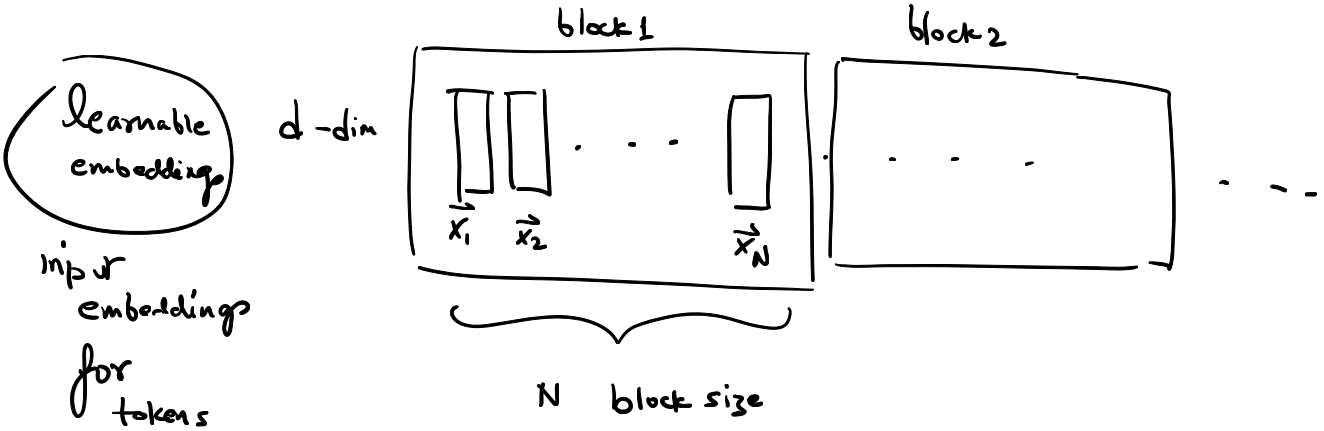
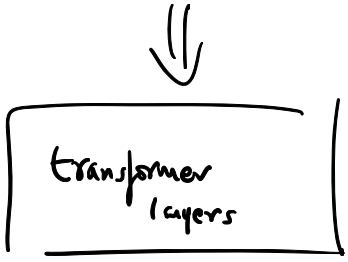
Transformers

dot-product attention

Input: block/"sequence"

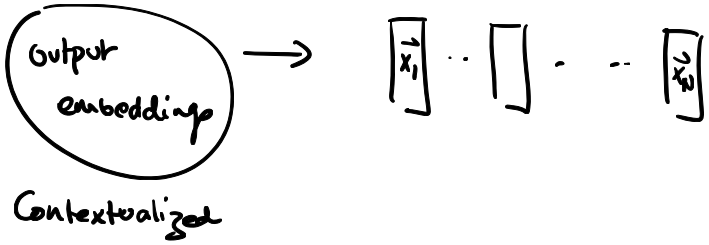


Cost per block $O(N^2)$

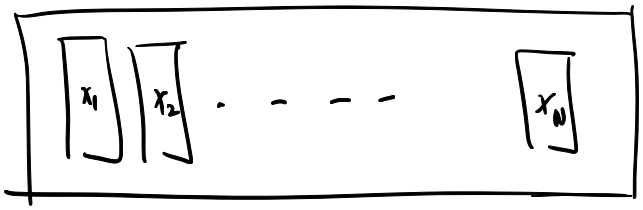


Same token

different embeddings in diff blocks



Self-attention



$\vec{x}_i$ → query representation $\vec{q}_i$

$\vec{x}_i$ → key representation $\vec{k}_i$

$\vec{x}_i$ → value representation $\vec{v}_i$

dimensionality

$\vec{x}_i$: d_{model} (512)

$\vec{q}_i$: d_k ($512/8 = 64$)

$\vec{k}_i$: d_k (64)

$\vec{v}_i$: d_v (64)

$\vec{x}_i$ ← attention

↳ $\vec{q}_i$ vs all keys $\vec{k}_j$ for all $j = 1, \ldots, N$

attention vector : $\vec{\alpha}_i = \text{Softmax}\left(\frac{\vec{q}_i^T \vec{k}_1}{\sqrt{d_k}}, \frac{\vec{q}_i^T \vec{k}_2}{\sqrt{d_k}}, \ldots, \frac{\vec{q}_i^T \vec{k}_N}{\sqrt{d_k}} \right)$

