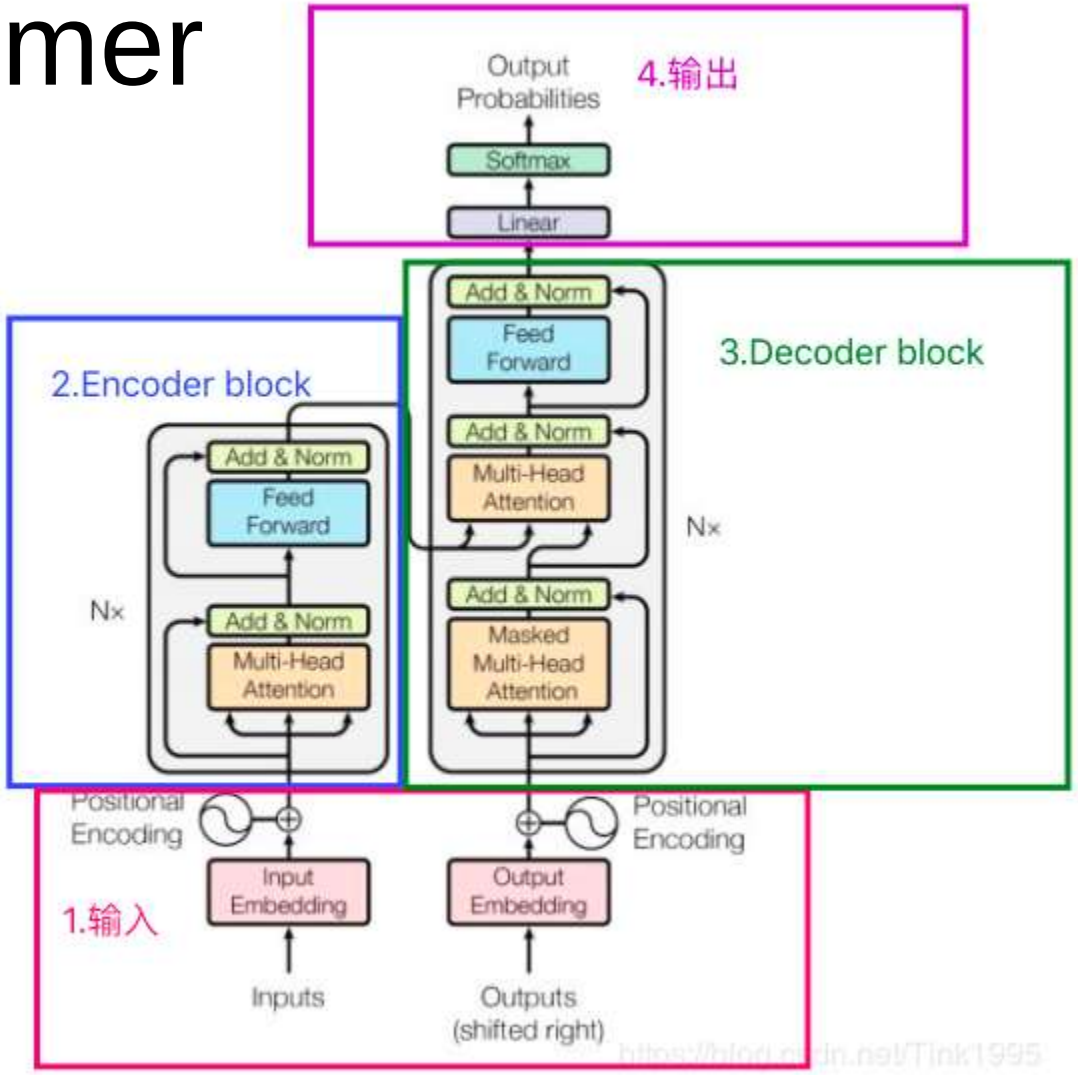
Transformer

Yu Jiening
Yu Jiening@umac.mo

um 澳大

Architecture of transformer

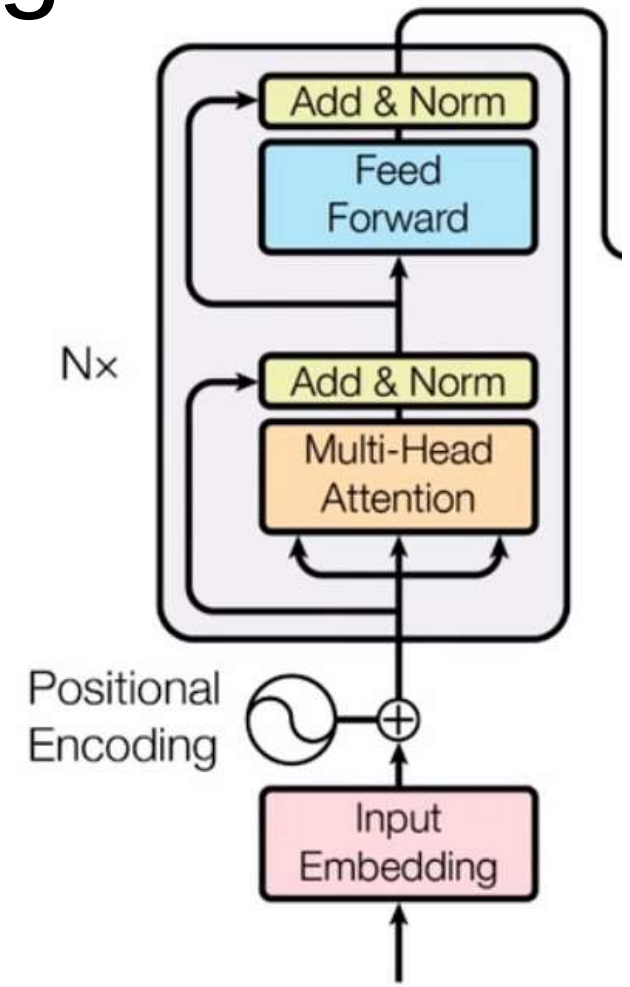
