基于SAE的金融LLM可解释性研究

# 技术路线图



# 语料库的选择

## 东方财富

FinGPT，中文语料库的使用，[open-compass/OpenFinData (github.com)](https://github.com/open-compass/OpenFinData)

OpenFinData是由东方财富与上海人工智能实验室联合发布的开源金融评测数据集。该数据集代表了最真实的产业场景需求，是目前场景最全、专业性最深的金融评测数据集。它基于东方财富实际金融业务的多样化丰富场景，旨在为金融科技领域的研究者和开发者提供一个高质量的数据资源。

优点：数据集模块多，缺点：数量少，为json文件，压缩大小为400KB，总共为1500条数据记录

## 轩辕开源

[Duxiaoman-DI/FinCorpus · Datasets at Hugging Face](https://huggingface.co/datasets/Duxiaoman-DI/FinCorpus)

中文金融数据集FinCorpus，包含上市公司公告、金融资讯、金融新闻和金融试题四个模块



## 最终选择

Anthropic在进行大模型可解释性分析时，使用The Pile数据集，并使用四千万的tokens来生成八十亿的数据集。所有在金融大模型领域进行可解释时，其金融语料库的内容需要丰富并具大规模的特点。

其中轩辕大模型开源的FinCorpus语料库，其数据丰富，包括上市公司公告、金融资讯、金融新闻以及金融试题，满足构建用于SAE训练的数据集的要求。

# 金融大模型的选择

## 基于GPT2微调

MLP层神经元为768个。

## Fingpt

基于ChatGLM2-6B进行微调。其MLP层神经元为4096个。

推理demo：[FinGPT/fingpt/FinGPT\_Sentiment\_Analysis\_v1/FinGPT\_v1.0/inferencing/infer.ipynb at master · AI4Finance-Foundation/FinGPT (github.com)](https://github.com/AI4Finance-Foundation/FinGPT/blob/master/fingpt/FinGPT_Sentiment_Analysis_v1/FinGPT_v1.0/inferencing/infer.ipynb)

## 轩辕

从头开始训练，和Fingpt类似的结构。不同在于fingpt-7B是在chatGLM/Llama2上微调，而轩辕-6B是采用类LLaMA架构，从零开始进行预训练的金融大模型。

## 大模型推理对比

### 问题一

输入: 某人向银行借款10万元，借款期限是36个月，年利率是6%，还款方式是等额本金，则第二个月的利息是多少，列出计算公式和答案

#### 轩辕

输出: 在等额本金还款方式下，每月还款金额由两部分组成：一部分是每月应还的本金，另一部分是剩余本金产生的利息。

首先，我们计算每月应还的本金金额。总借款额是10万元，借款期限是36个月，所以每月应还本金为：

每月应还本金 = 总借款额 / 还款月数

每月应还本金 = 100,000元 / 36月

每月应还本金 = 2777.78元（保留两位小数）

接下来，我们计算第二个月的利息。在等额本金还款方式中，第二个月的利息是基于初始借款额计算的。

第一个月的剩余本金 = 总借款额 - 第一个月已还本金

第一个月的剩余本金 = 100,00

## 最终模型的选取

需要考虑进行大模型可解释性分析时，模型结构对显存占用的消耗。

# SAE数据集的生成

## 使用HookedTransformer加载金融大模型

在SAE\_lens包中，是使用的Transformer\_lens包中的HookedTransformer来读取常见的预训练模型。而金融大模型不能直接通过HookedTransformer读取。

例如轩辕-6B模型，其使用Transformers包中的LlamaForCausalLM来读取其预训练模型，需要将LLaMA的权重转为HuggingFace的权重，并使用HookedTransformer来读取。

## 相关参考代码：

### 模型权重转换

[LLaMA.ipynb - Colab (google.com)](https://colab.research.google.com/github/TransformerLensOrg/TransformerLens/blob/main/demos/LLaMA.ipynb#scrollTo=sMkmqMjkVdMq)