$\sqrt{d_k}$: rescaling of dot-product

α_{ij} : attention $\vec{x}_j$ pays to $\vec{x}_i$ ($\vec{k}_j$) ($\vec{q}_i$)

Update the value representation $\vec{v}_i$: weighted sum of all value vectors (in the block)

$$\vec{v}_i = \sum_{j=1}^{N} \alpha_{ij} \vec{v}_j$$

$\vec{x}_i$: d_{model}

W^Q : $(W^Q)^T \vec{x}_i = \vec{q}_i$

W^K : $(W^K)^T \vec{x}_i = \vec{k}_i$

W^V : $(W^V)^T \vec{x}_i = \vec{v}_i$

$(W^Q)^T \vec{x}_1$

$d_k \times d_{model}$ $\quad d_{model} \times 1$

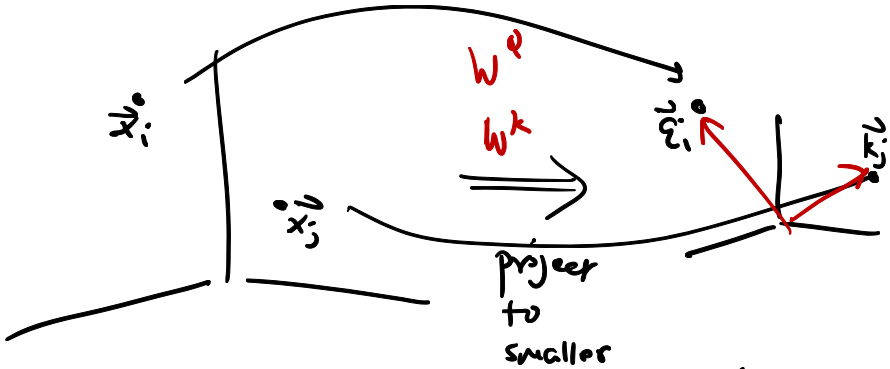
$d_k \times 1$

learnable parameters

W^Q : $d_{model} \times d_k$

W^K : $d_{model} \times d_k$

W^V : $d_{model} \times d_v$

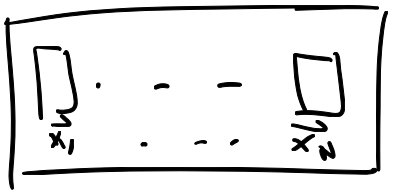


project to smaller space

512 d input space

64 d attention space

Matrix Approach : All attention for all elements of the block



X : $N \times d_{model}$ matrix

W^Q $\quad W^K$ $\quad W^V$

$Q = X \cdot W^Q$

$(N \times d_{model})(d_{model} \times d_k)$