- Consists of four parts: input, output, encoder, decoder.



Positional Encoding

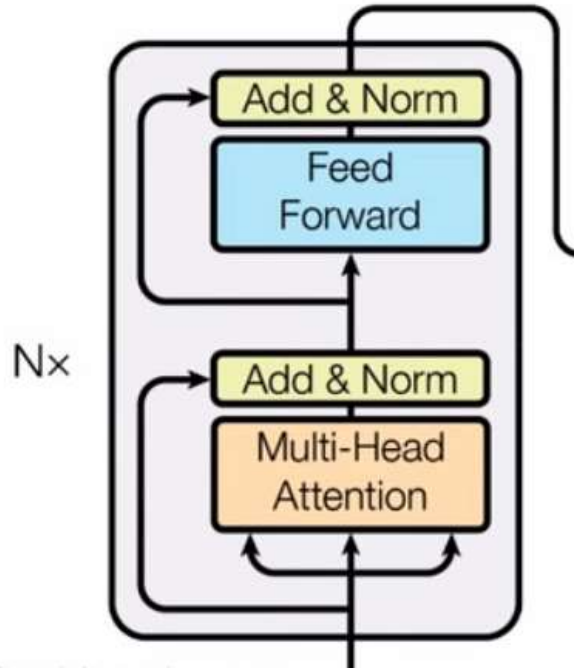
$$PE(pos, 2i) = sin(\frac{pos}{10000^{\frac{2i}{d_{model}}}})$$

