

- Attention：本质是一种加权矩阵乘法

即：预测结果 = 矩阵权重 \times 值

N 和 d 的取值

例：用腰围预测体重

腰围	key	体重	Value
51	k_1	40	v_1
56	k_2	43	v_2
58	k_3	46	v_3

现需预测 腰围 = 55 的体重，如何预测？

方法1：通过建立回归模型预测：体重 $y = c + a \cdot \text{腰围} + \epsilon$

通过最小二乘估计 or 极大似然估计求得参数 a 使得模型的预测值 \hat{y} 和实际值 y 最小

$$\text{即有 loss function} = (y - \hat{y})^2 = (y - ax - c)^2$$

$$\text{最小二乘} = \frac{\partial f}{\partial a} = -2a(y - ax - c) = 0 \text{ 求得 } \hat{a}$$

回归的思路是先根据数据 (key 和 value) 建立与原始数据最相似的模型

再代入想知道的问题 (Query) 进行询问 / 预测

方法2：利用 Query 与现有自变量数据 (key) 做相似性对比

得出权重，该权重表示 Query 和 key 与每一个数据的相似性

再用该权重与每个对应的 Value 的值做加权求和，得到 Query 的预测结果

① 定义权重函数：一般使用 softmax

$$k_i \text{ 关于 } q \text{ 的权重 } \alpha(q, k_i) = \text{softmax}\left(-\frac{1}{2}(q - k_i)^2\right) = \frac{\exp(-\frac{1}{2}(q - k_i)^2)}{\sum_{j=1}^n \exp(-\frac{1}{2}(q - k_j)^2)}$$

根据 Q 想知道的预测值即为

$$D = \alpha(q, k_1) \cdot v_1 + \alpha(q, k_2) \cdot v_2 + \dots + \alpha(q, k_n) \cdot v_n$$

$$\Rightarrow D = \sum_{i=1}^n \alpha(q, k_i) \cdot v_i \quad n = \text{样本量}$$

二、多维 Softmax Attention 和复杂度

1. 上例仅使用简单的单自变量，单因变量问题

通过 Attention 机制可将问题拓展至多自变量，多因变量问题

例如通过年龄、身高、胖瘦预测体重、血压

可将其矩阵化，通过线性矩阵解

假设所有自变量为 k_{nd} ，所有因变量为 V_{np} n 特征量， d 为自变量个数， p 为因变量个数

则对于 Q_{nd} 的权重为 $\text{softmax}(Q \cdot k^T)$

为防止梯度消失，加上缩放因子，变为 $\text{softmax}(\frac{Q \cdot k^T}{\sqrt{d}})$

因此最终预测值为 $O = \text{softmax}(\frac{Q \cdot k^T}{\sqrt{d}}) \cdot V$

上式的 $\frac{Q \cdot k^T}{\sqrt{d}}$ 本质上是个相似性权重，因此同样也能进一步细化

$$O_i = \sum_{j=1}^n \frac{\text{sim}(Q_i, k_j)}{\sum_{j=1}^n \text{sim}(Q_i, k_j)} \cdot V_j, \text{ 其中 sim = simulating = Gauss kernel Function = } \exp(Q_i^T / \sqrt{d})$$

softmax Att. 的复杂度的计算：

1) $O(Qk^T) = O(n \times d \times n) = O(nd^2)$ ，其中 $Qk^T \in \mathbb{R}^{nm}$

2) $O(\text{softmax}(Qk^T)) = O(n^2)$ ，因为 $Qk^T \in \mathbb{R}^{nm}$

3) $O(\text{softmax}(Qk^T)V) = O(n \times n \times d) = O(nd^2)$

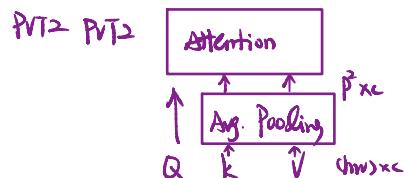
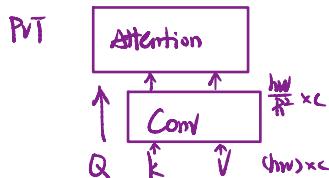
因此复杂度为 $O(n^2d)$

在 NLP 中一般没有大规模不了解程度，但在 CV 中则可能很大

因此有很多 CV 的方法解决该问题：Sparse Attn. (PVT 和 PVT2)、Window Attn. (Swin Transformer)

2. PVT、PVT2 和 Swin Transformer 技术分析

1) PVT 和 PVT2 本质都是对 k 和 V 下采样，来降低对图像信息的计算复杂度问题



但无法解决过大而导致输出信息有效性不足的问题（同秩化）

对于 Q 和 K：

行数 = 特征量

列数 = 自变量个数

例如 $Q = 56 \times 56$, 但 k 和 V 的采样率为 7×7 ,

也就是说最后的输出结果 D 是由 V 的 56×56 个线性组合构成的

因此 D 的秩会很小

2) Window Attn.: 为了解决所列的 window 之间无法计算 / 通信的问题, 提出了 Smith window Attn 方法

为了避免无效信息, 在此基础上又加入了掩码机制 (上次介绍过)

在不降低精度的前提下降低复杂度来提升计算速度和储存压力

综上所述, 基于 Softmax Attn. 计算复杂度一般为 $O(n^2d)$.