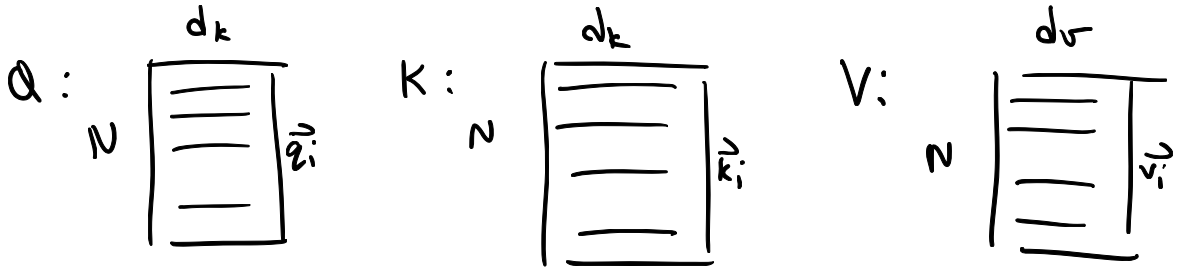
$= (N \times d_k)$

$K = X \cdot W^K$

$(N \times d_k)$

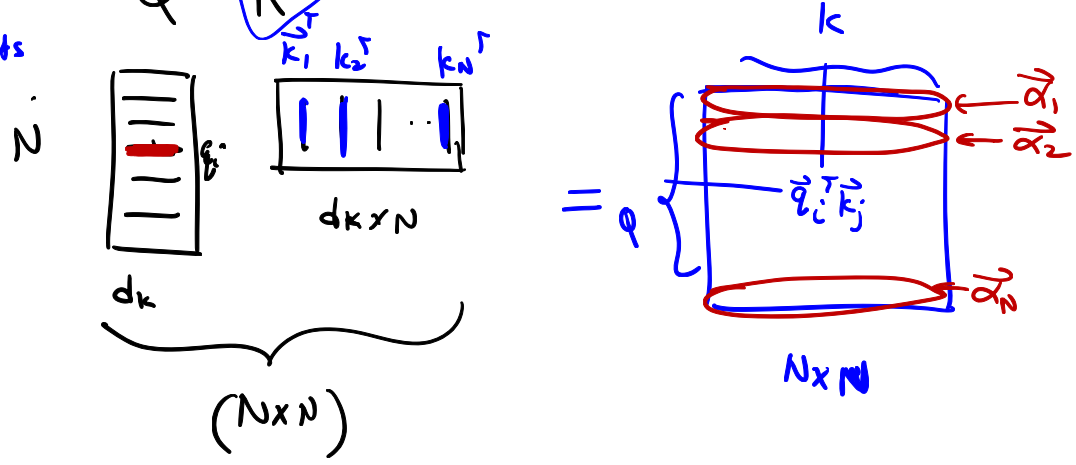
$V = X \cdot W^V$

$(N \times d_v)$



all pair-wise dot products

$$= Q \cdot K^T$$



$$A = \text{softmax}\left(\frac{Q \cdot K^T}{\sqrt{d_k}}\right)$$

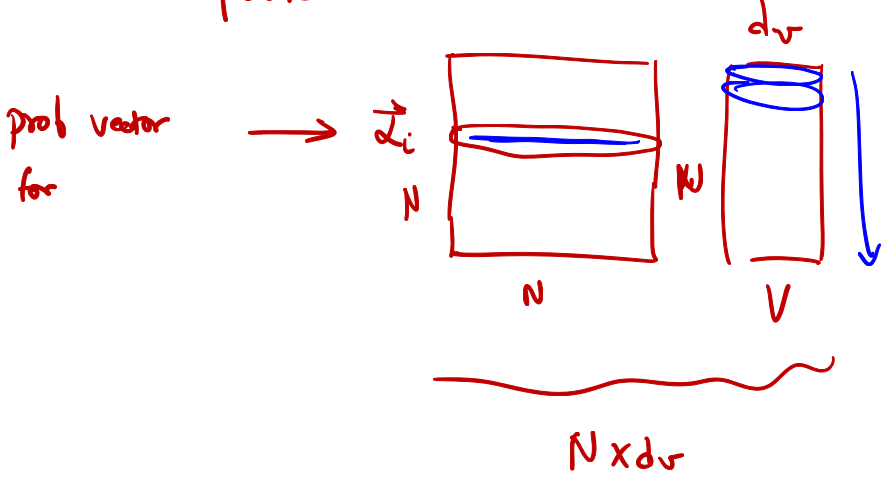
(rowwise)

α_{ij}

$O(N^2)$

$$V: \quad A \cdot V$$

updated



$$V' = \underbrace{\left[\text{Softmax}\left(\frac{Q \cdot K^T}{\sqrt{d_k}}\right)\right] \cdot V}_{A}$$

$\times$ $[\vec{x}_1\ \vec{x}_2\ \cdots\ \vec{x}_N]$ $\quad d_{model}$

self-attention layer

W^0

$W^Q\ W^K\ W^V$

Self-Attention

W^Q, W^K, W^V $\quad (d_k, d_v)$

$Q\ K\ V$

d_v ? V' : $\text{soft}\left(\frac{QK^T}{\sqrt{d_k}}\right) \cdot V$

$N \times d_v$

apply W^0 to map d_v to d_{model}

$V' \longrightarrow V' \cdot W^0 = X'$

$N \times d_v \quad (N \times d_v \quad d_v \cdot d_{model}) \quad (N \times d_{model})$

