# PR #36603 完整报告

sgl-project/sglang

fix(kimi-k3): preserve dense ModelSlim MLA weights

合并时间：2026-08-28 17:15

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/36603

## 执行摘要

- 一句话：修复 Kimi-K3 ModelSlim MLA 权重被误删问题
- 推荐动作：值得精读。该 PR 展示了一个典型的能力膨胀导致的行为错误及修复方式：通过拆分粒度更细的能力标志，避免语义混淆。适合对量化加载、模型构建配置有深入兴趣的工程师阅读。

## 功能与动机

Kimi-K3 ModelSlim W4A8 多节点精度任务在加载 351 个 checkpoint 分片后失败，报错 ValueError: Kimi-K3 MLA K projection must remain GGUF Q4_0。#35314 为 Kimi-K3 ExpertPack 路径添加了 split-GGUF K/V 支持，但错误地通过 supports_kimi_k3_quantized_latent_projections 来触发该路径。ModelSlim 虽然声明支持该能力（用于 latent-MoE 线性层），但其 MLA checkpoint 中 kv_b_proj.weight 仍是普通密集 FLOAT 张量，导致模型构建时误删 kv_b_proj，注册了未加载的 GGUF 占位符，最终在加载后 Q4_0/Q2_K 校验失败。

## 实现拆解

1. 新增能力标志：在 ExpertPackConfig（python/sglang/srt/layers/quantization/expert_pack.py）中新增类属性 supports_kimi_k3_split_gguf_kv_b = True，明确标识使用 split-GGUF MLA K/V 表示。
2. 引入判断函数：在 kimi_k3.py 中新增 _uses_split_gguf_kv_b(quant_config)，通过 getattr(quant_config, 'supports_kimi_k3_split_gguf_kv_b', False) 判断，避免直接依赖旧的通用能力。
3. 调整构建逻辑：在 KimiK3MLAAttention.__init__ 中，将原来的 split_gguf_kv_b = getattr(quant_config, 'supports_kimi_k3_quantized_latent_projections', False) 改为调用 _uses_split_gguf_kv_b，使得只有在显式声明 split-GGUF 能力时才删除 kv_b_proj，ModelSlim 保持密集路径。
4. 新增回归测试：在 test_kimi_k3_gguf.py 新增 test_split_kv_capability_is_expert_pack_specific，断言 ModelSlim 不启用 split-GGUF，而 SimpleNamespace 显式启用时才返回 True，防止回归。

关键文件：

- python/sglang/srt/models/kimi_k3.py（模块 模型层；类别 source；类型 data-contract；符号 _uses_split_gguf_kv_b, KimiK3MLAAttention.init）：核心修改文件，添加 _uses_split_gguf_kv_b 并调整 KimiK3MLAAttention 的构建逻辑，直接修复 ModelSlim 密集 MLA 权重被误删的问题。
- python/sglang/srt/layers/quantization/expert_pack.py（模块 量化模块；类别 source；类型 core-logic；符号 ExpertPackConfig）：新增 supports_kimi_k3_split_gguf_kv_b 能力标志，确保 ExpertPack 路径继续使用 split-GGUF，与 ModelSlim 区分开。
- test/registered/expert_pack/test_kimi_k3_gguf.py（模块 测试；类别 test；类型 test-coverage；符号 test_split_kv_capability_is_expert_pack_specific）：新增回归测试，验证 ModelSlim 和 split-GGUF 的能力区分，防止回归。

关键符号：_uses_split_gguf_kv_b, KimiK3MLAAttention.init

## 关键源码片段

### python/sglang/srt/models/kimi_k3.py

核心修改文件，添加 _uses_split_gguf_kv_b 并调整 KimiK3MLAAttention 的构建逻辑，直接修复 ModelSlim 密集 MLA 权重被误删的问题。

```
# python/sglang/srt/models/kimi_k3.py ( 关键片段 )
from sglang.srt.layers.quantization.base_quantization import QuantizationConfig

def _uses_split_gguf_kv_b(quant_config: Optional[QuantizationConfig]) -> bool:
    """判断 K3 是否使用 split-GGUF 存储 MLA K/V，只有显式启用该能力的配置才返回 True。"""
    return bool(getattr(quant_config, "supports_kimi_k3_split_gguf_kv_b", False))

class KimiK3MLAAttention(DeepseekV2AttentionMLA):
    """带输出门的 MLA 注意力层，仅用于 K3。"""

    def __init__(self, config, layer_idx, quant_config=None, ...):
        # ModelSlim 仅量化 latent 投影，而 MLA KV 投影仍是单个密集张量；
        # 只有 GGUF expert pack 才会拆分 K/V，因此必须用专用标志判断。
        split_gguf_kv_b = _uses_split_gguf_kv_b(quant_config)
        self.all_reduce_fusion = all_reduce_fusion
        self.use_output_gate = getattr(config, "mla_use_output_gate", False)
        self._disable_npu_fused_split_qk_norm = True # 数值不等价，禁用融合路径
        super().__init__(...)
        if split_gguf_kv_b:
            del self.fused_qkv_a_proj_with_mqa
            del self.kv_b_proj # 仅 split-GGUF 场景才需要删除
            ...
        # 否则保留 kv_b_proj，确保 ModelSlim 密集权重正常加载
```

