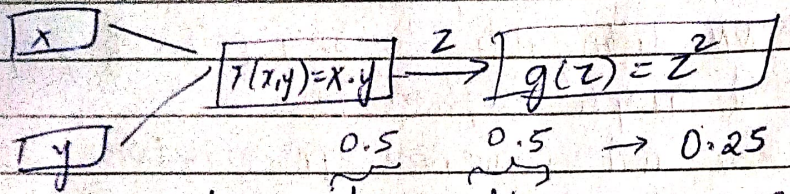
10/8/25.

Attention is all you need.

Problems with RNN

1) Slow computation
2) Vanishing or exploding gradients



$$\frac{dg}{dx} = \frac{dg}{df} \times \frac{df}{dx}$$ (Chain rule)

Computation chain is long, too much.

Either very big no. or very small no.

>1 multiplied <1 multiplied

3) Difficulty in accessing info from long time ago.

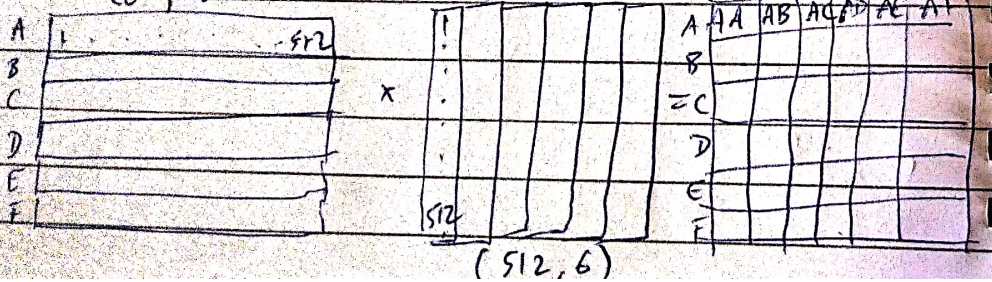
Introducing the Transformer.

Encoder Decoder.

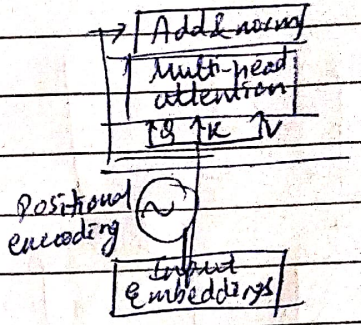
Notations

Input matrix sequence (d model)

(6, 512)



Encoder



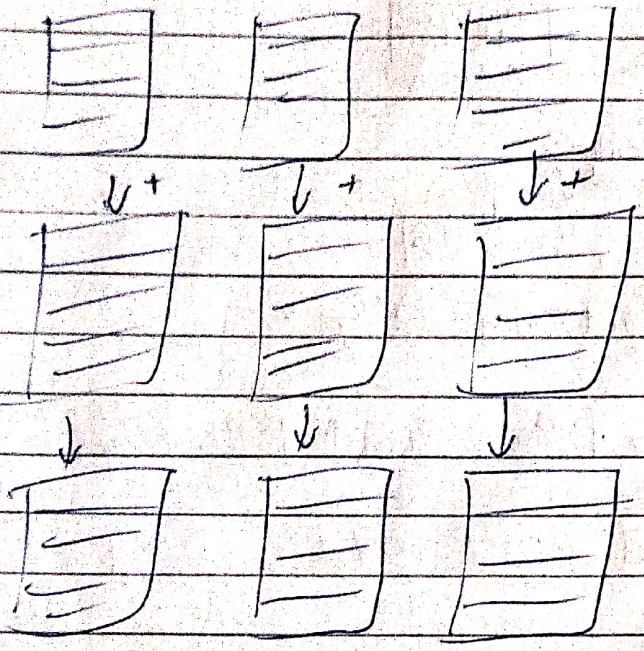
What is an input embeddings?

dmodel = 512.