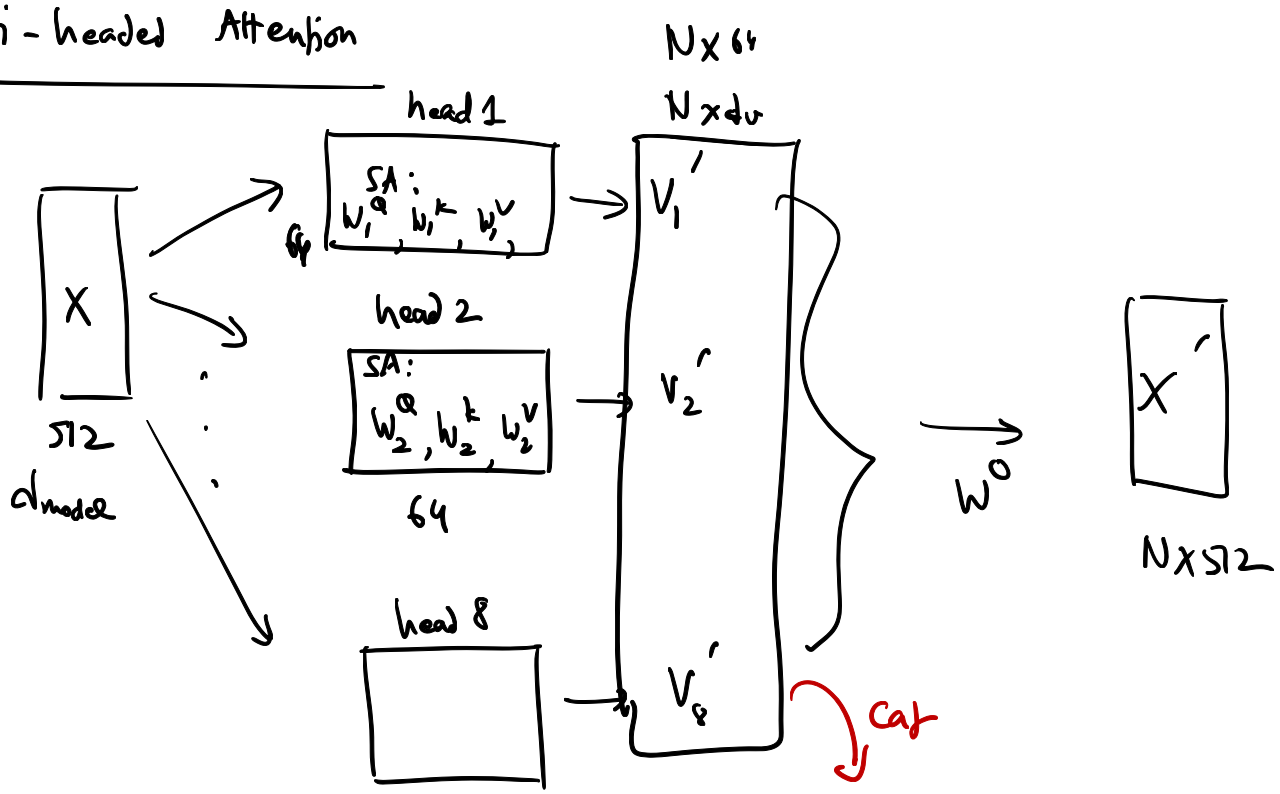
512 $\quad$ 64 $\quad$ 512 $\quad$ 512

$X \rightarrow$ SA $\rightarrow X' \rightarrow$ SA $\rightarrow X'' \rightarrow \cdots$

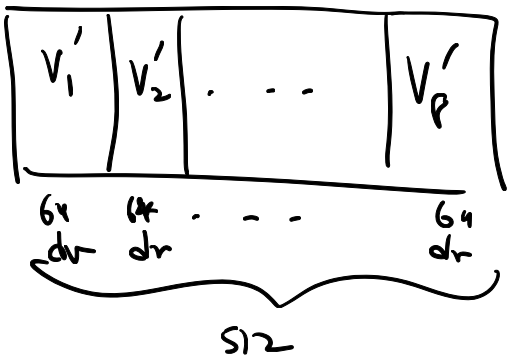
self attention

(Single head)

