Multi-Head Attention

$$X = \begin{bmatrix} [1 \dots 1024] \\ [1 \dots 1024] \\ [1 \dots 1024] \\ [1 \dots 1024] \end{bmatrix}$$

Sequence of 4 items where each item is represented as a vector with 1024 dimensions.
Suppose number of heads $h = 8$

Our goal with MHA is to transform the initial sequence of uncontextualized embeddings into a sequence of contextualized embeddings.

VISION TRANSFORMER

$$X = \begin{bmatrix} [1 \dots 1024] \\ [1 \dots 1024] \\ [1 \dots 1024] \\ [1 \dots 1024] \end{bmatrix} \begin{matrix} \text{PATCH 1} \\ \text{PATCH 2} \\ \text{PATCH 3} \\ \text{PATCH 4} \end{matrix} \Rightarrow X = \begin{bmatrix} [1 \dots 1024] \\ [1 \dots 1024] \\ [1 \dots 1024] \\ [1 \dots 1024] \end{bmatrix} \begin{matrix} \text{PATCH 1, 2, 3, 4} \\ \text{PATCH 1, 2, 3, 4} \\ \text{PATCH 1, 2, 3, 4} \\ \text{PATCH 1, 2, 3, 4} \end{matrix}$$

LANGUAGE MODEL

$$X = \begin{bmatrix} [1 \dots 1024] \\ [1 \dots 1024] \\ [1 \dots 1024] \\ [1 \dots 1024] \end{bmatrix} \begin{matrix} \text{I} \\ \text{LOVE} \\ \text{PEPPERONI} \\ \text{PIZZA} \end{matrix} \Rightarrow X = \begin{bmatrix} [1 \dots 1024] \\ [1 \dots 1024] \\ [1 \dots 1024] \\ [1 \dots 1024] \end{bmatrix} \begin{matrix} \text{I} \\ \text{I LOVE} \\ \text{I LOVE PEPPERONI} \\ \text{I LOVE PEPPERONI PIZZA} \end{matrix}$$