转为LLaMA的权重为HuggingFace的权重，并使用HookedTransformer来读取。

# SAE用于可解释性分析



## 占用显存分析

### SAE模型占用显存



M=神经元数量，N=特征数量，F=数据精度。

总的显存占用=(M \* B \* F + M \* N \* F \* 2 + M \* B \* F + N \* B \* F + M \* F + N \* F)/(1024\*\*3)GB

之下表格为B=1024，F=4时的情况。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| M/N | 2\*\*12(4K) | 2\*\*15(32K) | 2\*\*17(131K) | 2\*\*20(1M) | 2\*\*22(4M) | 2\*\*25(34M) |
| 512 | 0.03 | 0.25 | 1.00 | 8.00 |  |  |
| 768 | 0.04 | 0.31 | 1.25 | 10.00 | 40.02 | 320.13 |
| 1024 | 0.05 | 0.38 | 1.50 | 12.01 | 48.02 | 384.13 |
| 4096 | \ | 1.15 | 4.53 | 36.03 | 144.04 | 1152.15 |

### 数据shuffle占用显存

sae\_lens的shuffle机制，并不是对全部数据进行统一shuffle，而是分批载入的，二合一后分为二

buffer\_size=store\_batch\_size\_prompts \* (n\_batches\_in\_buffer / 2) \* context\_size \* d\_model

buffer\_count=9+5\*2。

总的显存占用=buffer\_count \* buffer\_size





### 实际消耗

n\_batches\_in\_buffer简称为nbib，expansion\_factor简称为ef

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| model | 大小 | d\_model | ef | 激活 | dtype | batchsize | nbib | GPU |
| tiny-stories-1L-21M | 13M | 1024 | 16 | Relu | float32 | 4096 | 64 | 13.2 |
| 13M | 1024 | 16 | topk | float32 | 4096 | 64 | 13.6 |
| 13M | 1024 | 16 | Relu | float16 | 4096 | 64 |  |
| 13M | 1024 | 16 | topk | float16 | 4096 | 64 | 9.7 |
| 13M | 1024 | 16 | topk | float16 会导致loss变为nan | 1024 | 64 | 9.8 |
| 13M | 1024 | 16 | topk | float32 | 4096 | 32 | 9.6 |
| gpt2-small | 85M | 768 | 16 | topk | float32 | 4096 | 64 | 18.2 |
| 85M | 768 | 16 | topk | float32 | 4096 | 32 | 14.1 |
| 85M | 768 | 16 | topk | float32 | 1024 | 64 | 18.3 |

## SAE的变体

### gated SAE

### TopK SAE

## SAE评价指标分析

如何评估SAE，目前存在L0和Loss Recovered两个指标 。L0是SAE编码中非零元素的平均数。Loss Recovered是用重建的激活替换GPT或者其他大模型的原始激活并测量不完美重建的额外损失。这两个指标之间通常存在跷跷板效应，需要进行权衡。毕竟SAE会选择降低重建精度以增加稀疏性。

### Anthropic对特征的评价分析[1] ：

1. 手动检查：这些特征是否看起来可解释？
2. 特征密度：我们发现“实时”特征的数量和它们触发的标记百分比是极其有用的指南。（详情请参阅附录。）
3. 重构损失：自动编码器如何很好地重构 MLP 激活？最终目标是解释 MLP 层的功能，因此 MSE 损失应该很低。
4. 玩具模型：拥有玩具模型，可以在其中了解基本事实，从而可以清晰地评估自动编码器的性能。

基于以上分析，结合目前开源工具SAE\_lens，本研究拟采用以下几种评估指标对sae训练性能进行评估。（get\_downstream\_reconstruction\_metrics函数和get\_sparsity\_and\_variance\_metrics函数）

### SAE模型在重建任务中的指标

在金融大模型中使用自编码器（SAE）来替换某一层的激活，并通过前向钩子（forward hook）进行重建。

通过大模型计算原始 logits （logits：神经网络输出的未经过激活函数的原始分数）和交叉熵损失，然后定义一个标准的替换钩子函数，用于在前向传播过程中替换某一层的激活，最后使用这个钩子函数运行大模型，获取重构的 logits 和交叉熵损失，分析重构logits和原始logits之间的kl散度和交叉熵损失。

