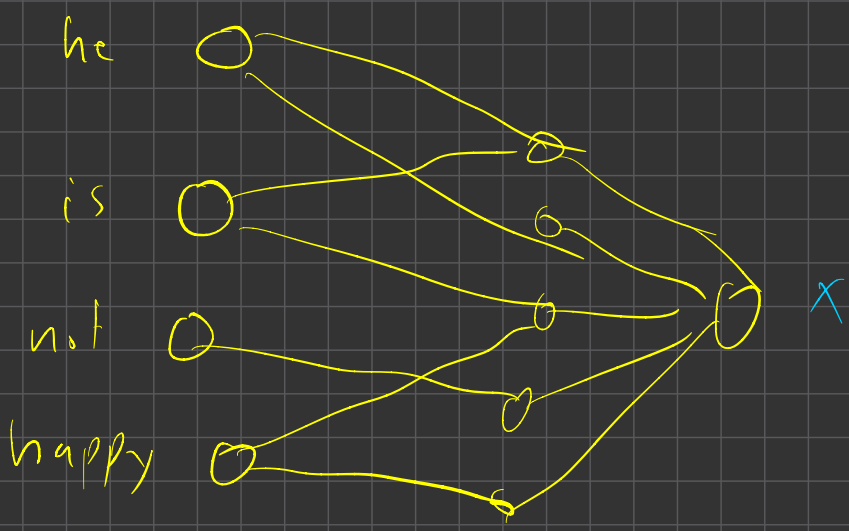
• RNN : example : We can do this method (neural network)



- First we need to do for each word (One hot encoding) not good
if we do that we have to much vectors.

example :-

he	is	not	happ	maybe more words
1	0	0	0	
0	1	0	0	
0	0	1	0	
0	0	0	1	

⟹ Payable to high computational power.

• RNNs have short memory.

- remember just current things and forget previous.

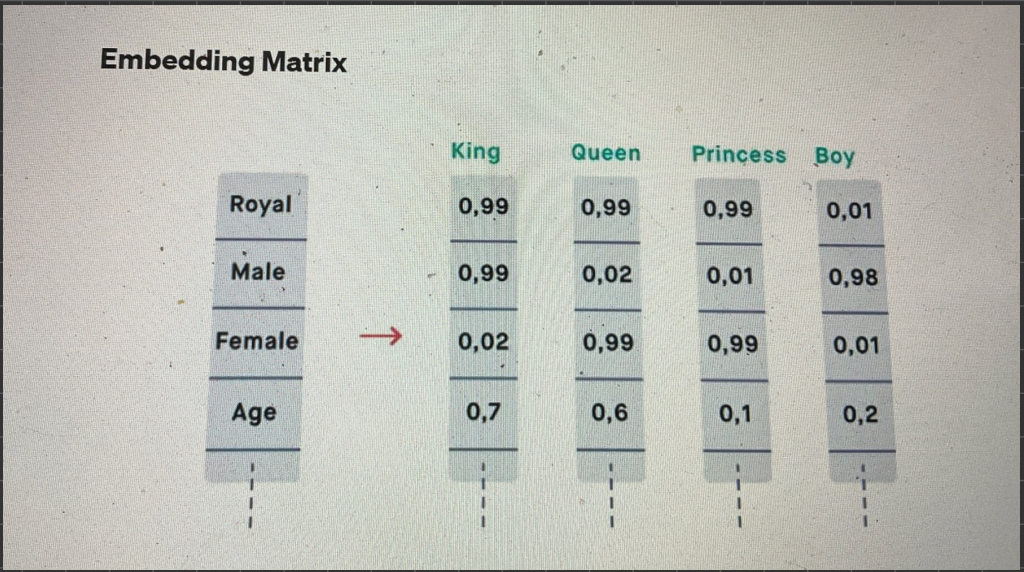
exp:- there is 30 words. Only the last 3 word are remembeird.

