PR #24875 完整报告

sgl-project/sglang

Support Intern-S2-Preview

合并时间：2026-05-10 22:17

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/24875

执行摘要

- 一句话：支持 Intern-S2-Preview 多模态模型
- 推荐动作：该 PR 是 SGLang 添加新模型的典型范例，结构清晰（继承 + 注册），推荐开发者阅读以了解模型集成流程。可重点关注配置文件、路由注册和多模态处理器的修改模式。

功能与动机

根据 PR 描述，支持 HuggingFace 上的 internlm/Intern-S2-Preview 多模态 MoE 模型，使得 SGLang 能够以分布式方式推理该模型。

实现拆解

1. 新增配置类（python/sglang/srt/configs/interns2preview.py）：定义 InternS2PreviewConfig 和 InternS2PreviewVisionConfig，均继承自 Qwen3.5 MoE 对应配置类，指定 model_type 为 intern_s2_preview，并映射 vision_config 和 text_config 子配置。
2. 新增模型类（python/sglang/srt/models/interns2preview.py）：定义 InternS2PreviewForConditionalGeneration 直接继承 Qwen3_5MoeForConditionalGeneration，作为模型入口并注册为 EntryClass。
3. 多模态处理器注册（python/sglang/srt/multimodal/processors/qwen_vl.py）：导入 InternS2PreviewForConditionalGeneration 并添加到 QwenVLImageProcessor.models 列表，同时在 get_mm_data 和 process_mm_data_async 方法中增加 model_type 分支，以支持视频时间戳等功能。
4. 模型路由注册（python/sglang/srt/configs/model_config.py, model_runner.py, server_args.py, hf_transformers/common.py）：将新模型添加到生成模型列表、草稿模型候选、配置类型联合以及信任远程代码支持，确保模型能被正确识别和调度。
5. 解聚编码器适配（python/sglang/srt/disaggregation/encode_server.py）：在 _process_mm_items 方法中，将 intern_s2_preview 添加到支持视频时间戳计算的模型列表，避免视频处理被跳过。同时在其他文件中微调了 import 和条件分支以覆盖新模型。

关键文件：

- python/sglang/srt/configs/interns2preview.py（模块 配置层；类别 source；类型 core-logic；符号 InternS2PreviewVisionConfig, init, InternS2PreviewConfig）：定义模型配置类，继承 Qwen3.5 MoE 配置，指定 model_type 和子配置映射，是架构核心。

- python/sglang/srt/models/interns2preview.py（模块 模型定义；类别 source；类型 data-contract；符号 InternS2PreviewForConditionalGeneration）：定义模型类，直接继承 Qwen3_5MoeForConditionalGeneration，作为模型入口并注册为 EntryClass。
- python/sglang/srt/multimodal/processors/qwen_vl.py（模块 多模态处理；类别 source；类型 dependency-wiring）：多模态处理器注册新模型，使其能复用 QwenVL 的图像 / 视频处理流程，并扩展视频时间戳分支。
- python/sglang/srt/disaggregation/encode_server.py（模块 解聚编码器；类别 source；类型 core-logic）：解聚编码器视频时间戳处理分支需要覆盖新模型，否则视频特征提取会跳过时间戳计算。
- python/sglang/srt/configs/model_config.py（模块 模型注册；类别 source；类型 data-contract）：将 InternS2PreviewForConditionalGeneration 注册为生成模型和草稿模型候选，是模型发现和调度路由的关键。

关键符号：InternS2PreviewVisionConfig, init, InternS2PreviewConfig, InternS2PreviewForConditionalGeneration