### python/sglang/srt/layers/quantization/expert_pack.py

新增 supports_kimi_k3_split_gguf_kv_b 能力标志，确保 ExpertPack 路径继续使用 split-GGUF，与 ModelSlim 区分开。

```
# python/sglang/srt/layers/quantization/expert_pack.py ( 关键片段 )
```

```
class ExpertPackConfig(GGUFConfig):
    """Expert Pack 专用量化配置。"""

    is_fp4_experts = True
    # ModelSlim 也声明支持 latent 投影量化，但 MLA 仍为密集权重，
    # 因此不能复用此标志来判断 split-GGUF。
    supports_kimi_k3_quantized_latent_projections = True
    # 新增独立标志，仅 ExpertPack 使用 split-GGUF 存储 MLA K/V。
    supports_kimi_k3_split_gguf_kv_b = True

    def __init__(self, store: ExpertPackStore) -> None:
        super().__init__()
        self.store = store
```

### test/registered/expert_pack/test_kimi_k3_gguf.py

新增回归测试，验证 ModelSlim 和 split-GGUF 的能力区分，防止回归。

```
# test/registered/expert_pack/test_kimi_k3_gguf.py ( 关键片段 )
from sglang.srt.layers.quantization.modelslim.modelslim import ModelSlimConfig
from sglang.srt.models.kimi_k3 import _uses_split_gguf_kv_b

class TestKimiK3GGUFMapping(unittest.TestCase):
    def test_split_kv_capability_is_expert_pack_specific(self) -> None:
        # ModelSlim 声明 latent 投影量化，但仍使用密集 MLA，不应触发 split-GGUF。
        self.assertTrue(ModelSlimConfig.supports_kimi_k3_quantized_latent_projections)
        self.assertFalse(_uses_split_gguf_kv_b(ModelSlimConfig))
        # 显式声明 split-GGUF 能力时应返回 True。
        self.assertTrue(
            _uses_split_gguf_kv_b(
                SimpleNamespace(supports_kimi_k3_split_gguf_kv_b=True)
            )
        )
```

## 评论区精华

Issue 评论区主要包含作者的详细验证说明，无 Review 分歧。作者在评论中报告了四节点 64 NPU 硬件验证结果：351/351 分片加载成功，无原始 MLA 后加载错误，服务器正常启动，GSM8K 200 题准确率 97.0%。

- 暂无高价值评论线程

## 风险与影响

- 风险：
  1. 回归风险：修改了 KimiK3MLAAttention 的构建逻辑，若其他量化路径（如 GGUF/ExpertPack）依赖旧行为，可能受影响。但本 PR 通过新增独立标志并保持 ExpertPackConfig 显式启用，降低了风险。

2. 测试覆盖：新增测试仅验证能力标志判断，未覆盖实际 ModelSlim 权重加载 E2E 流程（依赖硬件）。
3. 多节点验证：四节点 NPU 验证结果报告成功，但未在 CUDA 环境跑 GGUF/ExpertPack E2E（作者说明硬件限制）。 - 影响：用户侧：修复了 Kimi-K3 ModelSlim 用户无法加载 checkpoint 的问题，使其能在多节点 NPU 上正常推理。系统侧：新增配置能力标志，不影响其他模型或量化路径。团队侧：提供了清晰的回归测试，防止未来重蹈覆辙。影响范围限于 Kimi-K3 MLA 量化加载路径。 - 风险标记：核心加载路径变更，缺少正式 E2E 测试，NPU 特定修复

## 关联脉络

- PR #35314 Support deepseek v4 and kimi k3 on ssd: 本 PR 修复了 #35314 引入的 split-GGUF K/V 能力误用问题，新增独立能力标志，并保持其 ExpertPack 路径行为不变。
- PR #36211 [k3] declare packed_modules_mapping on KimiK3ForConditionalGeneration: 同属 Kimi-K3 量化模型加载的修复，涉及模型配置与量化表示的匹配问题。