Multi-headed Attention



C^V : N



$h \cdot d_v$

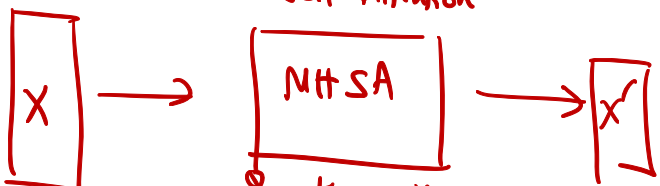
↑ # of heads ← Value dim

$$X' = C^V \cdot W^O$$

$$(N \times hd_v)(hd_v \times d_{model})$$

$$8 \cdot 64 \times 512$$

Multi-head self Attention

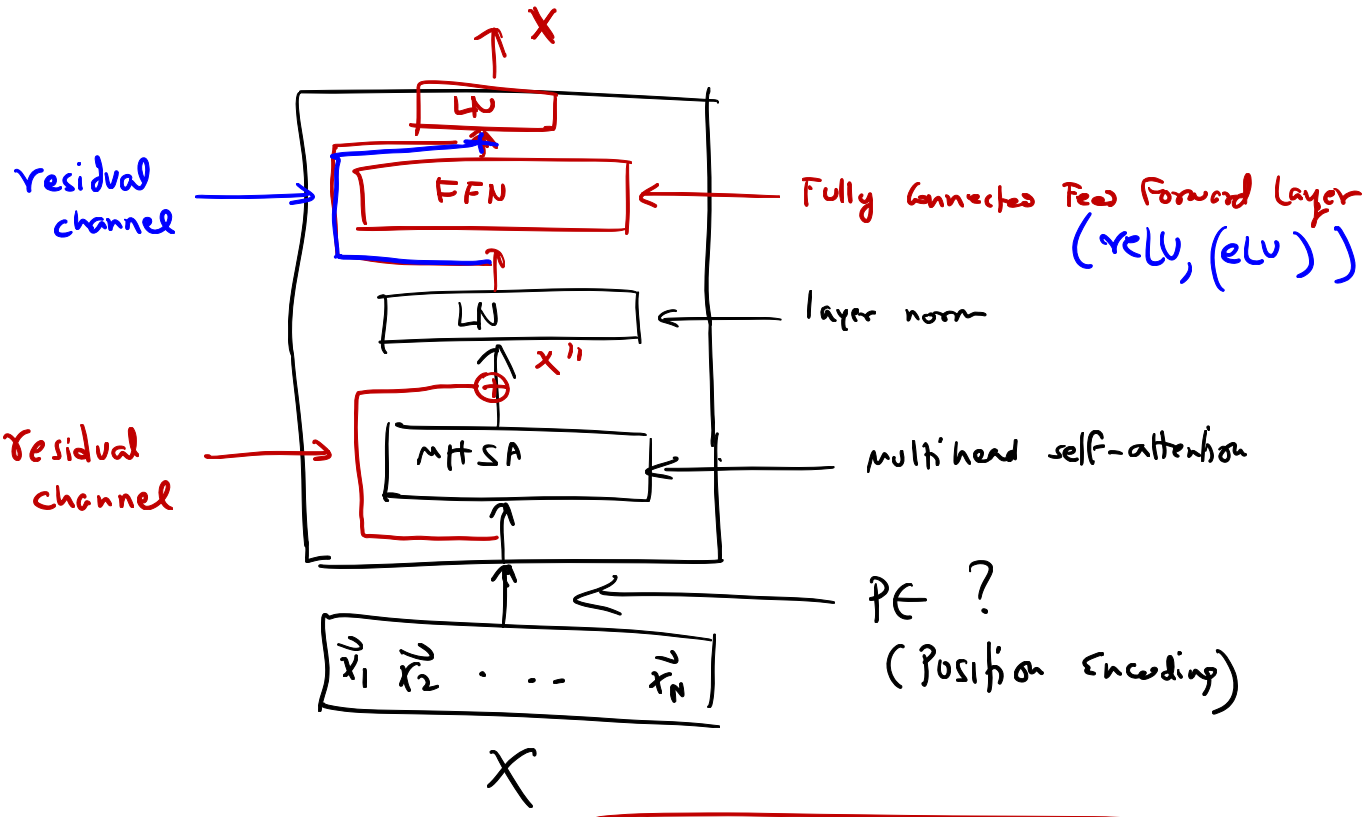


$W_i^Q \; W_i^K \; W_i^V$

$i = 1 \ldots h$

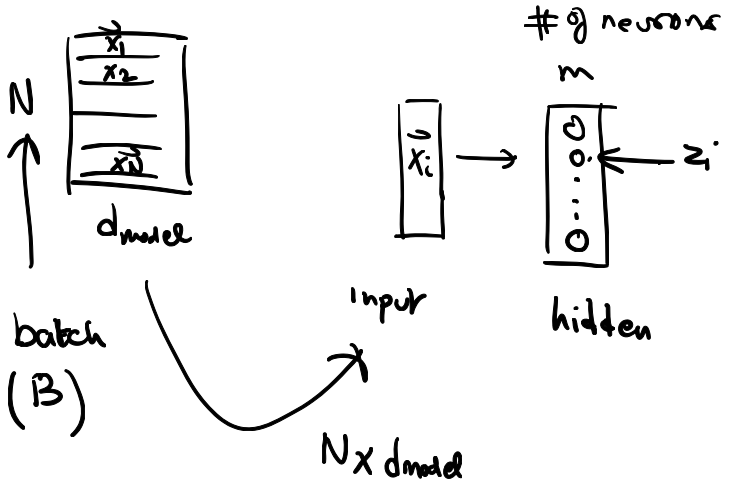
W^O

Transformer Block



Residual channel: $X'' = X + \boxed{X' = MHSA(X)}$

Batch-Norm



Layer Norm

$$z_i = \alpha_1 \left(\frac{z_i - \mu_1}{\sigma_1} \right) + \beta_1$$

$(\vec{x}_1)$

$$\mu_1 = \frac{1}{m} \left(z_1(x_1) + z_2(x_1) + \cdots + z_m(x_1) \right)$$

x_1

$$\sigma_1 = \sqrt{\frac{1}{m} \sum_i \left(z_i(x_1) - \mu_1 \right)^2}$$