Soul: LSTM

*Word Embeding:-

The main advantage of using word Embedding is that it allows words of similar context to be grouped together and dissimilar words are positioned far away from each other. With help of an Embedding matrix.

- The similarity of 2 words can be found ⇒ (Cosine similarity).



* Pre-processing for Embedding matrix:-

There are a lot of algorithms for this:
1. One Hot Encoding
2. Term Frequency-Inverse Document Frequency
3. Tokenization (Text to Sequence)

- Tokenization:- Assigning a number for each number unique word.

example:-

["what is your name", "how are you", "where are you"]

After tokenize:-

what: 1, is: 2, your: 3, name: 4, how: 5, are: 6, you: 7, where: 8

- Tokenized of first sentence : [1,2,3,4]
- " " second " : [5,6,7]
- " " third " : [8,6,7]

Word2vec : is a neural network that creates Word Embeddings (a vector that represents a word in numeric form). To present all the words in a database of document. A word embedding will capture many different parts of word including its semantic, syntactical similarity and relation to other words.

- we have 50,000 words with using word2vec we can represent 30 word instead of 50,000 words.

* Cosine Similiray :

- $\cos(0)$ degree $\vec{=}1$ :: correlation is high.
- $\cos(90) \vec{=} 0$ :: there is no correlation.
- $\cos(180) \vec{=} -1$ :: correlation is opposite.

