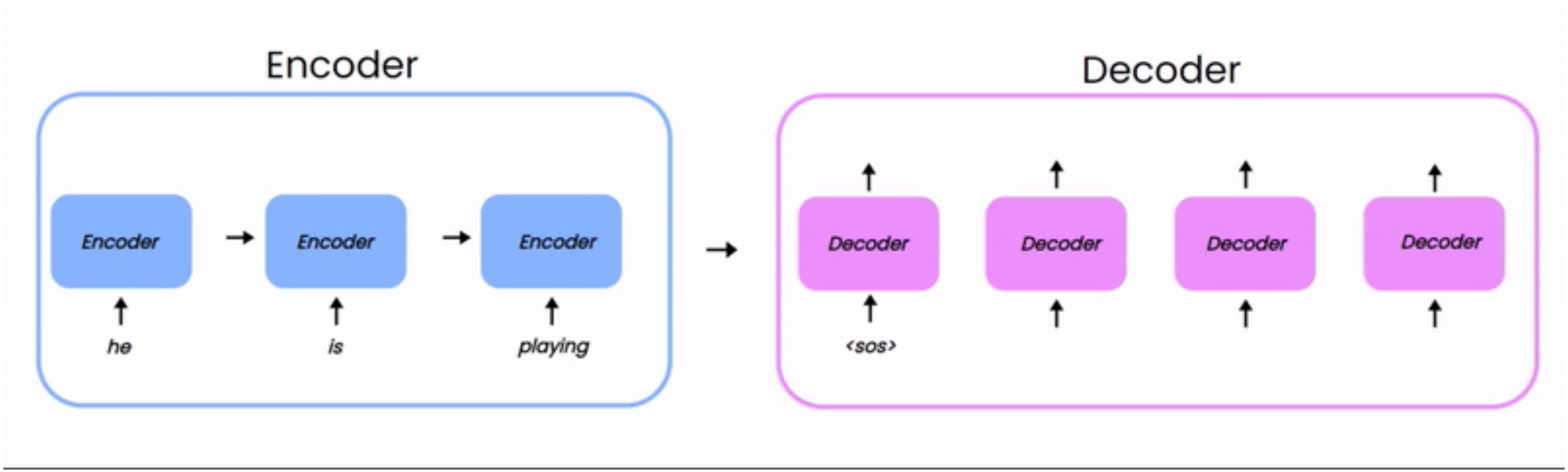
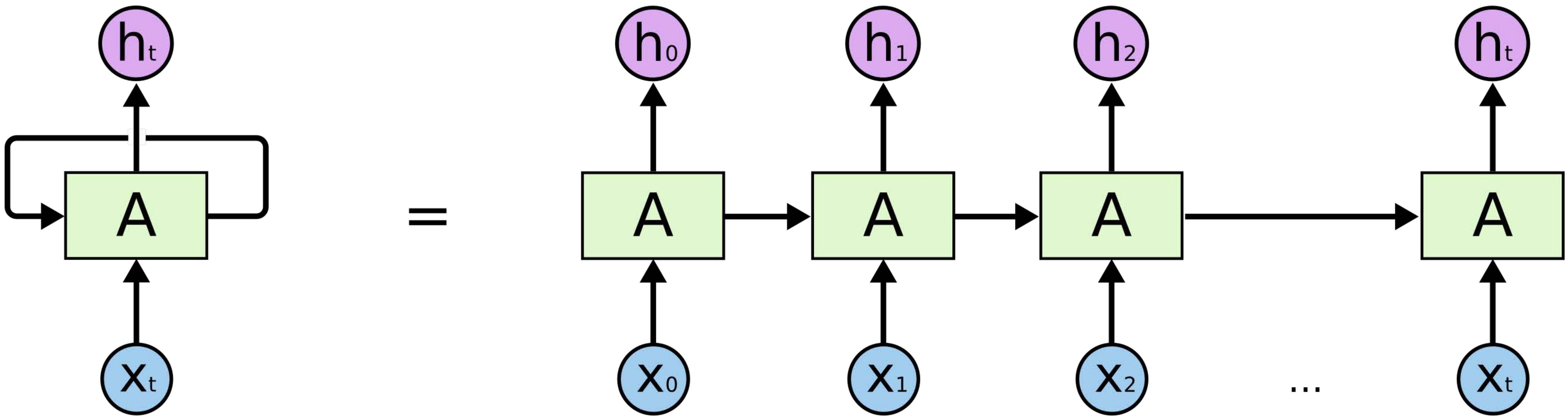
ENM 3600: Data-driven Modeling

Lecture #15: Recurrent neural networks

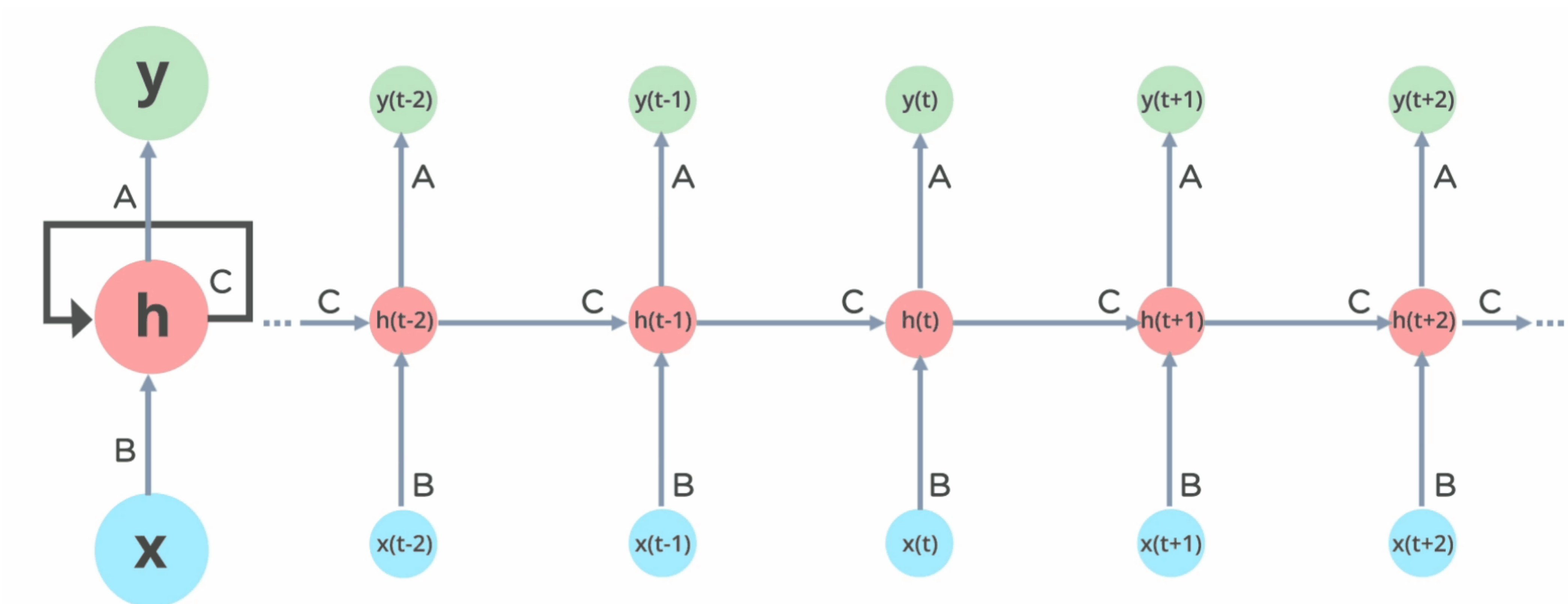
Modeling of sequence data



RNNs



An unrolled recurrent neural network.



http://colah.github.io/posts/2015-08-Understanding-LSTMs/

RNN limitations

- Sequential processing is slow (can't parallelize)
- Long-range dependency issues (vanishing gradients)
- Limited context window in practice

  "The cat is on the table"

  "The cat, who belongs to my mother, is on the table".

E.g.: when translating we need to remember "cat" is the subject even after processing many words.

- A need for better models:
  - Language translation requiring full sentence context
  - Document summarization
  - Question answering

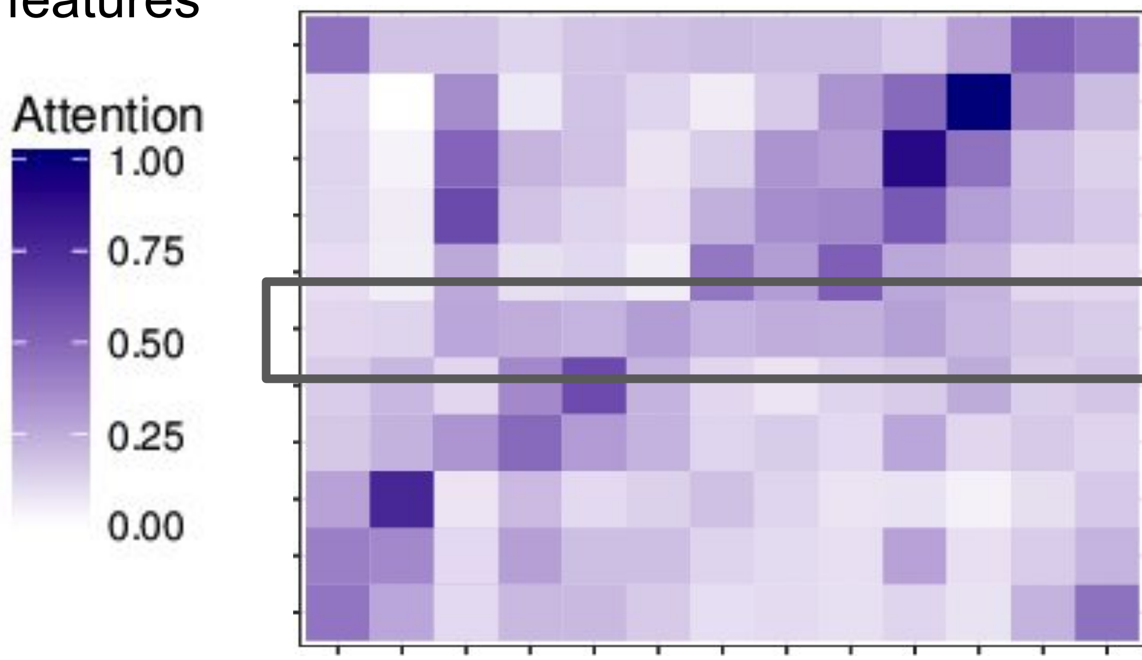
  - Protein Structure Prediction (long amino acid sequences)
  - Molecular Property Prediction (multi-atom interactions)
  - Climate Science (long timescales)
  - Astronomical Data Analysis (vast spatial and time scales)
  -

Key Innovation: Attention

Bahdanau Attention

Input sentence $[u_1, \ldots, u_i, \ldots, u_{T_u}]$

Input features $v_1 \quad \cdots \quad v_j \quad \cdots \quad v_{T_u}$



$\varphi_1, \ldots, \varphi_i, \ldots, \varphi_{T_s}$

Attention weights φ_{ij}

$$\begin{bmatrix} c_1 \\ \vdots \\ c_i \\ \vdots \\ c_{T_s} \end{bmatrix} \rightarrow c_i = \sum_{j=1}^{T_u} \varphi_{ij} v_j$$

Context vector

$p(s_i \mid c_i, u)$ Probability of word s_i at output position i

$[s_1, \ldots, s_i, \ldots, s_{T_s}]$ Output sentence

Bahdanau, Dzmitry, Kyung Hyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate." *3rd International Conference on Learning Representations, ICLR 2015*. 2015.

From Sequential to Parallel Processing

- Compare how humans read vs how RNNs process text
  - Humans: Quick glances at relevant parts
  - RNNs: Must process word by word
- Attention is "looking" at all words simultaneously

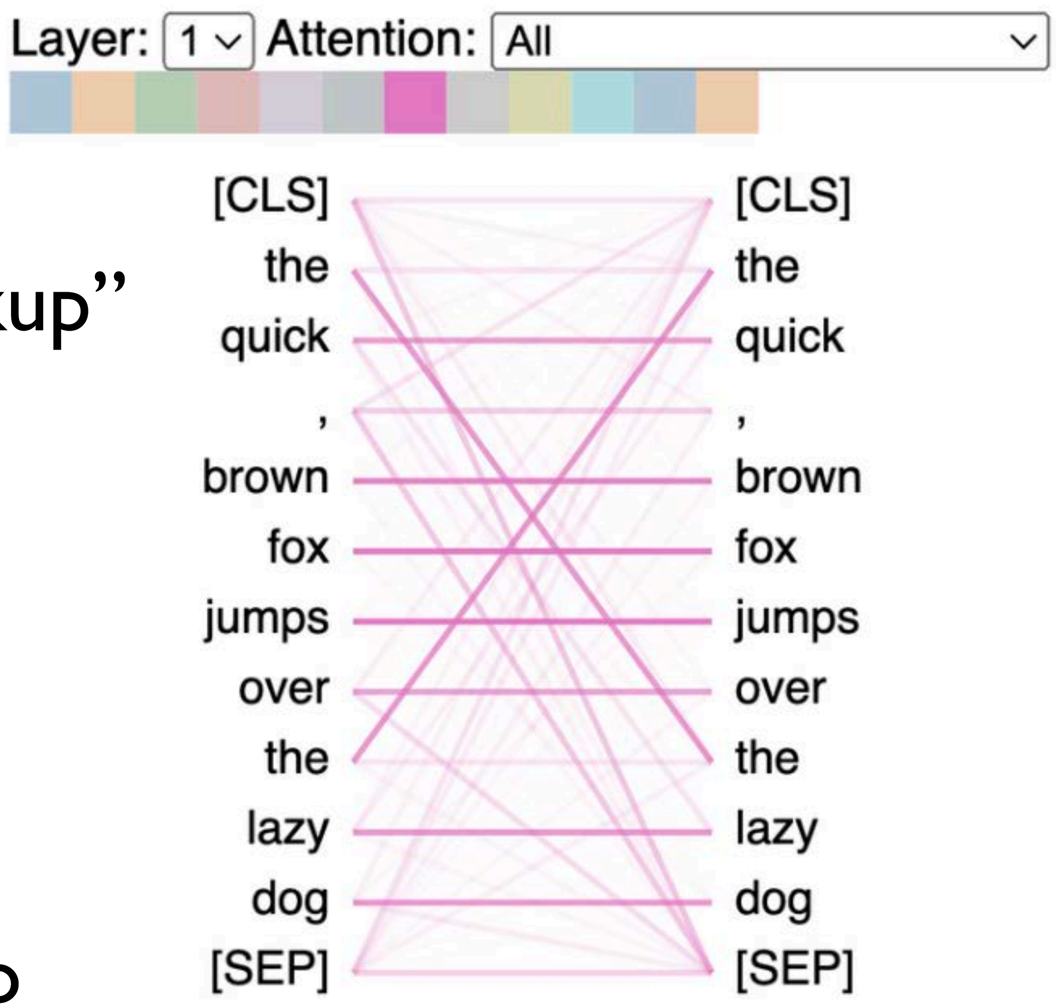
Key innovation: Attention

Think of attention as a "smart dictionary lookup”