（1）KL散度

i.对原始 logits和重构logits进行softmax变换，得到原始分布original\_probs和重构分布new\_probs。

ii. 计算原始分布和重构分布的对数概率log\_original\_probs和log\_new\_probs

iii.计算kl散度。kl\_div = original\_probs \* (log\_original\_probs - log\_new\_probs)

（2）交叉熵损失（ce loss）

计算加入sae替换后的重构损失ce\_loss\_with\_sae和原始损失original\_ce\_loss。

### L2范数

L2范数由以下公式得到：

其中，表示第i个样本的输入激活向量。

该研究拟计算输入sae和输出激活的L2范数以及他们的比值作为评估指标。

### 稀疏度度量

（1）L0范数

L0范数用于表示非零特征的个数。

其中，表示第i个样本的第j个特征激活，是指示函数，当条件为真时取值为1，否则取值为0。即，L0范数表示非零特征的个数。

（2）L1范数

其中，表示第i个样本的第j个特征激活。即，L1范数表示特征激活的绝对值之和。

### 方差

（1）解释方差。

解释方差由以下公式得到。

其中，表示第i个样本的第j个输入激活，表示第i个样本的第j个输出激活, 表示第j个特征激活在所有样本中的平均值。

（2）均方误差

均方方差由以下公式得到。

其中，表示第i个样本的第j个输入激活，表示第i个样本的第j个输出激活。

**特征可解释性**

[sparse\_autoencoder/sparse\_autoencoder/explanations.py at main · openai/sparse\_autoencoder (github.com)](https://github.com/openai/sparse_autoencoder/blob/main/sparse_autoencoder/explanations.py) 提供了一种解释模型激活值的方法，使用 Trie 数据结构高效存储和查询模式，并通过显著性值计算来确定每个 token 的重要性。

核心功能说明

Trie 数据结构：用于高效存储和查询模式。Trie 的每个节点代表一个 token 或者特殊标识符（例如 \_ANY\_TOKEN），叶节点存储激活（\_SALIENCY\_KEY）。

激活值计算与存储：通过模型函数计算每个 token 序列的激活值，并存储在 Trie 中。显著性值计算：通过替换 token 并计算变化后的激活值来确定每个 token 的显著性。最小子序列确定：从 token 序列末尾开始，找到能解释激活值的最小子序列。

**重建可解释性**

如果我们只使用模型中我们可以解释的部分，我们会牺牲多少性能？

这建议将我们的下游损失和解释指标结合起来，通过使用我们的解释来模拟自动编码器潜伏，然后在解码后检查下游损失。该指标还有一个优点，即它以原则性的方式同时重视召回率和精确度，并且还更重视激活更密集的潜在变量的召回率。