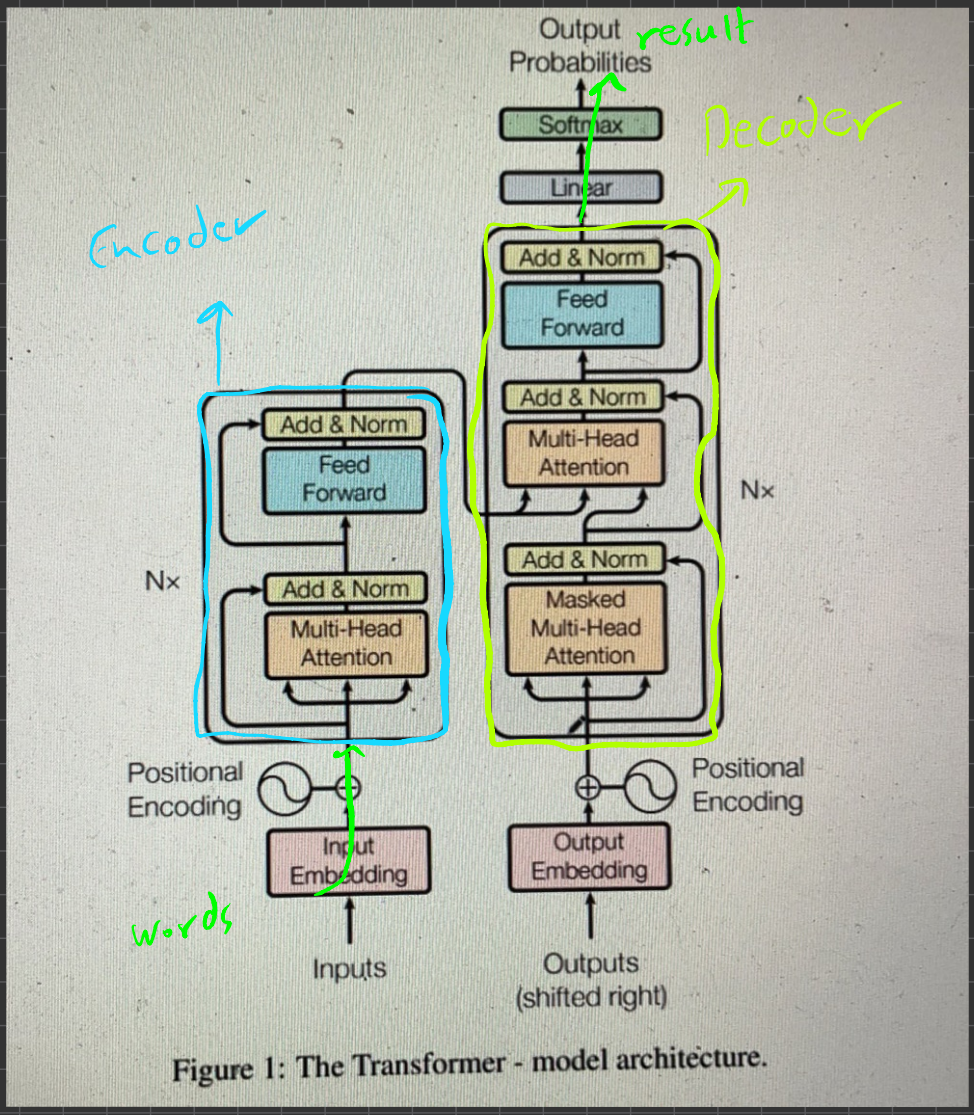


Figure 1: The Transformer - model architecture.

1. input Embedding:- (Embedding layers):.

returns sentences to words using tokenization.

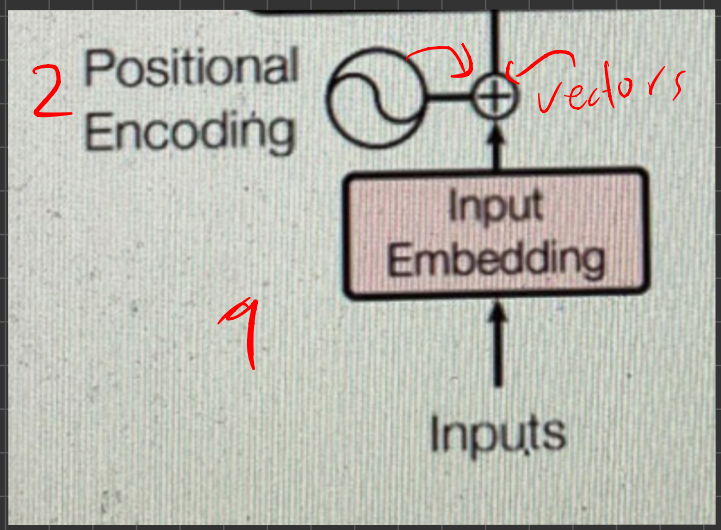
Then to vectors.

2. Positional Encoding. (index of vector)

$$PE_{2i} = \sin\left(\frac{Pos}{1000^{\frac{2i}{d}}}\right) \quad \text{even} \qquad i = 1, 2, 3$$

