

# Transformers

## Plan

Review

Self-Attention

Cross-Attention

Positional Encoding

## Logistics

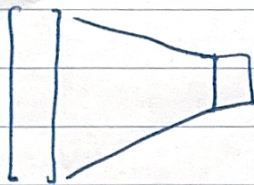
check in

scribe

Zoom

## Review

Motivation: words as vectors



one-hot encoding

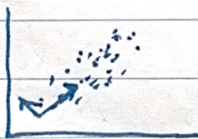
$$f: \mathbb{R}^{|\mathcal{V}|} \rightarrow \mathbb{R}^d$$

$(x, x^+)$  close  $\Rightarrow f(x) \cdot f(x^+) \approx$  large

$(x, x^-)$  far  $\Rightarrow |f(x) \cdot f(x^-)| \approx$  small

↻ Contrastive learning  $\subseteq$  unsupervised learning

↙ Principal Component Analysis



capture variation of data  
via eigenvectors

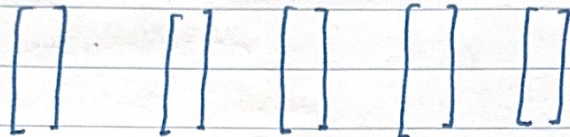
$$X^T X = \sum_{i=1}^n \lambda_i v^{(i)} v^{(i)T} \quad \text{for } v^{(i)} \cdot v^{(j)} = \begin{cases} 1 & \text{if } i=j \\ 0 & \text{else} \end{cases}$$

$$\max_{v: \|v\|_2=1} \|Xv\|_2^2 \leftarrow v^T X^T X v \leftarrow \sum_{i=1}^n \lambda_i [v^T v^{(i)}]^2$$



Motivation: sentences as vectors

"Vermont is chilly and beautiful"



How can we understand sequences of vectors?

↳ Recurrent networks

↳ LSTM

Attention! (self first)  $X \in \mathbb{R}^{n \times d}$

Goal: Combine similar words/tokens

Queries:  $W^{(Q)} X = Q$

$W^{(Q)} \in \mathbb{R}^{r \times d}$

Keys:  $W^{(K)} X = K$

Values:  $W^{(V)} X = V$

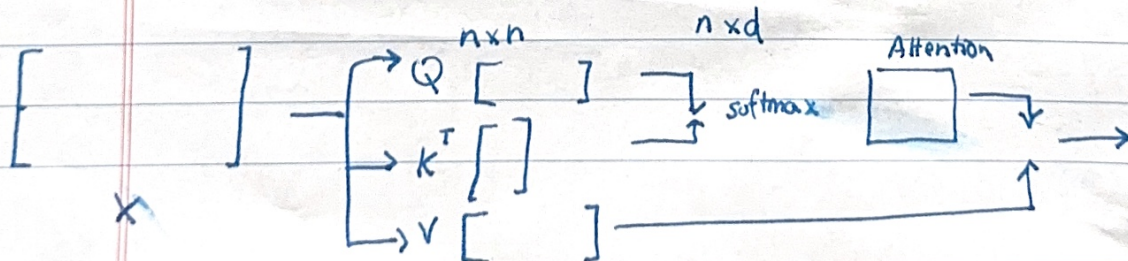
$W^{(V)} \in \mathbb{R}^{d \times d}$

attention:  $\underset{\substack{\uparrow \\ \text{to rows}}}{\text{softmax}(Q K^T)} = \begin{bmatrix} \vdots \\ \vdots \\ \vdots \end{bmatrix} \begin{matrix} \swarrow \\ \vdots \\ \searrow \end{matrix} Q_i^T K_j$

result:  $\text{softmax}(Q K^T) V$

$n \times n$

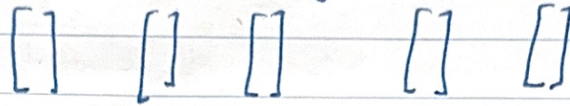
$\sum_{i=1}^n \text{sim}(j,i) v_i$



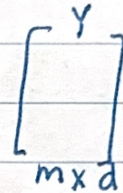
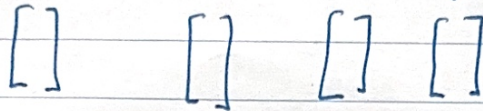


# Cross-Attention!

"Vermont is chilly and beautiful"



"Vermont es fria y"



