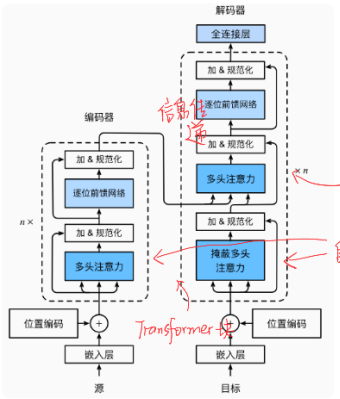
Transformer

- **Transformer架构**



基于编码器-解码器架构
处理序列对
完全基于注意力

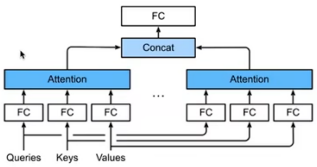
信息传递

编码器输出作 key-value

自注意力

Transformer块

- 多头注意力(Multi-head attention)



类似CNN中多个卷积核

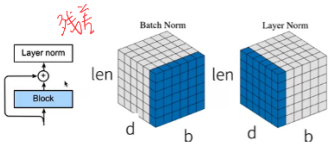
$q \in R^{d_q}$, $k \in R^{d_k}$, $v \in R^{d_v}$

头i的学习参数: $W_i^{(q)} \in R^{p_q \times d_q}$, $W_i^{(k)}$, $W_i^{(v)}$

头i的输出 $h_i = f(W_i^{(q)}q, W_i^{(k)}k, W_i^{(v)}v)$

输出的学习参数 $W_o \in R^{p_o \times hp_v}$

  - 对同一个key, value, query，希望抽取不同信息
    - 例如短距离和长距离的关系
  - 多头注意力使用h个独立的注意力池化
  - 合并各个头的输出得到最后输出: $W_o \begin{bmatrix} h_1 \\ \vdots \\ h_k \end{bmatrix} \in R^{p_o}$
- 带掩码(Masked)的多头注意力
  - 解码器对序列中一个元素输出时，不应该考虑该元素之后的元素
  - 可以通过掩码来实现，即计算xi输出时假设当前序列长为l (valid_lens)
- 基于位置的前馈网络（FFN）
  - 将输出形状由(b,n,d)变为(bn,d)：batch，n是序列长度，d是qkv长度
  - 作用两个全连接层
  - 输出形状变回(b,n,d)
  - 等价于核窗口为1的一维卷积层，用于非线性变换
- Add和层归一化（LN）



  - Add就是加了一个残差块X，目的是为了防止在深度神经网络训练中发生退化问题（即通过增加网络的层数，Loss逐渐减小，然后趋于稳定达到饱和，然后再继续增加网络层数，Loss反而增大）
  - 批量归一化(BN)对每个特征/通道元素进行归一化，不适合序列长度变化的NLP应用
  - 层归一化对于每个样本里所有元素进行归一化
- 信息传递
  - 编码器输出y1…yn
  - 将其作为解码器中第i个Transformer块中多头注意力的key和value，query来自目标序列，类似seq2seq
  - 意味着编码器和解码器中块的个数和输出纬度是一样的

- **预测**
  - 预测t+1个输出时，解码器中输入已知的前t个预测值，作为key-value，同时第t个预测值还作为query。是顺序的