### Rome\_main程序功能概述

#### ****输入****

**模型与分词器**

model: 预训练因果语言模型（如GPT-2、GPT-J）

tok: 与模型匹配的分词器

示例：AutoModelForCausalLM.from\_pretrained("gpt2")

**编辑请求列表**

每个请求为字典结构，包含：

prompt: 待编辑知识的模板（如"{}的国籍是"）

subject: 编辑主体（如"爱因斯坦"）

target\_new: 新知识的目标值（如{"str": "德国"}）

**超参数配置**

ROMEHyperParams对象定义：

layers: 需要修改的Transformer层号（如[5,6]）

rewrite\_module\_tmp: 权重模块路径模板（如"transformer.h.{}.mlp"）

context\_template\_length\_params: 上下文生成模板长度参数

#### ****输出****

**修改后的模型**

包含新知识权重的模型副本，保持原始架构

生成示例：model.generate("爱因斯坦的国籍是") → "德国"

**原始权重备份**​（可选）

返回被修改权重的原始值字典，用于恢复

键为权重路径（如transformer.h.5.mlp.weight），值为张量

### 运行逻辑

#### ****1.模型初始化与复制****

通过deepcopy创建模型副本（若copy=True），确保原始模型不受影响

加载所有待修改层的权重到字典weights，并创建原始权重的备份weights\_copy

#### 2.****逐请求知识编辑****

对每个编辑请求执行：

**上下文模板生成**  
调用get\_context\_templates生成多样化输入模板（如"{}", "The capital is. {}"），用于后续梯度计算

**分层权重更新**  
按层号遍历各Transformer层（如第5、6层）：

​左向量计算​（compute\_u）  
基于请求内容与上下文模板，计算MLP层的输入梯度方向

​右向量计算​（compute\_v）  
结合左向量与目标输出差异，计算权重更新的方向向量

​秩一矩阵生成  
通过外积left\_vector @ right\_vector生成低秩更新矩阵

​权重增量注入  
将更新矩阵按形状匹配后叠加到目标权重：

#### 3. ​****权重恢复与增量返回****

完成所有层更新后，将模型权重恢复至初始状态（保证算法无副作用）

返回各层的增量向量对deltas，供主函数apply\_rome\_to\_model实际应用修改

#### 4. ​****模型最终更新****

遍历所有请求后，累计各层的权重增量

若开启return\_orig\_weights，返回被修改权重的原始值用于回滚

### 示例执行流程

**输入请求**

requests = [{ "prompt": "{}的发明者是", "subject": "电话", "target\_new": {"str": "安东尼奥·穆奇"}}]

**运行过程**

1. 在第5、6层MLP模块生成上下文模板集（如["{}", "重要发明包括. {}"]）
2. 计算各层的left\_vector（输入方向梯度）和right\_vector（输出差异方向）
3. 生成秩一更新矩阵并叠加到transformer.h.5.mlp.weight等指定权重
4. 最终模型对"电话的发明者是"生成"安东尼奥·穆奇"

### Hparams程序功能与运行逻辑解析

该代码定义了一个用于大规模语言模型知识编辑的ROME算法超参数配置类​（ROMEHyperParams），其核心功能是为模型权重更新提供细粒度控制参数，实现精准知识注入。以下是模块化解析：

#### ​****输入与输出说明****

#### ​****运行逻辑分阶段描述****

**1.参数初始化阶段**

通过@dataclass自动生成构造函数，继承HyperParams基类的通用参数（如学习率、正则化系数）

关键参数组：

​模型结构参数​（如layers指定目标层，rewrite\_module\_tmp定义权重路径模板）

​优化控制参数​（如v\_num\_grad\_steps梯度步数，v\_lr学习率）

​统计修正参数​（如mom2\_adjustment启用二阶矩修正）

*# 示例配置（参考网页2调参实践）*config = ROMEHyperParams( layers=[5, 6], *# 修改第5、6层权重* v\_lr=1e-3, *# 右向量学习率* mom2\_n\_samples=1000 *# 二阶矩统计采样数*)

**知识编辑执行阶段**

​分层权重更新​（按layers遍历目标层）：

​左向量计算：基于编辑请求和上下文模板（context\_template\_length\_params动态生成）计算输入梯度方向

​右向量优化：通过梯度下降（v\_num\_grad\_steps控制迭代次数）调整输出差异方向

*# 参考网页2参数空间设计*space = { 'v\_lr': hp.loguniform('v\_lr', np.log(0.0001), np.log(0.1)), 'clamp\_norm\_factor': hp.uniform(0.5, 2.0)}

**低秩更新注入**：生成秩一矩阵ΔW = u⊗v，叠加到目标权重（rewrite\_module\_tmp指定路径）

**稳定性保障机制**

​梯度裁剪：clamp\_norm\_factor限制梯度范数，防止数值不稳定（类似网页2早停机制）

​统计修正：mom2\_adjustment启用时，基于mom2\_dataset数据集计算激活分布协方差矩阵，修正更新方向

​上下文多样性：context\_template\_length\_params生成多长度模板（如[[15,3]]生成3个15词模板），增强编辑鲁棒性

#### ​****关键参数功能对照表****

| **参数组** | **核心参数** | **作用** | **技术原理** |
| --- | --- | --- | --- |
| ​**结构控制** | layers rewrite\_module\_tmp | 指定修改的Transformer层及权重路径 | 网页2层级参数优化 |
| ​**优化控制** | v\_num\_grad\_steps v\_weight\_decay | 右向量优化步数、L2正则强度 | 网页2梯度下降调参 |
| ​**统计修正** | mom2\_adjustment mom2\_n\_samples | 启用二阶矩修正、统计采样数 | 协方差矩阵稳定更新方向 |
| ​**上下文生成** | context\_template\_length\_params | 控制模板生成策略（长度、数量） | 动态参数空间扩展 |

#### ​****技术关联与最佳实践****

**参数空间设计**​（参考网页2经验）：

对连续参数（如v\_lr）使用对数尺度分布（hp.loguniform）

层级参数（如layers）采用hp.choice避免维度爆炸

**性能优化**：

启用mom2\_adjustment可提升更新稳定性，但需权衡计算开销（mom2\_n\_samples不宜过大）

设置clamp\_norm\_factor=1.2平衡梯度裁剪强度与收敛速度

**调试建议**：

优先验证layers参数有效性（中间层通常更敏感）

监控kl\_factor对模型原始知识保留的影响

该配置类通过模块化设计实现了ROME算法的高效控制，其参数体系融合了贝叶斯优化与统计修正思想，适用于需要精准修改模型知识的场景（如事实更新、逻辑修正）。