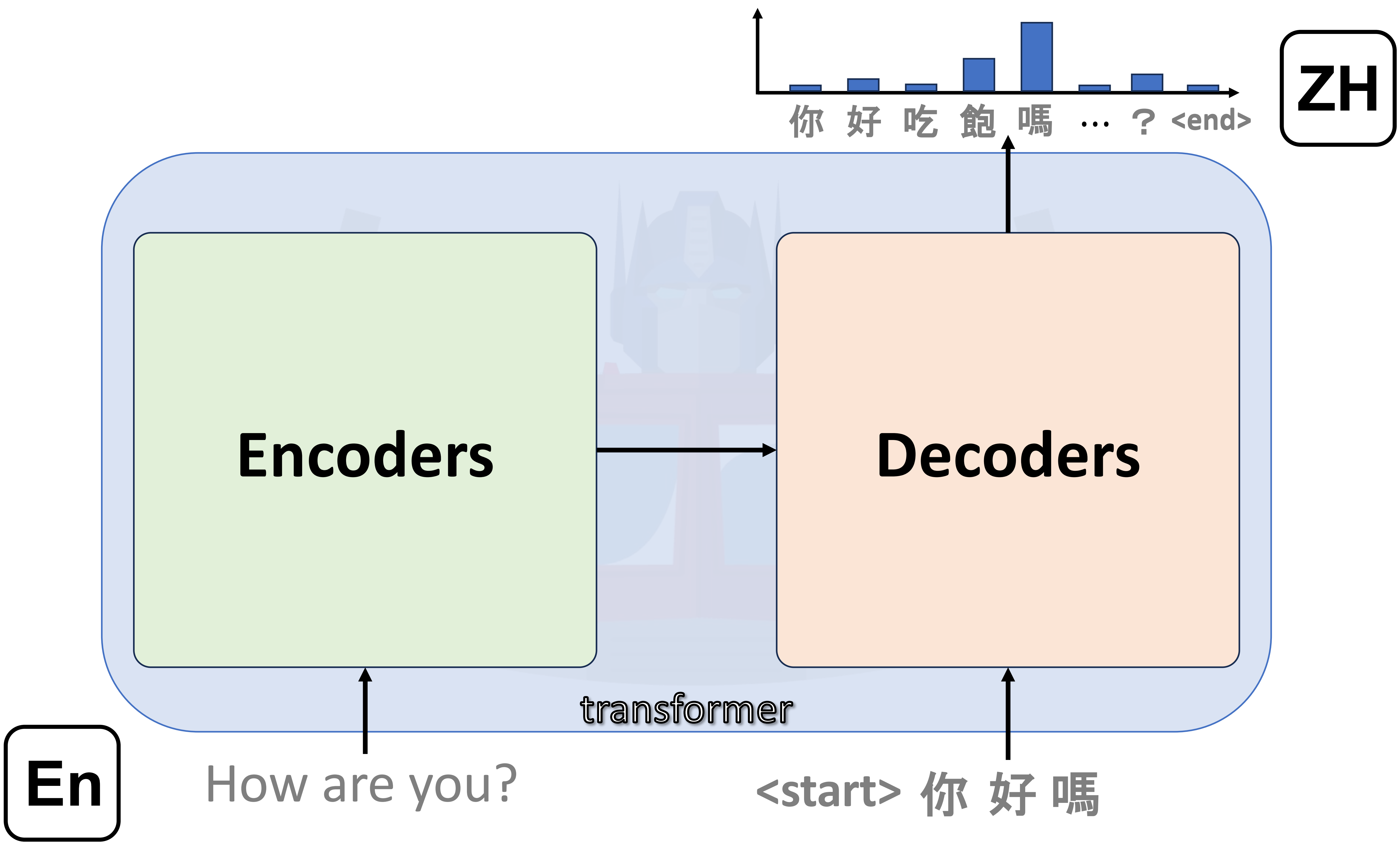
- Keys: What you're looking for
- Values: What you get back
- Queries: What you're asking for

Simple example: Translation attention

- Each word in target language might need to "look at" multiple source words



Transformers



Encoder-Decoder Auto-regressive Models

Slide by Jia-Bin Huang, University of Maryland College Park

Key Building Blocks

- Tokenization
- Token embeddings
- Position embeddings
- Self-attention & Multi-head self-attention
- MLPs
- Cross-attention

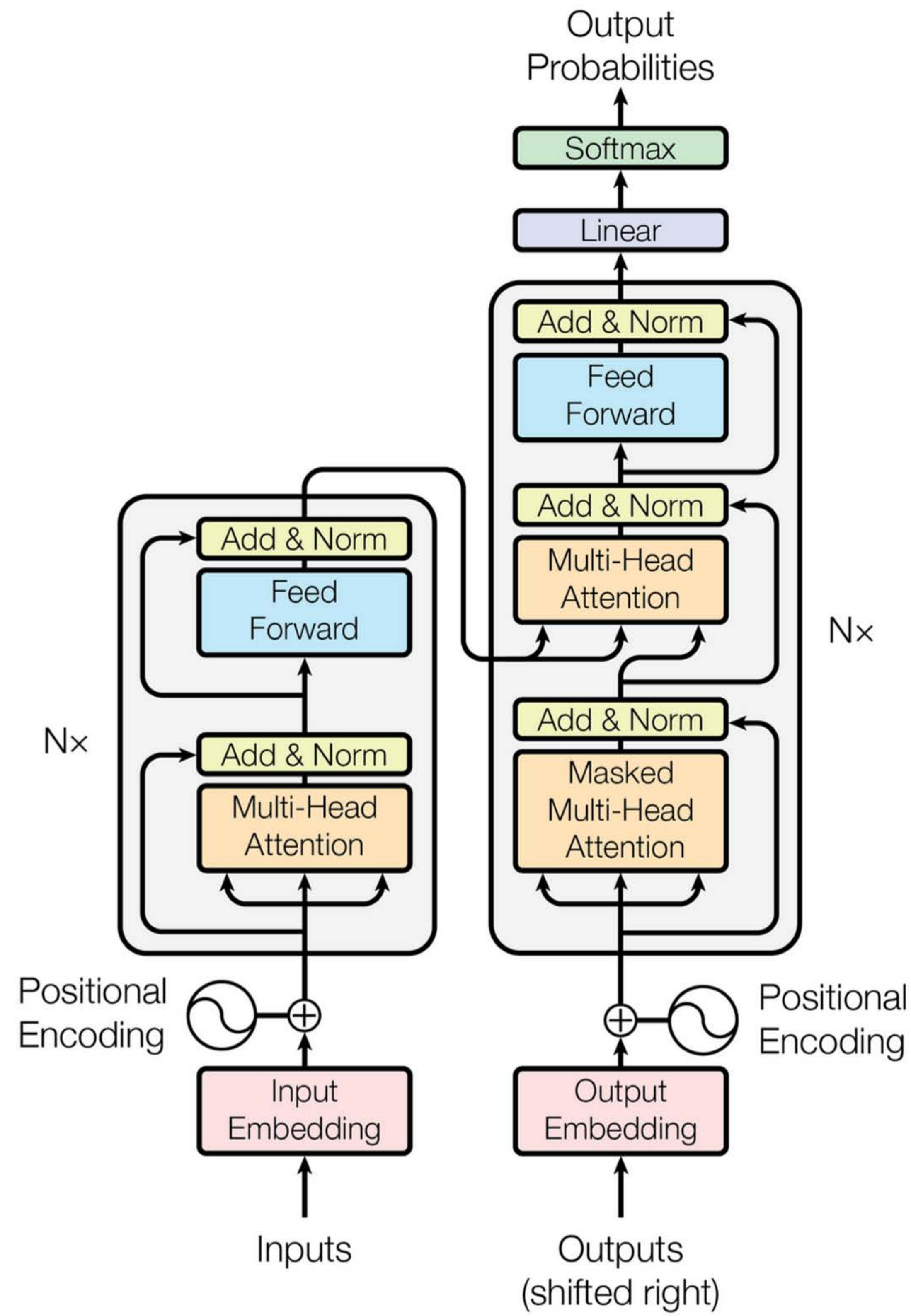


Figure 1: The Transformer - model architecture.

Vaswani, A. (2017). Attention is all you need. *Advances in Neural Information Processing Systems.*

Tokenization in NLP

Many	words	map	to	one	token	,	but	some	don	't	:	indiv	isible	.
8607	4339	2472	311	832	4037	11	719	1063	1541	956	25	3687	23936	13

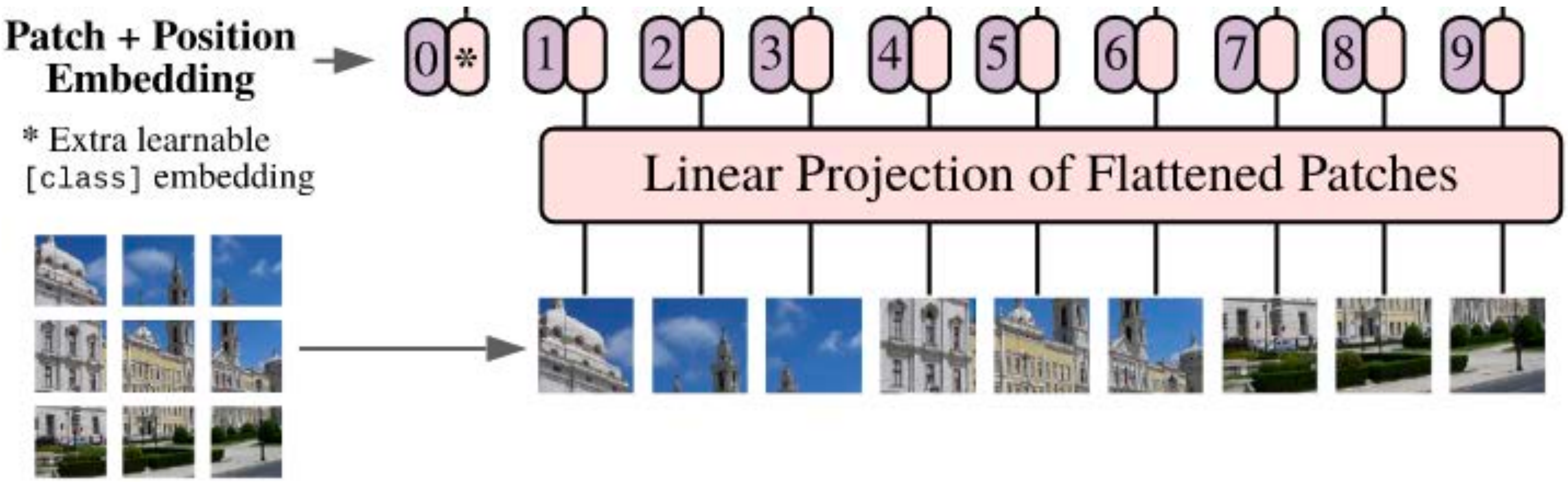
One-hot encoding

$$
\begin{array}{ccccccc}
 & \text{cat} & \text{dog} & \text{bear} & \text{cow} & & \text{indiv} \\
\text{\# tokens} & \begin{bmatrix}1\\0\\0\\0\\0\\\vdots\\0\end{bmatrix} & \begin{bmatrix}0\\1\\0\\0\\0\\\vdots\\0\end{bmatrix} & \begin{bmatrix}0\\0\\1\\0\\0\\\vdots\\0\end{bmatrix} & \begin{bmatrix}0\\0\\0\\1\\0\\\vdots\\0\end{bmatrix} & & \begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\0\end{bmatrix}
\end{array}
$$

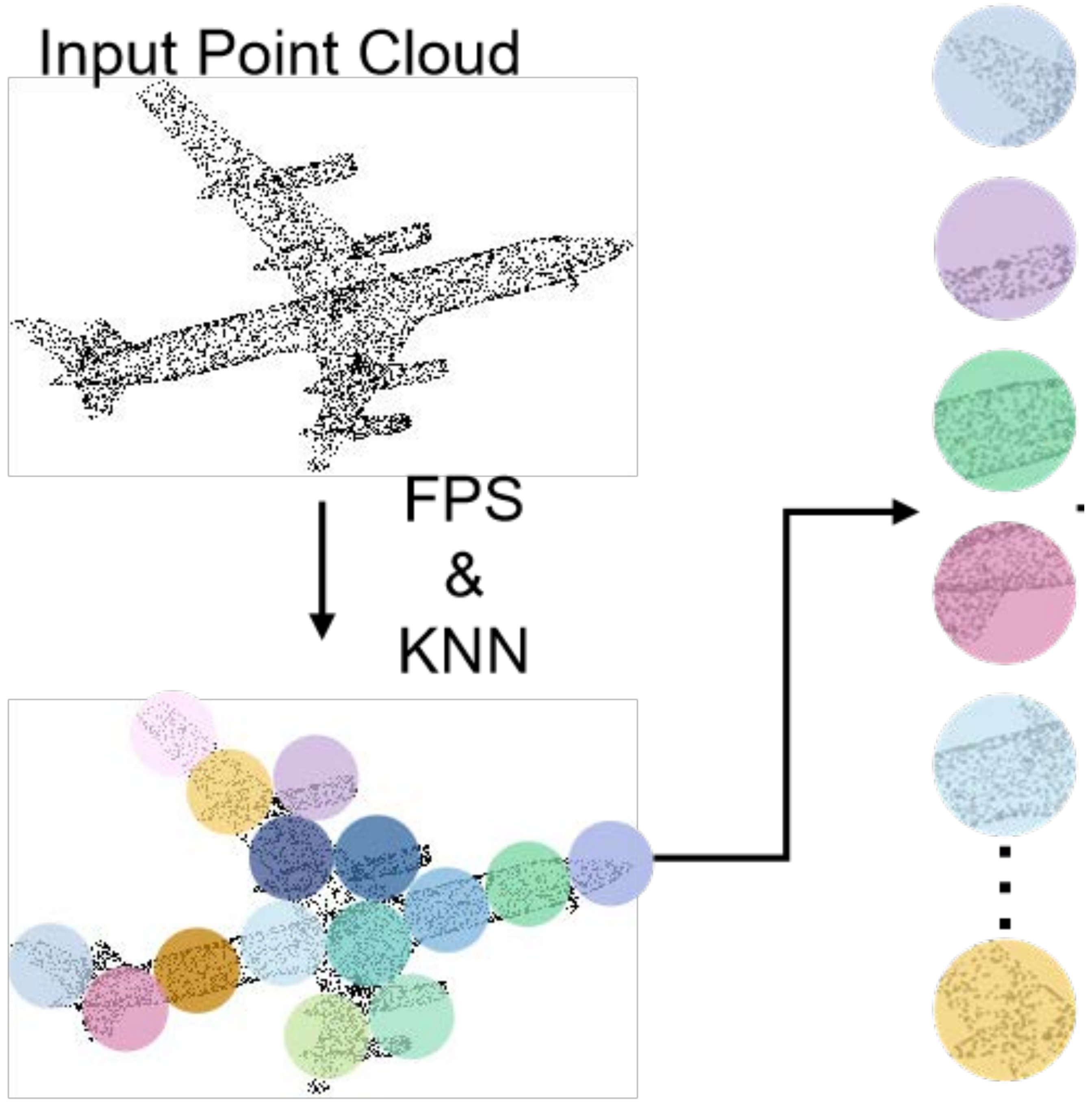
Value 1 at 3687th entry

Slide by Jia-Bin Huang, University of Maryland College Park

Tokenization in Vision



Tokenization in Point Clouds



Tokenization in NLP

Many words map to one token, but some don't: indivisible.