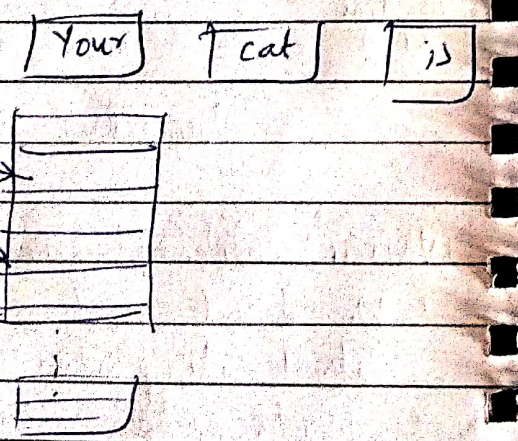
What is positional encoding?

- We want each word to carry some information about its position in the sentence.
- Words appear close → close } to model
  distant → distant.

- Positional encoding represent patterns for model to learn.



How is it calculated, positional embedding?



Why trig func? Why cos & sin?

What is self-attention?

→ Self attention allows the model to relate words to each other

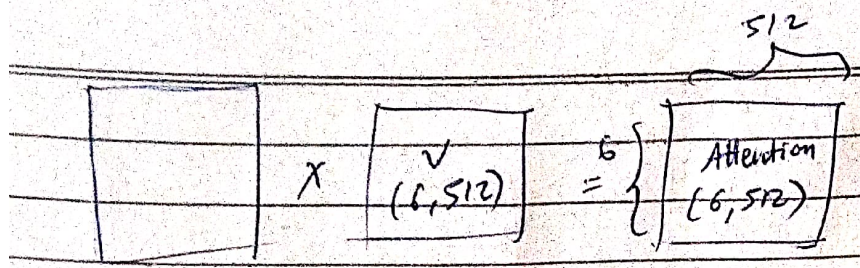
$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_K}}\right)V$$

→ In this simple case, we consider the sequence length seq = 6 and $d_{model} \div d_K = 512$.

→ The matrices Q, K, V are just the input sentence.

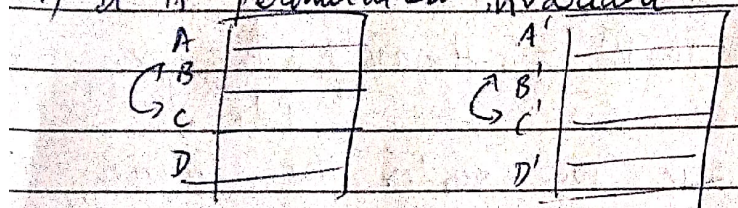
$$\left(\frac{\begin{bmatrix} Q \\ (6,512) \end{bmatrix} \times \begin{bmatrix} K^T \\ (512,6) \end{bmatrix}}{\sqrt{512}}\right) \text{softmax} = \begin{bmatrix} \quad \\ \quad \end{bmatrix}_{(6,6)}$$

sums up every value to 1 (scale it down)



Self - Attention in detail

1) It is permutation invariant.



2) Doesn't require any parameter.

3) We expect value along the diagonal to be maximum. (dot prod of each value with itself)

4) If we don't want two words to interact, then substitute $-\infty$ in their value. Softmax replaces $-\infty$ to zero (0).

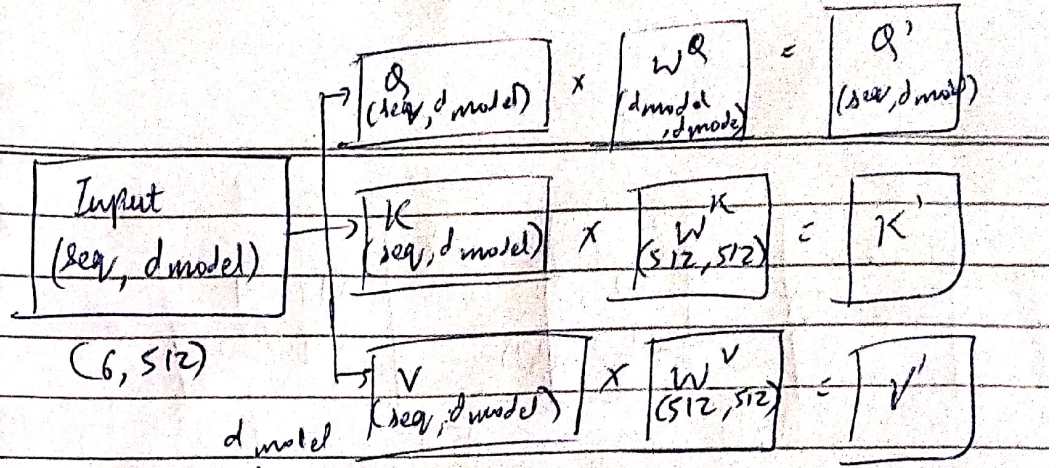
What is Multi-Head Attention ?

