

Transformer 注意力系列（一）：注意力位置关系建模的研究

朱梦

初稿于 2025 年 7 月 3 日，修改于 2025-07-04

1. CNN vs. RNN vs. 注意力

在博客《卷积感受野大小的研究》中分析出，卷积神经网络（Convolutional Neural Network, CNN）模型需要堆叠多层才能捕获全局位置依赖关系。但是，面向每层都需要捕捉全局位置感受野的任务场景，CNN 模型无法胜任。在博客《RNN 位置关系建模的研究》中分析出，非线性循环神经网络（Recurrent Neural Network, RNN）模型需要递归才能捕获全局位置依赖关系，无法充分利用设备的并行特性。针对此问题，线性 RNN 被提出，即能捕获全局位置依赖关系，又具有 $\mathcal{O}(s)$ 推理效率。然而，不管是 CNN 模型还是 RNN 模型均为隐式建模位置关系，而注意力模型则是显示建模位置关系，直觉上会更好。任何硬币都有两面，更好的建模性能往往意味着更高的计算代价 $\mathcal{O}(s^2)$ ，反过来更高的计算代价也往往意味着更好的建模性能。

2. 注意力层

设查询 $Q \in \mathbb{R}^{s_Q \times d_K}$ ，键 $K \in \mathbb{R}^{s_K \times d_K}$ ，值 $V \in \mathbb{R}^{s_K \times d_V}$ ，那么点积注意力定义为：

$$\text{Attn}(Q, K, V) = \text{Softmax}(\kappa QK^T)V \quad (1)$$

其中， κ 表示缩放因子。图1形象地展示了点积注意力的计算流程。

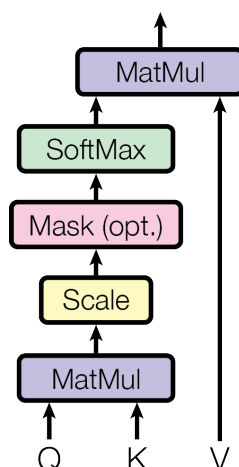


图1 缩放点积注意力

如何理解这种结构呢？不妨逐个向量来看：

$$\text{Attn}(q_i, K, V) = \sum_{j=1}^{s_K} \frac{1}{Z} e^{\kappa \langle q_i, k_j \rangle} v_j = \sum_{j=1}^{s_K} \frac{e^{\kappa \langle q_i, k_j \rangle}}{\sum_{m=1}^{s_K} e^{\kappa \langle q_i, k_m \rangle}} v_j \quad (2)$$

其中， Z 表示归一化因子。式(2)表示，先通过 q_i 和各个 k_j 作点积并 Softmax 的方式，以得到 q_i 和各个 v_j 的相似度，然后，加权求和得到一个 d_V 向量。图2形象地展示了如何理解这种结构。

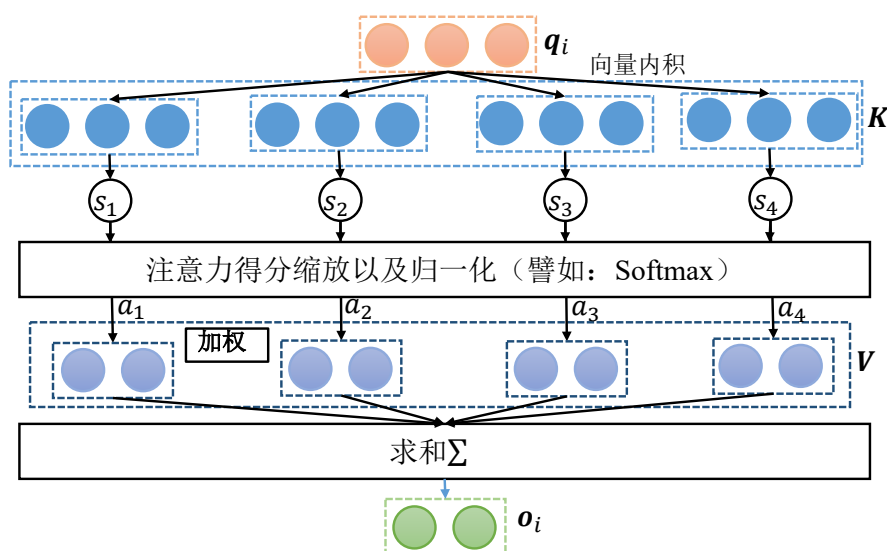


图2 逐向量理解缩放点积注意力

3. 多头注意力层

多头注意力层（Multi-Head Attention, MHA）把 Q K V 通过矩阵参数仿射投影，然后再做 Attention。此过程重复做 h 次，结果拼接起来。整个过程为：

$$\begin{aligned} \text{MHA}(Q, K, V) &= \text{Concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_h), \\ \text{head}_i &= \text{Attn}(Q\Theta_{Q,i}, K\Theta_{K,i}, V\Theta_{V,i}) \end{aligned} \quad (3)$$

图3形象地展示了 MHA 的计算流程。

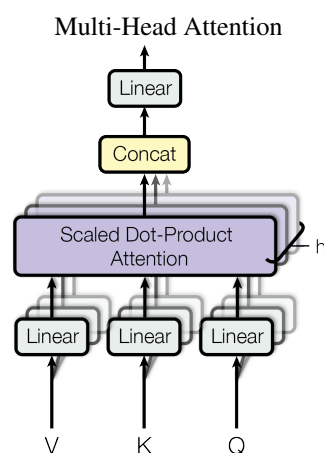


图3 逐向量理解缩放点积注意力

4. 自注意力层

当 $Q = K = V$ 时，即为自注意力（Self-attention）。在 Google 的论文中，所用的为多头自注意力。