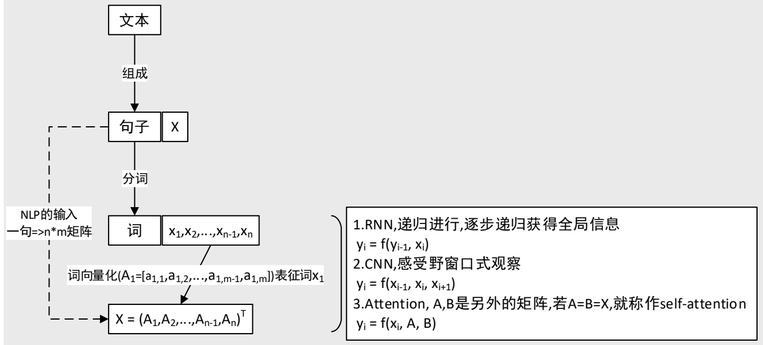
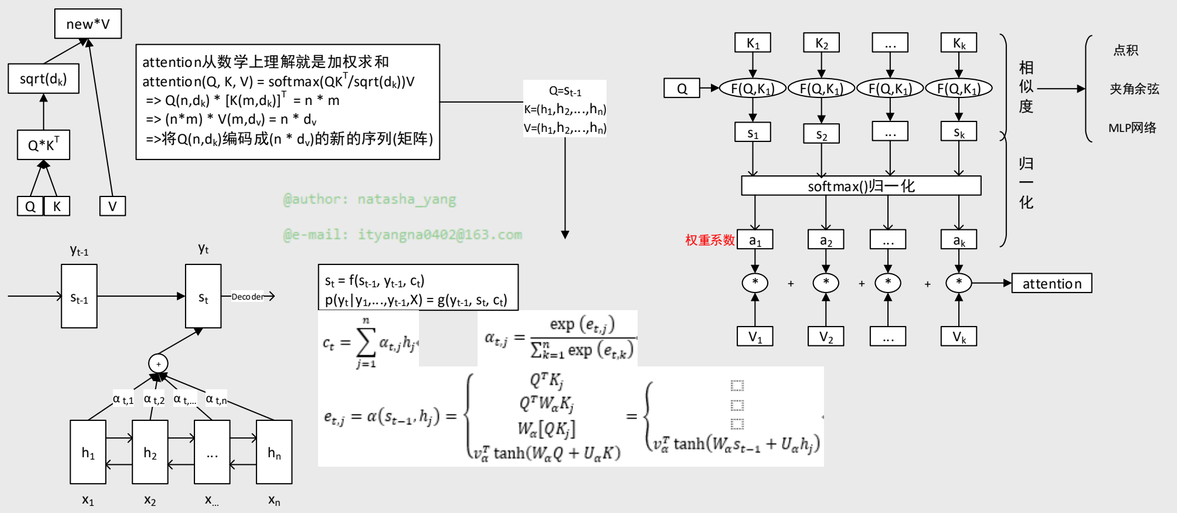
文本在自然语言中的处理

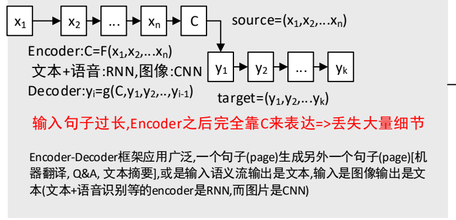


其实Attention就是一个Q和K比较(求相似度,比较的方法又很多),比较之后得到的相似度就是V的权重,然后归一化之后求加权和
其实主要就是找到Attention的Q,K,V分别是谁
在seq2seq中Q是decoder种当前t(实际用的t-1时刻)的HiddenState,K和V是encoder的所有HiddenState
含义就是当前的输出和所有的输入比较求和输入之间的相似度(比较的方式又很多)
transformer中encoders中的所有encoder中的Q,K,V分别是输入和Q,K,V各自的参数矩阵求得的
decoders中的所有decoder中的Q是来自于之前的输出,K,V是encoders的输出
那local方式的attention就是一个变种,不是对比所有而是对比一个窗口的中那些

details

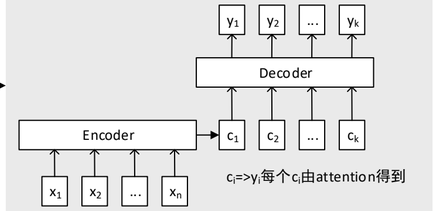


RNN(seq2seq)

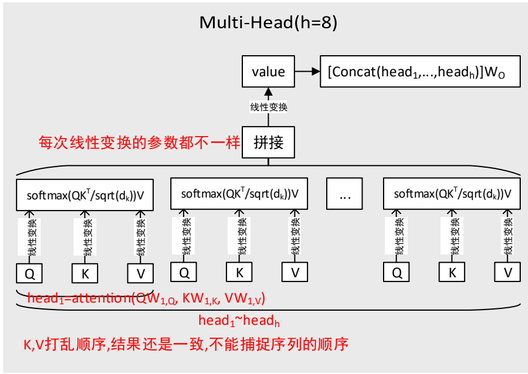


改进

attention加持



Multi-Head(h=8)



拼接(维度不同)
相加(维度相同)

位置向量

每个位置编号,每个编号对应一个向量

id为p的位置映射成一个d_{pos}维的位置向量

$$\begin{cases}PE_{2i}(p)=\sin\left(\frac{p}{10000^{\frac{2i}{d_{pos}}}}\right)\\PE_{2i+1}(p)=\cos\left(\frac{p}{10000^{\frac{2i}{d_{pos}}}}\right)\end{cases}$$

若第100个词p=100想生成D=8的向量
i=0 => 2i/D=0 => 10000^0=1 =>[sin(100),cos(100)]
i=1 => 2i/D=¼ => 10000^¼ => [sin(10000^¼),cos(10000^¼)]
i=2 => 2i/D=½ => 10000^½=100 => [sin(1),cos(1)]
[sin(100),cos(100),sin(10000^¼),cos(10000^¼),sin(1),cos(1),PE_6,PE_7]