## SAE的参数选择

### 其他论文的参数选择

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | [1] | [2] | | [3] | | |
| d\_in | 512 |  | |  | | |
| batch\_size |  | 2048、  4096 | |  | | |
| train\_step |  | 200k | |  | | |
| 重构loss | L2 |  | |  | | |
| 隐藏层loss | L1 |  | |  | | |
| 优化器 |  | Adam  beta1=0.9  beta2=0.999 | |  | | |
| 学习率 |  | 5e-5 | |  | | |
| L1系数 |  | 5 | | [5e-4,1e-2] | | |
| gradient norm |  | 1 | |  | | |
| Bias初值 |  |  | | 样本的几何均值 | | |
| 减少死特征 |  |  | | AuxK，类似鬼梯度  init\_encoder\_as\_decoder\_transpose | | |
| 死特征的数量 |  | 1M：2%  4M：35%  34M：64% | |  | | |
|  | | | [1] | | [2] | [3] |
| d\_in | | | 512 | |  |  |
| This must match the size of the hook in the model where the SAE is trained. | | | | | | |
| expansion\_factor | | | 8,16 | |  |  |
| **d\_sae**: The hidden layer of the SAE will have size **`expansion\_factor \* d\_in`** | | | | | | |
| train\_batch\_size\_tokens | | |  | | 2048、  4096 |  |
| l1\_coefficient | | |  | | 2，5 |  |
| **This controls how much sparsity the SAE will have after training.** | | | | | | |
| training\_tokens | | |  | |  |  |
| The total tokens used for training. total\_training\_steps \* train\_batch\_size\_tokens | | | | | | |
| train\_batch\_size\_tokens | | |  | |  |  |
| The batch size used for training. Adjust this to keep the GPU saturated. | | | | | | |
| context\_size | | |  | | 1024 |  |
| training batch will be tokens of size train\_batch\_size\_tokens x context\_size | | | | | | |
| train\_step | | |  | | 200k |  |
| b\_dec\_init\_method | | |  | | geometric\_median |  |
| geometric\_median: Initializing the decoder bias with an estimate of the geometric median of the activations. | | | | | | |
| **normalize\_sae\_decoder** | | |  | |  |  |
| **noise\_scale** | | |  | |  |  |
| **apply\_b\_dec\_to\_input** | | |  | |  |  |
| **decoder\_orthogonal\_init** | | |  | |  |  |
| **decoder\_heuristic\_init** | | |  | | Anthropic |  |
| **init\_encoder\_as\_decoder\_transpose** | | |  | |  |  |
| finetuning\_tokens | | |  | | 是一个较大的数，在run..lpynb文件中，和Activation  有关 |  |
| normalize\_activations | | |  | |  |  |
| prepend\_bos标记 | | |  | |  |  |
| autocast(自动类型转换，可以节省显存大小) | | |  | |  |  |
| autocast\_llm | | |  | |  |  |
| scale\_sparsity\_penalty\_by\_decoder\_norm | | |  | |  |  |
| mse\_loss\_normalization | | |  | | 有两种可选的的方式，便准方式和归一化方式 |  |
| l1\_warm\_up\_steps | | |  | |  |  |
| lr | | |  | |  |  |
| lr\_scheduler\_name | | |  | | 三种方式 |  |
| lr\_warm\_up\_steps | | |  | |  |  |
| lr\_end(当lr\_scheduler\_name 设置为consineannealing时，需要设置lr\_end  ) | | |  | |  |  |
| lr\_decay\_steps | | |  | |  |  |
| n\_restart\_cycles | | |  | |  |  |
| finetuning\_method | | |  | |  |  |
| use\_ghost\_grads | | |  | |  |  |
| feature\_sampling\_window | | |  | |  |  |
| dead\_feature\_window | | |  | |  |  |
| dead\_feature\_threshold | | |  | |  |  |
| 重构loss | | | L2 | |  |  |
| 隐藏层loss | | | L1 | |  |  |
| 学习率 | | |  | | 5e-5 |  |
| gradient norm | | |  | | 1 |  |
| Bias初值 | | |  | |  | 样本的几何均值 |
| ghost grads | | |  | |  | AuxK，类似 |

初始化解码器的偏差为几何中心b\_dec\_init\_method

初始化编码器和解码器相同的方向？可能指的就是init\_encoder\_as\_decoder\_transpose？

解码器的潜在方向缩放为单位范数？*normalize\_sae\_decoder*

辅助损失？*use\_ghost\_grads*

Adam epsilon: ε = 6.25 × 10−10

L(C)和L(N)是什么东东？

### 参数选择的建议

1. 在训练中出现的死亡神经元的数量似乎取决于许多因素，包括但不限于:学习速率(太高)；批量大小(过低)；数据集冗余(每个上下文的标记太多或同一数据集上重复的历元)；训练步骤数(太多)；使用的优化器的问题[1] 。
2. 文献[1] 中尤其提到低的学习率会导致更多有效的特征以及更低的loss。

# 特征可视化

## 相关参考代码

### github开源代码1：

[GitHub - callummcdougall/sae\_vis: Create feature-centric and prompt-centric visualizations for sparse autoencoders (like those from Anthropic&apos;s published research).](https://github.com/callummcdougall/sae_vis)

[文件夹 - Google 云端硬盘](https://drive.google.com/drive/folders/1sAF3Yv6NjVSjo4wu2Tmu8kMh8it6vhIb)