Many	words	map	to	one	token	,	but	some	don	'	t:	indiv	isible	.
8607	4339	2472	311	832	4037	11	719	1063	1541	956	25	3687	23936	13

One-hot encoding

$$
\begin{array}{ccccccc}
 & \text{cat} & \text{dog} & \text{bear} & \text{cow} & & \text{indiv} \\
\text{\# tokens} & \begin{bmatrix}1\\0\\0\\0\\0\\\vdots\\0\end{bmatrix} & \begin{bmatrix}0\\1\\0\\0\\0\\\vdots\\0\end{bmatrix} & \begin{bmatrix}0\\0\\1\\0\\0\\\vdots\\0\end{bmatrix} & \begin{bmatrix}0\\0\\0\\1\\0\\\vdots\\0\end{bmatrix} & & \begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\0\end{bmatrix}
\end{array}
$$

Value 1 at 3687th entry

Slide by Jia-Bin Huang, University of Maryland College Park

Token Embeddings

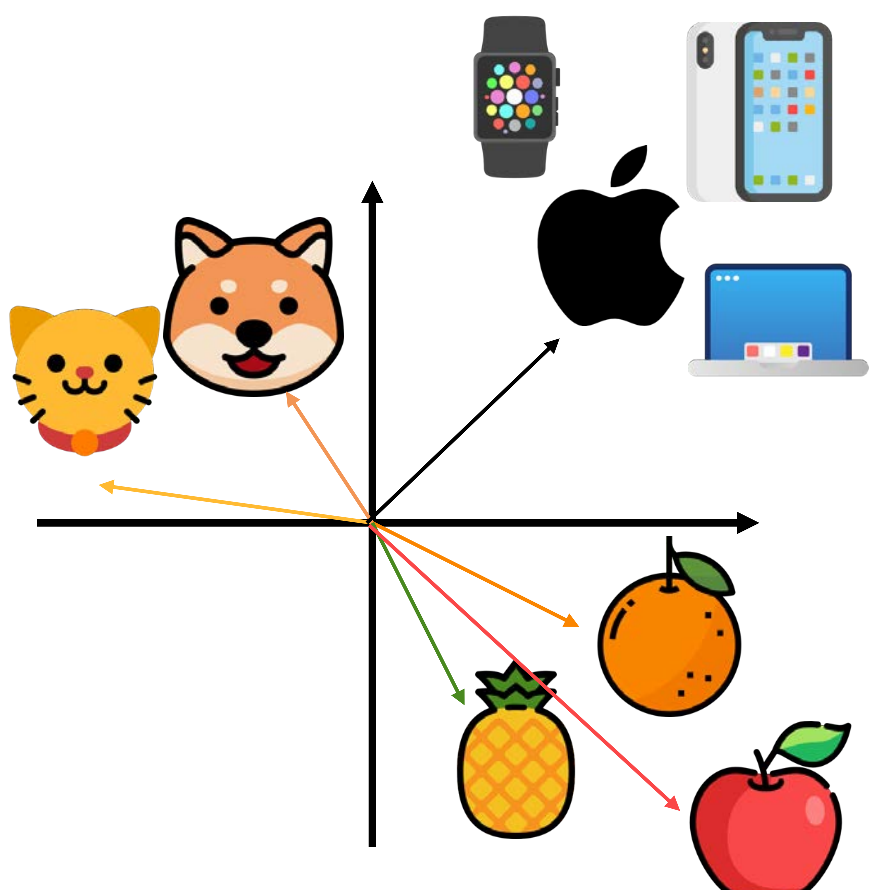
$$
\begin{array}{ccccc}
\text{cat} & \text{dog} & \text{bear} & \text{cow} & \text{indiv} \\
\begin{bmatrix}1\\0\\0\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\1\\0\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\1\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\1\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\0\end{bmatrix}
\end{array}
$$

Value 1 at 3687th entry

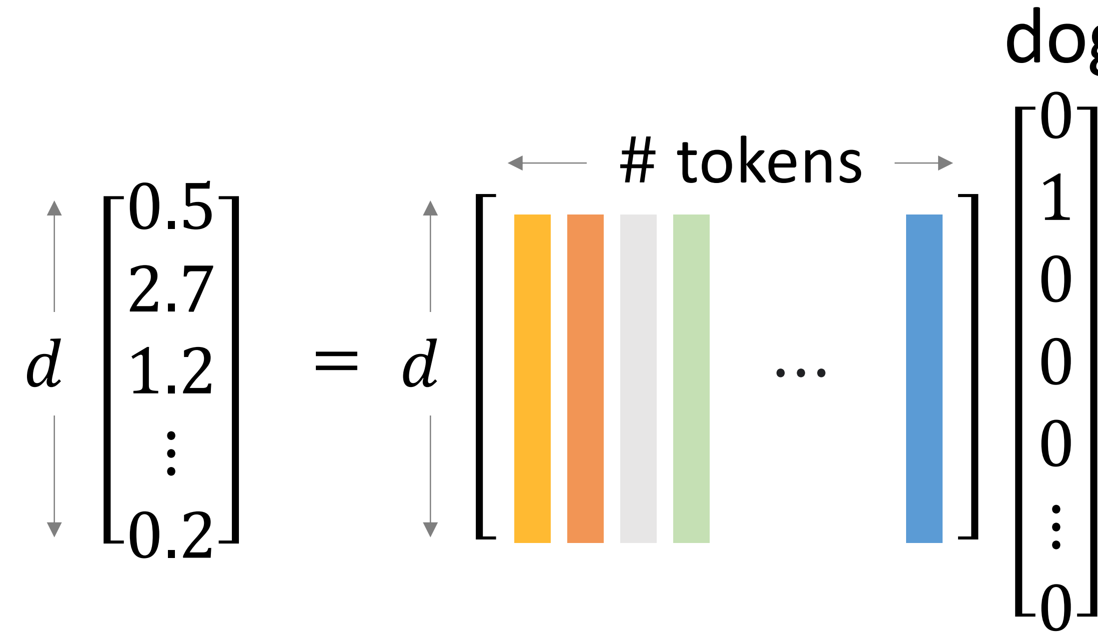
I bought an **apple** and an orange.

I bought an **apple** watch.

Apple



Embedding Space



Embedded token

Embedding Matrix

Slide by Jia-Bin Huang, University of Maryland College Park

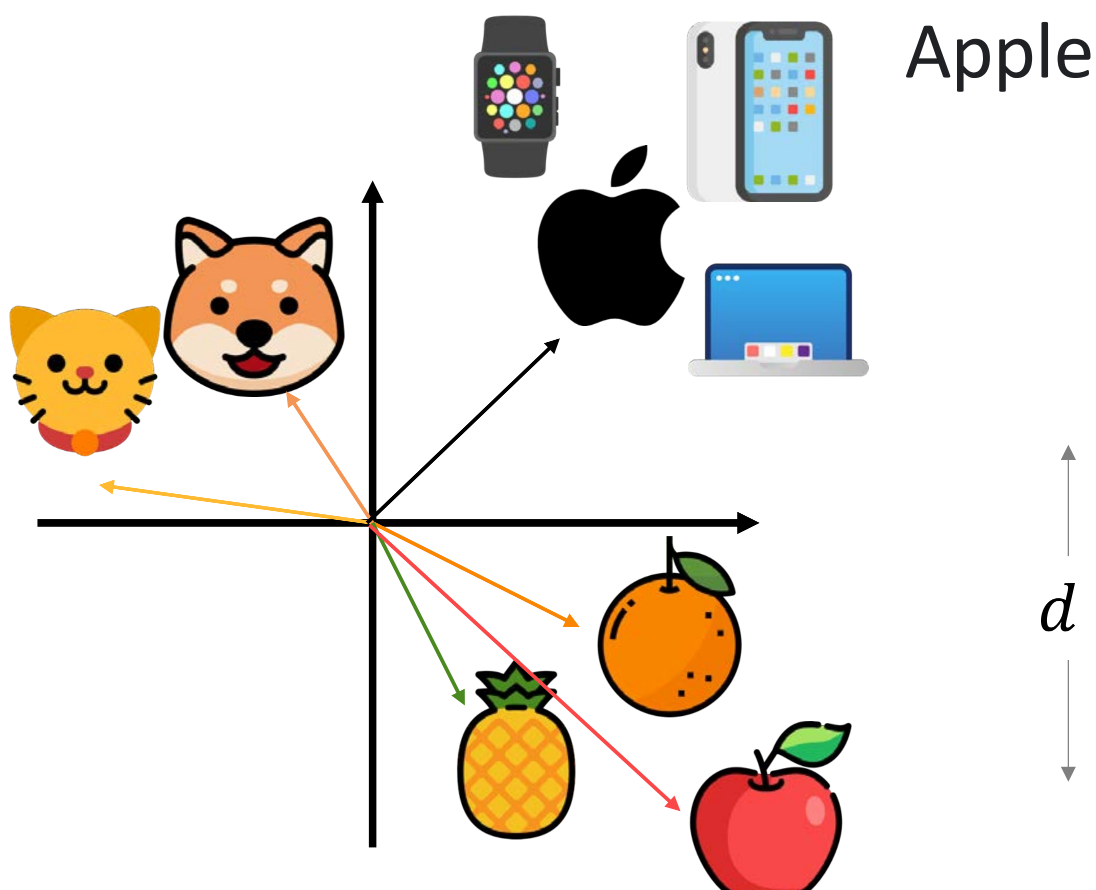
Token Embeddings

$$
\begin{array}{ccccc}
\text{cat} & \text{dog} & \text{bear} & \text{cow} & \text{indiv} \\
\begin{bmatrix}1\\0\\0\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\1\\0\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\1\\0\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\1\\0\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\0\end{bmatrix}
\end{array}
$$

Value 1 at 3687th entry

I bought an **apple** and an orange.

I bought an **apple** watch.



Embedding Space

$$
d \begin{bmatrix}0.5\\2.7\\1.2\\\vdots\\0.2\end{bmatrix} = d
$$

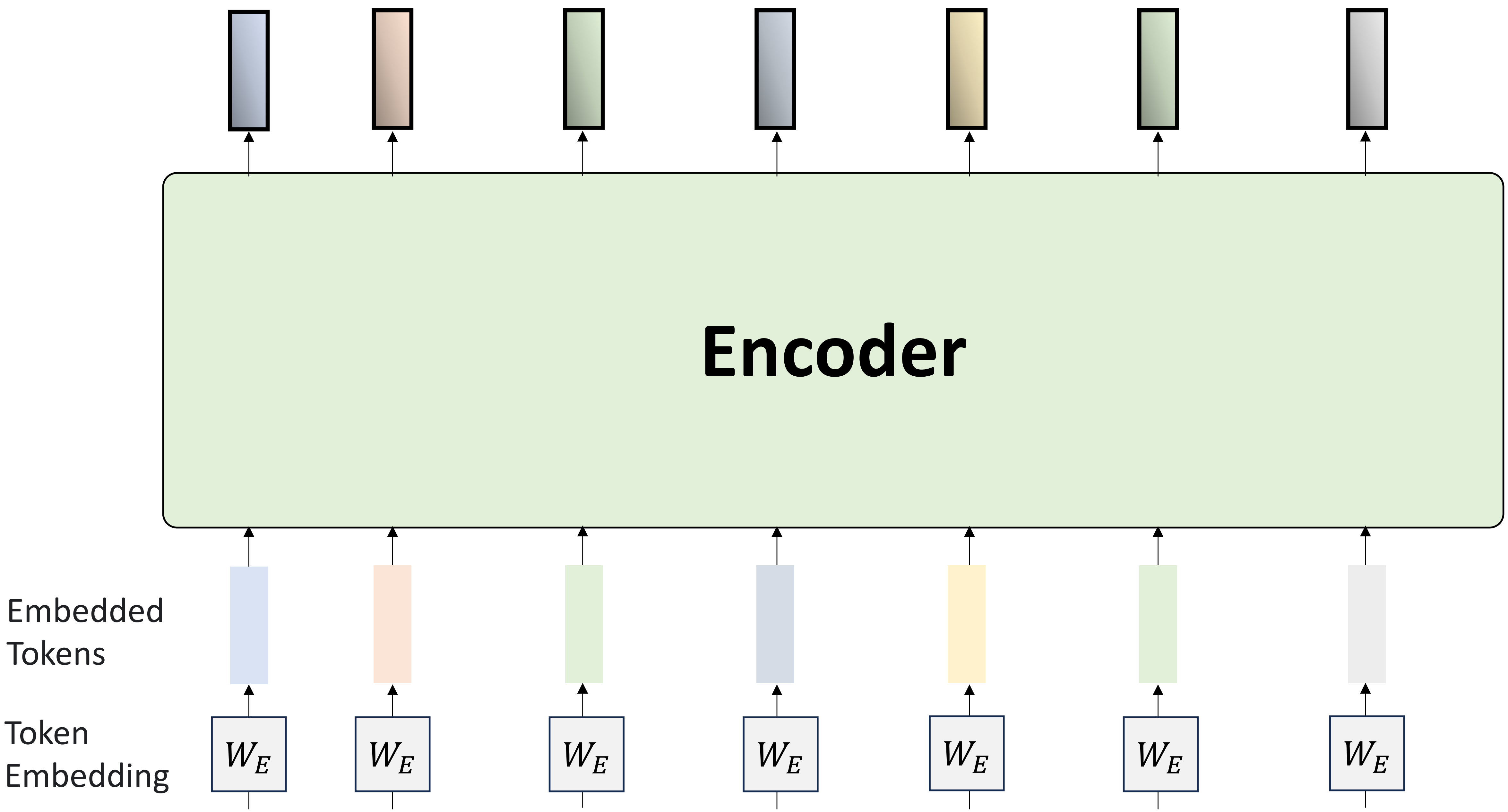
$$
\begin{array}{c}
\text{dog} \\
\begin{bmatrix}0\\1\\0\\0\\0\\\vdots\\0\end{bmatrix}
\end{array}
$$

