve_family_tool_argument_serialization, test_qwen35_and_qwen36_resolve_family_template, test_qwen36

关键源码片段

miles/utils/chat_template_utils/tito_tokenizer.py

核心源码文件：新增 Qwen36TITOTokenizer 类，拆分 Qwen35/Qwen36 的 FIXED_TEMPLATE 注册，并在 TITOTokenizerType 枚举与 get_tokenizer_class 工厂中新增 qwen36 分支，是本次功能的注册入口。

所有 Qwen 家族共享同一个 token 边界逻辑：

当 pretokenized 前缀以 "<lim_endl>" 结尾时，需要补一个换行再拼接增量 token。
该逻辑定义在 Qwen3TITOTokenizer.merge_tokens 中，
Qwen3.5 / Qwen3.6 通过纯继承复用，不重写任何 token 级行为。

```
class Qwen35TITOTokenizer(Qwen3TITOTokenizer):
```

```
    """Qwen3.5 模板 + Qwen3 token 边界。
```

```
    工具参数保持标量序列化（boolean 输出为 False、null 输出为 None），  
    只把 thinking 历史控制从 clear_thinking=False 改为 preserve_thinking=True，  
    与新 Qwen 合约保持一致。  
    """
```

```
    tool_call_parser = "qwen3_coder"
```

```
    FIXED_TEMPLATE = FixedTemplate(  
        template="qwen3.5_fixed.jinja",  
        extra_kwargs={"preserve_thinking": True},  
        allowed_append_roles=frozenset({"tool", "user", "assistant"}),  
    )
```

```
class Qwen36TITOTokenizer(Qwen3TITOTokenizer):
```

```
    """Qwen3.6 模板 + Qwen3 token 边界。
```

```
    使用 Qwen3.6-35B-A3B 的 JSON 风格序列化（boolean 为 false、null 为 null），  
    并通过 preserve_thinking=True 保留历史 reasoning。  
    注意 3.5 与 3.6 的序列化合约不同，必须使用各自的固定模板，  
    不能共享同一个 jinja，否则会静默改变另一家族的 tool-call token。  
    """
```

```
    tool_call_parser = "qwen3_coder"
```

```
    FIXED_TEMPLATE = FixedTemplate(  
        template="qwen3.6_fixed.jinja",  
        extra_kwargs={"preserve_thinking": True},  
        allowed_append_roles=frozenset({"tool", "user", "assistant"}),  
    )
```

```
class TITOTokenizerType(StrEnum):
```

```
    # ... 其他家族枚举省略 ...
```

```
    QWEN3 = "qwen3"
```

```
    QWEN35 = "qwen35"
```

```
    QWEN36 = "qwen36" # 新增枚举，供 --tito-model qwen36 解析
```

```
    QWENNEXT = "qwennext"
```

```
    # ...
```

```
    @classmethod
```

```
    def get_tokenizer_class(cls, t: TITOTokenizerType) -> type[TITOTokenizer]:
```

```
"""把 --tito-model 名称解析为具体的 TITOTokenizer 子类。"""
```

```
match t:
```

```
    case cls.DEFAULT:
```

```
        return TITOTokenizer
```

```
    case cls.QWEN3:
```

```
        return Qwen3TITOTokenizer
```

```
    case cls.QWEN35:
```

```
        return Qwen35TITOTokenizer
```

```
    case cls.QWEN36:
```

```
        return Qwen36TITOTokenizer
```

```
    case cls.QWENNEXT:
```

```
        return QwenNextTITOTokenizer
```

```
    # ... 其余家族分支省略 ...
```

评论区精华

Shi-Dong 在新增的 e2e 测试 `test_qwen36.py` 的 `ModelConfig` 处留下唯一一条技术评论：

My agent told me that this Config is not compatible with [PR 2711](#). Might worth a heads-up when you merge.

评论未展开具体冲突点，作者也未在评论区回复；PR 最终合并且两个 e2e 测试（qwen35、qwen36）均在 stage-c-4-gpu-h200 上通过，说明未触发实际阻塞，但合并顺序与后续 PR 2711 的兼容性仍需留意。此外，从提交历史可见一个重要设计取舍：首版曾将 qwen35 与 qwen36 统一映射到 Qwen3.6 语义，随后第二个提交（`fix(tito): split Qwen3.5 and Qwen3.6 templates`）将其拆开，因为共享模板会静默改变 Qwen3.5 的 boolean/null 工具调用 token——这正是本 PR 的核心争议点和最终结论。

- 新 e2e 测试与 PR 2711 的兼容性提醒 (question)：作者未在评论区回复，但 PR 已合并且两个 e2e 测试均通过，说明未形成实际阻塞；合并顺序与 PR 2711 的后续协调仍值得关注。

风险与影响

- 风险：
 1. Qwen3.5 行为语义变化：Qwen35TITOTokenizer 的固定模板 kwargs 从 `clear_thinking=False` 改为 `preserve_thinking=True`，虽然新单测覆盖了序列化输出，但 thinking 历史的保留策略变化可能影响已有 Qwen3.5 会话的渲染结果，需关注线上回归。
 2. 上游模板漂移：qwen3.6_fixed.jinja 固定自特定 HF revision (995ad96)，若上游 tokenizer_config 的模板演进，需要手动同步，否则会出现与新版本模型的渲染偏差。
 3. 与 PR 2711 潜在冲突：reviewer 明确提示新 e2e 测试的 ModelConfig 与 PR 2711 不兼容，若合并顺序不当可能导致后续 CI 或配置层面问题。
 4. CI 成本：新增 e2e 测试在 H200 上运行约 2 小时，且注册了 CUDA 与 ROCm 双套 CI，延长了 PR 验证周期。
 - 影响：用户侧：session server 新增 `--tito-model qwen36` 能力，qwen35 的渲染行为调整为更贴合新版 Qwen 合约，两个家族的工具调用渲染与 reasoning 保留均更符合模型原生语义。系统侧：改动集中于 `miles/utils/chat_template_utils` 的注册表与模板文件，注册表驱动的设计使 GLM、Kimi、

DeepSeek 等其他家族完全不受影响。团队侧：按家族拆分模板并配序列化锁定测试的模式，为后续新增模型家族（如 Qwen3.7 或类似换代）提供了清晰范本。 - 风险标记：核心模板渲染变更，Qwen3.5 语义调整，与 PR 2711 潜在冲突，新增 e2e 测试 CI 成本高

关联脉络

- PR #2717 add DeepSeek-V4-Flash-0731 support and mxfp4->fp8 converter: 同为给新模型家族扩展会话 / 模板支持，可参照其模型注册、快照测试与文档配套模式。
- PR #2710 Log compaction-aware rollout metrics: 本 PR 新增 e2e 测试的 CI gate 依赖 rollout/tito_session_mismatch_rate/v1 与 v2 指标，与 rollout 指标口径直接相关。
- PR #2604 fix(rollout): raise server readiness timeout to 120s: 同为 session-server 与 TITO 渲染链路的稳定性改动，长时 e2e 测试的可靠性与其相关。