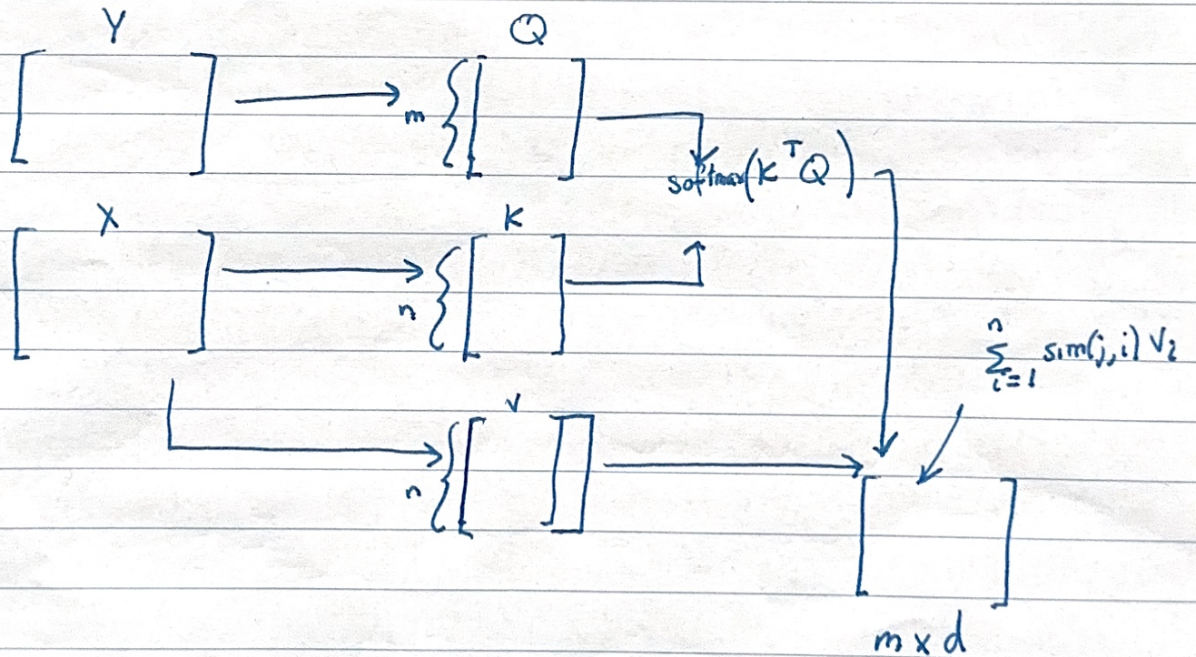
Goal: Represent sequence as linear combo of another

Queries:  $W^{(Q)} X = Q$

$W^{(K)} X = K$

$W^{(V)} X = V$

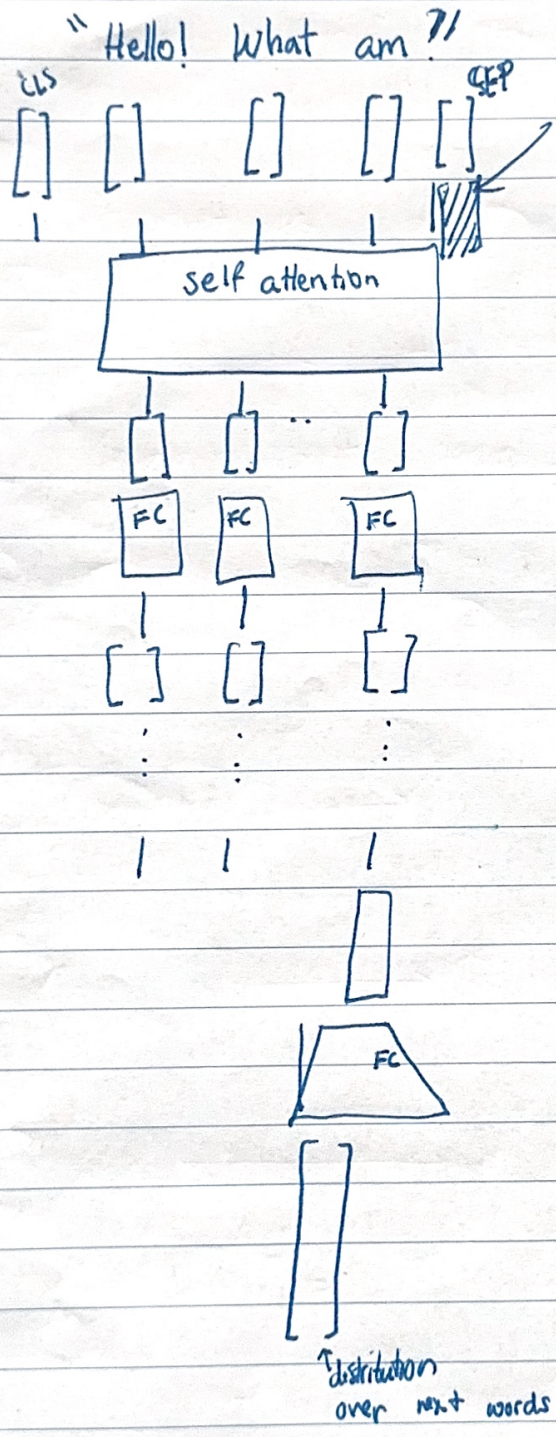
$X \in \mathbb{R}^{n \times d}$   
 $W^{(Q)} \in \mathbb{R}^{n \times d}$   
 $W^{(K)} \in \mathbb{R}^{n \times d}$   
 $W^{(V)} \in \mathbb{R}^{d \times d}$





Make upwards direction?

# Large Language Models



## Positional Encoding?

We represent time as  
 11:24am Tuesday, Jan 14, 2025  
 rather than  
 1,065,066,880 min since 0 BC

- ↳ min captures schedule
- ↳ hour captures time of day
- ↳ day captures schedule
- ↳ date captures schedule
- ↳ month captures time of year
- ↳ year captures years passed

"Hello! What am I?"

0 1 2 = t