$$PE(pos, 2i+1) = cos(\frac{pos}{10000^{\frac{2i}{d_{model}}}})$$



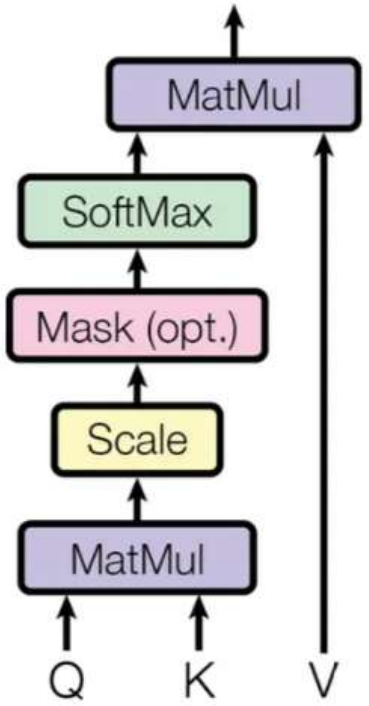
Encoder

- Multi-Head Attention
- Add & Norm
- Feed Forward

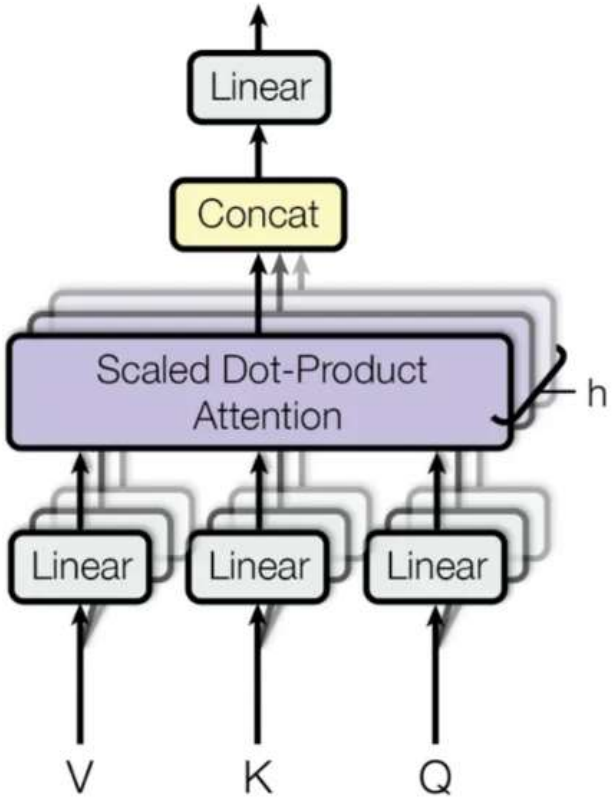


Multi-Head Attention

Scaled Dot-Product Attention