为了解决复杂度的问题也有许多课题研究.

但由于复杂度的根源是 Softmax Attn. 的 Gauss kernel Function

则也有研究提出使用线性变换降低复杂度

三. Linear Attention

上文提到, Softmax Attn. 可表示为 $D_i = \sum_{j=1}^n \frac{\text{Sim}(Q_i, k_j)}{\sum_{j=1}^n \text{Sim}(Q_i, k_j)} V_j$

以 V_i 为例: 有 $Q = xW_Q$, $k = xW_K$, $V = xW_V$, $W \in \mathbb{R}^{C \times C}$, $x \in \mathbb{R}^{m \times d}$

则复杂度化简为 $O(nd^2)$

但如果将计算复杂度上升的 Gauss kernel 换成其他的 kernel Function 呢?

引入 Linear Attn.: $\text{Sim}(Q, k) = \phi(Q) \cdot \phi(k)^T$. $\phi(\cdot)$ 可为 ReLU, 也可为 $\text{elu} = \begin{cases} x, & x > 0 \\ \alpha e^x - 1, & x \leq 0 \end{cases}$ 且称 elu 为线性核

则公式可直接用矩阵乘法化简:

$$D_i = \sum_{j=1}^n \frac{\phi(Q_i) \phi(k_j)^T}{\sum_{j=1}^n \phi(Q_i) \phi(k_j)^T} V_j$$

$$= \sum_{j=1}^n \frac{\phi(Q_i) \phi(k_j)^T V_j}{\sum_{j=1}^n \phi(Q_i) \phi(k_j)^T}$$

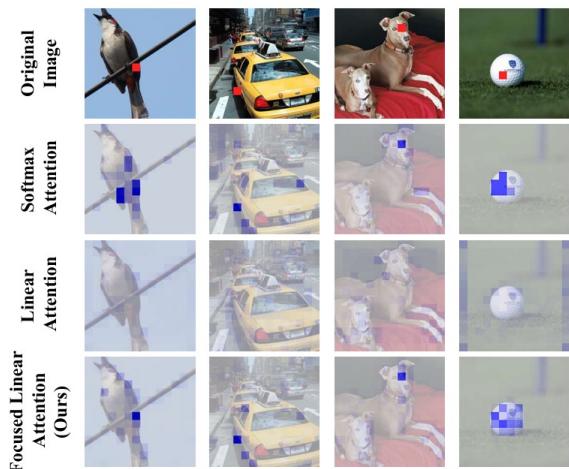
$$= \frac{\phi(Q_i) \left(\sum_{j=1}^n \phi(k_j)^T V_j \right)}{\phi(Q_i) \left(\sum_{j=1}^n \phi(k_j)^T \right)}$$

因此复杂度变为 $O(n \times d \times d) = O(nd^2)$, 相比 D_{softmax} 下降了一个 n 的数量级

但效果也降低了.

因此，于篇章设计了一种基于 Linear Attn. 但效果像 Softmax Attn. 的模型

四. Focused Linear Attn.



sharp \rightarrow softmax 突出

$$\text{softmax}(6, 2, 2) = [0.96, 0.02, 0.02]$$

smooth \rightarrow Average 不突出

$$\text{Normalization}(6, 2, 2) = [0.3, 0.14, 0.14]$$

因此，能设计一种不同于 Gauss kernel Function (softmax Attn.) 方法。

但同样能放大二者之间区别 (Trick)?

作者基于上述工作提出 Focused Function f_p

$$\text{有 } \text{Sim}(Q_i, k_j) = \phi_p(Q_i) \phi_p(k_j)^\top$$

$$\phi_p(x) = f_p(\text{relu}(Qx)), f_p(x) = \frac{\|x\|}{\|x\|_{\text{sum}^2}} \cdot x^{\text{sum}^2}, x^{\text{sum}^2} \text{ 是 } x \text{ 中元素的累加运算}$$

则问题变成了：为什么要设计这样的 Function? 有什么作用?