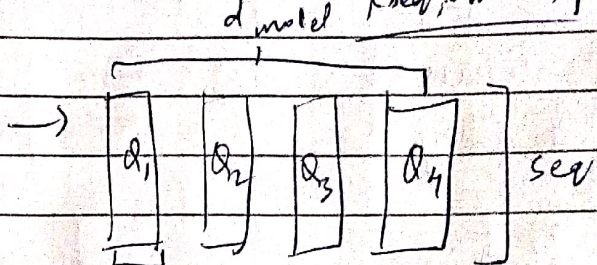
seq = sequence length = 6

dmodel = size of embedding vector. = 512

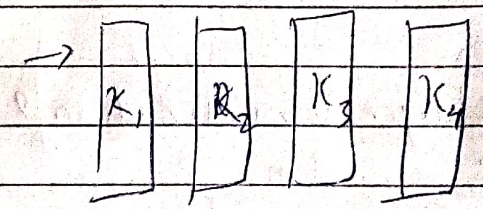
h = no. of heads

$d_k = d_v = d_{model}/h$

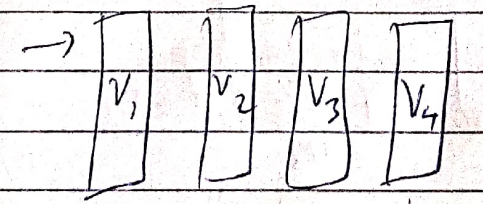




$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

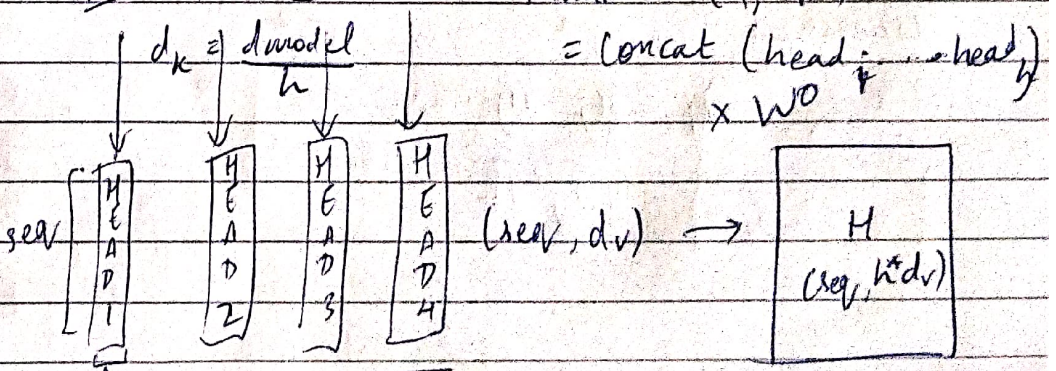


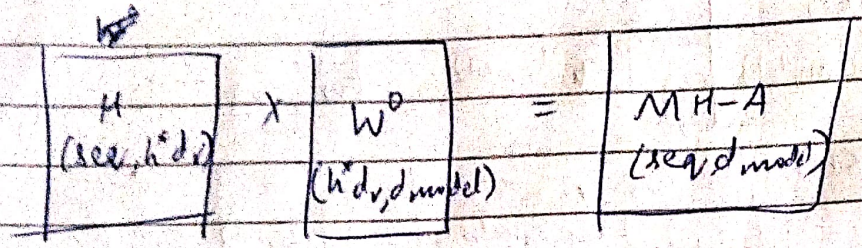
$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$



$d_v = d_k = \frac{d_{model}}{h}$

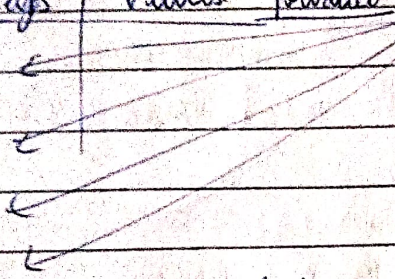
$\text{Multihead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots \text{head}_h) \times W^O$