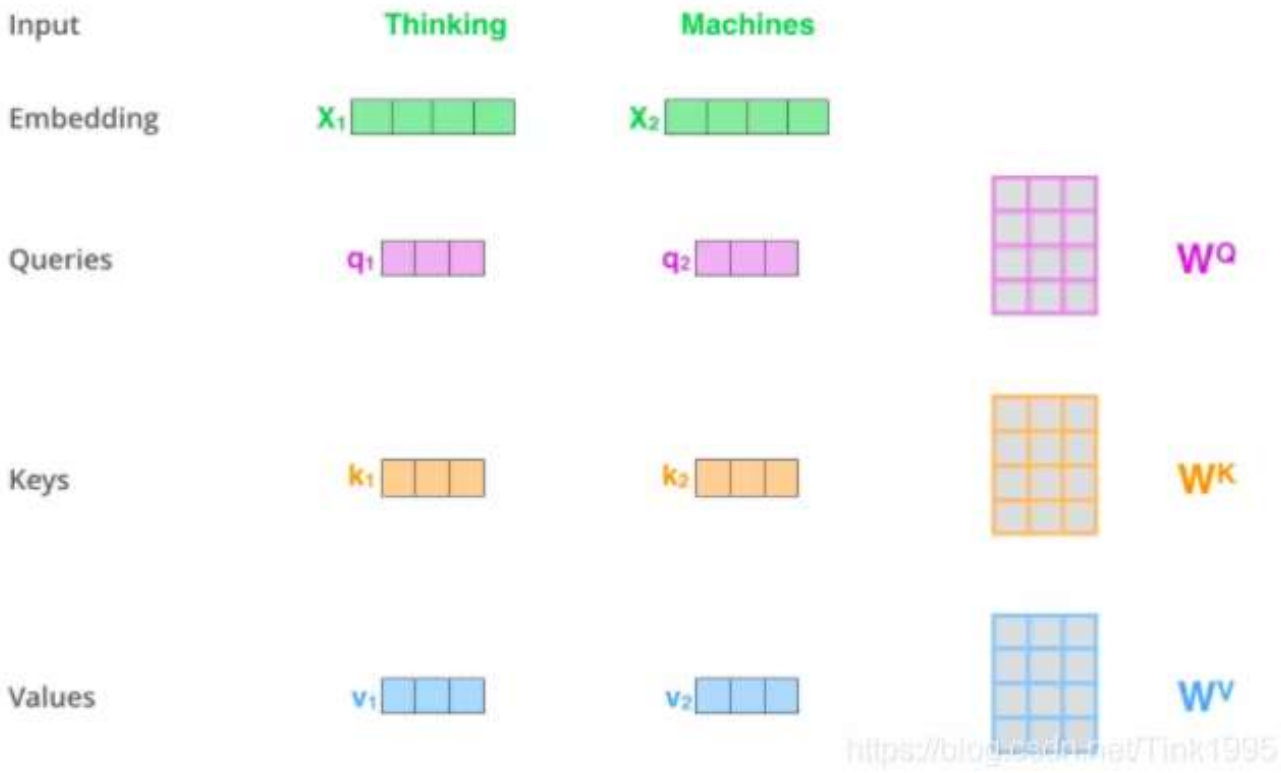
Multi-Head Attention

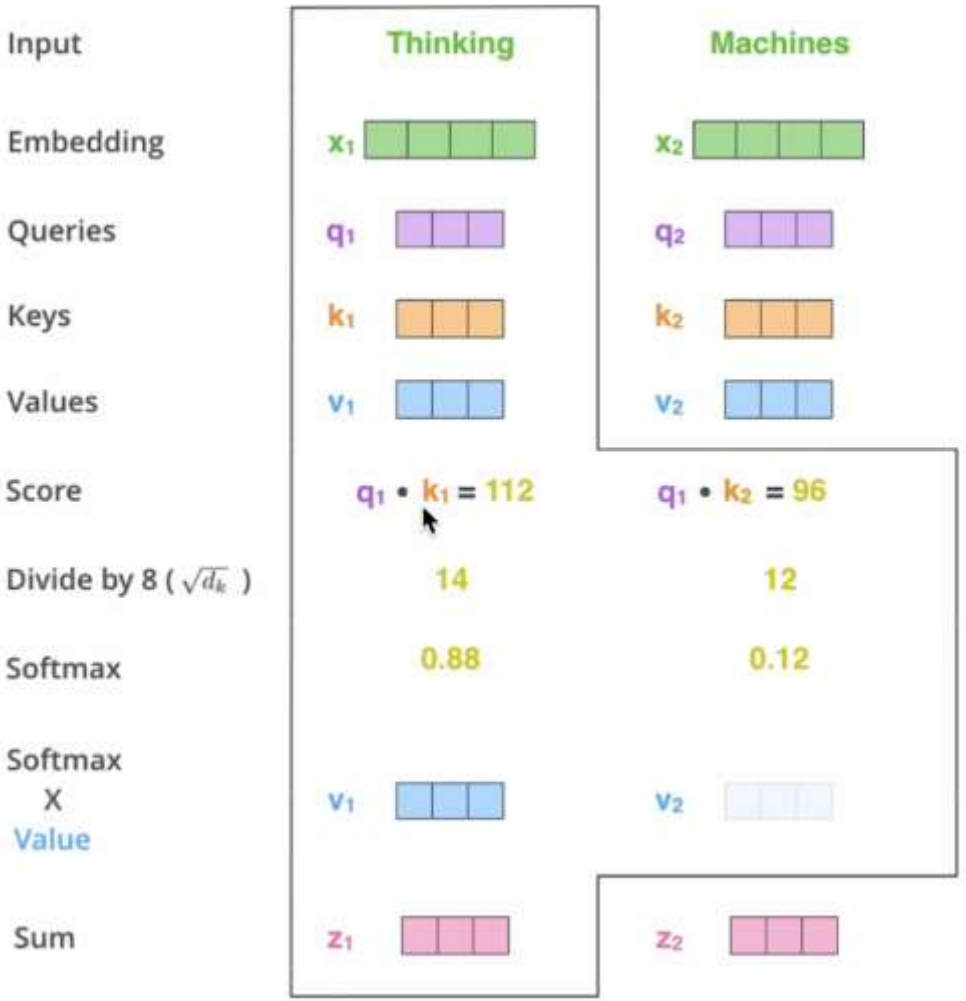


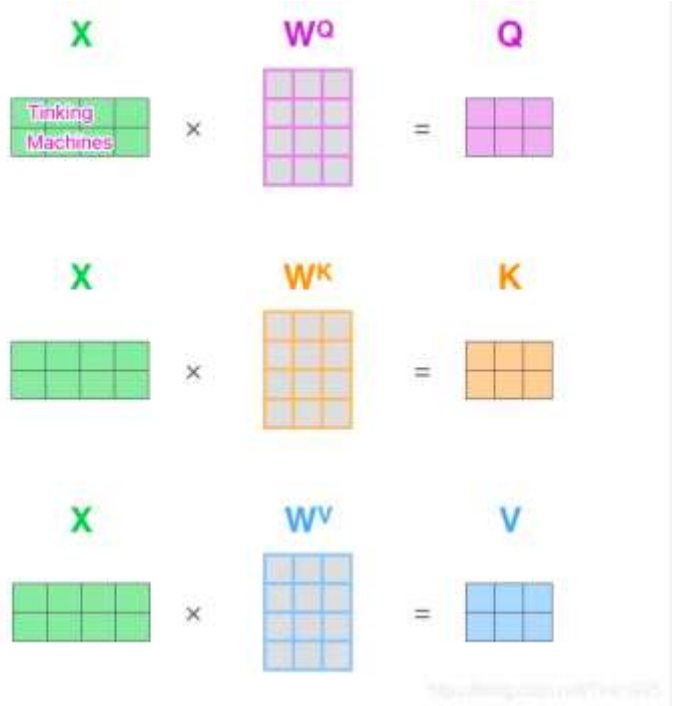
$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

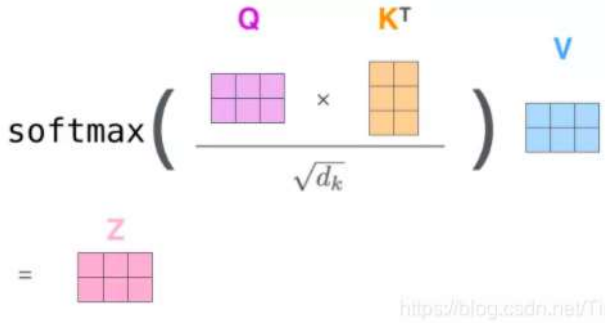
$$MultiHead(Q,K,V) = Concat(head1,\dots headh)W_0$$

