

# Introduction to transformers

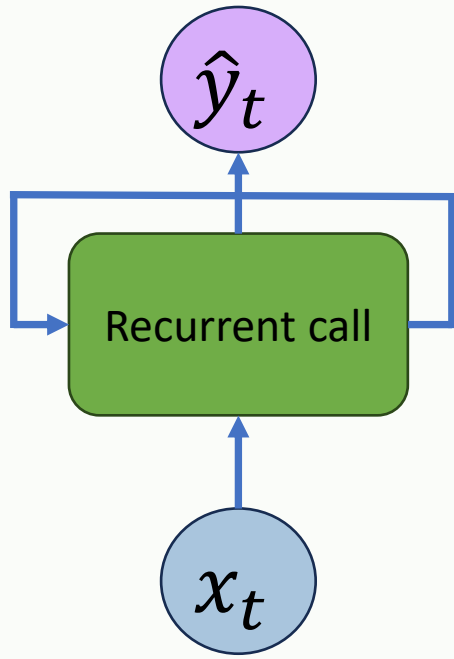
Peyman Tahghighi

Winter 2024

# Learning objectives

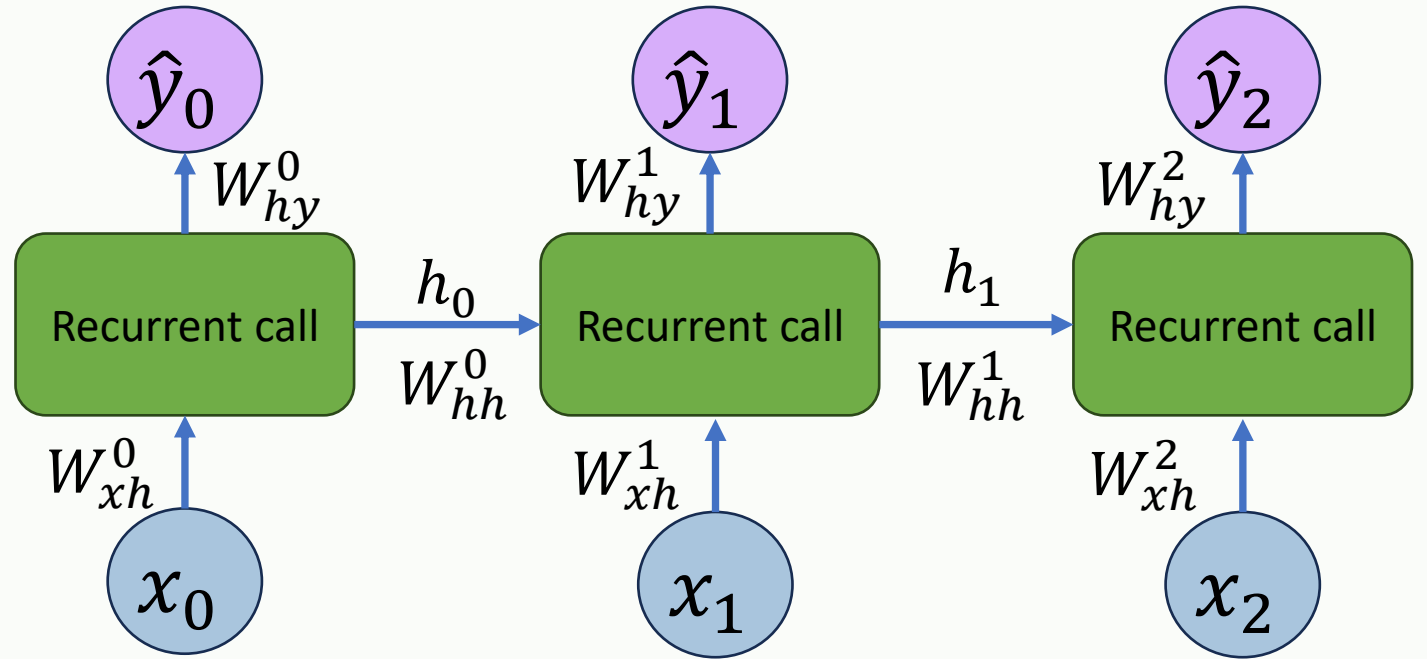
- Why do we need an attention mechanism?
- How does attention work?
- What are encoders and decoders in transformers?
- Vision transformers.

