PR #25401 完整报告

sgl-project/sglang

Add output_gate_type to Qwen3NextConfig and update models to utilize it

合并时间：2026-05-19 00:18

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/25401

执行摘要

- 一句话：为 Qwen3 Next 和 Qwen3.5 模型添加可配置的输出门激活类型。
- 推荐动作：值得精读，特别是对 Qwen3 系列模型进行定制推理的团队。建议关注 Qwen3_5TextConfig 是否需要同步添加字段，以及 self.output_gate_type or self.activation 的简化写法是否更优。

功能与动机

支持 Qwen3.6 Air 等模型变体对输出门控使用不同的激活函数（如 silu 变体）。PR 描述中的准确性测试显示 Qwen3.6 Air 在 GSM8K 上达到 96.5% 准确率，Qwen3.5-35B-A3B 达到 87.5%，验证了该配置的有效性。

实现拆解

1. 新增配置字段: 在 python/sglang/srt/configs/qwen3_next.py 的 Qwen3NextConfig 类中，增加 output_gate_type 参数，默认值为 None，并在 __init__ 方法中将其保存为实例属性。
2. 模型层适配 - Qwen3Next: 在 python/sglang/srt/models/qwen3_next.py 的 Qwen3NextLinearAttention.__init__ 中，从配置读取 output_gate_type 并存储为实例属性；修改 RMSNormGated 和 FusedRMSNormGated 的 activation 参数：若 output_gate_type 不为 None 则使用它，否则使用 self.activation（即 hidden_act）。
3. 模型层适配 - Qwen3.5: 在 python/sglang/srt/models/qwen3_5.py 的 Qwen3_5LinearAttention.__init__ 中，执行相同的读取和条件逻辑，但仅修改 RMSNormGated（Qwen3.5 未使用 FusedRMSNormGated）。
4. 无配置 / 测试 / 部署变更: 未涉及 Qwen3_5TextConfig 的修改（Review 已指出此风险），也未添加单元测试。

关键文件：

- python/sglang/srt/configs/qwen3_next.py（模块 模型配置；类别 source；类型 core-logic；符号 Qwen3NextConfig.init）：定义了新的 output_gate_type 配置字段，是整个功能的入口配置。
- python/sglang/srt/models/qwen3_next.py（模块 模型层；类别 source；类型 data-contract；符号 Qwen3NextLinearAttention.init）：在 Qwen3NextLinearAttention 中读取并使用 output_gate_type，是功能核心实现。

- python/sglang/srt/models/qwen3_5.py（模块 模型层；类别 source；类型 data-contract；符号 Qwen3_5LinearAttention.init）：在 Qwen3_5LinearAttention 中增加类似逻辑，但缺少配套配置类更新（风险点）。

关键符号：Qwen3NextConfig.init, Qwen3NextLinearAttention.init, Qwen3_5LinearAttention.init

关键源码片段

python/sglang/srt/configs/qwen3_next.py

定义了新的 output_gate_type 配置字段，是整个功能的入口配置。

```
# python/sglang/srt/configs/qwen3_next.py
class Qwen3NextConfig(PretrainedConfig):
    # ... 省略其他字段 ...
    # 新增字段：输出门控的激活函数类型，若为 None 则使用 hidden_act
    output_gate_type: Optional[str] = None

    def __init__(
        self,
        vocab_size=151936,
        # ... 省略其他参数 ...
        hidden_act="silu",
        output_gate_type=None, # 新增参数，默认 None
        # ...
    ):
        super().__init__()
        # ... 省略其他赋值 ...
        self.hidden_act = hidden_act
        self.output_gate_type = output_gate_type # 保存到实例
        # ...
```

