PR #34951 完整报告

sgl-project/sglang

[Diffusion] Native ERNIE prompt enhancer

合并时间：2026-08-16 09:59

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/34951

执行摘要

- 一句话：ERNIE PE 原生化，支持分层 offload
- 推荐动作：值得精读。核心看点有两个：一是 sdpa.py 的 cached decode mask 语义修正，这是任何用 PyTorch SDPA 做增量解码时容易踩坑的通用正确性问题；二是 Ministral3ForCausalLM 的 offload 契约与 tied weights 处理，可作为扩散原生模型迁移的样板。维护其他 diffusion 模型的工程师建议核对 sdpa.py 变更对其解码数值的影响，并确认现有契约测试覆盖。

功能与动机

原实现（pe_loader.py）通过 AutoModelForCausalLM 在 diffusion server 进程内加载 HF PE 模型，并带 FA2 → SDPA 回退分支，文档此前也标注该路径 "may not provide optimal performance"。PR body 给出三点目标：replace the in-process ERNIE prompt enhancer HF causal LM with the native Ministral3 decoder；align cached causal SDPA with FlashAttention lower-right semantics；expose PE decoder layers to component layerwise offload。即去掉 HF 包装与 attention_implementation 回退的不确定性、修复 cached decode 时 SDPA 与 FlashAttention 的 mask 语义分歧、并让 26 层 PE 解码器可被逐层 offload 以服务显存受限部署。

实现拆解

1. 原生 Ministral3 解码器（mistral_3.py）：新增 Ministral3ForCausalLM，继承 Ministral3PreTrainedModel、GenerationMixin、LayerwiseOffloadableModuleMixin，复用仓库已有的 MistralModel → MistralDecoderLayer → MistralAttention 结构；声明 _tied_weights_keys = {"lm_head.weight": "model.embed_tokens.weight"} 与 layer_names = ["model.layers"]，实现 get/set input/output embeddings 与支持 logits_to_keep 的 forward。构造 MistralModel(config, allow_cudnn_sdp=False) 以保持与 HF 逐位一致，该开关贯穿 attention/decoder/model 三层（默认仍为 True，不影响既有 Mistral3 模型）。
2. Llama-4 query 缩放（mistral_3.py）：新增 _get_llama_4_attn_scale，从 config.rope_parameters 读取 llama_4_scaling_beta 与 original_max_position_embeddings；MistralAttention.forward 新增可选 position_ids 入参，在 RoPE 之后、写入缓存之前用 scale = 1 + beta * log(1 + floor(position / original_max)) 缩放 query_states。这是 Ministral3 长上下文注意力的关键数值组件，公

式与 HF 参考逐项对应。

3. SDPA *cached decode* 语义修正（sdpa.py）：SDPAImpl.forward 在 *causal=True* 且 query 长度 != key 长度时，放弃 is_causal=True，改为手工构造对角线为 key_length - query_length 的 tril(bool) mask，实现与 FlashAttention 一致的右下角对齐；单 query 解码（q_len = 1）时无需 mask。该修复影响所有经过该后端的 diffusion causal SDPA 解码路径。
4. PELoader 接线（pe_loader.py）：load_customized 从 AutoModelForCausalLM（含 FA2/SDPA 回退）切换为 Ministral3ForCausalLM.from_pretrained；PEModelWrapper 改为继承 nn.Module 与 LayerwiseOffloadableModuleMixin，注册 layer_names = ["model.model.layers"]，*generate* 在 set_forward_context(current_timestep=0, attn_metadata=None) 下执行，to 改为调用 super().to() 以正确移动参数。
5. 测试与文档：新增 test_ministral3_generation.py，覆盖 query scale 公式、SDPA 右下角对齐（q_len 参数化 1/2）、native 与 HF 的 prefill + generate 数值一致、以及两层 offload 暴露契约（Ministral3ForCausalLM.layer_names 与 PEModelWrapper.layer_names）。models_with_pe.mdx 更新默认实现说明并新增 --layerwise-offload-components pe 用法与延迟 / 显存权衡；cookbook 同步更新。