Embedded token

Embedding Matrix

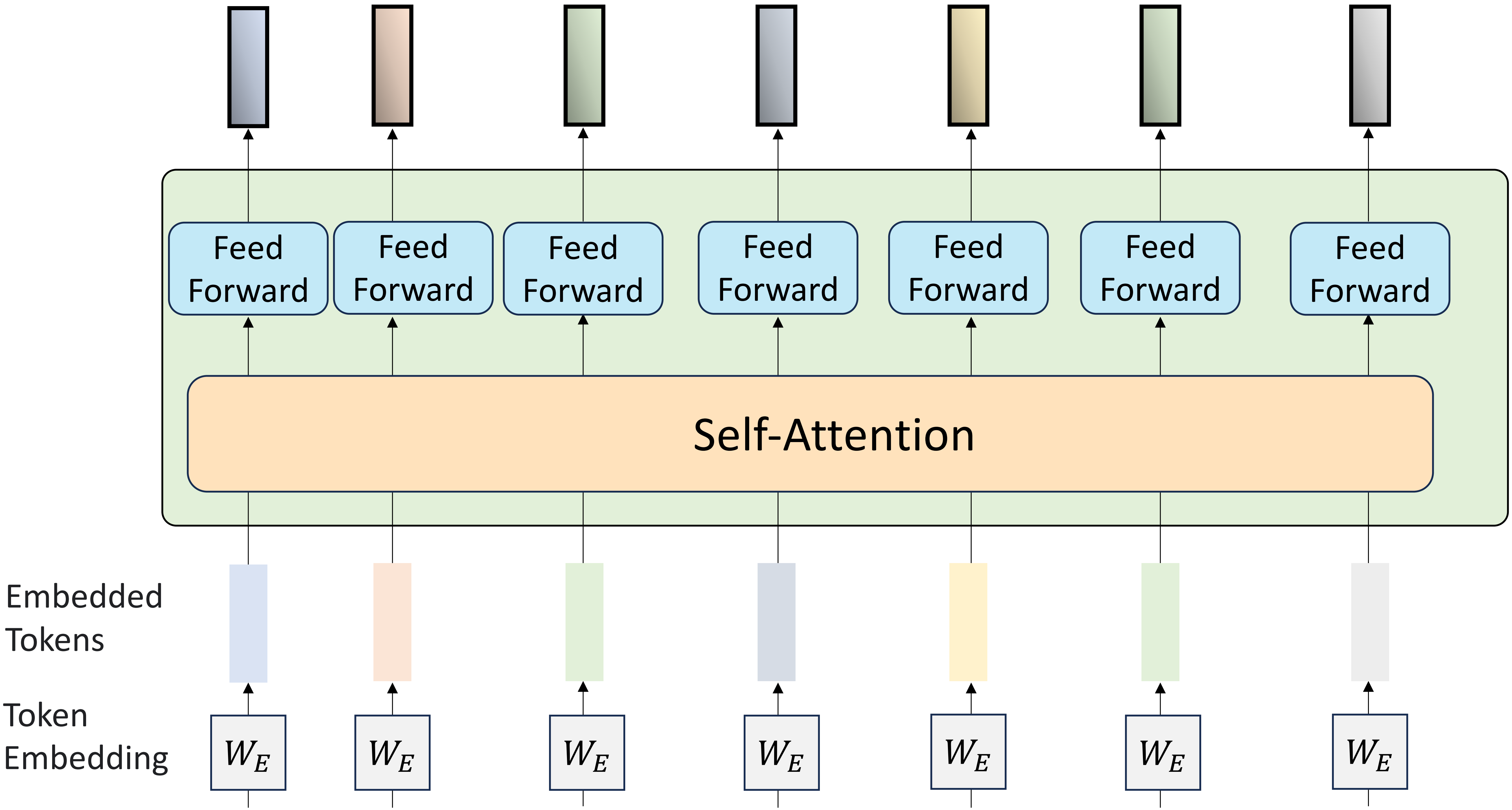
Slide by Jia-Bin Huang, University of Maryland College Park

Encoder



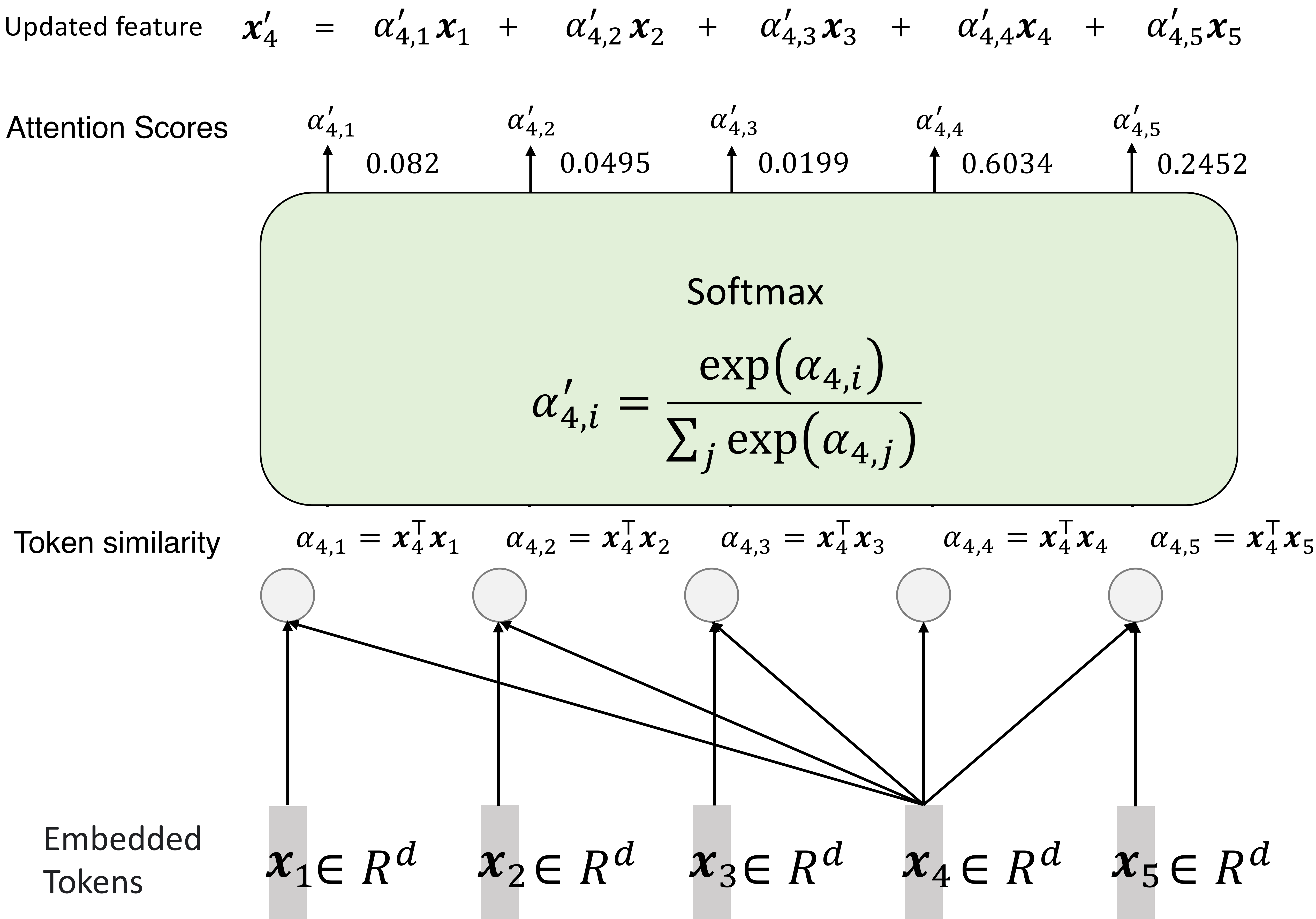
Slide by Jia-Bin Huang, University of Maryland College Park

Encoder



Slide by Jia-Bin Huang, University of Maryland College Park

Self-Attention



Self-Attention

Updated feature $\boldsymbol{x}'_4 = \boxed{W^O}\left(\alpha'_{4,1} v_1 + \alpha'_{4,2} v_2 + \alpha'_{4,3} v_3 + \alpha'_{4,4} v_4 + \alpha'_{4,5} v_5 \right)$

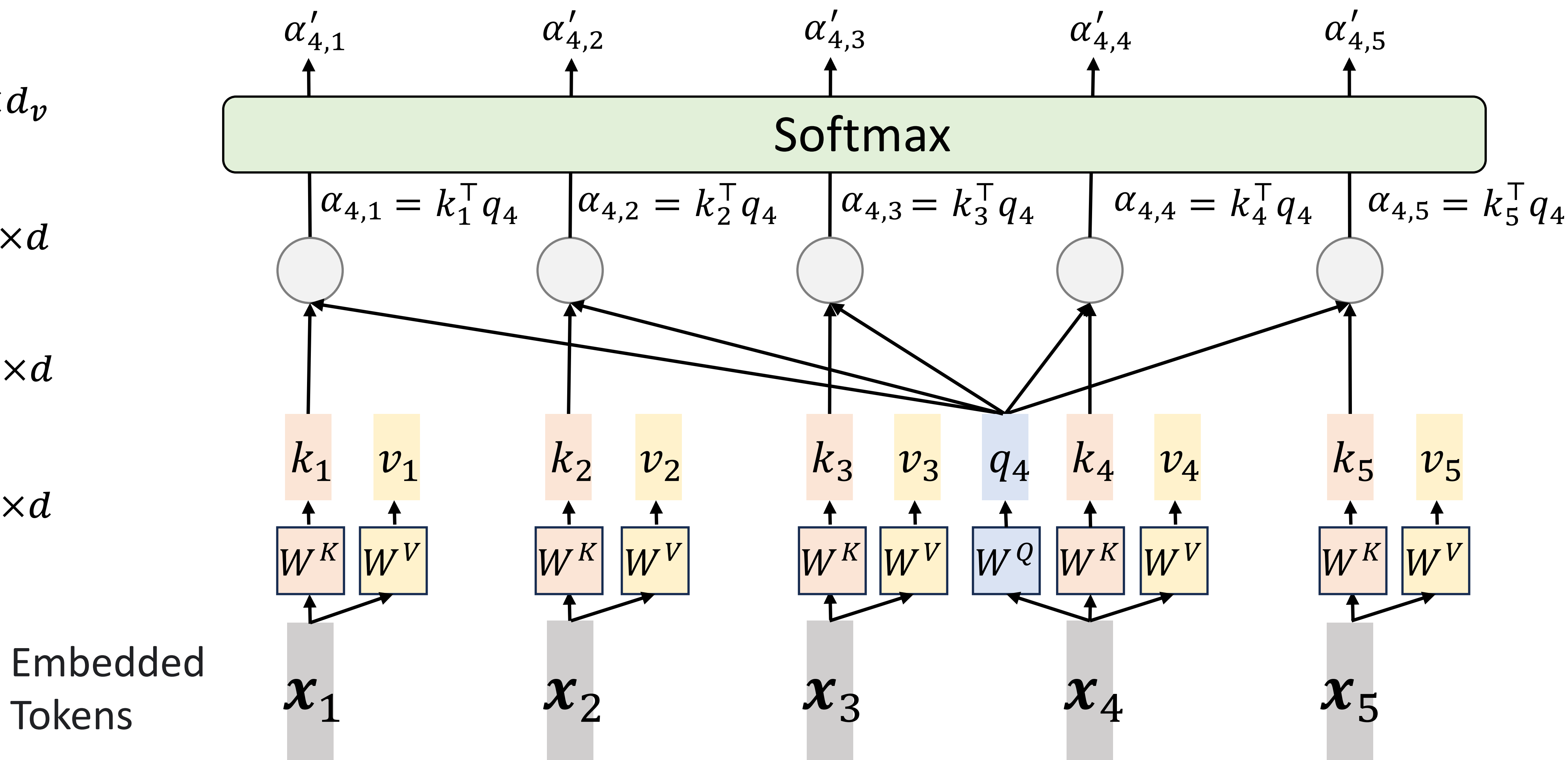
$$= \sum_i \alpha'_{4,1} \left(\boxed{W^O} \boxed{W^V} \right) \boldsymbol{x}_i$$