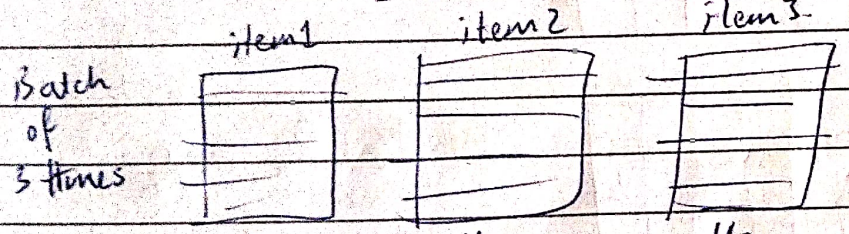


→ Why Query, Keys and Values?

Keys < Values dot product Query

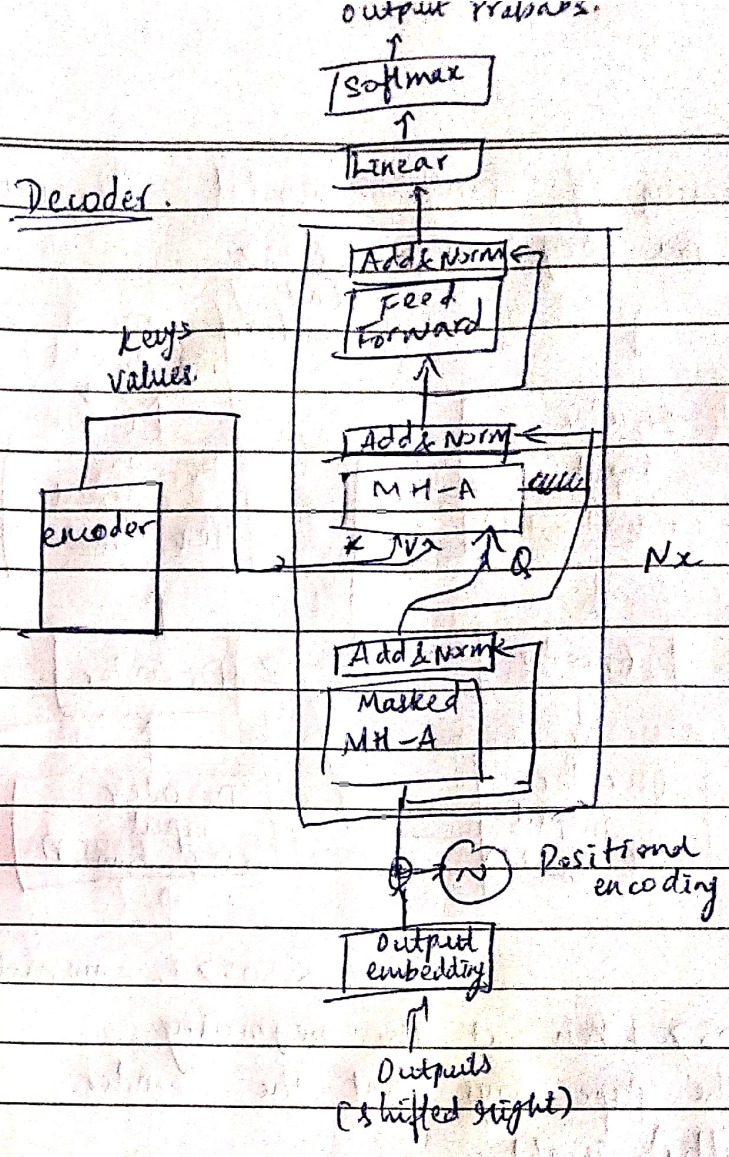


→ Add & norm → What is layer Normalization?