# Recurrent Neural Networks



Output vector

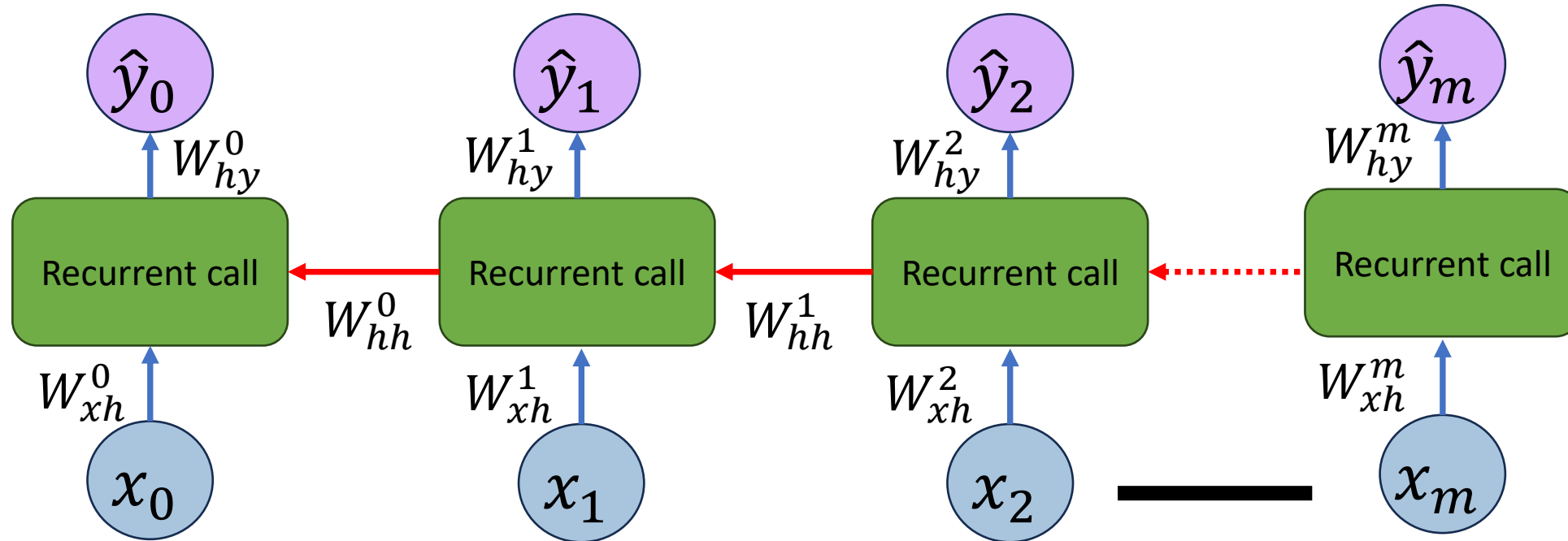
$$\hat{y}_t = W_{hy}^T h_t$$



Hidden state update

$$h_t = \tanh(W_{hh}^T h_{t-1} + W_{xh}^T x_t)$$

# Recurrent Neural Networks

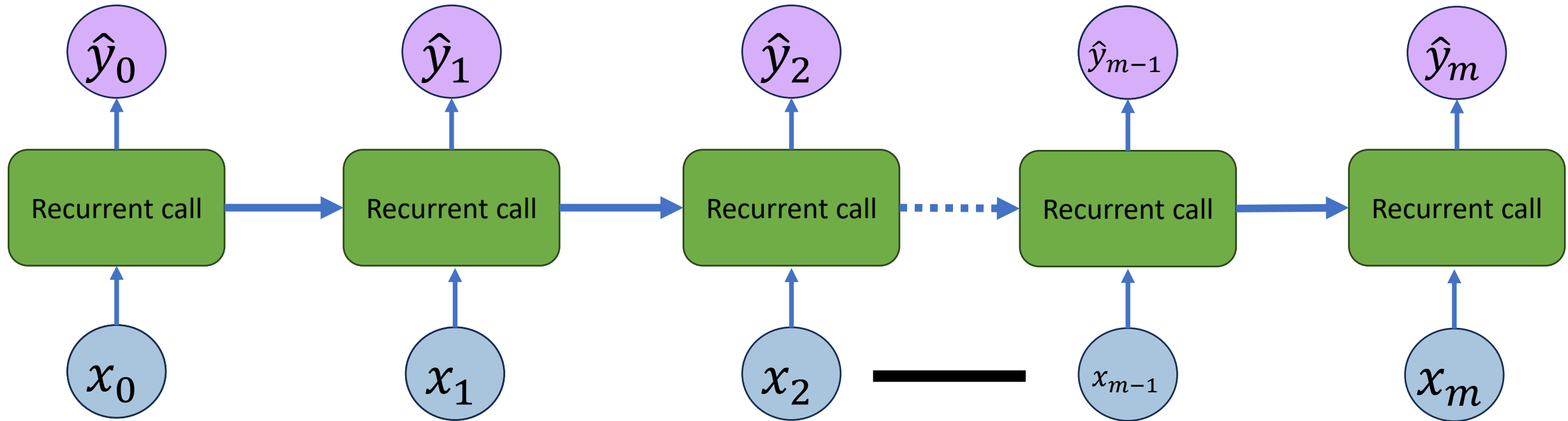


Many values  $> 1$ :  
Exploding gradients

Many values  $< 1$ :  
Vanishing gradients

# Recurrent Neural Networks

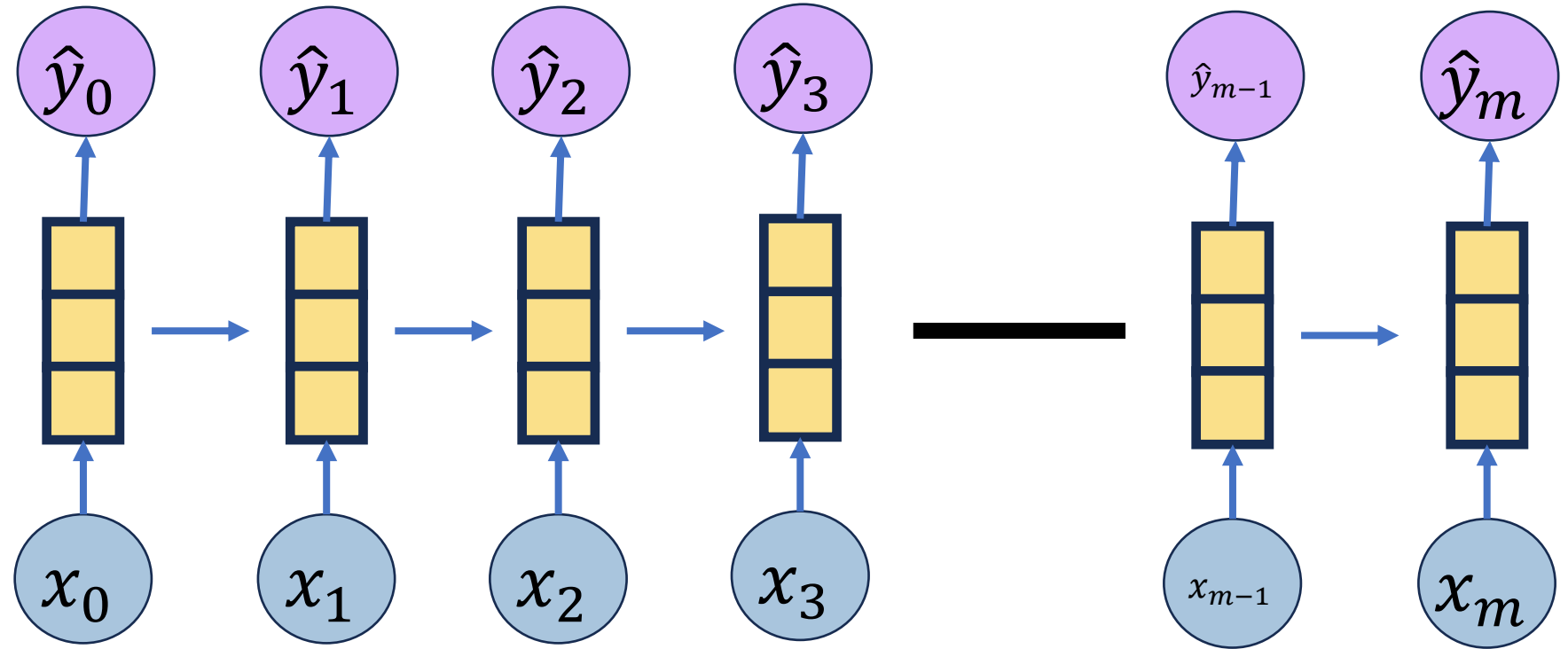
I grew up in France.... And I speak fluent....



