改进大规模训练稀疏自编码器的方法

Ref： https://mp.weixin.qq.com/s/iZHPnnIncVFa8QJOuH8qFg

神经网络中的激活通常表现出不可预测和复杂的模式，且每次输入几乎总会引发很密集的激活。而现实世界中其实很稀疏，在任何给定的情境中，人脑只有一小部分相关神经元会被激活。

研究人员开始研究稀疏自编码器，这是一种能在神经网络中识别出对生成特定输出至关重要的少数“特征”的技术，类似于人在分析问题时脑海中的那些关键概念。

在 OpenAI超级对齐团队的这项研究中，他们推出了一种基于 **TopK**激活函数的新稀疏自编码器（SAE）训练技术栈，消除了特征缩小问题，能够直接设定 L0（直接控制网络中非零激活的数量）。

具体来看，他们使用 **GPT-2 small**和 **GPT-4** 系列模型的残差流作为自编码器的输入，选取网络深层（接近输出层）的残差流，如 GPT-4的 5/6层、 GPT-2 small的第 8层。

并使用之前工作中提出的基线 **ReLU**自编码器架构，编码器通过 ReLU激活获得稀疏 latent z，解码器从 z 中重建残差流。损失函数包括重建 MSE损失和 L1正则项，用于促进 latent稀疏性。

$$z = \mathrm{ReLU}(W_{\mathrm{enc}}(x - b_{\mathrm{pre}}) + b_{\mathrm{enc}})$$

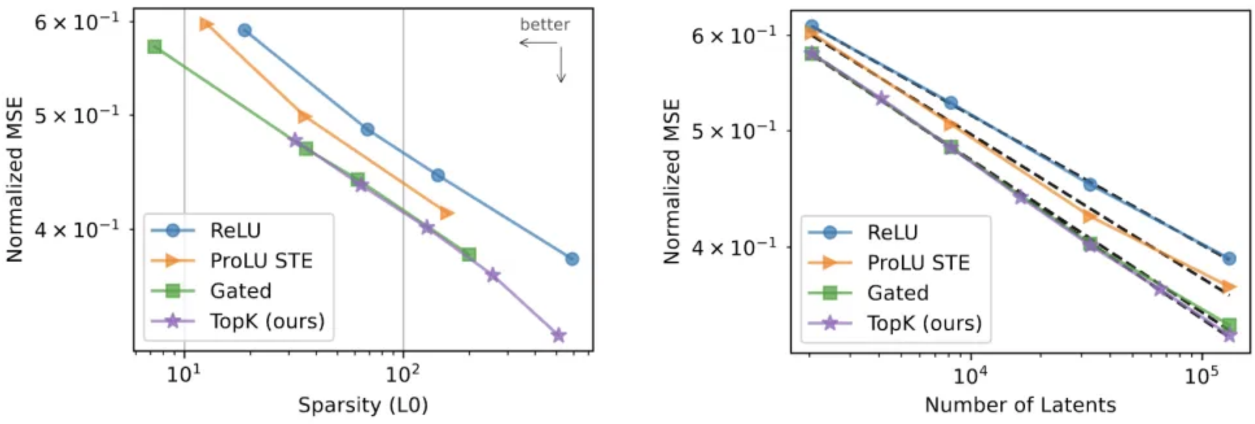
$$\hat{x} = W_{\mathrm{dec}} z + b_{\mathrm{pre}}$$

公众号 · 量子位

然后，团队提出使用 TopK激活函数代替传统 L1正则项。 TopK在编码器预激活上只保留最大的 k 个值，其余清零，从而直接控制 latent稀疏度 k。

$$z = \mathrm{TopK}(W_{\mathrm{enc}}(x - b_{\mathrm{pre}}))$$

不需要 L1正则项，避免了 L1导致的激活收缩问题。实验证明，TopK相比 ReLU 等激活函数，在重建质量和稀疏性之间有更优的权衡。



(a) At a fixed number of latents ($n = 32768$), TopK has a better reconstruction-sparsity trade off than ReLU and ProLU, and is comparable to Gated.

(b) At a fixed sparsity level ($L_0 = 128$), scaling laws are steeper for TopK than ReLU.[6]

Figure 2: Comparison between TopK and other activation functions.

公众号 · 量子位

此外，自编码器训练时容易出现大量latent永远不被激活（失活）的情况，导致计算资源浪费。

团队的解决方案包括两个关键技术：

- 将编码器权重初始化为解码器权重的转置，使latent在初始化时可激活。
- 添加辅助重建损失项，模拟用top-kaux个失活latent进行重建的损失。

如此一来，即使是**1600**万**latent**的大规模自编码器，失活率也只有**7%**。

最后，论文一作表示稀疏自编码器的问题仍然远未解决，这项研究中的SAE只捕获了GPT-4行为的一小部分，即使看起来单义的latent也可能难以精确解释。而且，从表现优异的SAE到更好地理解模型的行为，还需要大量的工作。

Revision #1
Created 11 January 2025 09:44:04 by Colin
Updated 11 January 2025 09:44:07 by Colin