包含feature-centric的可视化，复现Anthropic的可视化，即我们查看一个特定feature并查看哪些tokens在该功能上触发最强。

以及以prompt-centric的可视化，查看一次特定prompt并根据各种不同的指标查看哪些feature在该prompt上触发最强。

### github开源代码2：

[sparse\_autoencoder/sae-viewer at main · openai/sparse\_autoencoder (github.com)](https://github.com/openai/sparse_autoencoder/tree/main/sae-viewer)

### 可视化网站

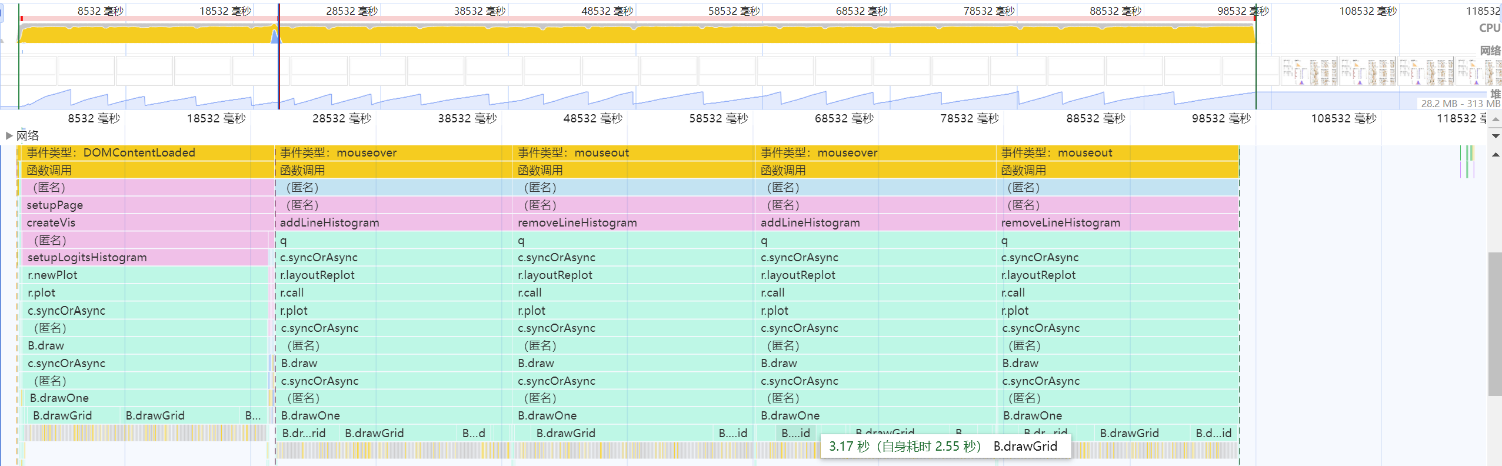
[Neuronpedia](https://www.neuronpedia.org/)

## sae\_vis包，性能分析

生成部分和html打开部分

### 可视化生成性能

### 可视化结果，HTML查看性能



## 可视化示例

# 自动解释性评分

## 参考代码

[automated-interpretability/neuron-explainer/demos/generate\_and\_score\_explanation.ipynb at main · openai/automated-interpretability (github.com)](https://github.com/openai/automated-interpretability/blob/main/neuron-explainer/demos/generate_and_score_explanation.ipynb)

# 引导

## 相关参考代码

### Github开源代码1：

[SAELens/tutorials/using\_an\_sae\_as\_a\_steering\_vector.ipynb at main · jbloomAus/SAELens (github.com)](https://github.com/jbloomAus/SAELens/blob/main/tutorials/using_an_sae_as_a_steering_vector.ipynb)

## 引导输出

### 输出1

set\_seed(42)  
cfg = config(  
 model\_name="/root/data/sae/LLMmodel/XuanYuan-6B-Chat",  
 sae="/root/data/sae/sae\_checkpoint/pcc1n73m/final\_3072000",  
 hook\_point="blocks.0.hook\_mlp\_out",  
 feature\_index=12,  
)  
example\_prompt = "作为一个独立的法人实体，寿险公司的法人财产权是如何实现的？"