where$headi = Attention(Q_i,K_i,V_i)$



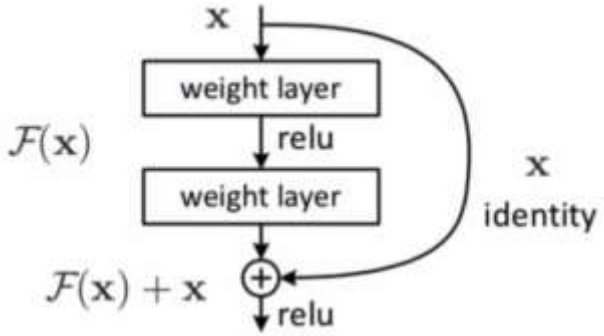




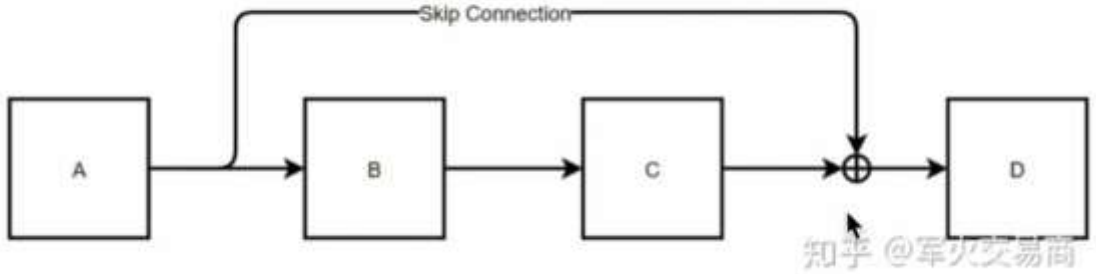


$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

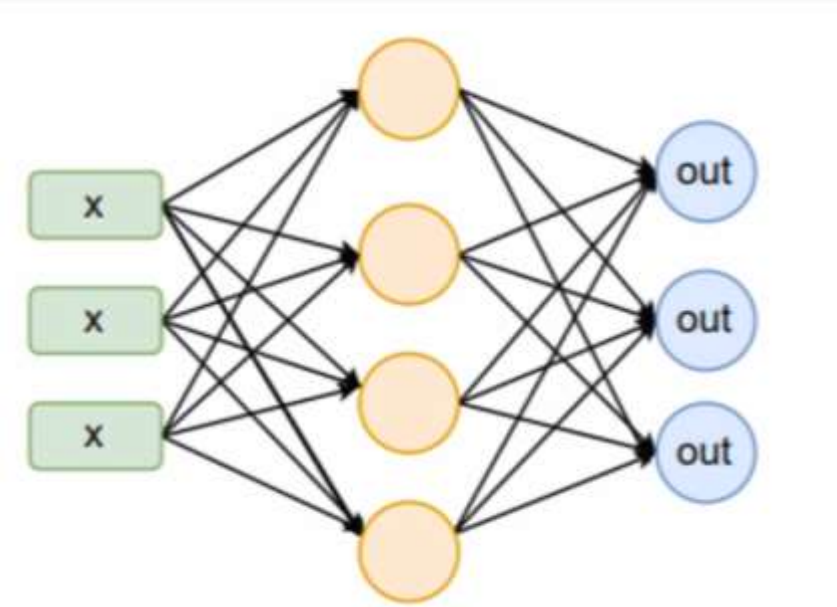
Add & Norm



$$D_{in} = A_{out} + C(B(A_{out}))$$



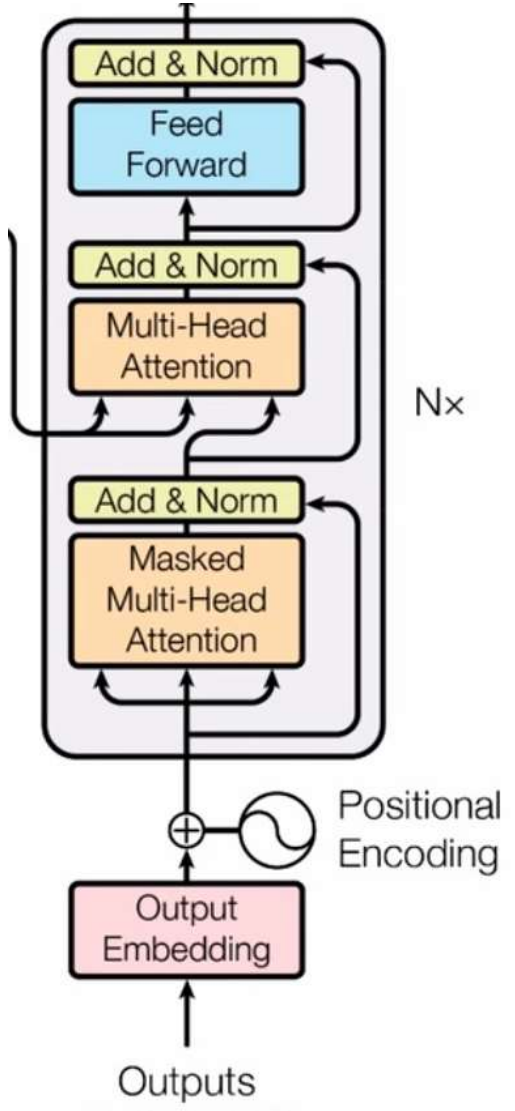
Feed Forward



MLP structure

