PR #35418 完整报告

sgl-project/sglang

[Diffusion] Support MiniMax-H3 pruned safetensors checkpoints

合并时间：2026-08-20 19:34

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/35418

执行摘要

- 一句话：支持 MiniMax-H3 pruned 与 Comfy FP8 safetensors 加载
- 推荐动作：值得精读。重点学习三处设计： 1) comfy_fp8.py 中 FP8 存储与全精度计算分离的逐层分派，平衡显存与数值语义； 2) minimax_h3_weights.py 在模型构建前完成全部校验，把误导性缺权重错误提前转化为明确格式错误； 3) transformer_load_utils.py 的 HF 文件引用解析，用正则与启发式判断处理三种输入形态。

功能与动机

用户难以直接加载 MiniMax-H3 官方发布的 pruned BF16 与 Comfy FP8 scaled checkpoint，缺少自动识别会导致加载时出现误导性的缺权重错误。本 PR 在 #35370 通用 GGUF 支持之上，用 checkpoint 自描述元数据驱动加载，避免重复实现 GGUF 加载器，同时按 Comfy full_precision_matrix_mult 标记保留数值语义并维持 FP8 显存收益。

实现拆解

1. Checkpoint 自描述检测：新增 minimax_h3_weights.py，在模型构建前用 safe_open 扫描 safetensors，读取 adaln_t_table 形状识别 pruned BF16，读取 comfy_quant JSON 标记识别 Comfy FP8；对 FP8 权重做标记完整性校验，对 int8_tensorwise 或缺失 scale 的 checkpoint 提前抛出明确错误。
2. 新增 Comfy FP8 量化配置：comfy_fp8.py 定义 ComfyFp8Config 与 ComfyFullPrecisionFp8LinearMethod，按 full_precision_matrix_mult 标记逐层分派——普通层复用原生静态 FP8 线性路径，标记层用反量化权重做普通 GEMM，但权重仍以 FP8 驻留显存，避免永久展开为 BF16。
3. 加载流程接入：transformer_loader.py 在 MiniMaxH3DiTModel 分支中调用检查函数，把 checkpoint_quant_config 传入 resolve_transformer_quant_load_spec；Comfy FP8 拒绝 FSDP 推理，并用 comfy_quant_key_filter 过滤元数据键；同时把 adaln_curve_shape 回写到 dit_config。
4. 路径解析与配置冲突：transformer_load_utils.py 新增 HF 文件引用解析（直接 URL 或 owner/repo/path/file.safetensors 引用）和 is_comfy_fp8 属性，resolve_transformer_quant_load_spec 新增 checkpoint_quant_config 参数，与显式 --quantization 冲突时报错。

5. 测试与文档：test_transformer_quant.py 增加 8 个单元测试覆盖检查、分派、数值与冲突场景；test_minimax_h3_dit_contract.py 扩展 FP8 grouped-QKV TP 复制覆盖；cookbook 与 quantization.mdx 文档同步更新。

关键文件：

- python/sglang/multimodal_gen/runtime/layers/quantization/comfy_fp8.py（模块 量化层；类别 source；类型 core-logic；符号 ComfyFullPrecisionFp8LinearMethod, ComfyFp8Config, create_weights, process_weights_after_loading）：新增 Comfy FP8 按层量化分派核心，是 pruned FP8 checkpoint 加载的关键实现单元。
- python/sglang/multimodal_gen/runtime/loader/minimax_h3_weights.py（模块 加载器；类别 source；类型 core-logic；符号 comfy_quant_key_filter, inspect_minimax_h3_safetensors, resolve_minimax_h3_checkpoint_quantization, validate_minimax_h3_checkpoint_variant）：新增 checkpoint 自描述检测与校验，识别 pruned BF16 与 Comfy FP8，并在构建前拒绝不支持的格式。
- python/sglang/multimodal_gen/runtime/loader/transformer_load_utils.py（模块 加载器；类别 source；类型 core-logic；符号 is_comfy_fp8, resolve_transformer_safetensors_to_load, resolve_transformer_quant_load_spec, _HF_SAFETENSORS_URL_RE）：新增 HF 文件引用解析与 is_comfy_fp8 属性，是通用加载路径的关键改动点。
- python/sglang/multimodal_gen/runtime/loader/component_loaders/transformer_loader.py（模块 加载器；类别 source；类型 dependency-wiring；符号 load_customized）：将 checkpoint 检测、校验、量化配置解析接入 MiniMaxH3DiTModel 的加载流程，并处理 FSDP 冲突与键过滤。
- python/sglang/multimodal_gen/test/unit/test_transformer_quant.py（模块 测试；类别 test；类型 test-coverage；符号 test_resolve_transformer_safetensors_to_load_uses_hf_file_reference, test_inspect_minimax_h3_safetensors_detects_curve_and_comfy_format, test_inspect_minimax_h3_fp8_validates_required_scales, test_minimax_h3_comfy_int8_fails_before_weight_loading）：新增 8 个单元测试，覆盖 checkpoint 检查、格式分派、全精度数值与配置冲突等核心场景。
- python/sglang/multimodal_gen/test/unit/test_minimax_h3_dit_contract.py（模块 测试；类别 test；类型 test-coverage；符号 test_native_weight_names_and_grouped_qkv_reorder）：扩展 grouped-QKV TP 复制测试以覆盖 FP8 权重，验证 TP1/2/4 下的 FP8 加载正确性。
- python/sglang/multimodal_gen/runtime/models/dits/minimax_h3.py（模块 模型定义；类别 source；类型 data-contract）：对 DiT 模型做数据契约层面的小调整，适配 pruned AdaLN 曲线配置回写。
- docs/cookbook/diffusion/MiniMax/MiniMax-H3.mdx（模块 文档；类别 docs；类型 documentation）：新增 pruned BF16 与 scaled-FP8 加载命令、兼容性与不支持格式说明。
- docs/docs/sglang-diffusion/quantization.mdx（模块 文档；类别 docs；类型 documentation）：更新扩散模型量化矩阵，区分通用 GGUF 与 Comfy scaled-FP8 格式。