$$\hat{x}_j = \frac{x_j - \mu_j}{\sqrt{\sigma_j^2 + \varepsilon}}$$

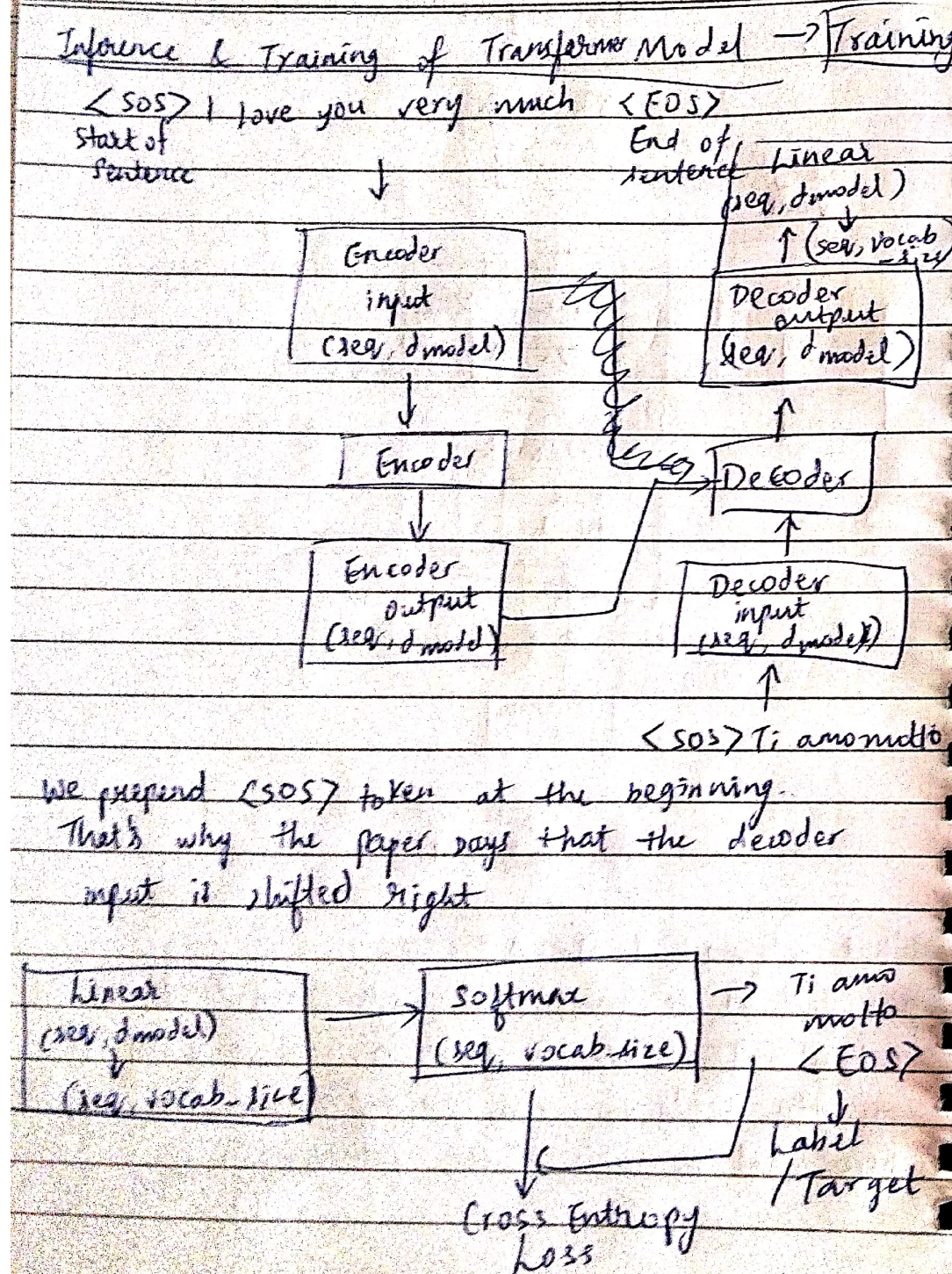
Decoder.



What is masked MH-A?

Our goal is to make the mode causal: it means output at a certain position can only depend on the words on the prev positions. The mode must not be able to see future words.

Inference & Training of Transformer Model → Training



Time step = 1

It all happens in 1 time step !

INFERENCE

I love you very much → Ti amo molto.

Time step = 3 mul-to.

Time step = 4 <EOS>.

Inference Strategy

→ We used greedy, at every step, the word with maximum softmax value. (does not perform very well)

→ Beam search is better.

↳ top B values taken → each of the top B values, next ~~best~~ words are predicted.