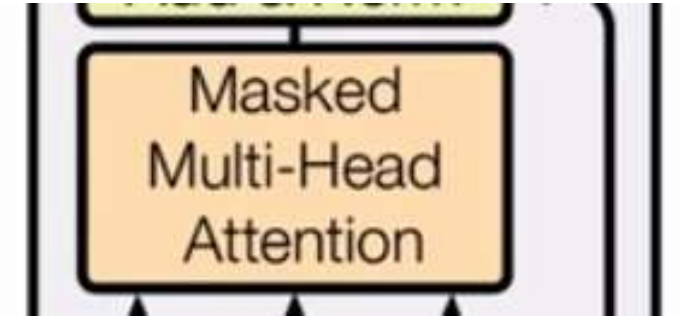
Decoder

- Masked Multi-Head Attention
- Multi-Head Attention
- Add&Norm
- Feed Forward



Masked Multi-Head Attention

- The point of masking is to mask out future information, making the trained model more accurate



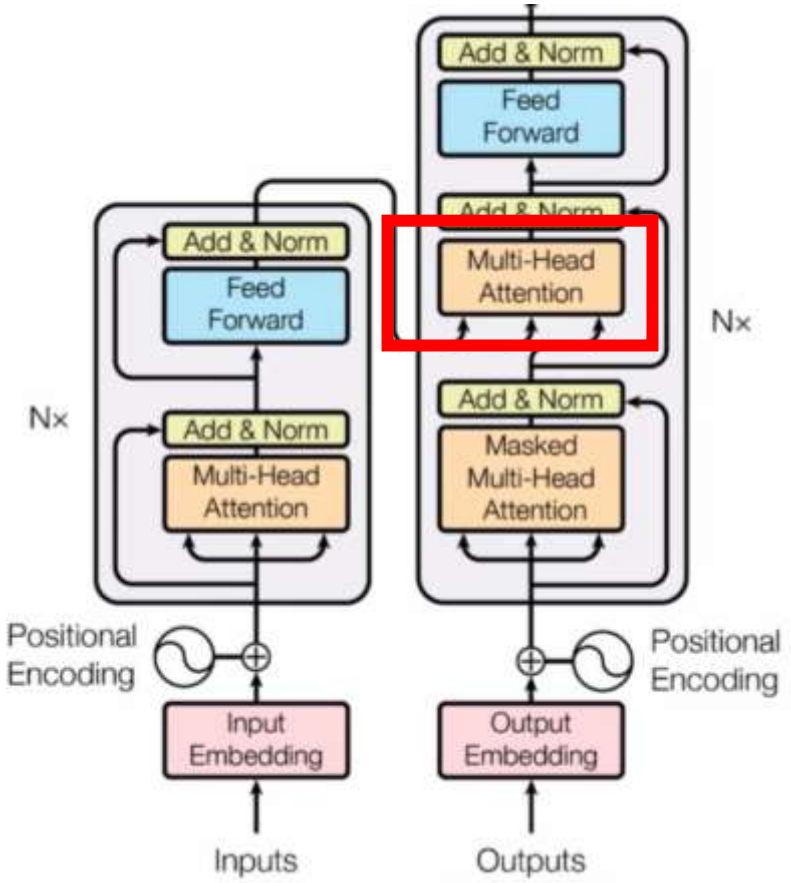
Interactive Attention

Calculation of Q, K, V for self-attention:

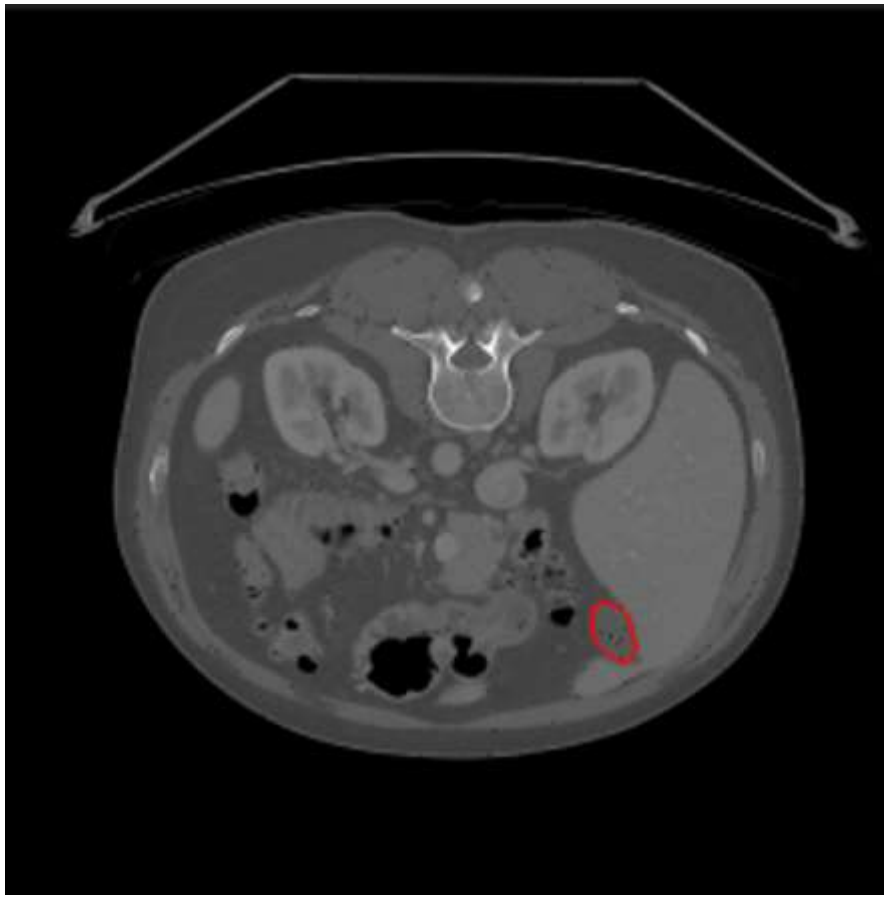
$$Q = W_q X + b_q$$
$$K = W_k X + b_k$$
$$V = W_v X + b_v$$

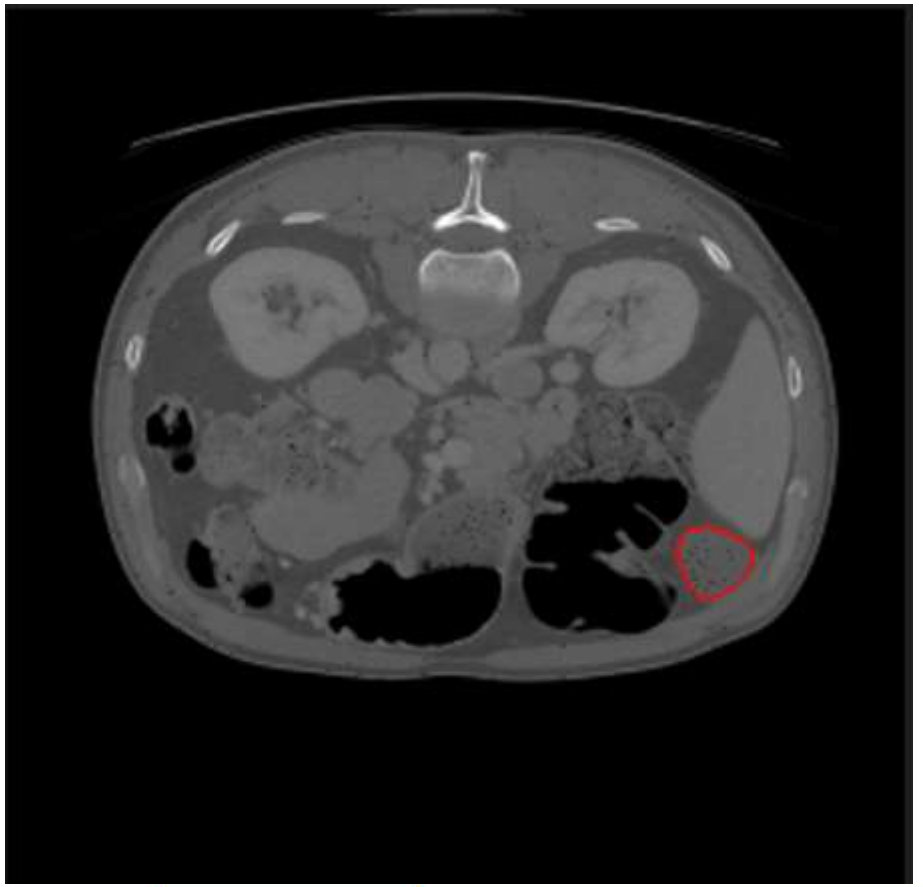
Calculation of Qfor interactive attention