d: Embedding size

$$PE_{2i+1} = \cos\left(\frac{Pos}{1000^{\frac{2i}{d}}}\right) \quad \text{odd}$$



- Uniqueness for position vectors

1. Periodic function (sin, cos).
2. different frequencies (i). (shift for words)
3. Omega(10,000) لمساعدة على عدم حصول تكرار

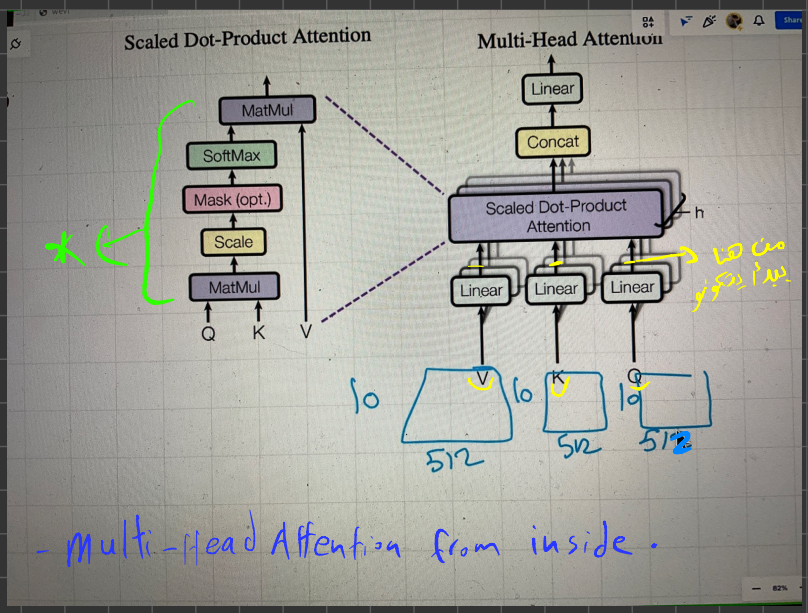
3. Self Attention: (Multi-Head Attention)