STEP 1: from X to Q, K, V

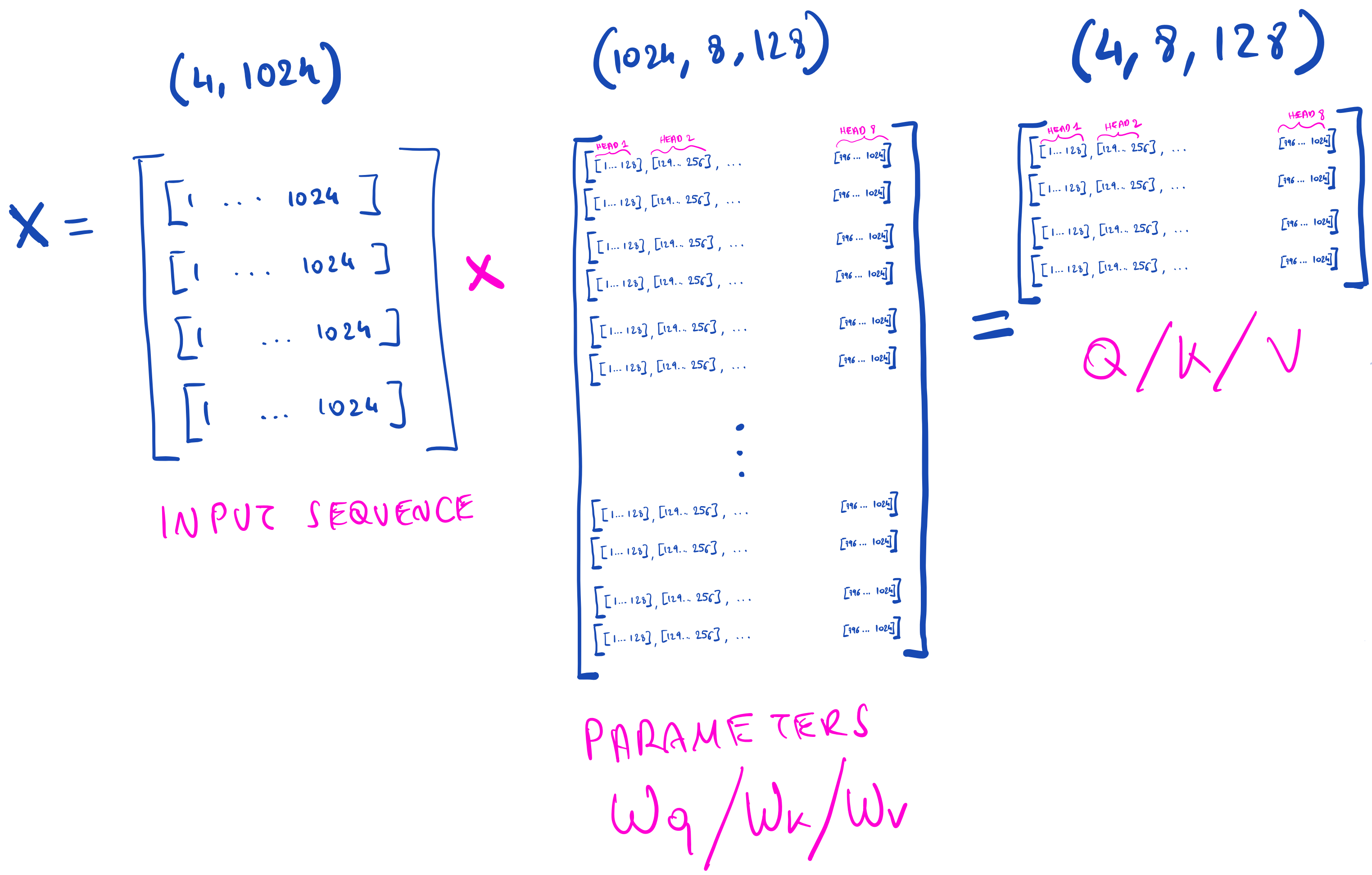
$$Q = X \times W_q = (4, 1024) \times (1024, \overbrace{8, 128}^{1024}) = (4, \overbrace{8, 128}^{1024})$$

$$K = X \times W_k = (4, 1024) \times (1024, 8, 128) = (4, 8, 128)$$

$$V = X \times W_v = (4, 1024) \times (1024, 8, 128) = (4, 8, 128)$$

In $(4, 1024)$: 4 = SEQUENCE, 1024 = HIDDEN_SIZE

In $(4, 8, 128)$: 4 = SEQUENCE, 8 = N_HEAD, 128 = HEAD_DIM = 1024/N_HEAD



STEP 2: TREAT EACH HEAD INDEPENDENTLY!

Q : (4, 8, 128)
K : (4, 8, 128)
V : (4, 8, 128)

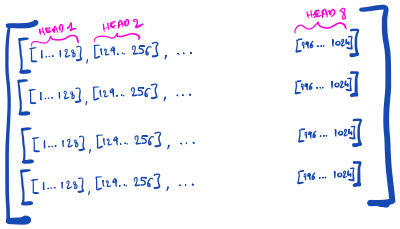
TRANSPOSE ⟹

(8, 4, 128)
(8, 4, 128)
(8, 4, 128)

8 → each head...

4, 128 → ... will compute the attention scores independently from other heads by using a part of the entire embedding.

(4, 8, 128)

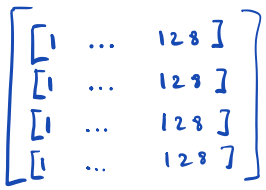


Q / K / V

⟹

(8, 4, 128)

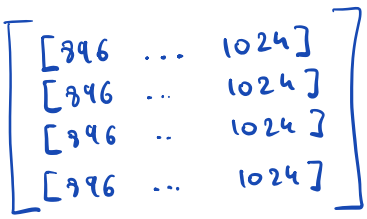
HEAD 1



HEAD 2

HEAD 7

HEAD 8



WHY?