# Goal of Sequence Modeling

Limitations of RNN:

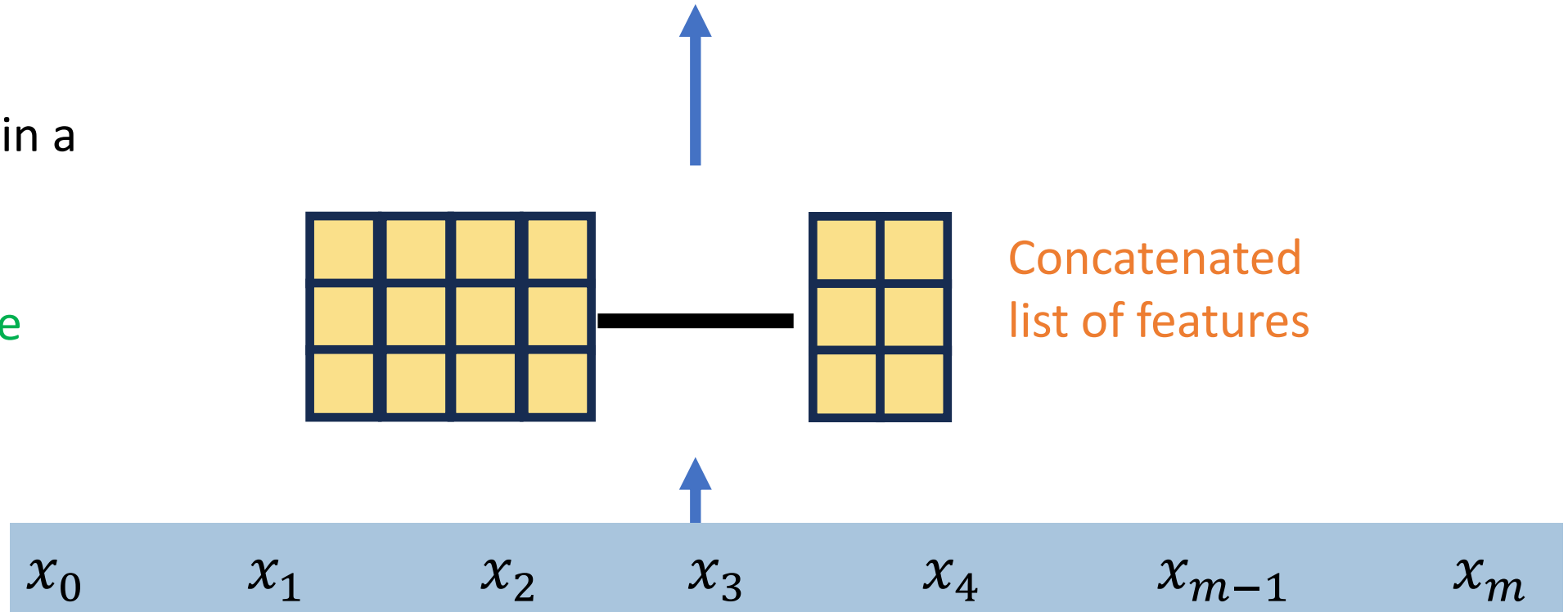
- Weak Long- Term Memory.
- Vanishing/Exploding Gradients.
- Not easy to parallelize.



# Goal of Sequence Modeling

Feed Everything in a  
dense network:

- No Recurrence
- Not scalable
- No order



# Attention, the core of transformers



1. Identify which parts to attend to
2. Extract the features with higher attention



# What is attention?

YouTube search results for "deep learning". The search bar contains "deep learning" and the search button is labeled "Query (Q)".

Three video results are shown, each with a thumbnail and a title:

- GIANT SEA TURTLES • AMAZING CORAL REEF FISH • 12 HOURS of THE BEST RELAX MUSIC** (Key  $K_1$ )
- MIT 6.S191 (2020): Introduction to Deep Learning** (Key  $K_2$ )
- The Kobe Bryant Fadeaway Shot** (Key  $K_3$ )

The video titled "MIT 6.S191 (2020): Introduction to Deep Learning" is highlighted with a purple box, indicating it is the Value (V).

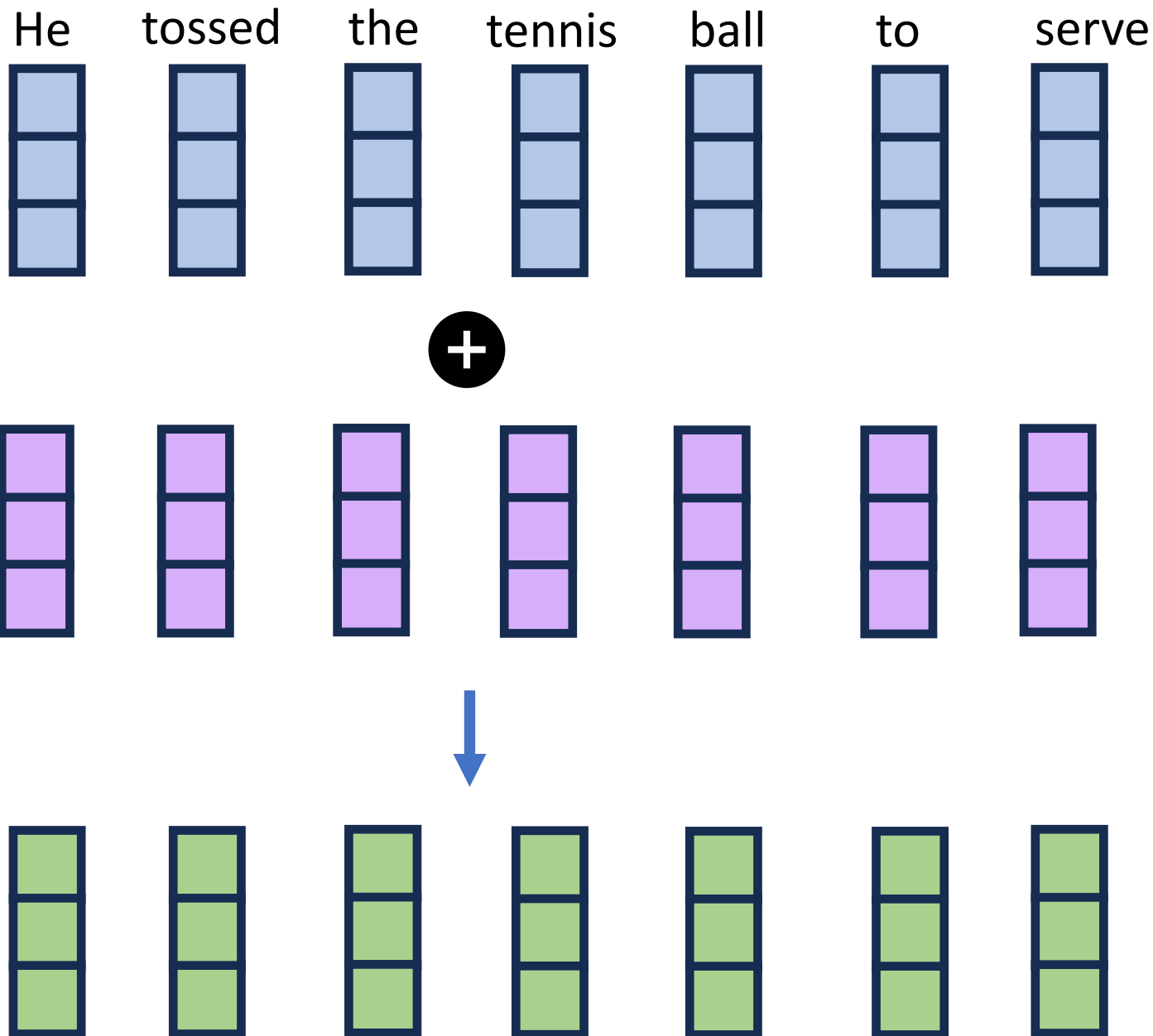
1. **Attention mask:** How similar each key is to the query?