$W^O \in R^{d \times d_v}$

$W^Q \in R^{d_k \times d}$

$W^K \in R^{d_k \times d}$

$W^V \in R^{d_v \times d}$



Self-Attention

Single-head attention

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

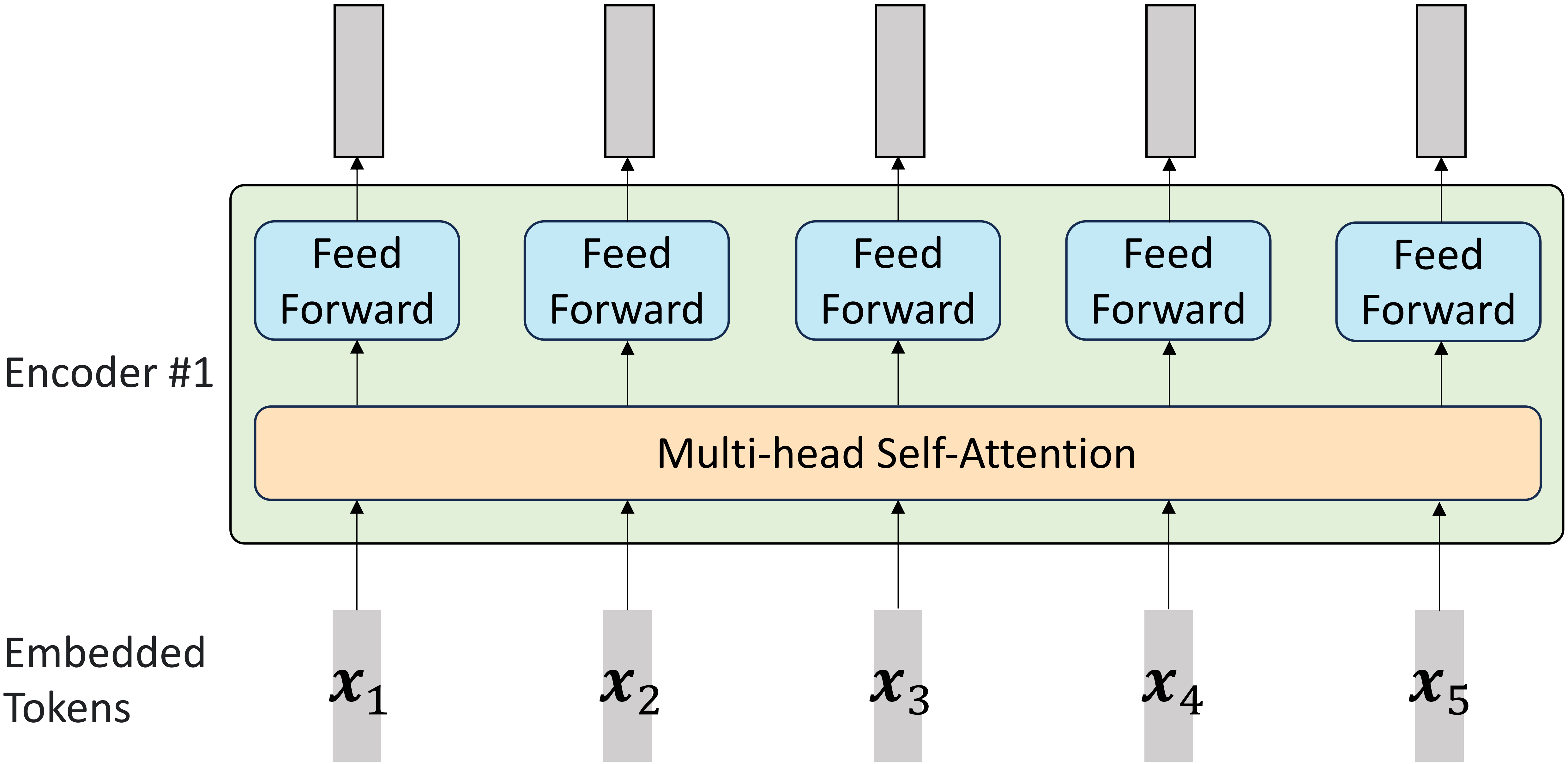
$$Q = W^Q \begin{bmatrix} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \boldsymbol{x}_3 & \boldsymbol{x}_4 & \boldsymbol{x}_5 \end{bmatrix}$$

$$K = W^K \begin{bmatrix} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \boldsymbol{x}_3 & \boldsymbol{x}_4 & \boldsymbol{x}_5 \end{bmatrix}$$

$$V = W^V \begin{bmatrix} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \boldsymbol{x}_3 & \boldsymbol{x}_4 & \boldsymbol{x}_5 \end{bmatrix}$$

Slide by Jia-Bin Huang, University of Maryland College Park

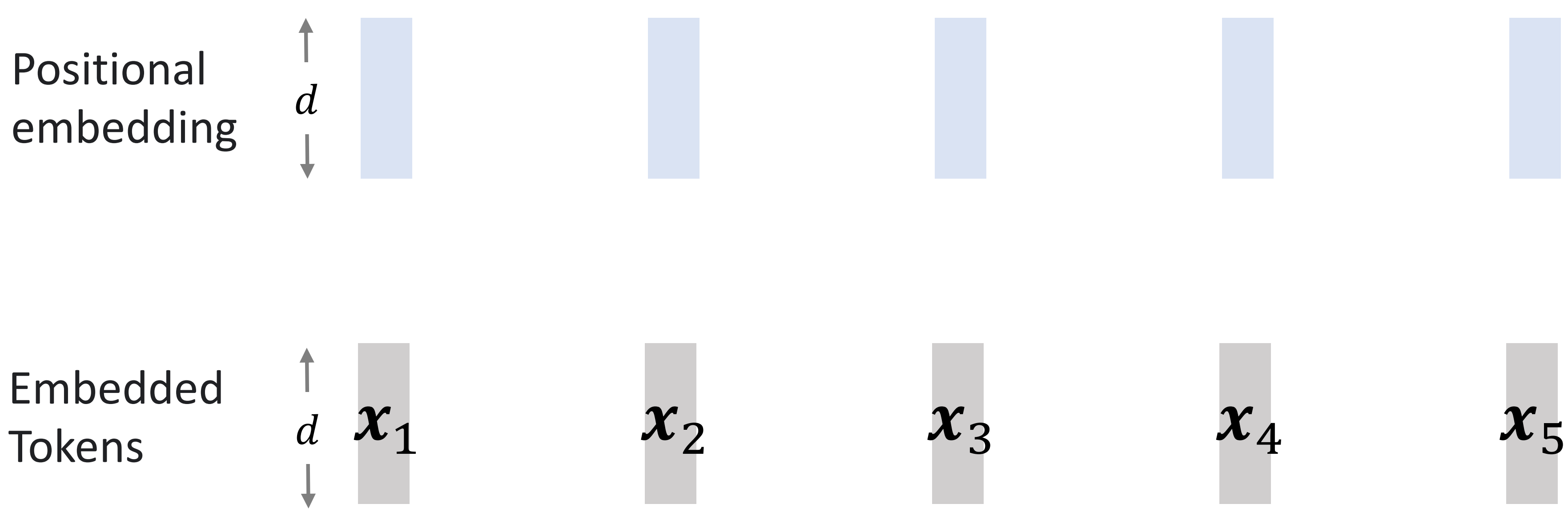
MLPs



Slide by Jia-Bin Huang, University of Maryland College Park

Positional Encoding

	Position k	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
	2^3	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	← Slow oscillating
Dimension	2^2	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	
	2^1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	
	2^0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	← Fast oscillating



Slide by Jia-Bin Huang, University of Maryland College Park

Positional Encoding

$$\mathbf{S}_i = \left[\sin(\omega_1 i), \cos(\omega_1 i), \sin(\omega_2 i), \cos(\omega_2 i), \ldots, \sin(\omega_{e/2} i), \cos(\omega_{e/2} i)\right]$$

Position k

Angular frequency

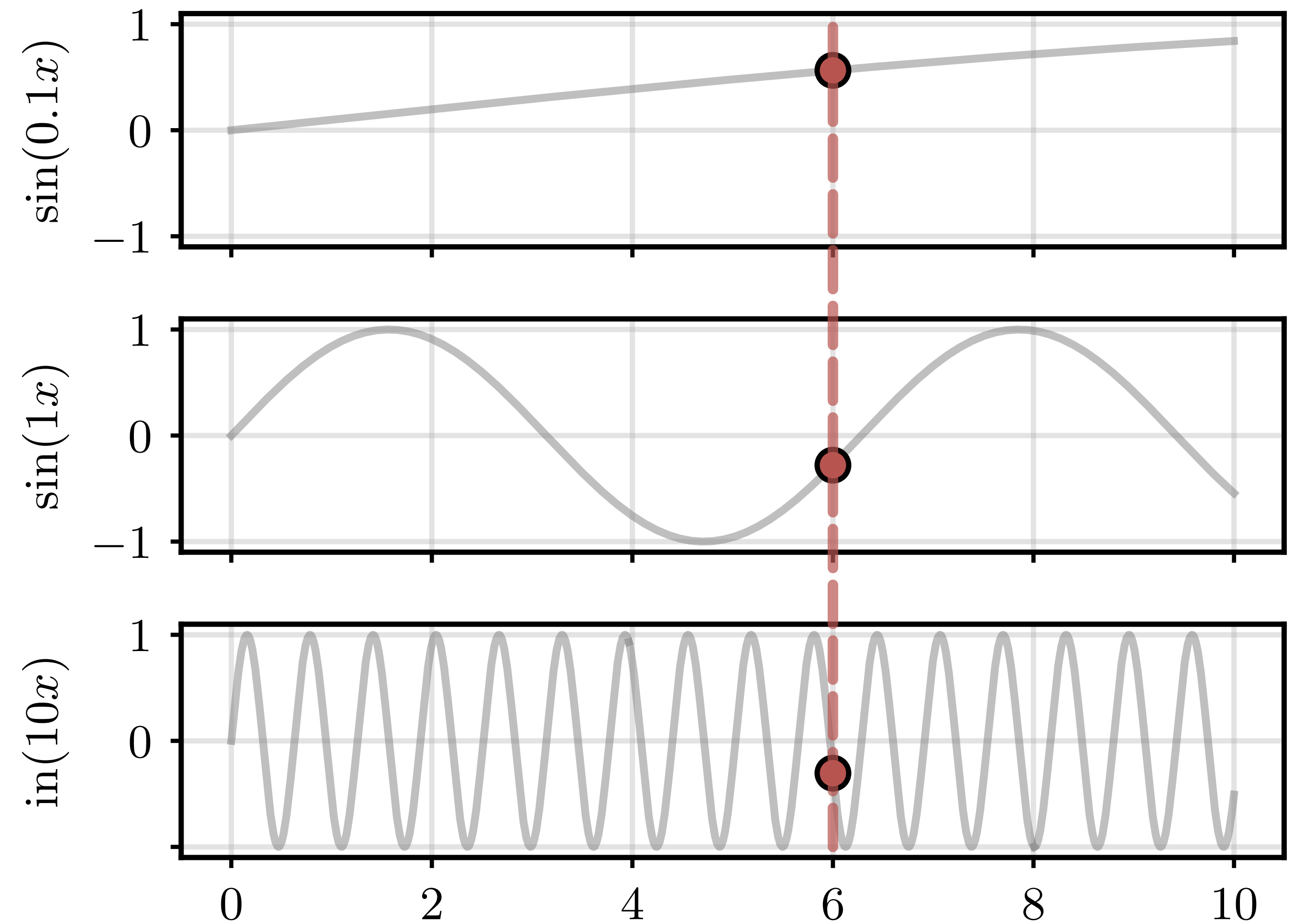
$w_i = N^{-2i/d}$

$N = 100{,}000$

$$d \quad \begin{bmatrix} \sin(w_0 k) \\ \cos(w_0 k) \\ \sin(w_1 k) \\ \cos(w_1 k) \\ \vdots \\ \vdots \\ \sin\left(w_{\frac{d}{2}-1} k\right) \\ \cos\left(w_{\frac{d}{2}-1} k\right) \end{bmatrix}$$