关键源码片段

python/sglang/srt/configs/interns2preview.py

定义模型配置类，继承 Qwen3.5 MoE 配置，指定 model_type 和子配置映射，是架构核心。

```
# 继承自 Qwen3.5 MoE 配置
from sglang.srt.configs.qwen3_5 import (
    Qwen3_5MoeConfig,
    Qwen3_5MoeTextConfig,
    Qwen3_5MoeVisionConfig,
)

# 视觉配置，直接复用父类逻辑
class InternS2PreviewVisionConfig(Qwen3_5MoeVisionConfig):
    model_type = "intern_s2_preview"

    def __init__(self, **kwargs):
        super().__init__(**kwargs)

# 整体模型配置，指定子配置的映射关系
class InternS2PreviewConfig(Qwen3_5MoeConfig):
    model_type = "intern_s2_preview"
    sub_configs = {
        "vision_config": InternS2PreviewVisionConfig,
        "text_config": Qwen3_5MoeTextConfig,
    }

    def __init__(self, **kwargs):
        super().__init__(**kwargs)
```

python/sglang/srt/multimodal/processors/qwen_vl.py

多模态处理器注册新模型，使其能复用 QwenVL 的图像 / 视频处理流程，并扩展视频时间戳分支。

```
# 导入新模型
from sglang.srt.models.interns2preview import InternS2PreviewForConditionalGeneration

# 在 QwenVLImageProcessor 的 models 注册列表中增加
class QwenVLImageProcessor(SGLangBaseProcessor):
    supports_transformers_backend = True
    models = [
        Qwen2VLForConditionalGeneration,
        Qwen2_5_VLForConditionalGeneration,
        Qwen3VLForConditionalGeneration,
        Qwen3VLMoeForConditionalGeneration,
        Qwen3_5ForConditionalGeneration,
        Qwen3_5MoeForConditionalGeneration,
        InternS2PreviewForConditionalGeneration, # 新增 Intern-S2-Preview
        Qwen3OmniMoeForConditionalGeneration,
    ]

# 在 get_mm_data 中添加 model_type 分支
if (self.model_type in [
    "qwen3_vl",
    "qwen3_vl_moe",
    "qwen3_5",
    "qwen3_5_moe",
    "intern_s2_preview", # 新增模型类型
] and video_timestamps is not None):
    input_ids, offsets, modality_list = self.build_input_ids_with_timestamps(...)
```

python/sglang/srt/configs/model_config.py

将 InternS2PreviewForConditionalGeneration 注册为生成模型和草稿模型候选，是模型发现和调度路由的关键。

```
# 在 is_generation_model 中注册为生成模型
"Qwen3VLMoeForConditionalGeneration",
"Qwen3_5ForConditionalGeneration",
"Qwen3_5MoeForConditionalGeneration",
"InternS2PreviewForConditionalGeneration", # 新增

# 在 _config_draft_model 中添加草稿模型支持
if is_draft_model and self.hf_config.architectures[0] in [
    "Qwen3_5ForConditionalGeneration",
    "Qwen3_5MoeForConditionalGeneration",
    "InternS2PreviewForConditionalGeneration", # 新增，允许作为草稿模型
]:
    self.hf_config.architectures[0] = "Qwen3_5ForCausalLMMTP"
    self.hf_config.num_nextn_predict_layers = 1
```

评论区精华

无实质性讨论，PR 获得维护者 ispobock 的批准，未产生 review 评论。

- 暂无高价值评论线程

风险与影响

- 风险：主要风险包括：未添加单元测试验证模型精度和功能；新模型依赖 Qwen3.5 MoE 的稳定性，若基类存在 bug 会直接影响 Intern-S2-Preview；多处路由修改可能遗漏某些场景（如 TD 解聚、多模态处理流程）；模型需要 trust_remote_code 且可能需要特定的 CUDA 版本和 attention backend。由于继承了成熟架构，风险较低，但仍建议补充测试。
- 影响：对用户：新增 Intern-S2-Preview 模型可用，启动命令已给出。对系统：对现有模型无影响，代码改动集中在新增文件，修改均添加条件分支或列表元素。对团队：后续需补充该模型的 CI 测试和精度基准，以保障质量。
- 风险标记：缺少测试覆盖，模型精度未验证，依赖 Qwen3.5 MoE 稳定性

关联脉络

- PR #24826 [spec decoding] support kimi-k2.5-eagle3-mla: 同为新增模型支持的 PR，修改了 model_config.py 和 hf_transformers/common.py 等重叠文件，体现了相似的模型注册模式。