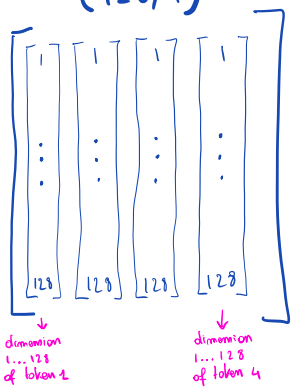
1) We want to parallelize the computation
2) Each head should learn to relate tokens (or patches) differently

STEP 3: CALCULATE THE ATTENTION FOR EACH HEAD IN PARALLEL

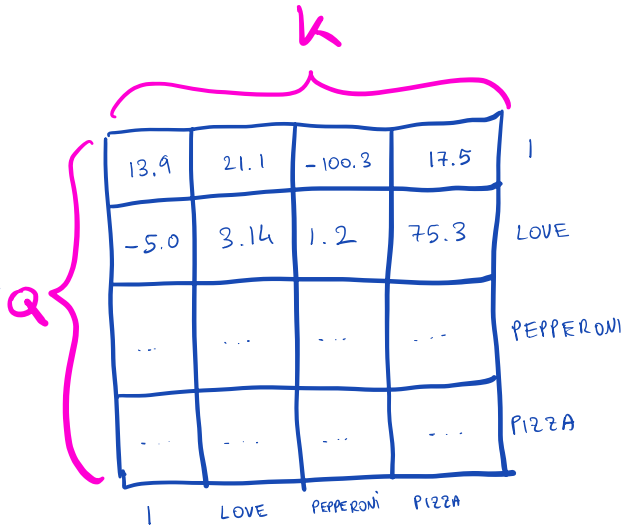
Q_{HEAD_1} = (4, 128)

[1 ... 128] -> dimension 1...128 of token 1
[1 ... 128] -> dimension 1...128 of token 2
[1 ... 128]
[1 ... 128] -> dimension 1...128 of token 4

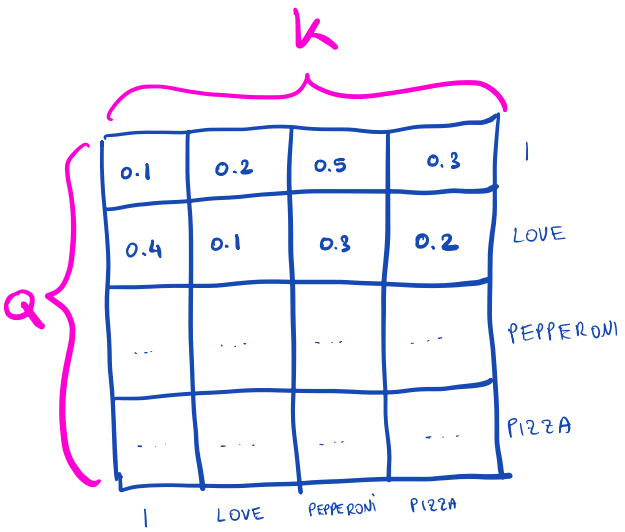
$K^T_{HEAD_1}$ = (128, 4)



$$\frac{Q * K^T}{\sqrt{d_{head}}}$$



$$\text{softmax}\left(\frac{Q * K^T}{\sqrt{d_{head}}}\right)$$



BRO, WHERE IS YOUR MASK?

$$\text{softmax}\left(\frac{Q \times K^T}{\sqrt{d_{head}}} + MASK\right) =$$

Q \ K	1	LOVE	PEPPERONI	PIZZA
1	1.0	0	0	0
LOVE	0.6	0.4	0	0
PEPPERONI	0.2	0.4	0.4	0
PIZZA	0.4	0.2	0.3	0.1

STEP 4: MULTIPLY BY THE V SEQUENCE

Q \ K	1	LOVE	PEPPERONI	PIZZA
1	1.0	0	0	0
LOVE	0.6	0.4	0	0
PEPPERONI	0.2	0.4	0.4	0
PIZZA	0.4	0.2	0.3	0.1

(4,4)

$\times$

$$\begin{bmatrix} [1 \ \dots \ 128] \\ [1 \ \dots \ 128] \\ [1 \ \dots \ 128] \\ [1 \ \dots \ 128] \end{bmatrix} \begin{matrix} 1 \\ \text{LOVE} \\ \text{PEPPERONI} \\ \text{PIZZA} \end{matrix}$$