关键符号：ComfyFullPrecisionFp8LinearMethod.apply, ComfyFp8Config.get_quant_method, inspect_minimax_h3_safetensors,

resolve_minimax_h3_checkpoint_quantization, validate_minimax_h3_checkpoint_variant,comfy_quant_key_filter, TransformerQuantLoadSpec.is_comfy_fp8,resolve_transformer_safetensors_to_load, resolve_transformer_quant_load_spec,transformer_loader.load_customized

评论区精华

该 PR 没有 review 评论，作者在 PR body 中记录了关键验证细节：通过检查官方 Comfy FP8 元数据（200 个 per-layer 标记，其中 50 个 full-precision 层）确认分派逻辑；H100 端到端加载 20,958,205,608 字节的 Comfy FP8 checkpoint，实例化 19.61 GB pruned DiT，2 步 1344x768 请求产出 107 帧视频加立体声，48.43 秒完成，21,236 MB 峰值显存。多 GPU 端到端验证因共享宿主在 NCCL 初始化时杀掉额外 worker 进程而留给 CI，TP1/2/4 的 grouped-QKV 复制路径由单元测试覆盖。

- 暂无高价值评论线程

风险与影响

- 风险：
  1. FP8 数值风险：ComfyFullPrecisionFp8LinearMethod.apply 直接在 x.dtype 下反量化权重做 F.linear，未走量化感知路径，数值对齐依赖作者手工核对官方元数据和单次 H100 验证，缺少多卡与多 prompt 的回归矩阵。
  2. FSDP 兼容性限制：Comfy FP8 checkpoint 明确拒绝 FSDP 推理，若团队配置了 FSDP 加载会直接报错，可能影响既有部署脚本。
  3. 路径解析回归：resolve_transformer_safetensors_to_load 的 HF 引用判定涉及 os.path.exists 与后缀判断，对不存在的相对路径且不含 ./~ 前缀的文件可能误判为 HF 引用，需关注本地路径的兼容性。
  4. 多 GPU 验证缺口：PR body 说明多 GPU 端到端验证留给 CI，TP 下的 FP8 数值一致性仍需 CI 结果确认。 - 影响：对用户：可直接通过 --transformer-weights-path 传入本地文件、HF URL 或仓库内引用加载 MiniMax-H3 pruned BF16 与 Comfy FP8 scaled checkpoint，FL2VA 与 Ref2VA 均可使用，TP、Ulysses/Ring 序列并行与组件 offload 保持可用。对系统：加载流程新增 checkpoint 自描述元数据驱动模式，对后续其他模型的量化格式识别有参考价值；HF 文件引用解析对所有 diffusion transformer 加载路径生效。对团队：该 PR 补上了 MiniMax-H3 从 GGUF 到 pruned/FP8 的完整格式矩阵，配合后续短边校验放宽与 CI 一致性覆盖，形成较完整的模型支持闭环。 - 风险标记：FP8 数值依赖手工验证，多 GPU 端到端留待 CI，FSDP 兼容性限制，HF 引用解析可能误判本地路径

关联脉络

- PR #35370 [diffusion] feat: load GGUF transformer checkpoints (MiniMax-H3)：本 PR 明确构建在该 PR 的通用 GGUF 支持之上，复用其加载器与测试，不再重复实现 GGUF 路径。

- PR #35664 [diffusion] feat: warn on an unverified short edge instead of rejecting it for minimax-h3: 同一模型后续功能演进，放宽短边校验，与本 PR 共同完善 MiniMax-H3 加载兼容性。
- PR #35511 [diffusion] CI: add minimax-h3 ref2va audio consistency coverage and guard peak vram: 同模型 CI 一致性覆盖，验证本 PR 引入的加载路径在多 GPU 下的行为，并补充显存峰值守卫。
- PR #35626 [diffusion] fix: keep large vocab tables in host memory under layerwise offload: 同运行时的显存优化，与本 PR 的 FP8 显存收益目标互补，共同降低 multimodal_gen 峰值显存。