2. **Extract values based on attention:** Return a combination of values according to the mask.

# Self-attention

1- Encode positional information

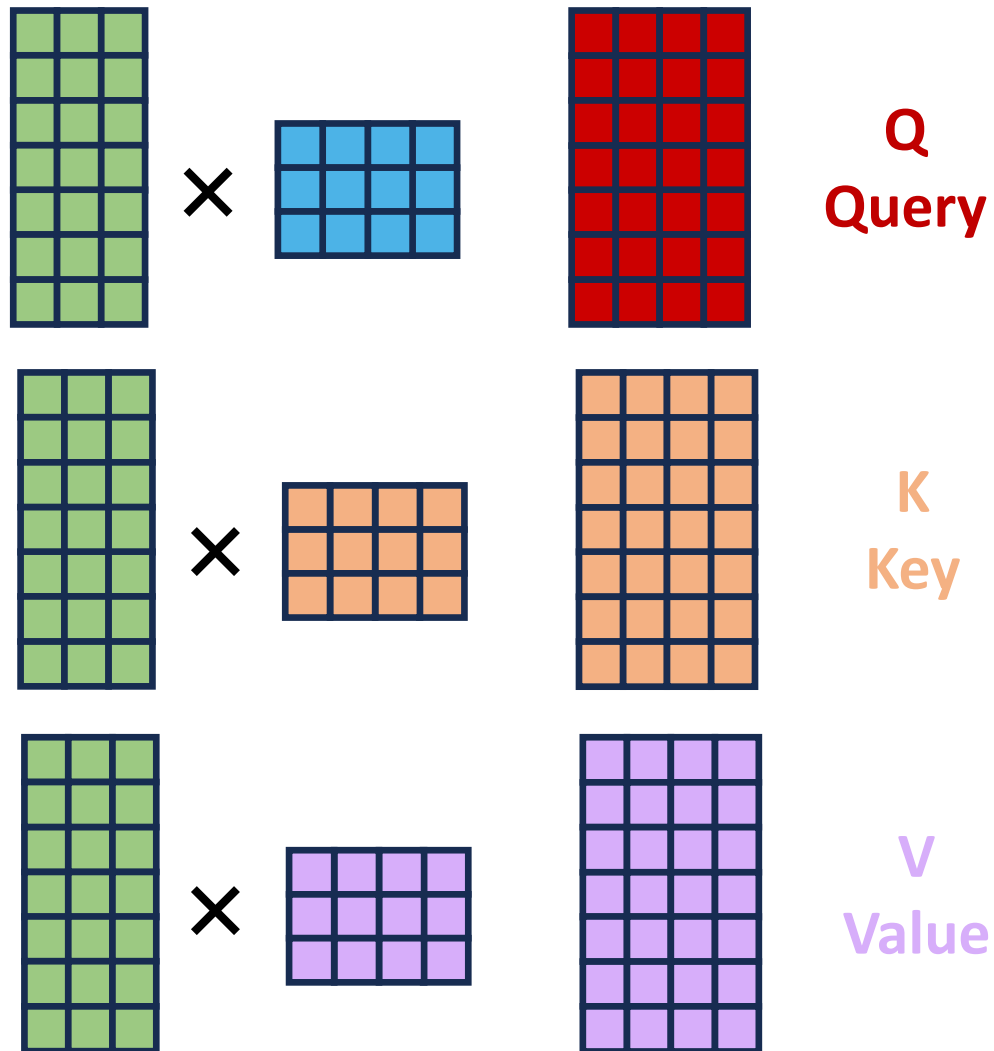
Positional encoding



# Self-attention

1- Encode positional information

2- Extract **Key**, **Query** and **Value**



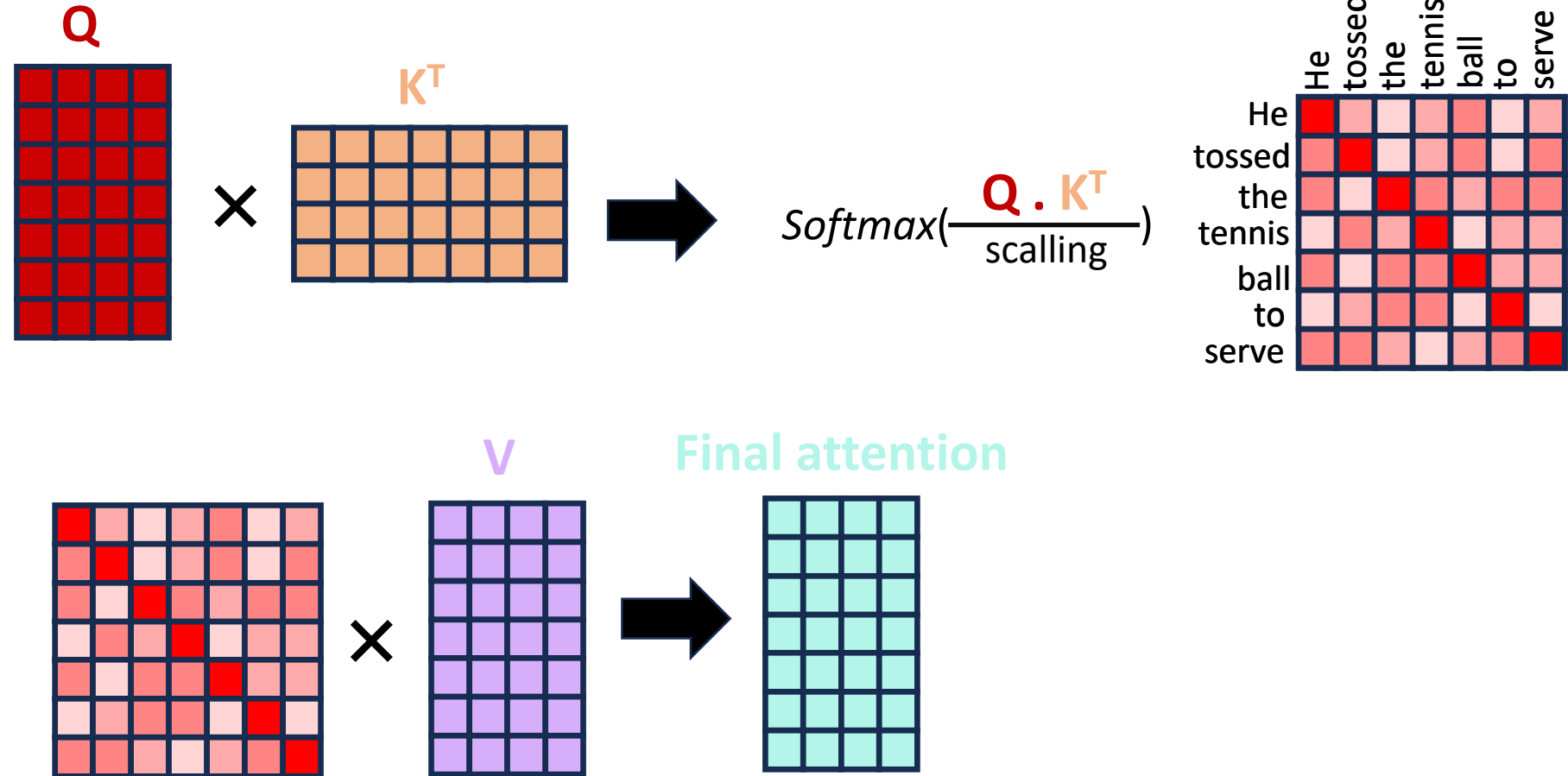
# Self-attention

1- Encode positional information

2- Extract **Key**, **Query** and **Value**