python/sglang/srt/models/qwen3_next.py

在 Qwen3NextLinearAttention 中读取并使用 output_gate_type，是功能核心实现。

```
# python/sglang/srt/models/qwen3_next.py 中的 Qwen3NextLinearAttention.__init__
self.conv_kernel_size = config.linear_conv_kernel_dim
self.layer_id = layer_id
self.activation = config.hidden_act
self.output_gate_type = config.output_gate_type # 新增：读取自定义门控激活函数
# ...
# 根据是否启用 piecewise CUDA graph 选择不同的 norm 实现
self.norm = (
    RMSNormGated(
        self.head_v_dim,
        eps=self.layer_norm_epsilon,
        group_size=None,
        norm_before_gate=True,
        device=torch.get_device_module().current_device(),
```

```
        dtype=config.torch_dtype,
        # 如果 output_gate_type 不为 None 则覆盖 activation 参数
        **({"activation": self.output_gate_type}
           if self.output_gate_type is not None
           else {}),
    )
    if not get_global_server_args().disable_piecewise_cuda_graph
    else FusedRMSNormGated(
        self.head_v_dim,
        eps=self.layer_norm_epsilon,
        # 同上：优先使用 output_gate_type，否则回退到 hidden_act
        activation=(
            self.output_gate_type
            if self.output_gate_type is not None
            else self.activation
        ),
        device=torch.get_device_module().current_device(),
        dtype=config.torch_dtype,
    )
)
```

python/sglang/srt/models/qwen3_5.py

在 Qwen3_5LinearAttention 中增加类似逻辑，但缺少配套配置类更新（风险点）。

```
# python/sglang/srt/models/qwen3_5.py 中的 Qwen3_5LinearAttention.__init__
self.conv_kernel_size = config.linear_conv_kernel_dim
self.layer_id = layer_id
self.activation = config.hidden_act
self.output_gate_type = config.output_gate_type # 新增：可能触发 AttributeError 如果 Qwen3_
5TextConfig 未定义该字段
# ...
self.norm = RMSNormGated(
    self.head_v_dim,
    eps=self.layer_norm_epsilon,
    group_size=None,
    norm_before_gate=True,
    device=torch.get_device_module().current_device(),
    dtype=config.torch_dtype,
    # 使用与 qwen3_next.py 相同的展开语法
    **({"activation": self.output_gate_type}
       if self.output_gate_type is not None
       else {}),
)
```

评论区精华

Review 中 gemini-code-assist[bot] 提出了两个主要问题：

- 正确性问题: Qwen3_5TextConfig 未添加 output_gate_type 字段，可能导致 AttributeError。但最终 PR 未修改该配置类，可能因为 Qwen3_5TextConfig 继承自 PretrainedConfig 且实际运行未报错，或该字段已在其他 PR 中隐式存在。
- 代码风格建议: 建议将 **({"activation": ...}) 展开写法简化为 activation=self.output_gate_type or self.activation，以提高可读性。PR 未采纳此建议。最终由 yizhang2077 批准合并。
- Qwen3_5TextConfig 缺少 output_gate_type 字段 (correctness): PR 未修改 Qwen3_5TextConfig，但最终仍被批准合并，可能因为实际运行中 PretrainedConfig 允许动态属性且该字段未触发错误。
- 代码风格：dict unpacking vs or 运算符 (style): PR 未采纳建议，保持原有展开写法。

风险与影响

- 风险：
  1. Qwen3.5 配置缺失风险: Qwen3_5TextConfig 未显式定义 output_gate_type 字段，如果该配置类使用 __slots__ 或严格属性检查（尽管 PretrainedConfig 通常允许动态属性），仍可能引发 AttributeError。当前仅在运行时可正常工作依赖于 attribute 的动态添加，但对静态类型检查不友好。
  2. 向后兼容性风险: 默认值为 None，不影响现有行为；但当 output_gate_type 被设置为新值时，需确保下层 RMSNormGated 能够正确处理该激活函数类型。
  3. 缺少测试覆盖: 无新增测试，回归风险由人工验证承担。 - 影响：影响范围: 仅影响 Qwen3 Next 和 Qwen3.5 系列模型的推理路径，对其它模型无影响。影响程度: 低到中。新增配置字段使模型爱好者能灵活切换输出门控激活函数，但对生产部署需确保配置兼容。团队影响: 需注意未来对 Qwen3_5TextConfig 的改动可能与本 PR 产生冲突。
- 风险标记：缺少测试覆盖，配置类未同步更新

关联脉络

- 暂无明显关联 PR