(4, 128)

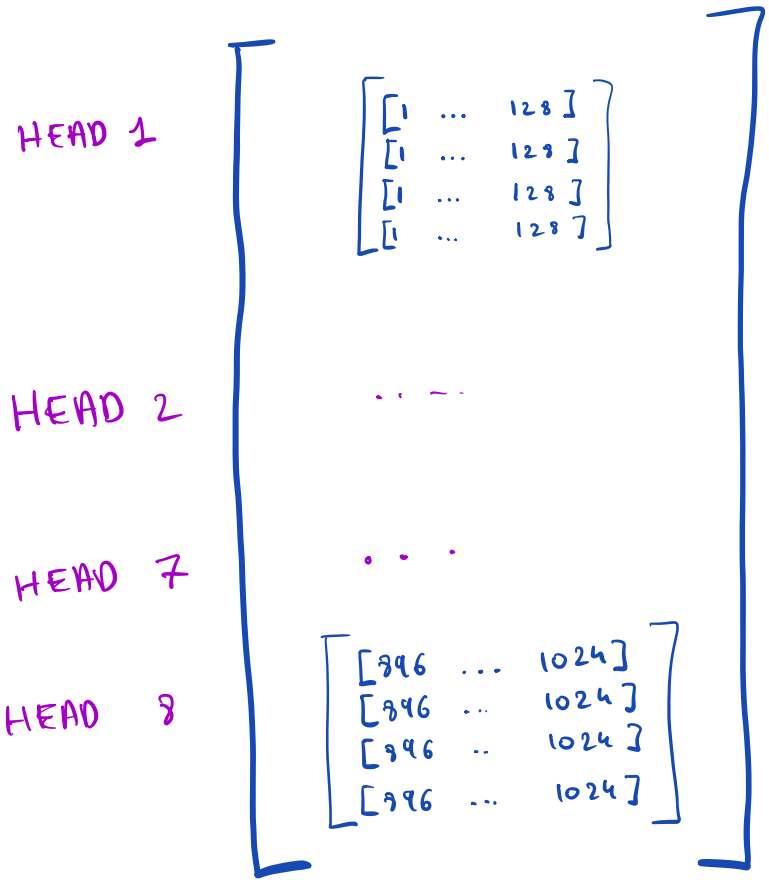
EACH ROW REPRESENTS A WEIGHTED SUM OF:

$$= \begin{bmatrix} [1 \ \dots \ 128] \\ [1 \ \dots \ 128] \\ [1 \ \dots \ 128] \\ [1 \ \dots \ 128] \end{bmatrix} \begin{matrix} \rightarrow 1 \\ \rightarrow 1 \ \text{LOVE} \\ \rightarrow 1 \ \text{LOVE PEPPERONI} \\ \rightarrow 1 \ \text{LOVE PEPPERONI PIZZA} \end{matrix}$$

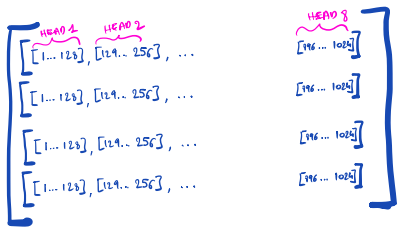
(4, 128)

STEP 5: TRANSPOSE BACK

(8, 4, 128)



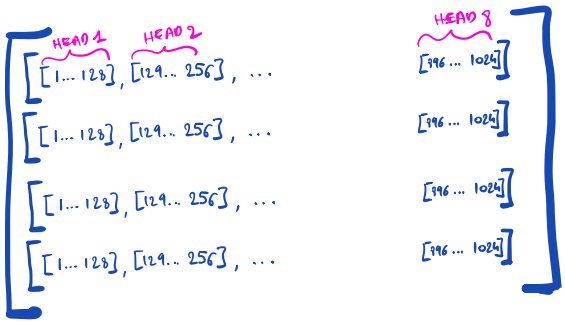
(4, 8, 128)