← Fast oscillating

← Slow oscillating



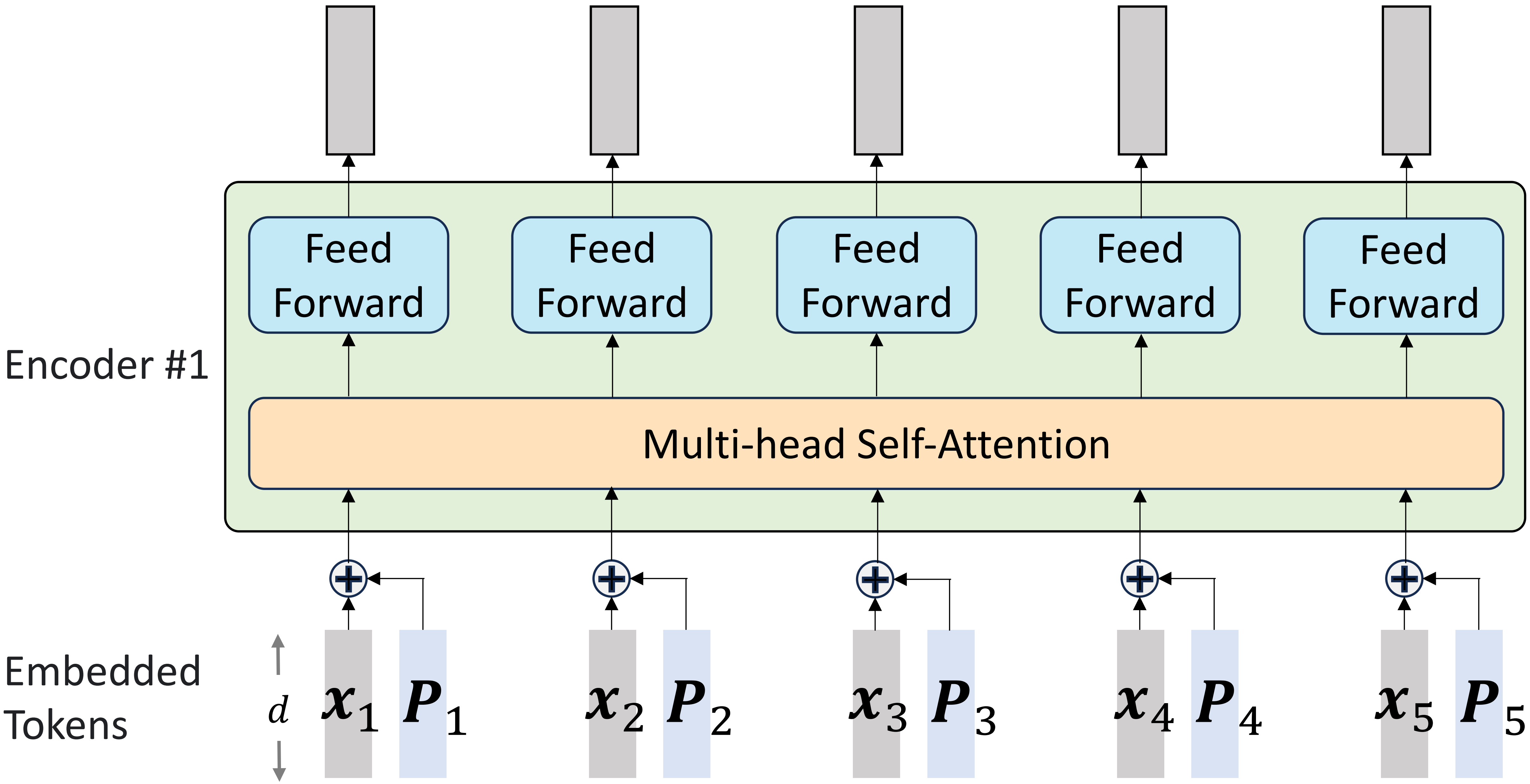
Normalized Range

Unique identifier, unlimited length

Relative positions as linear transform

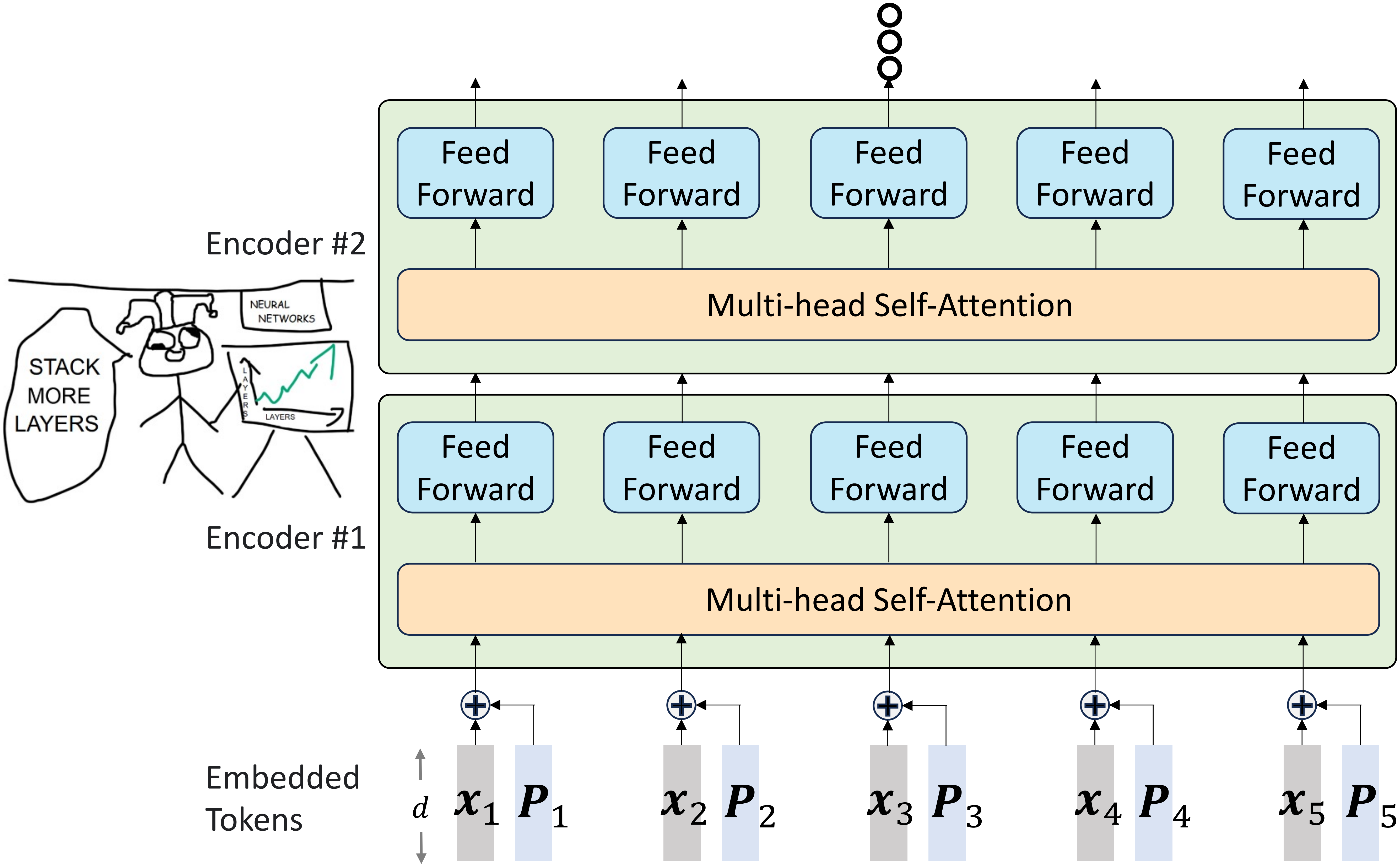
Slide by Jia-Bin Huang, University of Maryland College Park

Positional Encoding



Slide by Jia-Bin Huang, University of Maryland College Park

Transformer Encoder



Slide by Jia-Bin Huang, University of Maryland College Park

Transformer Encoder

Residual connection

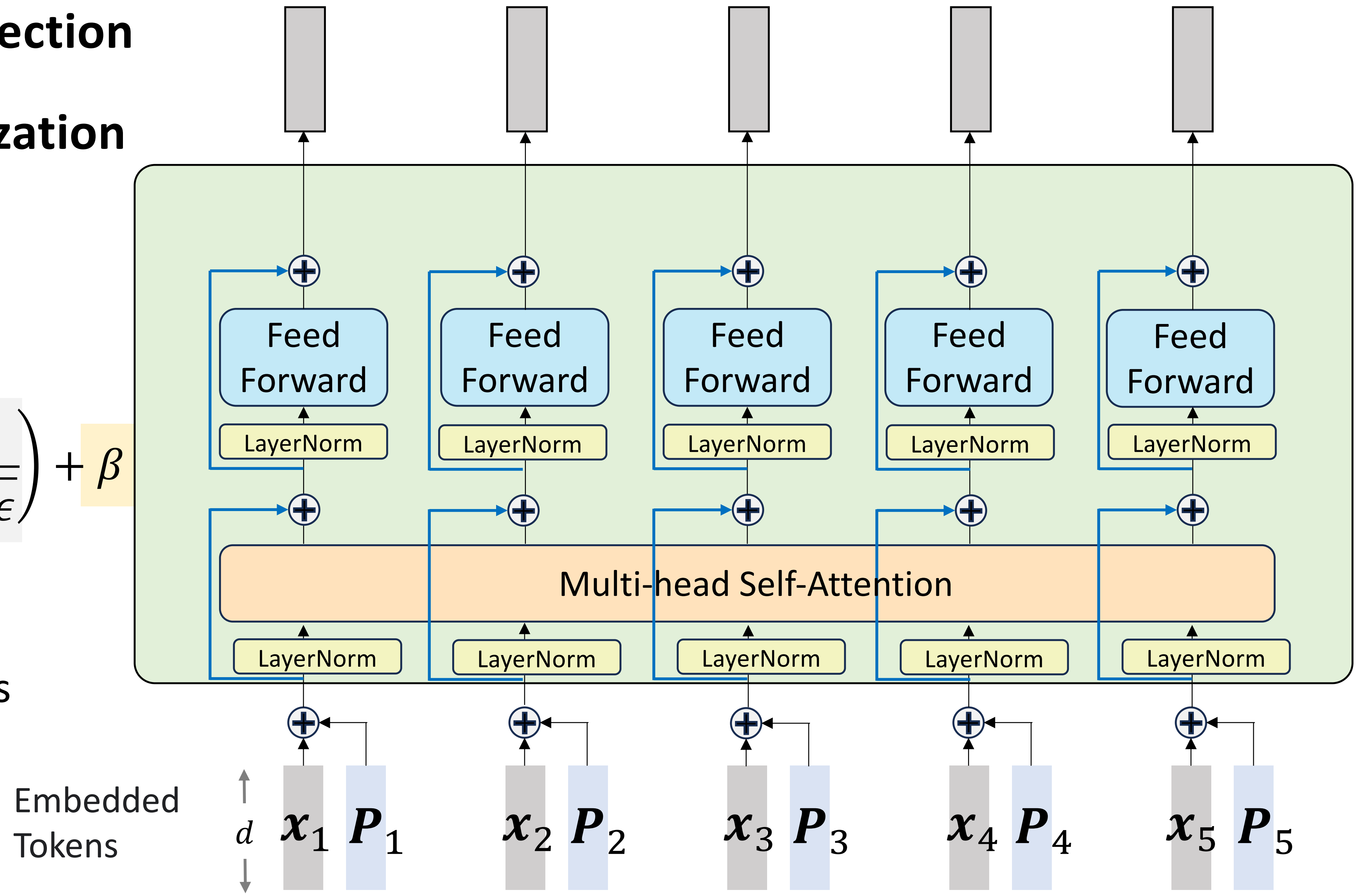
Layer normalization

$$\text{LayerNorm}(\boldsymbol{x}) =$$

$$\gamma\left(\frac{\boldsymbol{x} - \text{mean}(\boldsymbol{x})}{\sqrt{\text{Variance}(\mathbf{x}) + \epsilon}}\right) + \beta$$

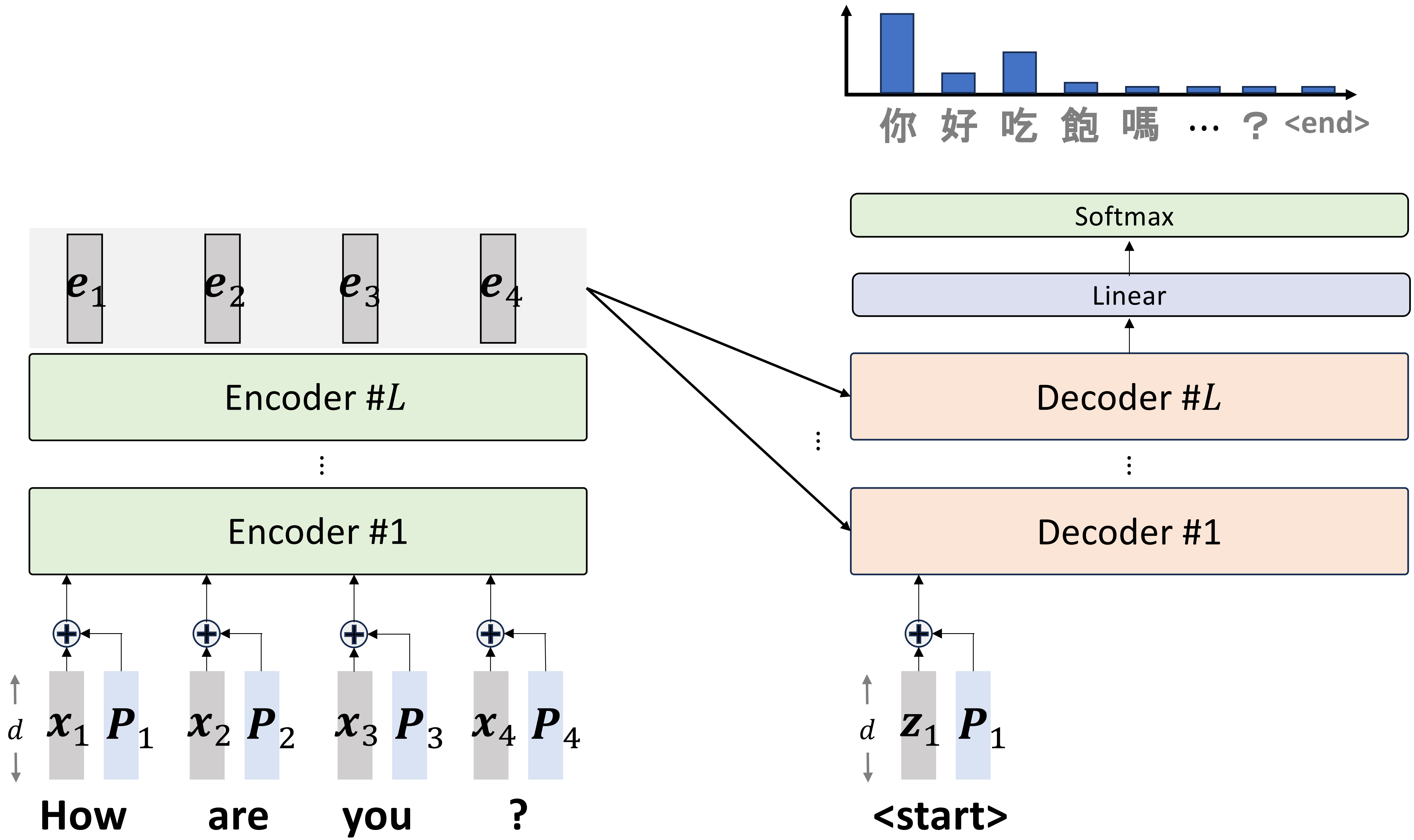
$\gamma, \beta \in R$

Learnable parameters



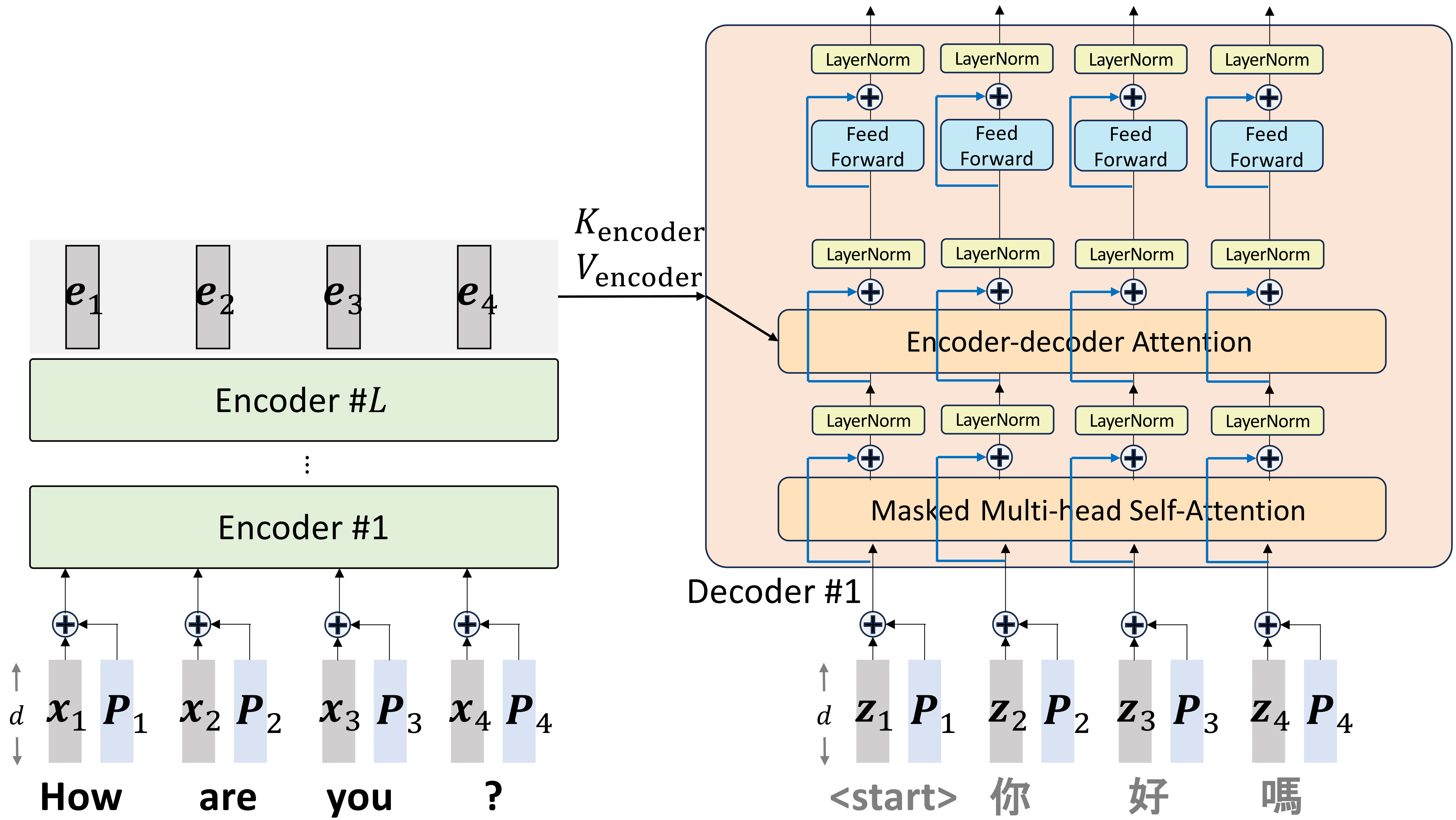
Slide by Jia-Bin Huang, University of Maryland College Park