3- Compute attention weights

4- Extract features with high attention



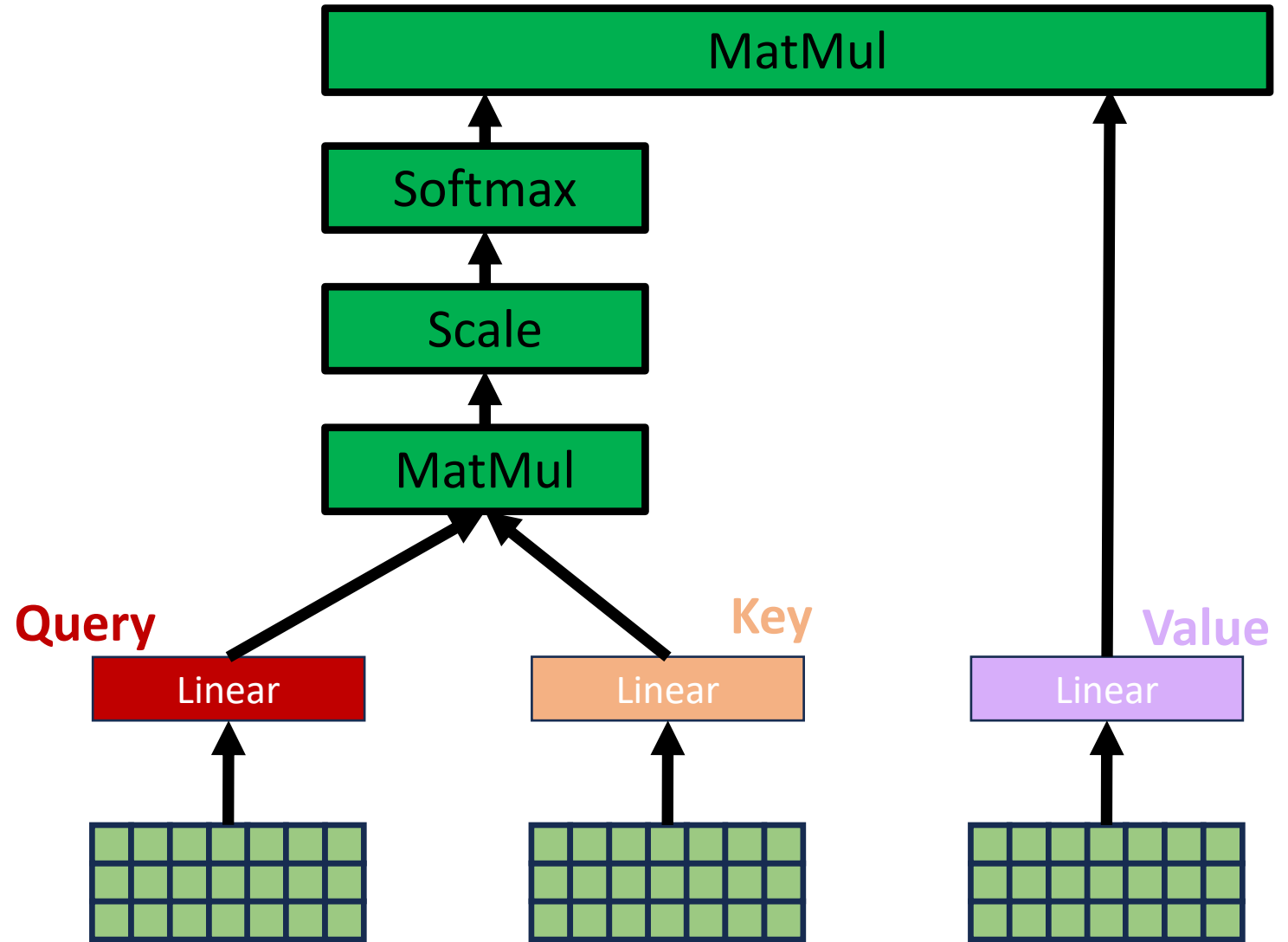
# Recap

1- Encode positional information

2- Extract **Key**, **Query** and **Value**

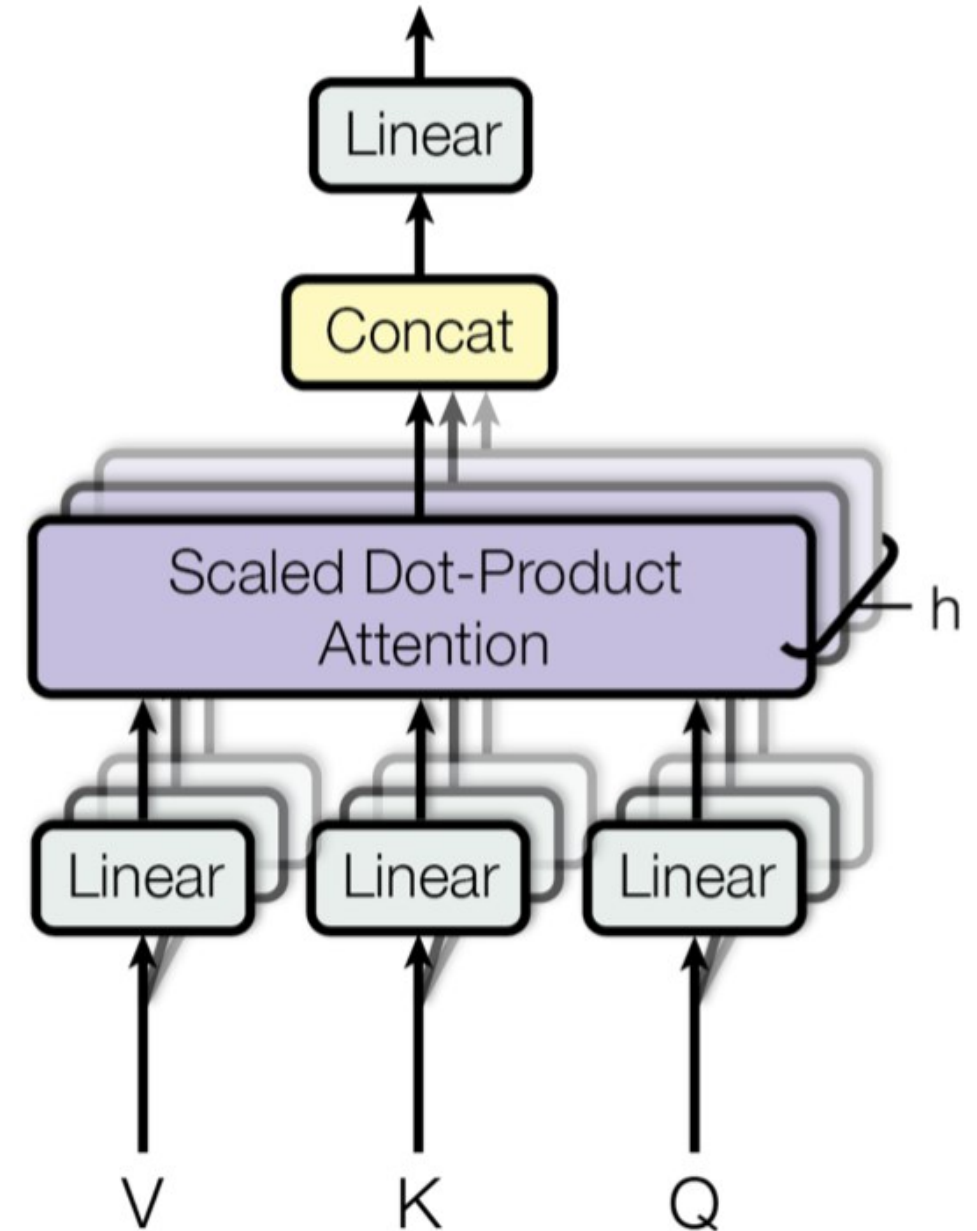
3- Compute attention weights

4- Extract features with high attention



# Multiheaded attention

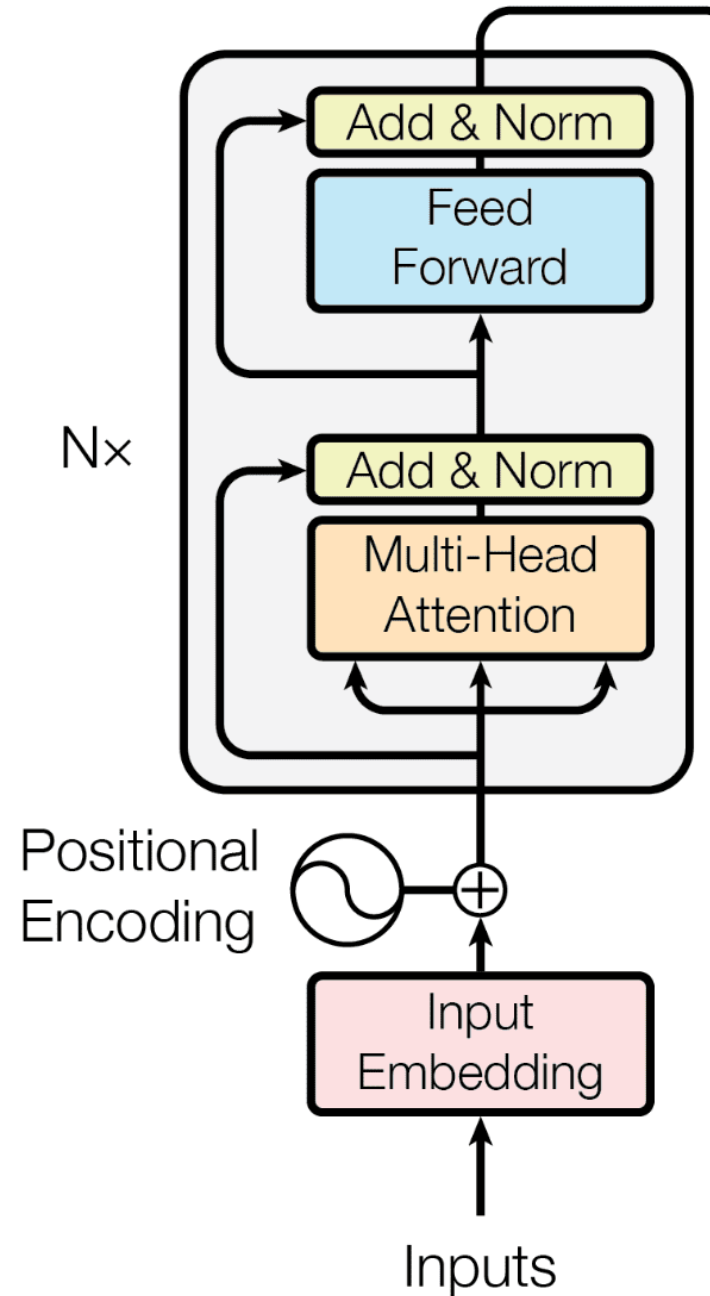
- Multiple parallel attention computation
- Helps with extracting diverse features