#### 开启引导之前：

87%|████████████████████████████████████████████████████████████████████████████████████████████████████████████████▍ | 886/1024 [01:53<00:17, 7.83it/s]

作为一个独立的法人实体，寿险公司的法人财产权是如何实现的？

寿险公司作为一个独立的法人实体，其法人财产权的实现是一个复杂的法律过程，涉及以下几个方面：

1. 公司设立：

- 根据相关法律法规，寿险公司需要经过国家金融监管部门的批准，并完成工商注册、税务登记等法律程序，取得法人资格。

- 公司设立过程**中**需要确定公司章程，明确公司的组织架构、经营范围、权力分配等，为公司的运营奠定基础。

2. 财产拥有：

- 寿险公司拥有自己的财产，包括现金、有价证券、投资性房产等，这些财产成为公司经营活动的物质基础。

- 公司对其财产享有占有、使用、收益和处分的权利，可以通过投资、借贷、抵押等方式增值和运用这些财产。

3. 权利行使：

- 寿险公司依据法律和公司章程，通过股东大会、董事会等治理结构行使其权利。

- 股东大会是公司的最高权力机构，决定公司的重大事项，如选举董事会、修改公司章程等。

- 董事会是公司的执行机构，负责公司日常经营活动的决策，包括选聘管理层、审议财务报告等。

4. 权利保护：

- 寿险公司的财产权利受到法律保护，任何侵犯公司财产权益的行为，如盗窃、挪用等，都将受到法律追究。

- 公司可以通过法律途径维护自己的权益，包括提起诉讼、申请仲裁等。

5. 财务透明：

- 寿险公司需要定期公布财务报告，包括资产负债表、利润表、现金流量表等，以保证财务信息的透明度。

- 投保人和股东等利益相关者可以通过查阅财务报告，了解公司的财务状况和经营成果，从而监督公司的运营。

6. 风险管理：

- 寿险公司面临市场风险、信用风险、流动性风险等多种风险，需要建立健全的风险管理体系，确保公司财产的安全。

- 公司可以通过购买保险、建立风险准备金等方式，对潜在的风险进行管理和控制。

7. 责任承担：

- 寿险公司作为独立法人实体，对外承担民事责任。

- 如果公司的行为给第三方造成损失，公司需要依法承担赔偿责任。

- 在公司清算或破产时，公司财产将用于清偿债务，剩余部分可能会分配给股东或债权人。

通过以上方式，寿险公司的法人财产权得以实现和保护，确保公司作为一个独立实体的合法地位和经营活动的有效进行。</s>

#### 开启引导之后：

71%|████████████████████████████████████████████████████████████████████████████████████████████▏ | 726/1024 [01:29<00:36, 8.13it/s]

作为一个独立的法人实体，寿险公司的法人财产权是如何实现的？

寿险公司作为一种特殊的法人实体，其法人财产权的实现主要依赖于以下几个方面：

1. 股权结构：寿险公司的资本构成通常包括股东投资的资本金，这些资本金代表了股东对公司的所有权。股东通过购买公司的股票或者以其他形式参与公司的投资，从而成为公司的所有者之一。股东的权益体现在他们所持有的股份比例上，并通过股东大会来行使对公司经营的决策权和监督权。这种所有权结构为寿险公司的法人财产权提供了基础。

2. 投资活动：寿险公司的主要经营活动之一是投资，通过购买债券、股票、房地产等资产来获取收益。公司的投资活动所获得的收益（如利息、股息和资本利得）构成了公司的利润来源。这些收益在扣除成本和税费后，剩余部分将成为公司的内部积累，即公司的法人财产。

3. 合同关系：寿险公司通过与客户签订保险合同，获得了客户支付的保费。这些保费构成了公司的现金流入，也是公司法人财产的重要组成部分。通过保险合同，公司承诺在未来的特定时间向客户支付保险金或给付保险利益，这是公司法人财产权的另一种体现。

4. 法律保护：寿险公司的法人财产权受到国家法律的保护。**在中国**，根据《**中华**人民共和国公司法》等相关法律法规，寿险公司作为独立的法人实体，享有独立的法人财产权，包括但不限于对其资产的占有、使用、收益和处分权。这些权利受到法律的保护，任何单位和个人不得侵犯。