$$z_i = \frac{z_i - \mu_i}{\sigma_i} \quad \text{(z-score)}$$

$$\rightarrow \mu_i = \frac{1}{N}\left(z_i(x_1) + z_i(x_2) + \dots z_i(x_N) \right)$$

mean response across the batch

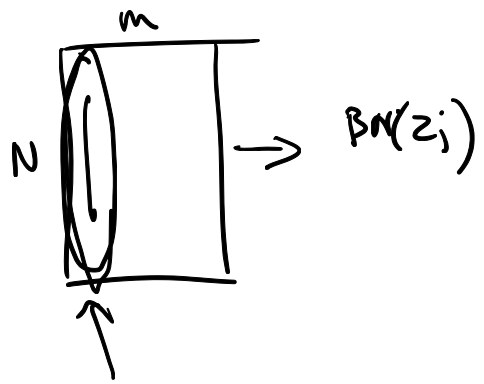
$$\sigma_i^2 = \frac{1}{N}\left((z_i(x_1) - \mu_i)^2 + \dots (z_i(x_N) - \mu_i)^2 \right)$$

$$\sigma_i = \sqrt{\sigma_i^2}$$

$$z_i = \alpha_i\left(\frac{z_i - \mu_i}{\sigma_i}\right) + \beta_i$$

α_i: gain, β_i: bias

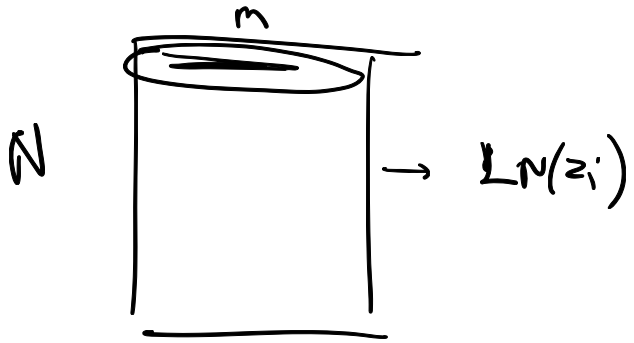
Batch Norm



avg across the batch elements

$$\mu = \frac{1}{N}\sum(\quad)$$

layer norm



avg across the layer (m)

$$\frac{1}{m}\sum(\quad)$$

Word2vec / Prot2vec

Vocab + Counts k-mer k=3

↳ do k-way sliding window

Collect the vocab + count in (defaultdict)

Negative sampling