PR #23199 完整报告

sgl-project/sglang

Add HunyuanVideo ModelOpt FP8 diffusion support

合并时间：2026-05-05 19:27

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/23199

执行摘要

- 一句话：为 HunyuanVideo 添加 ModelOpt FP8 量化支持
- 推荐动作：适合有意在 SGLang 中集成新量化模型的工程师阅读，重点关注 build_modelopt_fp8_transformer.py 中的命名映射机制和 ReplicatedLinear 的泛化设计。

功能与动机

HunyuanVideo 是强大的视频生成模型，但 BF16 推理开销较高。通过在 ModelOpt 工具链中引入 FP8 量化，能在几乎无损的情况下大幅降低去噪阶段计算时间，提升在线体验。

实现拆解

分四步实施：

1. 在 build_modelopt_fp8_transformer.py 中新增 HunyuanVideo 专用 BF16 保留层模式和运行时名称映射规则（约 30+ 条替换规则）。
2. 修改 hunyuanvideo.py 中双流注意力块，为 img_attn_qkv、txt_attn_qkv、txt_attn_proj、txt_mlp 等线性层添加 output_sizes 参数，以适配 FP8 量化权重的多分区加载。
3. 在 linear.py 中扩展 ReplicatedLinear 构造函数，添加可选 output_sizes 参数，替代原先的固定 [self.output_size] 分区方式；同时清理 weight_loader 中未使用的局部变量。
4. 在 testcase_configs.py 中注册 lmsys/hunyuanvideo-modelopt-fp8-sglang-transformer 仓库常量，在 gpu_cases.py 中添加 B200 CI 用例 hunyuanvideo_modelopt_fp8_t2v，并同步更新两份量化文档以及 Claude 技能文档。

关键文件：

- python/sglang/multimodal_gen/tools/build_modelopt_fp8_transformer.py（模块 构建工具；类别 source；类型 data-contract；符号 _module_name_variants, _map_hunyuanvideo_runtime_module_name, _get_runtime_module_name_mapper, _preferred_module_name）：核心转换工具，添加了 HunyuanVideo 的 BF16 保留模式和运行时名称映射规则
- python/sglang/multimodal_gen/runtime/models/dits/hunyuanvideo.py（模块 模型适配；类别 source；类型 data-contract）：HunyuanVideo 模型中双流注意力块添加 output_sizes 参数以支持 FP8 权重加载

- python/sglang/multimodal_gen/runtime/layers/linear.py（模块 线性层；类别 source；类型 core-logic）：扩展 ReplicatedLinear 以支持可选的 output_sizes 参数，使 FP8 量化权重能被正确分区加载
- python/sglang/multimodal_gen/benchmarks/bench_offline_throughput.py（模块 基准测试；类别 source；类型 core-logic）：修复 benchmark 解析方式以支持未知参数，适配 FP8 测试时的额外参数传递
- python/sglang/multimodal_gen/test/server/testcase_configs.py（模块 测试配置；类别 test；类型 test-coverage）：注册 HunyuanVideo FP8 的 transformer 路径常量
- python/sglang/multimodal_gen/test/server/gpu_cases.py（模块 GPU 测试；类别 test；类型 test-coverage）：为 B200 CI 添加 hunyuanvideo_modelopt_fp8_t2v 测试用例
- docs_new/docs/sglang-diffusion/quantization.mdx（模块 量化文档；类别 other；类型 documentation）：更新扩散模型量化文档，加入 HunyuanVideo 的 ModelOpt FP8 流程说明
- python/sglang/multimodal_gen/.claude/skills/sglang-diffusion-modelopt-quant/SKILL.md（模块 技能文档；类别 docs；类型 documentation）：更新 ModelOpt 量化技能文档，加入 HunyuanVideo 路径
- docs/diffusion/quantization.md（模块 扩散文档；类别 docs；类型 documentation）：旧版量化文档同步更新，保持与新文档一致

关键符号：_get_runtime_module_name_mapper, _map_hunyuanvideo_runtime_module_name, _preferred_module_name, ReplicatedLinear.init

关键源码片段

python/sglang/multimodal_gen/runtime/layers/linear.py

扩展 ReplicatedLinear 以支持可选的 output_sizes 参数，使 FP8 量化权重能被正确分区加载

```
class ReplicatedLinear(LinearBase):
    """复制的线性层，支持可选的 output_sizes 用于 FP8 量化权重分区加载。"""

    def __init__(
        self,
        input_size: int,
        output_size: int,
        bias: bool = True,
        skip_bias_add: bool = False,
        params_dtype: torch.dtype | None = None,
        quant_config: QuantizationConfig | None = None,
        # 新增：当 FP8 量化权重需要按多个输出分区加载时，通过 output_sizes 指定各分区大小
        output_sizes: list[int] | None = None,
        prefix: str = "",
    ):
        super().__init__(
            input_size,
            output_size,
            skip_bias_add,
```

```
        params_dtype,
        quant_config,
        prefix=prefix,
    )

    # 所有线性层必须支持量化方法
    assert self.quant_method is not None
    # 使用 output_sizes 或回退到整体单分区（原始行为）
    output_partition_sizes = output_sizes or [self.output_size]
    self.quant_method.create_weights(
        self,
        self.input_size,
        output_partition_sizes,
        self.input_size,
        self.output_size,
        self.params_dtype,
        weight_loader=self.weight_loader,
    )

    if bias:
        self.bias = Parameter(
            torch.empty(self.output_size, dtype=self.params_dtype)
        )
        set_weight_attrs(
            self.bias,
            {"output_dim": 0, "weight_loader": self.weight_loader},
        )
    else:
        self.register_parameter("bias", None)
```

评论区精华

唯一 Review 来自 mickqian，在 qwen_image.py 的 diff chunk 上评论 "duplicated contents?" 怀疑重复导入。作者未公开回复，但 PR 仍获得 Approved 并合并，推测后续提交已清理。

- 重复导入内容 (question): 未在评论区看到回复，但最终 PR 被 Approved，可能后续提交已修复或未构成问题。

风险与影响

- 风险：
  1. 名称映射表不完整：若 ModelOpt 导出权重名称有变动，可能导致加载失败，需依赖 CI 提前捕获。
  2. FP8 精度验证：仅 H100 验证，B200 和其他硬件（如 AMD）上精度需额外测试。
  3. ReplicatedLinear 回归：output_sizes 参数默认行为与之前一致，但已在构造器中改变 create_weights 的分区逻辑，可能影响所有使用该类且未指定 output_sizes 的扩散模型

（风险较低）。

4. CI 覆盖有限：仅 B200 单卡测试，未覆盖多卡或不同精度组合。 - 影响：用户可直接通过 --transformer-path lmsys/hunyuanvideo-modelopt-fp8-sglang-transformer 使用预量化模型，获得约 1.09x 端到端加速。团队新增了一套完整的 FP8 模型接入模板（命名映射 + BF16 保留 + output_sizes），便于未来扩展至其他扩散模型。 - 风险标记：名称映射表不完整可能失败，仅 H100 验证精度，ReplicatedLinear 泛化影响所有扩散模型，仅 B200 单卡 CI

关联脉络

- 暂无明显关联 PR