dict = {"a": 15, "b": 163}

Key Value

dict["a"] = 15

Query



- dot product: [ ]$_a$ [ ]$_b$

$[\ ]_{1\times 10} \cdot [\ ]_{10\times 1} = [\ ]_{1\times 1}$

$a \cdot b^T$

* cos similraty:

$$\cos(a,b) = \frac{a \cdot b}{\|a\| * \|b\|} \Rightarrow Sim(a,b)$$

example: $a \cdot b = 50$, $a \cdot c = 10$ → a closer to b because $a \cdot b = 50$ ↑

$$\Rightarrow \frac{Q_{10\times 512} \cdot K^T_{512\times 10}}{Scale} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ & & & \end{bmatrix}_{10\times 10}$$

مدى ترابط الكلمة مع بعضها

$$\Rightarrow \frac{Q \cdot K}{\sqrt{d_k}}$$

$(5,5,5)_3 = \sqrt{5^2+5^2+5^2} = \sqrt{3 * 5^2} = \sqrt{3} * 5$

d_k

10 × 10 → Attention filter

* $Attention(Q,K,V) = Softmax\left(\frac{Q \cdot K^T}{\sqrt{d_k}}\right) \cdot V$

* شرح الورقة

matrix 10 x 512 ج.ر معدل

* الأرقام موجودة بالورقة البحثية

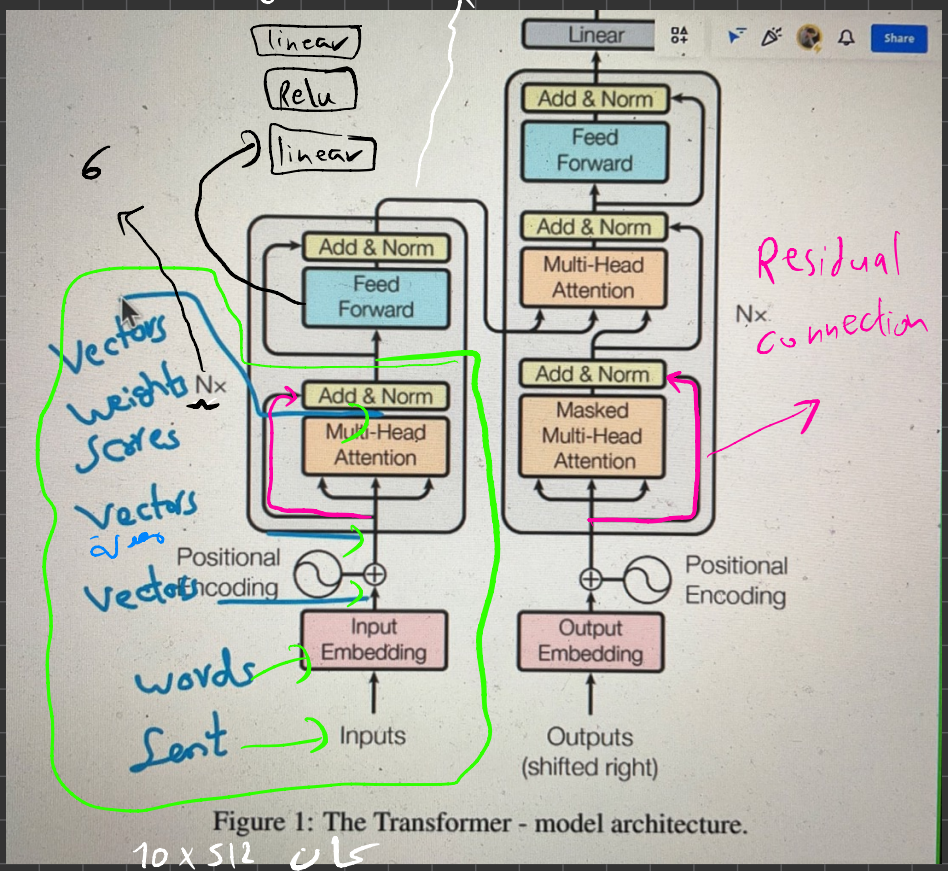


Figure 1: The Transformer - model architecture.

10 x 512 كان

عدل على Encoders shape

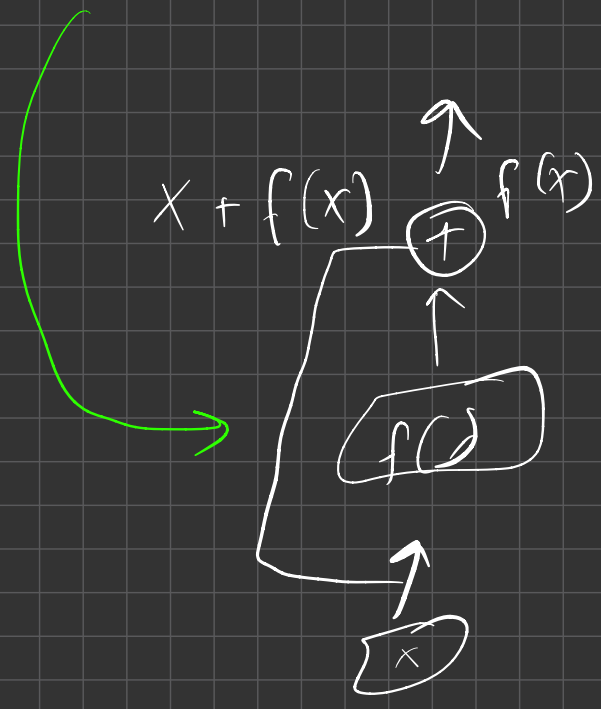
يعني كان 10 x 512 وطلع 10 x 512 ولكن معدلة

الـ Value

Seq-length

embed size

* Residual Connection:-

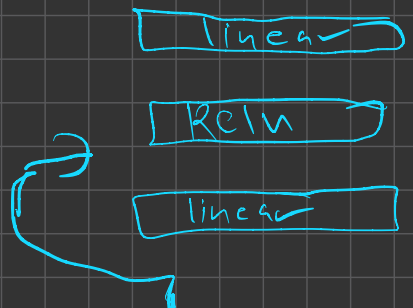


RNN يدخل كلمة كلمة

Tran. يدخل كل الكلمات مرة وحدة.

* Feed Forward :. Performed more encoding for values.