Decoders



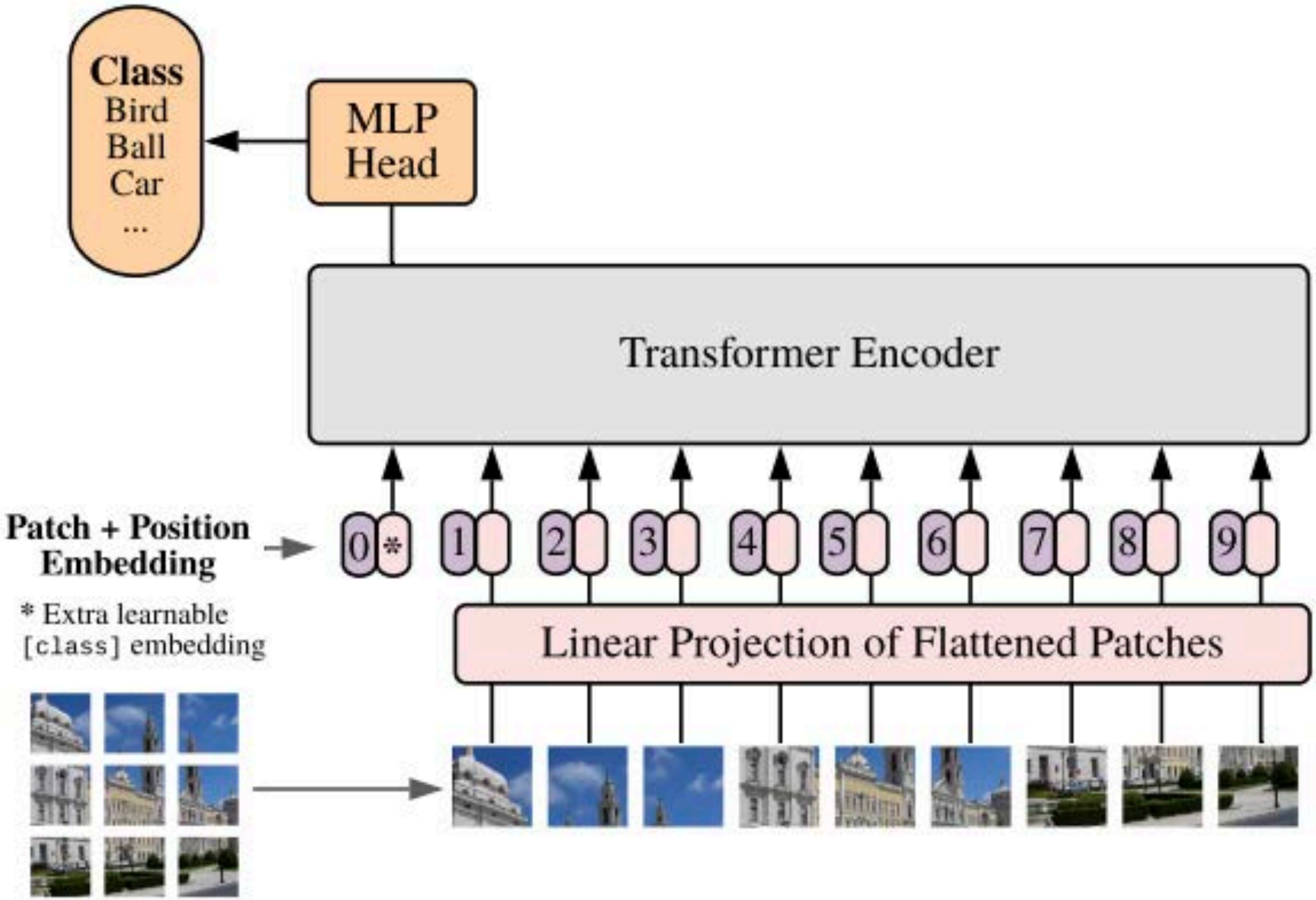
Slide by Jia-Bin Huang, University of Maryland College Park

Decoders



Slide by Jia-Bin Huang, University of Maryland College Park

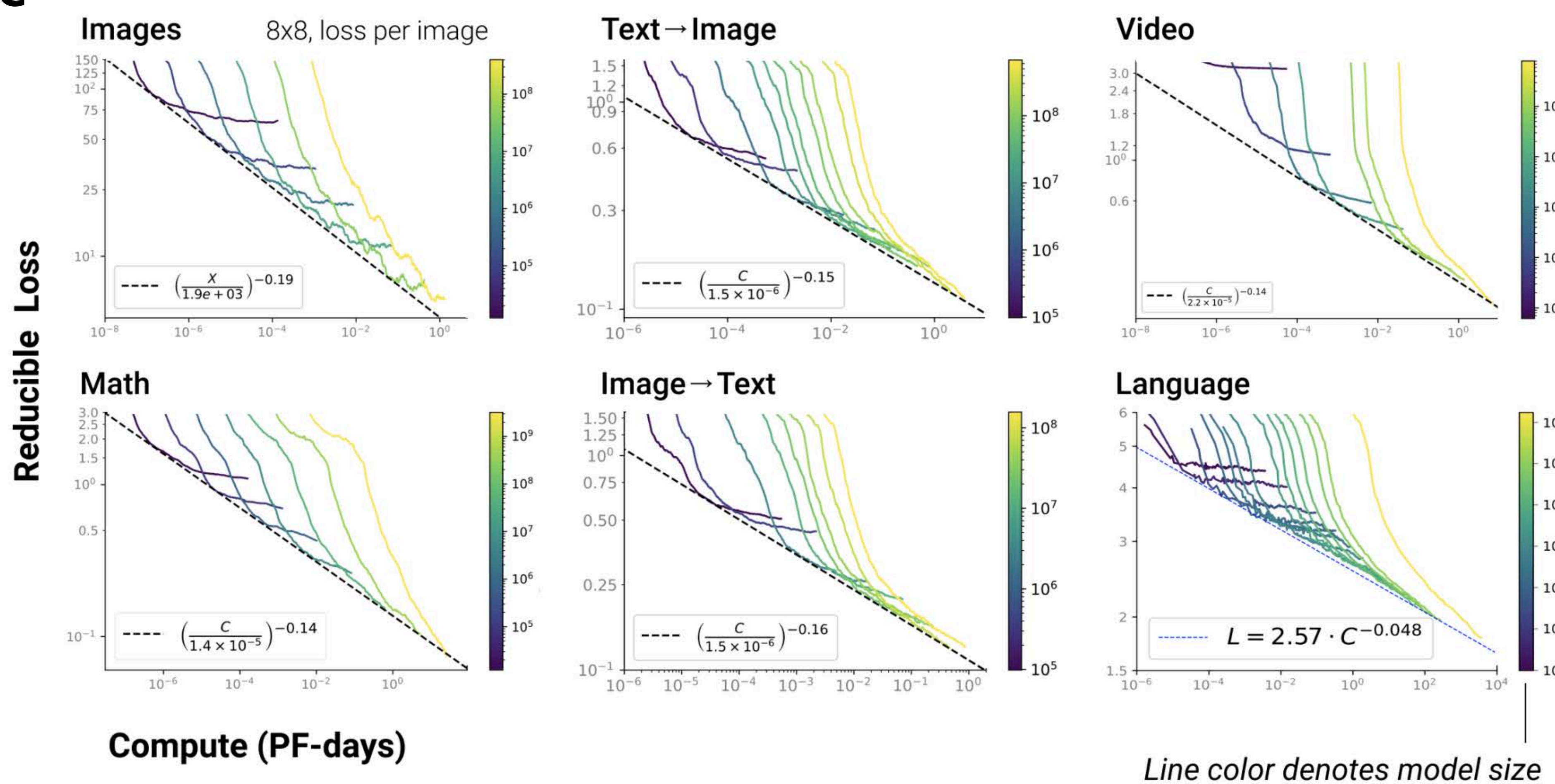
Vision Transformers



Dosovitskiy, A. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

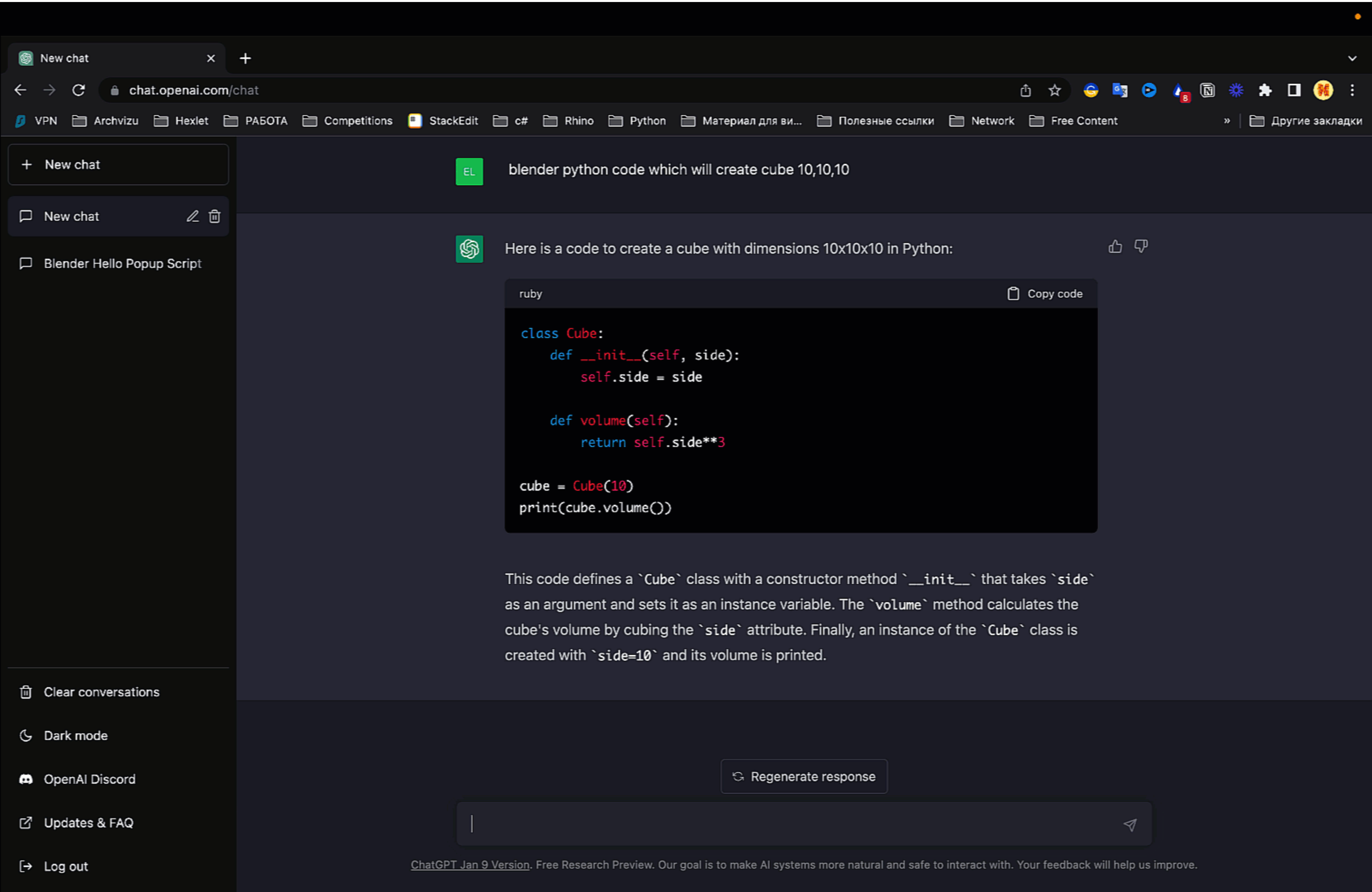
Advantages & Limitations

- Parallel processing
- Better at capturing long-range dependencies
- No vanishing gradient problem
- Scales well with data and compute
- Modality-agnostic