关键文件：

- python/sglang/multimodal_gen/runtime/models/encoders/mistral_3.py（模块 原生解码器；类别 source；类型 data-contract；符号 _get_llama_4_attn_scale, Ministral3ForCausalLM, Ministral3ForCausalLM.forward, MistralAttention.forward）：核心源码：新增原生 Ministral3ForCausalLM（含 Llama-4 query 缩放）、暴露 decoder 层给 offload、以 allow_cudnn_sdp=False 保证与 HF 逐位一致，是本次变更的主载体。
- python/sglang/multimodal_gen/runtime/layers/attention/backends/sdpa.py（模块 注意力后端；类别 source；类型 core-logic；符号 SDPAImpl.forward）：公共注意力后端正确性修复：*cached decode* 下把 is_causal 换成右下角对齐的 tril mask，使 SDPA 与 FlashAttention 语义一致，影响所有 diffusion causal SDPA 解码路径。
- python/sglang/multimodal_gen/runtime/loader/component_loaders/pe_loader.py（模块 组件加载；类别 source；类型 dependency-wiring；符号 PEModelWrapper, PEModelWrapper.generate, PEModelWrapper.to, PELoader.load_customized）：PE 组件接线：加载从 AutoModelForCausalLM 切到原生 Ministral3ForCausalLM, PEModelWrapper 改为 nn.Module + LayerwiseOffloadableModuleMixin 并注册 layer_names，*generate* 包 forward context。
- python/sglang/multimodal_gen/test/unit/test_ministral3_generation.py（模块 单元测试；类别 test；类型 test-coverage；符号 _config, test_ministral3_query_scale_matches_llama4_rule, test_causal_sdpa_uses_lower_right_alignment_for_cached_keys, test_native_ministral3_matches_hf_prefill_and_generation）：新增 5 项契约测试：query scale 公式、SDPA 右下角对齐（q_len 参数化 1/2）、native/HF prefill+generate 数值一致、以及两层 offload 暴露契约。
- docs/docs/sglang-diffusion/models_with_pe.mdx（模块 用户文档；类别 docs；类型 documentation）：更新 PE 部署文档：默认加载方式改为原生实现，新增

--layerwise-offload-components pe 用法及延迟 / 显存取舍说明。

- docs/cookbook/diffusion/Ernie-Image/Ernie-Image.mdx（模块 示例文档；类别 docs；类型 documentation）：cookbook 同步说明 PE 组件现在的 native 加载方式与 offload 选项。

关键符号：_get_llama_4_attn_scale, Ministral3ForCausalLM.forward, Ministral3ForCausalLM.init, MistralAttention.forward, PEModelWrapper.generate, PEModelWrapper.to, PELoader.load_customized, SDPAImpl.forward

关键源码片段

python/sglang/multimodal_gen/runtime/layers/attention/backends/sdpa.py

公共注意力后端正确性修复：cached decode 下把 is_causal 换成右下角对齐的 tril mask，使 SDPA 与 FlashAttention 语义一致，影响所有 diffusion causal SDPA 解码路径。

```
# SDPAImpl.forward 的核心修正：cached decode 时 query 长度小于 key 长度，
# PyTorch 的 is_causal=True 会按“左上角对齐”生成 mask，这与 FlashAttention
# 的“右下角对齐”语义不一致，导致与 FA/HF 结果偏差。因此在 q_len != k_len 时
# 放弃 is_causal，改用手工构造 tril mask：diagonal = key_length - query_length，
# 允许每个 query 只看到自身位置之前的 key（含全部 cached key），与 FA 一致。
def forward(
    self,
    query: torch.Tensor,
    key: torch.Tensor,
    value: torch.Tensor,
    attn_metadata: AttentionMetadata,
) -> torch.Tensor:
    # transpose 到 bs, heads, seq_len, head_dim
    query = query.transpose(1, 2)
    key = key.transpose(1, 2)
    value = value.transpose(1, 2)

    attn_mask = None
    is_causal = self.causal
    if self.causal and query.shape[-2] != key.shape[-2]:
        is_causal = False
        # 单 query 时该行天然全可见，无需 mask；多 query 时补右下角对齐 mask
        if query.shape[-2] > 1:
            query_length = query.shape[-2]
            key_length = key.shape[-2]
            attn_mask = torch.ones(
                query_length,
                key_length,
                dtype=torch.bool,
                device=query.device,
            ).tril(diagonal=key_length - query_length)

    attn_kwargs = {
        'attn_mask': attn_mask,
```

```
            'dropout_p': self.dropout,
            'is_causal': is_causal,
            'scale': self.softmax_scale,
        }
        if query.shape[1] != key.shape[1]:
            attn_kwargs['enable_gqa'] = True
        with self._sdpa_context(query):
            output = torch.nn.functional.scaled_dot_product_attention(
                query, key, value, **attn_kwargs
            )
        output = output.transpose(1, 2)
        return output
```