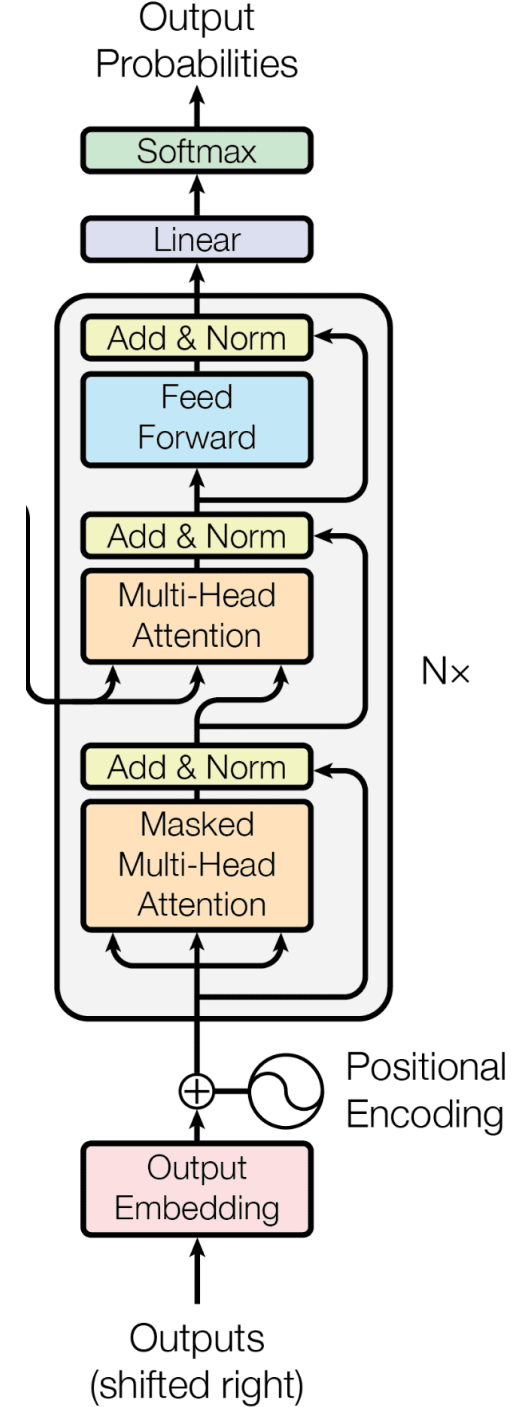
# Encoder bigger picture

$x + \text{atten}(x)$  is important for:

- Gradient flow
- Preserving information



# Decoder bigger picture



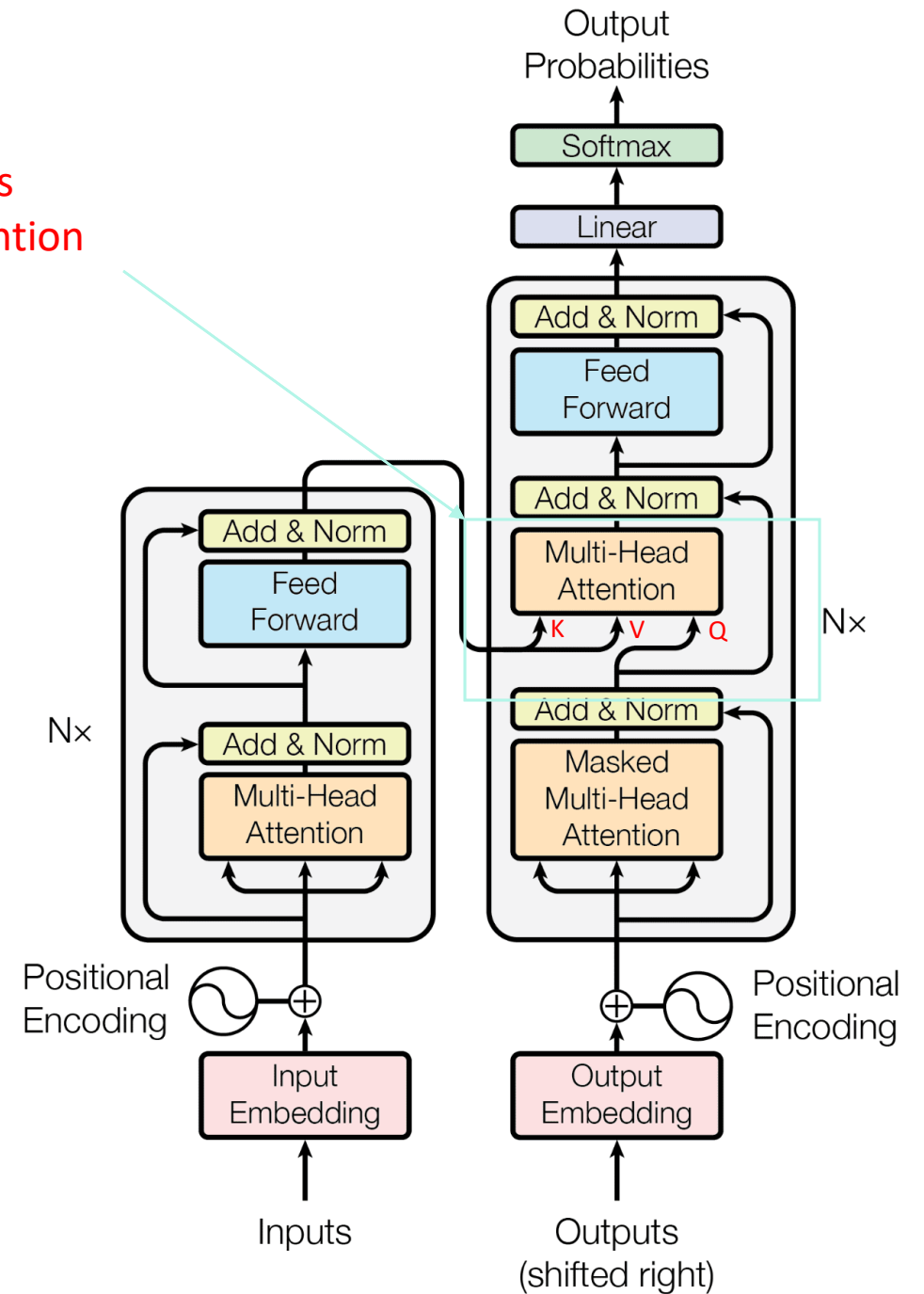


# Transformers

Q -> Input

K, V -> Conditional  
information

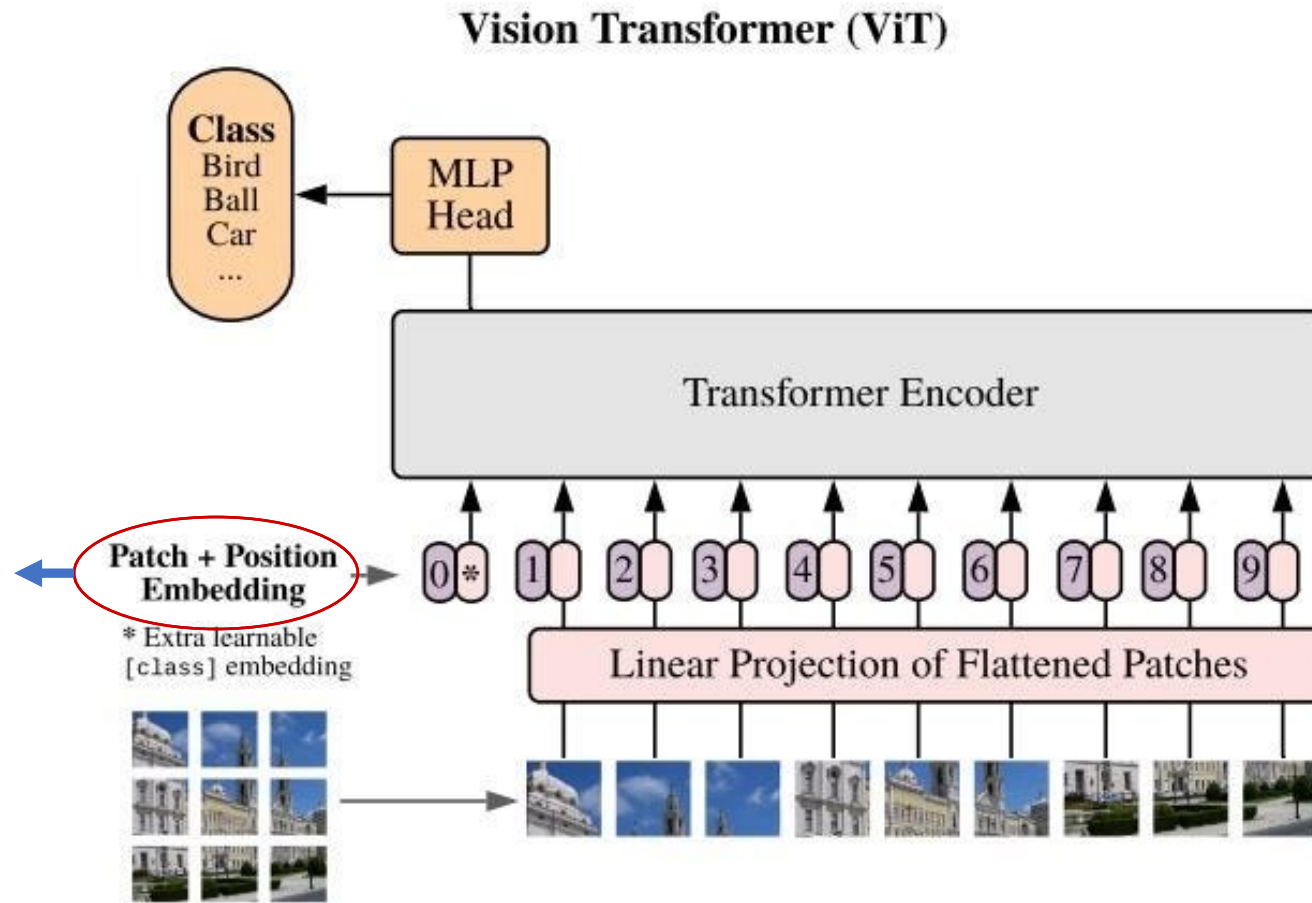
Cross  
attention



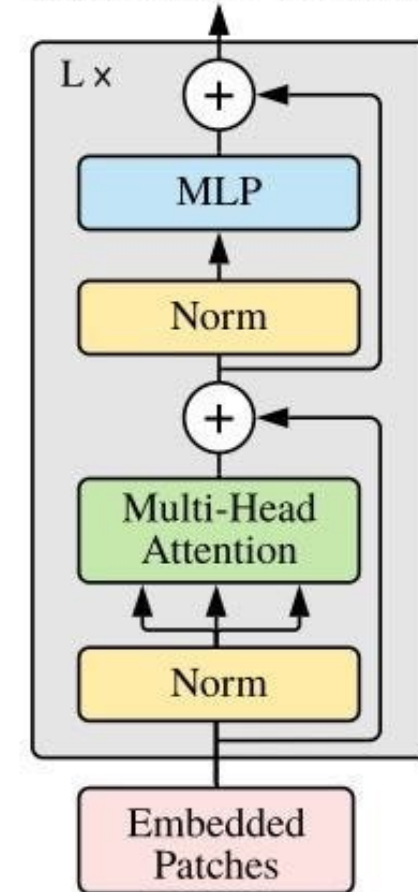
# Transformers Decoder



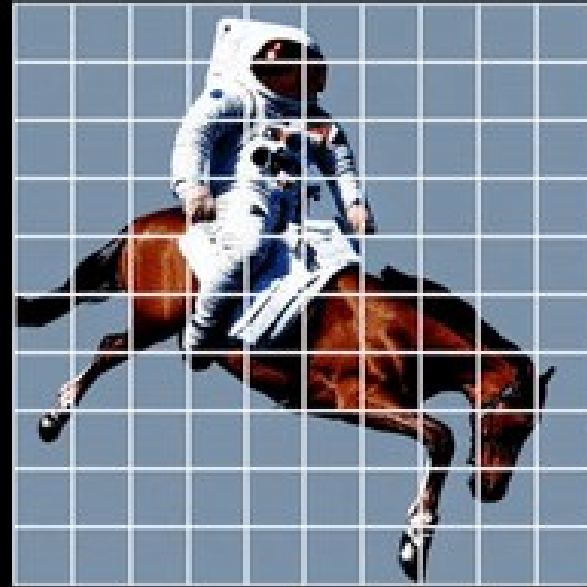
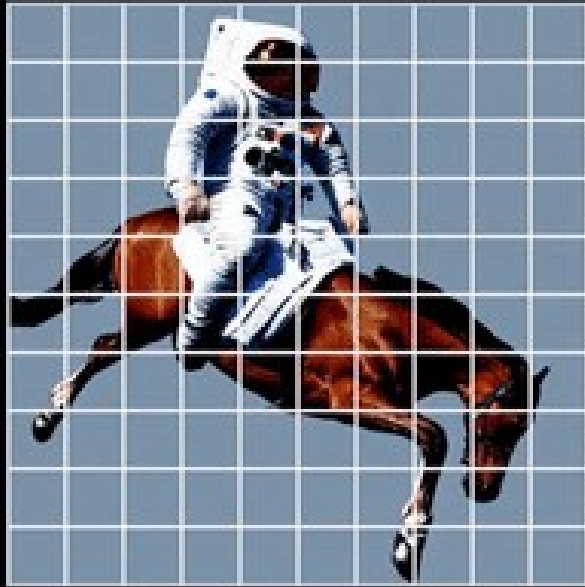
# Vision transformers



## Transformer Encoder



# Vision transformers



# Summary

- How attention mechanism solve challenges of RNNs.
- What are key, query and value mean in attention and how to calculate attention masks.
- Encoder and decoder.
- Vision transformers.

# Useful resources

- [https://www.youtube.com/watch?v=ySEx\\_Bqxxvvo&ab\\_channel=AlexanderAmini](https://www.youtube.com/watch?v=ySEx_Bqxxvvo&ab_channel=AlexanderAmini)
- <https://www.youtube.com/watch?v=kCc8FmEb1nY&pp=ygUIa2FycGF0aHk%3D>
- [https://www.youtube.com/watch?v=OyFJWRnt\\_AY&t=3634s&pp=ygUOcGFzY2FsIHVvdXBhcnQ%3D](https://www.youtube.com/watch?v=OyFJWRnt_AY&t=3634s&pp=ygUOcGFzY2FsIHVvdXBhcnQ%3D)