5. 公司治理：寿险公司通过建立健全的公司治理结构，确保法人财产权的有效实现。公司治理包括股东大会、董事会、监事会等机构，以及相应的议事规则和决策程序。这些机构和程序的设置旨在保障股东和其他利益相关者的权益，同时确保公司经营的合法性和效率性。通过公司治理结构，寿险公司能够合理分配和使用其法人财产，实现公司价值的最大化。</s>

### 输出二

set\_seed(42)  
cfg = config(  
 model\_name="/root/data/sae/LLMmodel/XuanYuan-6B-Chat",  
 sae="/root/data/sae/sae\_checkpoint/mb1j2uao/final\_768000",  
 hook\_point="blocks.0.hook\_mlp\_out",  
 feature\_index=3,  
)  
  
example\_prompt = "有关麦考利久期的正确表述是什么?"  
steering(cfg).run(sv\_prompt, example\_prompt)

有关麦考利久期的正确表述是什么?

#### 开启引导之前

正确的表述应该是：麦考利久期是衡量固定收益证券或者是一个投资组合平均到期时间的一种度量方法，它反映了投资者收回其在债券或者投资组合中的全部本金的加权平均时间。这种度量方法的核心在于考虑了各期现金流的现值以及其距离现在的时间点，而不仅仅是现金流的金额。

具体来说，麦考利久期的计算公式是：

D\_{mac}=1\times\frac{PV\_{1}}{PV\_{1}+PV\_{2}+\ldots+PV\_{n}}+2\times\frac{PV\_{2}}{PV\_{1}+PV\_{2}+\ldots+PV\_{n}}+\ldots+n\times\frac{PV\_{n}}{PV\_{1}+PV\_{2}+\ldots+PV\_{n}}

其中：

- \(D\_{mac}\) 表示麦考利久期

- \(PV\_{i}\) 表示第 \(i\) 期现金流的现值

- \(n\) 表示投资期限的总期数

通过这个公式，我们可以计算出投资组合或者是单一债券的平均到期时间，这个时间可以反映投资者在持有债券到期所需要的时间跨度。因为考虑了现金流的现值和时间因素，麦考利久期比修正久期更加全面，能够更好地反映出投资者的资金流动性风险。</s>

#### 开启引导之后：

39%|██████████████████████████████████████████████████▏ | 395/1024 [00:56<01:30, 6.96it/s]

有关麦考利久期的正确表述是什么?

正确答案：麦考利久期是衡量债券或其他金融工具的平均还款期限的一种度量方式，它是债券或贷款的平均收回成本的加权平均时间。其主要思想是将债券的每次现金流（包括利息支付和本金偿还）的现值乘以时间因子，然后求和得出债券的加权平均期限。

麦考利久期的计算公式为：D=C×[1-1/(1+y)^-n]+D×y×[1-1/(1+y)^-n]/(1+y)^n，其中，C是债券的年度现金流（即利息支付），y是债券的到期收益率，n是债券的期限。

具体来说，麦考利久期可以帮助投资者理解和比较不同期限的债券的利率风险。如果一个债券的麦考利久期比另一个债券的久期长，那么前者比后者更能抵御利率变动带来的风险。这是因为，当市场利率上升时，麦考利久期较长的债券的价格下跌的幅度会比麦考利久期较短的债券的价格下跌的幅度小。反之，当市场利率下降时，麦考利久期较短的债券的价格上涨的幅度会比麦考利久期较长的债券的价格上涨的幅度大。</s>

## 特征消融（另一种形式的引导）

# 参考文献

1. Bricken T, Templeton A, Batson J, et al. Towards monosemanticity: Decomposing language models with dictionary learning[J]. Transformer Circuits Thread, 2023, 2.
2. Templeton A. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet[M]. Anthropic, 2024.
3. Gao L, la Tour T D, Tillman H, et al. Scaling and evaluating sparse autoencoders[J]. arXiv preprint arXiv:2406.04093, 2024.