评论区精华

本 PR 没有任何 review 评论，唯一的 issue 评论是作者发布的 /tag-and-rerun-ci extra（触发 extra CI 重跑），最终由作者自行合入。技术结论主要来自 PR body 的自证数据：

- CI 重跑指令与无 review 状态 (other): extra CI 通过后由作者自行合入；无代码层面的设计交锋或未解决疑虑。

风险与影响

- 风险：
  - 公共注意力后端行为变更：sdpa.py 的改动不是 PR 专属路径，所有 diffusion 模型中 causal SDPA + cached decode 的输出都会从“左上角对齐”变为“右下角对齐”。这本身是朝向 FlashAttention 语义的修正，但若某个模型此前依赖旧行为或与手工 mask 叠加，可能产生输出变化，需关注 WanVideo 等其他 diffusion 模型的解码回归。
  - 移除 HF 回退路径：pe_loader.py 不再有 FA2 → SDPA 的 fallback；若目标 checkpoint 的 config 与 Ministral3 结构不兼容（缺 rope_parameters、自定义 remote code 等），加载会直接失败，报错信息更少。
  - 数值一致性强约束：max_abs = 0 的对齐依赖 bfloat16、mask 语义与缩放公式三者完全一致；未来任何对 sdpa.py 或 query scale 的调整都可能破坏该契约，需依托新增单元测试防回归。
  - 性能权衡：原生模型关闭 cudnn SDPA 以换取确定性；docs 明确 layerwise offload 会增加 prompt enhancement 延迟，属显存与延迟的取舍，文档未给出基准数据。
  - 流程风险：无 reviewer 审查、author 自合入，对共享后端的正确性修改而言交叉验证偏弱。
- 影响：
  - 用户 / 部署：ERNIE-Image 默认 in-process PE 改为原生实现，新增 --layerwise-offload-components pe 用法，显存受限场景可直接采用；文档与 cookbook 均已同步。
  - 系统：SDPA 后端 cached decode 语义统一为 FlashAttention 右下角对齐，影响所有 diffusion causal SDPA 解码路径，属于公共后端行为修正。

- 团队：延续 diffusion 原生化迁移（H3 VAE、Hunyuan3D、LTX-2.5 之后的 PE 组件），为后续其他扩散组件的原生化提供参照模板：模型类继承 LayerwiseOffloadableModuleMixin + layer_names 契约 + HF 逐位对齐测试。
- 风险标记：共享 SDPA 后端行为变更，移除 HF 加载回退路径，数值一致性强约束，layerwise offload 延迟权衡，无 review 交叉验证

关联脉络

- PR #34980 [Diffusion] Native Hunyuan3D Paint and Delight models: 同属 diffusion 原生模型迁移方向，共享"原生实现替换 HF/ 自定义 + 速率提升"的模式。
- PR #34949 [Diffusion] Route MiniMax H3 VAE attention through native backends: 注意力统一走原生后端，与本次 sdpa.py 语义修正同属 diffusion 注意力后端一致性收口。
- PR #34952 docs: define native diffusion model integration contract: 定义原生集成契约；本 PR 的 layer_names + LayerwiseOffloadableModuleMixin + HF parity 测试正是该契约的落地样板。
- PR #34891 fix(diffusion): scope attention backend fallback: 修复注意力后端回退过严问题，与本次 SDPA 后端语义修正共同构建 diffusion 注意力后端一致性。