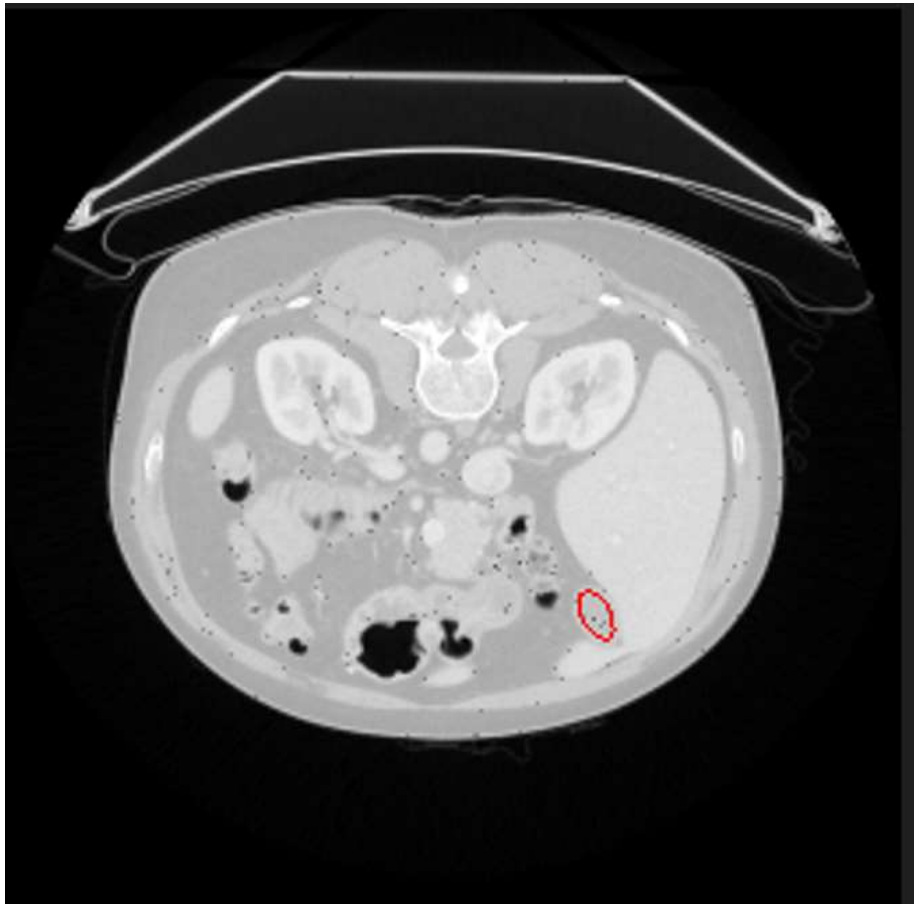
$$Q = W_q Out_{encoder} + b_q$$

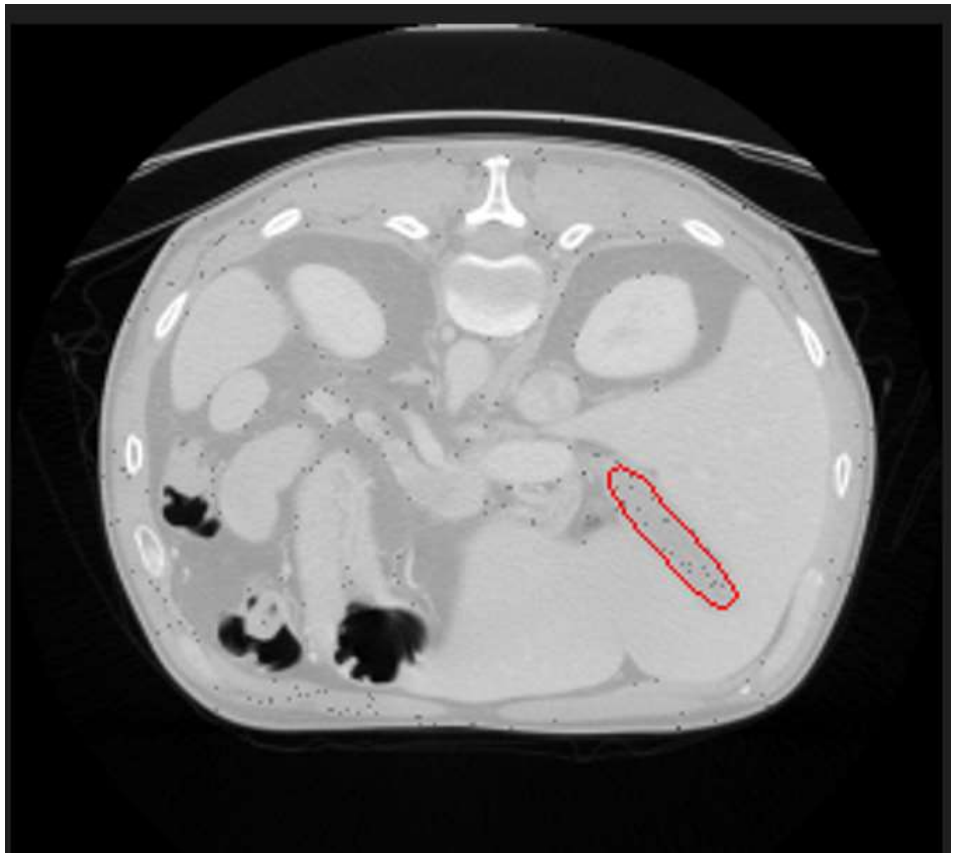


CT:









Thank You!