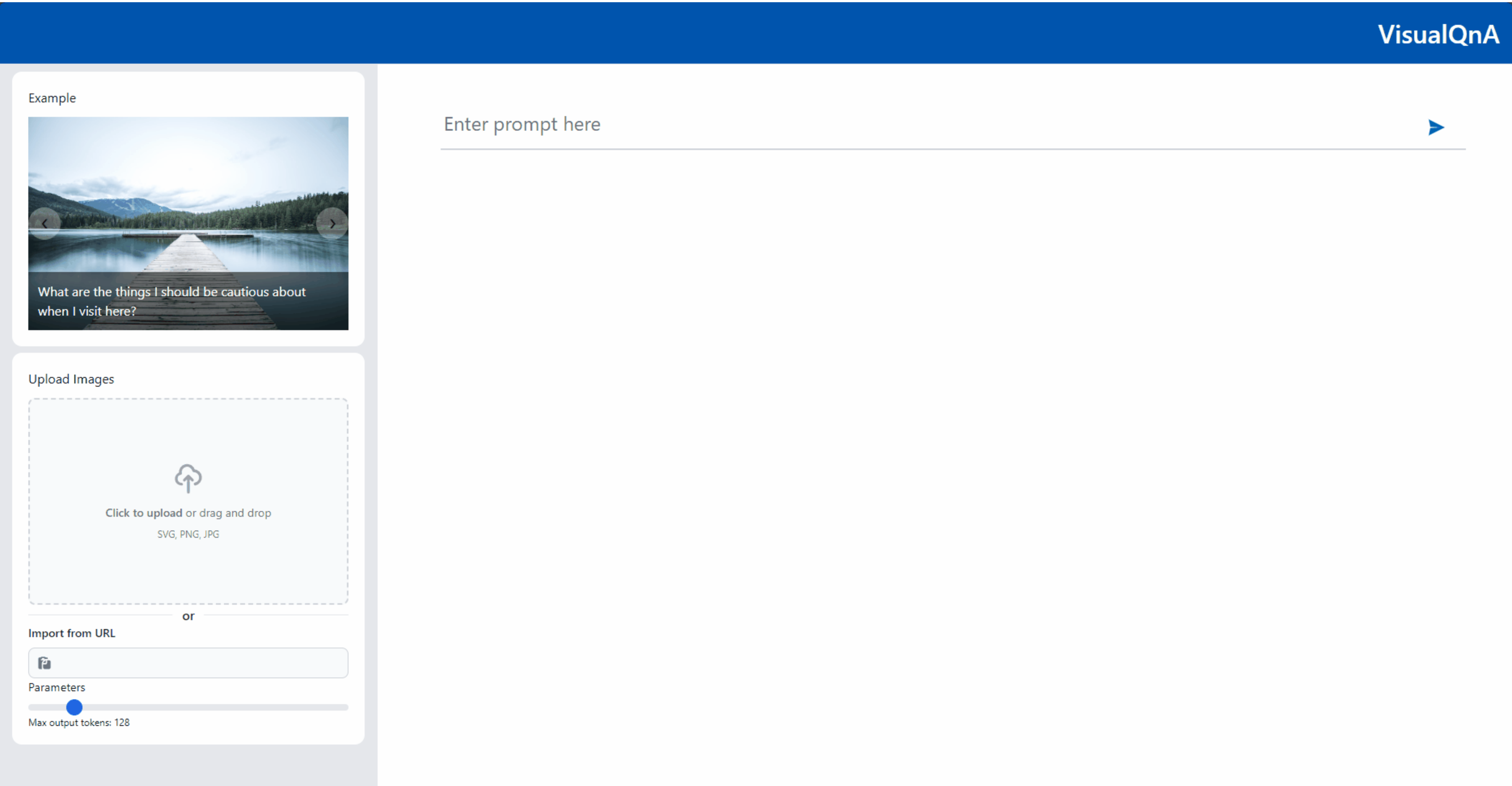


- Self-attention cost and memory scales quadratically with seq. length
- Needs a lot of data
- Expensive and difficult to train

Generative Pretrained Transformers

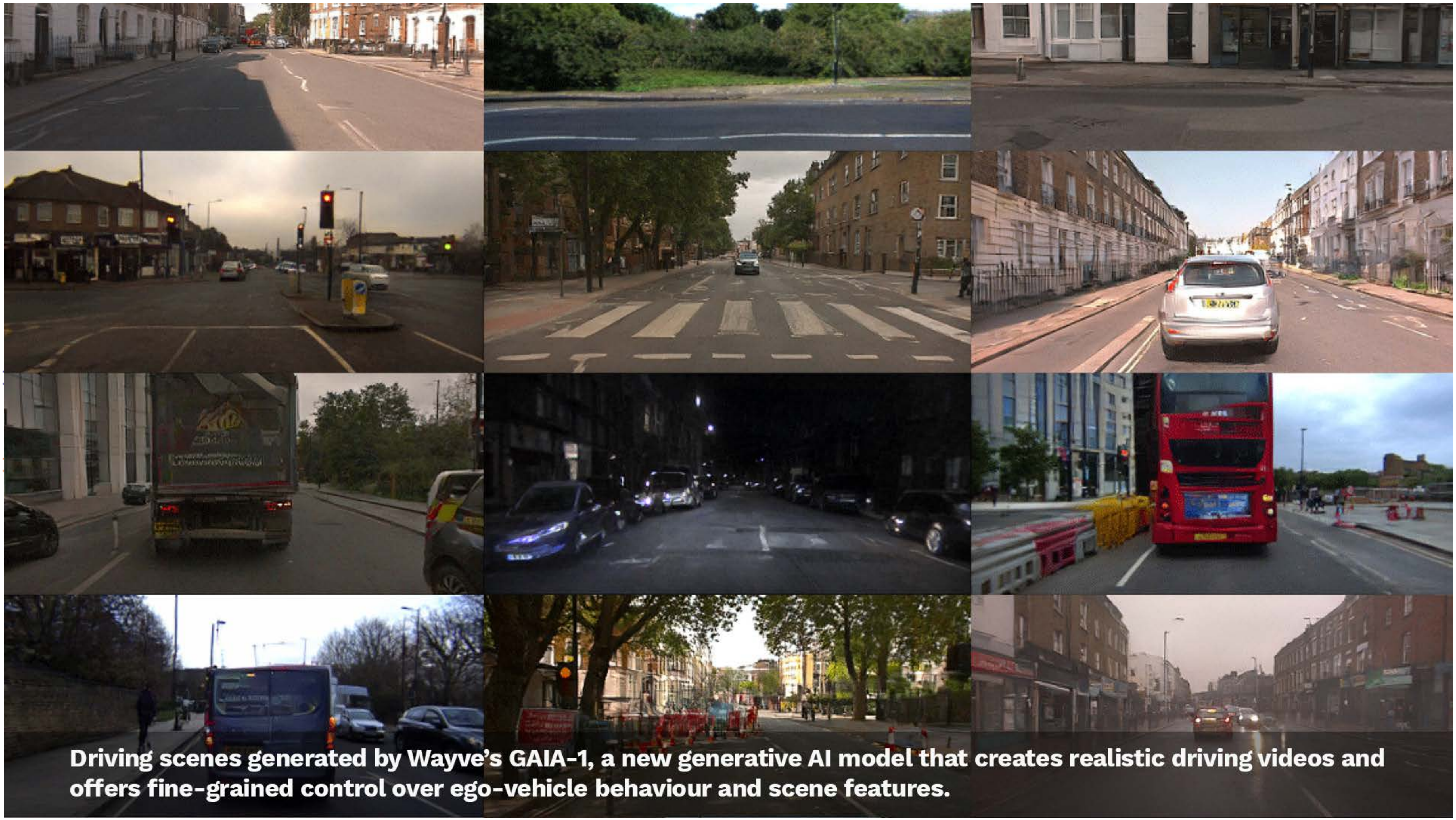


Multi-modal Transformers



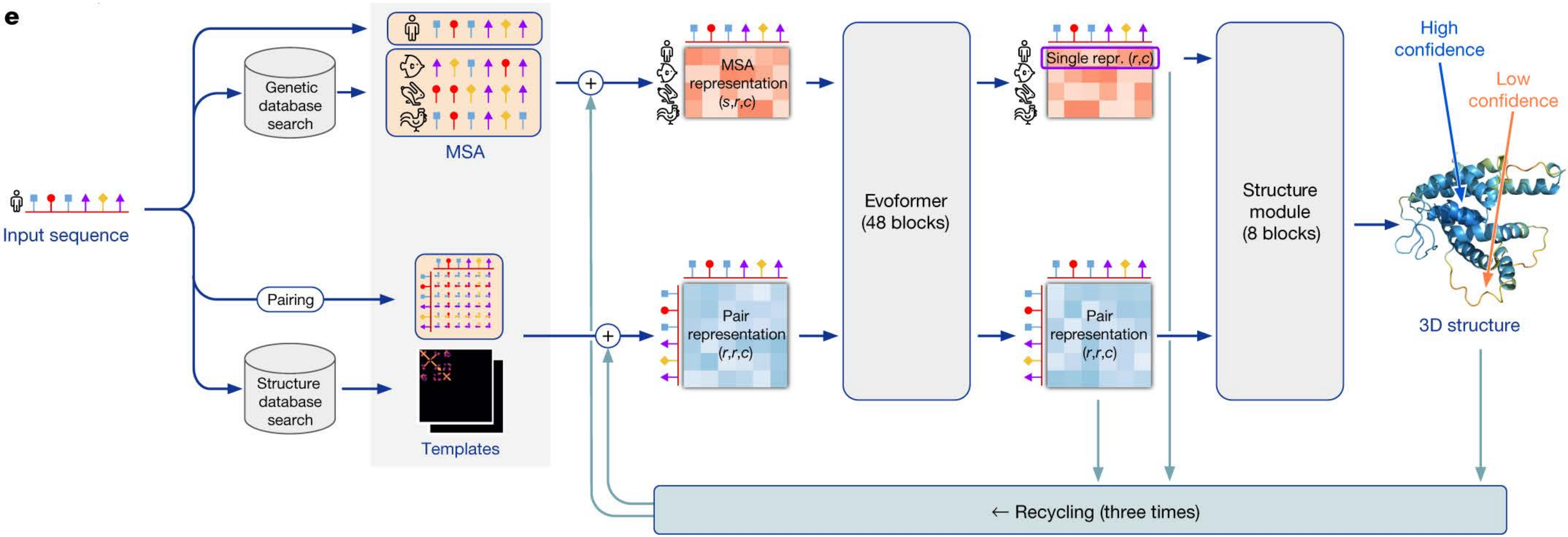
https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/

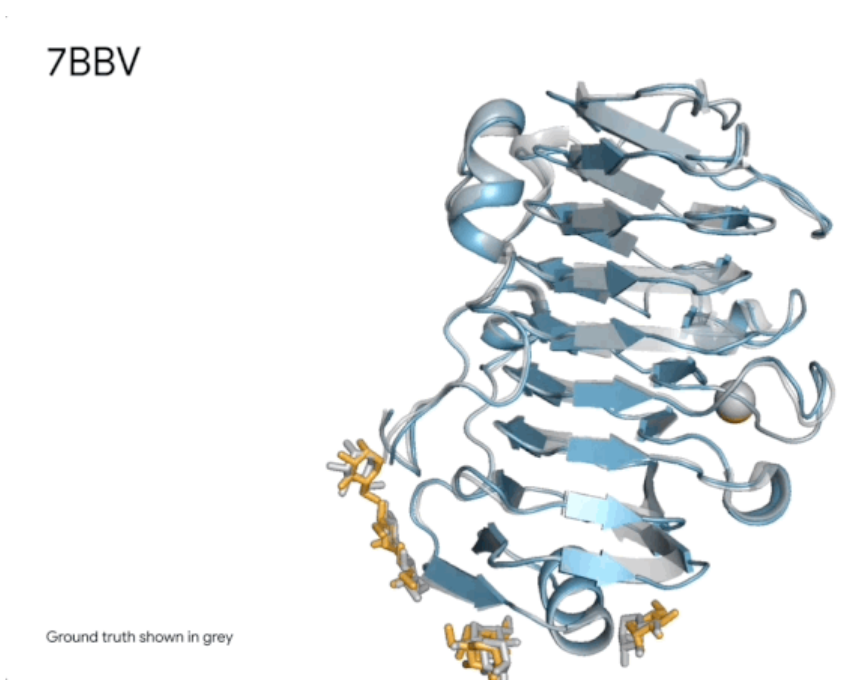
Multi-modal Transformers

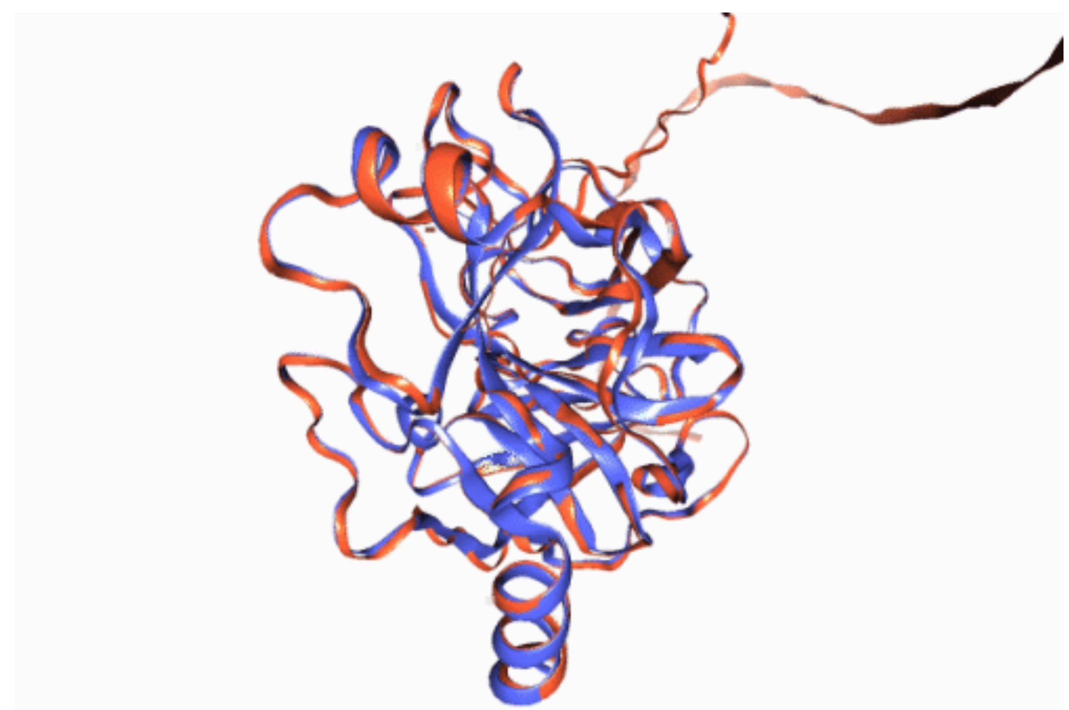


https://wayve.ai/thinking/introducing-gaia1/

AlphaFold 3







Abramson, J., Adler, J., Dunger, J., Evans, R., Green, T., Pritzel, A., ... & Jumper, J. M. (2024). Accurate structure prediction of biomolecular interactions with AlphaFold 3. Nature, 1-3.

UvA Tutorial Notebooks

Build your own GPT from scratch