设：有 $x = (x_1, \dots, x_n)$ 和 $y = (y_1, \dots, y_m) \in \mathbb{R}^c$, $x_i, y_j \geq 0$, 且 x 和 y 不正交也不共线，也就是 $0 < \langle x, y \rangle < \|x\| \|y\|$

且 x 在某维 m , y 在某维 n 取得最大值 x_m 和 y_n ,

$$\text{即 } x_m = \max_{1 \leq i \leq c} (x_i), y_n = \max_{1 \leq j \leq c} (y_j)$$

求证：H₁: 当 x 和 y 取得最大值的维度相同，即 $m=n$ ，则 $\langle \phi_p(x), \phi_p(y) \rangle > \langle x, y \rangle$ ， $\langle \cdot, \cdot \rangle$ 为内积

H₂: 当 x 和 y 取得最大值的维度不同，即 $m \neq n$ ，则 $\langle \phi_p(x), \phi_p(y) \rangle < \langle x, y \rangle$

$$\text{证明: } \phi_p(x) = f_p(\text{relu}(Qx)) = f_p(x), \text{ 则 } \|f_p(x)\| = \frac{\|x\|}{\|x\|_{\text{sum}^2}} \|x^{\text{sum}^2}\| = \|x\|$$

$$\text{同理, } \phi_p(y) = f_p(y), \|f_p(y)\| = \|y\|$$

$$\text{H1: } \langle \phi_p(x), \phi_p(y) \rangle = \langle f_p(x), f_p(y) \rangle \text{ 证明了 } f_p(\cdot) \text{ 不改变向量模长}$$

$$= \|f_p(x)\| \|f_p(y)\| \cdot \langle \frac{f_p(x)}{\|f_p(x)\|}, \frac{f_p(y)}{\|f_p(y)\|} \rangle$$

本文 Motivation

$$= \|x\| \cdot \|y\| \cdot \left\langle \frac{f_p(x)}{\|f_p(x)\|}, \frac{f_p(y)}{\|f_p(y)\|} \right\rangle, \text{ 其中 } \frac{f_p(\cdot)}{\|f_p(\cdot)\|} \text{ 是单位向量}$$

$$\text{而} \quad \left\langle \frac{f_p(x)}{\|f_p(x)\|}, \frac{f_p(y)}{\|f_p(y)\|} \right\rangle = \frac{\sum_{i=1}^n x_i^p \cdot y_i^p}{\sqrt{\sum_{i=1}^n x_i^{2p}} \sqrt{\sum_{i=1}^n y_i^{2p}}} \quad \text{上下同时除为偏的的最大值加和} y_n.$$

$$= \frac{\sum_{i=1}^n x_i^p / x_m^p \cdot y_i^p / y_n^p}{\sqrt{\sum_{i=1}^n x_i^{2p} / x_m^2} \sqrt{\sum_{i=1}^n y_i^{2p} / y_n^2}} \quad \text{标准化 } x_i \text{ 和 } y_i. \text{ 设 } \frac{x_i}{x_m} = a_i, \frac{y_i}{y_n} = b_i, \text{ 则得为}$$

$$\text{式} = \frac{\sum_{i=1}^n a_i \cdot b_i}{\sqrt{\sum_{i=1}^n a_i^{2p}} \sqrt{\sum_{i=1}^n b_i^{2p}}} \quad , \quad i=1, 2, \dots, n. \quad \text{关于 P 的单调性易证.}$$

其中 $a_i - b_i \in [0, 1]$, $a_m = 1$, $b_n = 1$.

$$\text{因此, } \sum_{i=1}^n a_i^p = \begin{cases} 1, & i=m \\ 0, & i \neq m \end{cases}, \quad \sum_{i=1}^n b_i^p = \begin{cases} 1, & i=n \\ 0, & i \neq n \end{cases}$$

根据老 x 和 y 的相似性.

当 $m=n$ 时 (x 和 y 相似时),

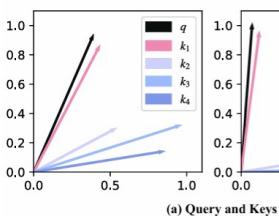
$$\begin{aligned} \text{有} \quad \left\langle \frac{f_p(x)}{\|f_p(x)\|}, \frac{f_p(y)}{\|f_p(y)\|} \right\rangle &= \frac{\sum_{i=1}^n x_i^p y_i^p}{\sqrt{\sum_{i=1}^n x_i^{2p}} \sqrt{\sum_{i=1}^n y_i^{2p}}} \\ &= \|x\| \cdot \|y\| \cdot \frac{1}{\sqrt{1} \sqrt{1}} = \|x\| \cdot \|y\|. \end{aligned}$$

由假设得: $\|x\| \cdot \|y\| > \langle x, y \rangle$. H1 得证

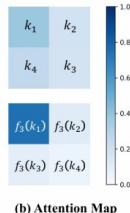
当 $m \neq n$ 时 (x 和 y 不相似时)

$$\text{有} \quad \left\langle \frac{f_p(x)}{\|f_p(x)\|}, \frac{f_p(y)}{\|f_p(y)\|} \right\rangle = \|x\| \cdot \|y\| \cdot \frac{1+0+0x}{\sqrt{1} \sqrt{1}} = 0 < \langle x, y \rangle \quad \text{H2 得证.}$$

因此, 该方法进一步放大了相似向量的相似性指标, 而缩小了不相似向量的相似性指标.



(a) Query and Keys



(b) Attention Map

- 不改变帧长
- 仅放大变化