هي عبارة عن:-



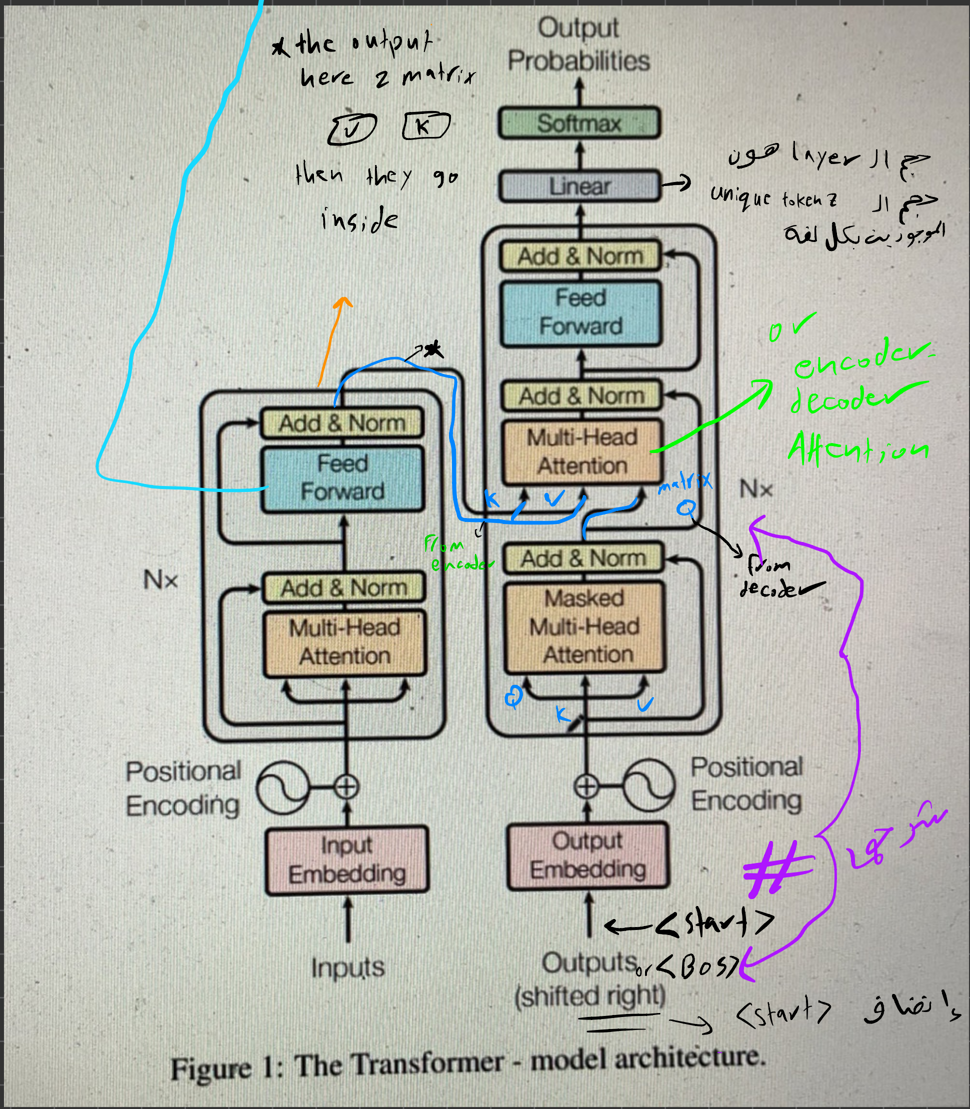


Figure 1: The Transformer - model architecture.

Decoder على :-

بيدخلها k,v من encoder

من الأسفل بنعطيه <start> token
<BOS>

بعدها بياخد ثلاث نسخ
(q,k,v) داخل
(multi-Head Attention)

وبعد عليه Add and Norm

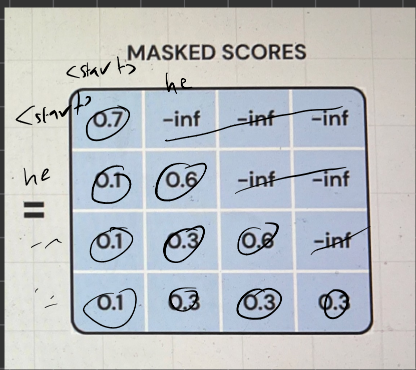
بيرجع matrix وهاي ال q

*Look ahead mask :-

	<start>	I	am	fine
<start>	0.7	0.1	0.1	0.1
I	0.1	0.6	0.2	0.1
am	0.1	0.3	0.6	0.1
fine	0.1	0.3	0.3	0.3

\+ mask (op)

0	$-\infty$	$-\infty$	$-\infty$
0	0	$-\infty$	$-\infty$
0	0	0	$-\infty$
0	0	0	0



* بيمنع Decoder من أنو يشوف Futur state .