STEP 6: CONCATENATE ALL THE HEADS

Given that each head is computing the contextualized embeddings using a "part" of each token we can concatenate all the result of all the heads back together

(4, 8, 128)

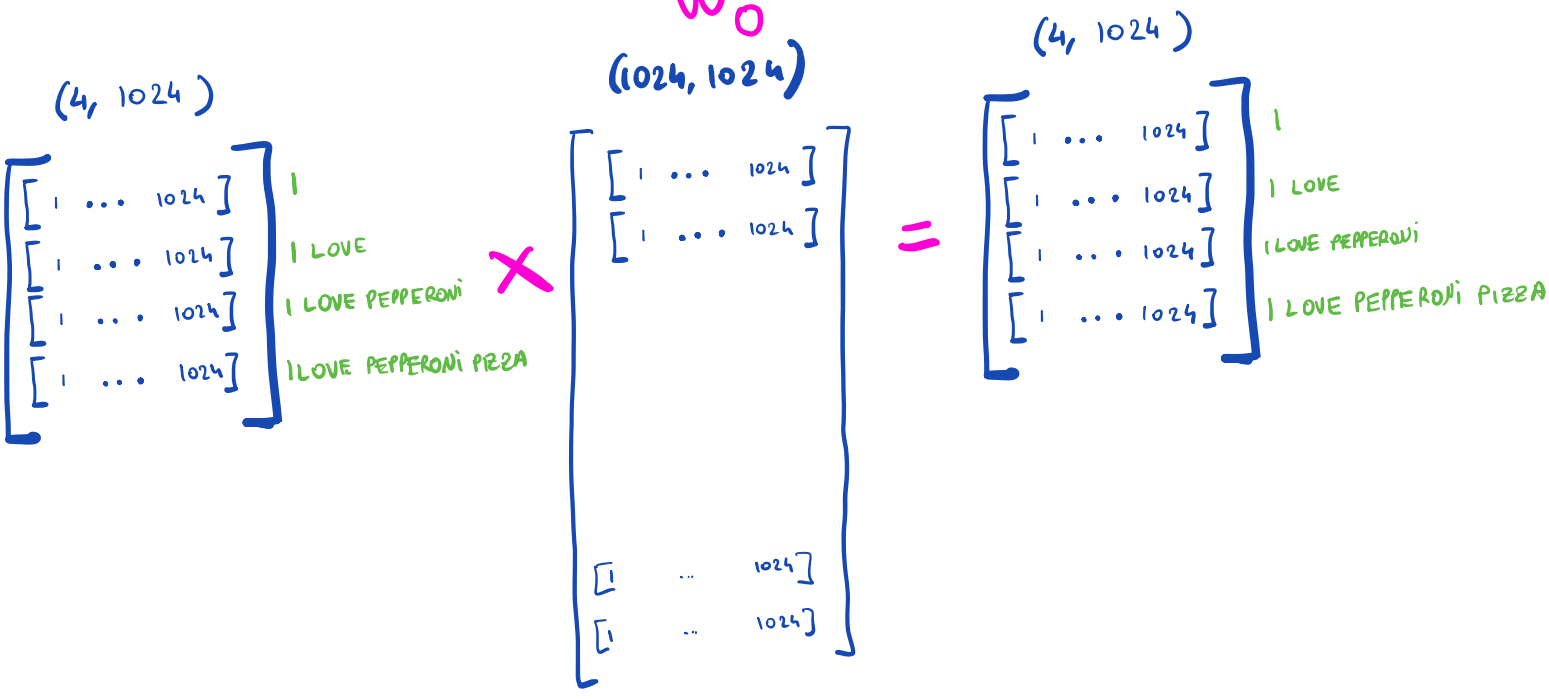


(4, 1024)

STEP 7: MULTIPLY BY W_o

PARAMETER

W_o



WHY MULTIPLY BY W_o ?

If we don't multiply by W_o, each group of 128 dimensions will be independent from each other, as they are the result of the concatenation of independent heads. Multiplying by W_o gives the chance to each head to "mix" with other heads.