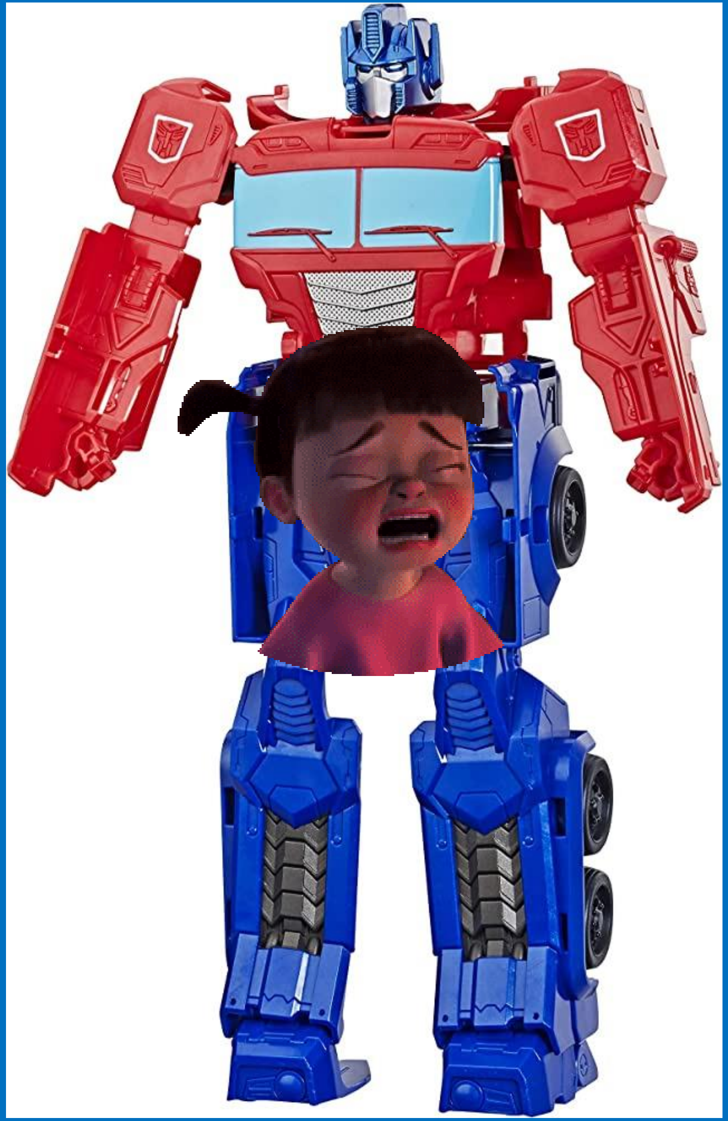
Computer Vision

by Piotr Krzemiński

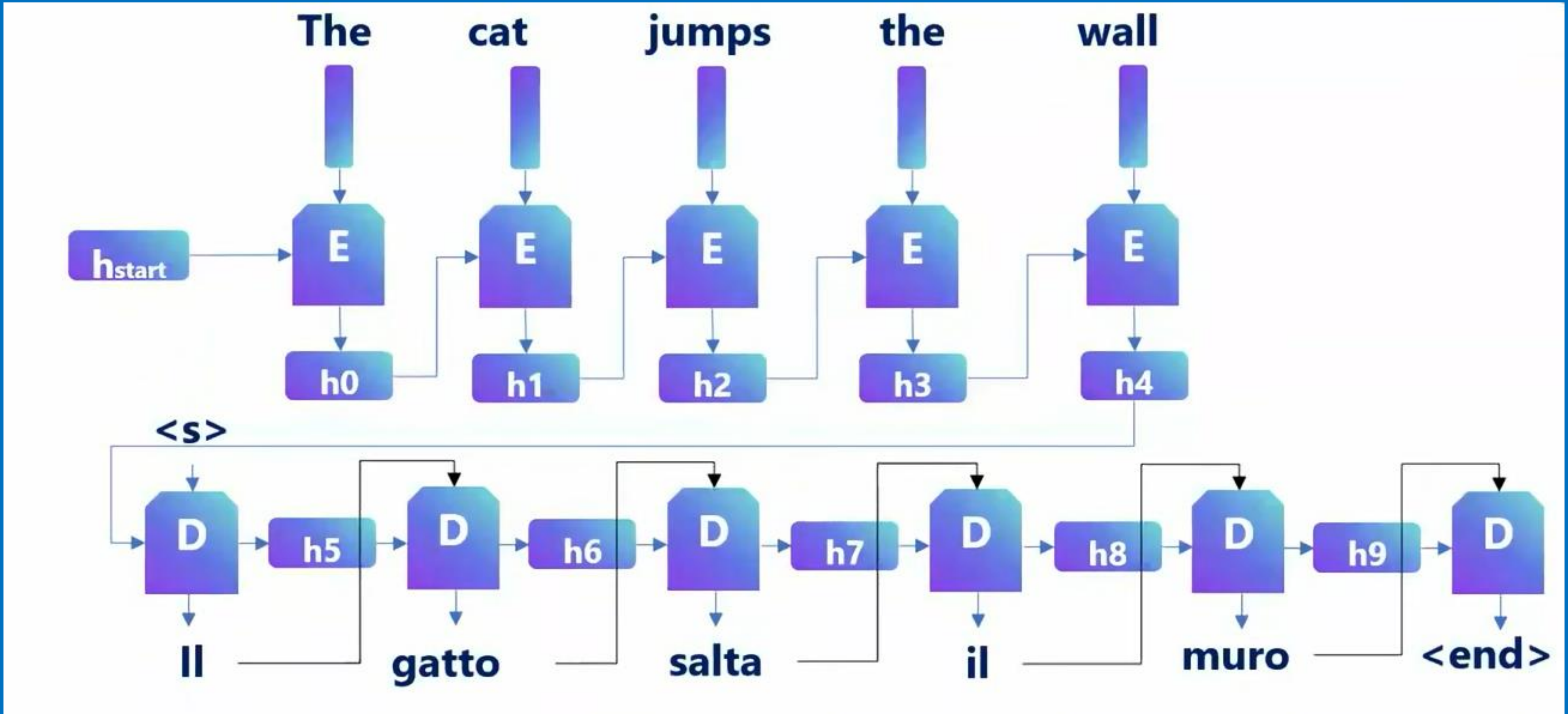
TRANSFORMERS
THE LAST KNIGHT
MEGATRON
PRIME 1 STUDIO
HASBRO and it's logo, TRANSFORMERS and all related characters are trademarks of Hasbro and are used with permission.
© 2017 Hasbro. All Rights Reserved.
© 2017 Paramount Pictures Corporation. All Rights Reserved. Licensed by Hasbro.
AUTHENTIC TRANSFORMERS
Licensed by Hasbro

VS

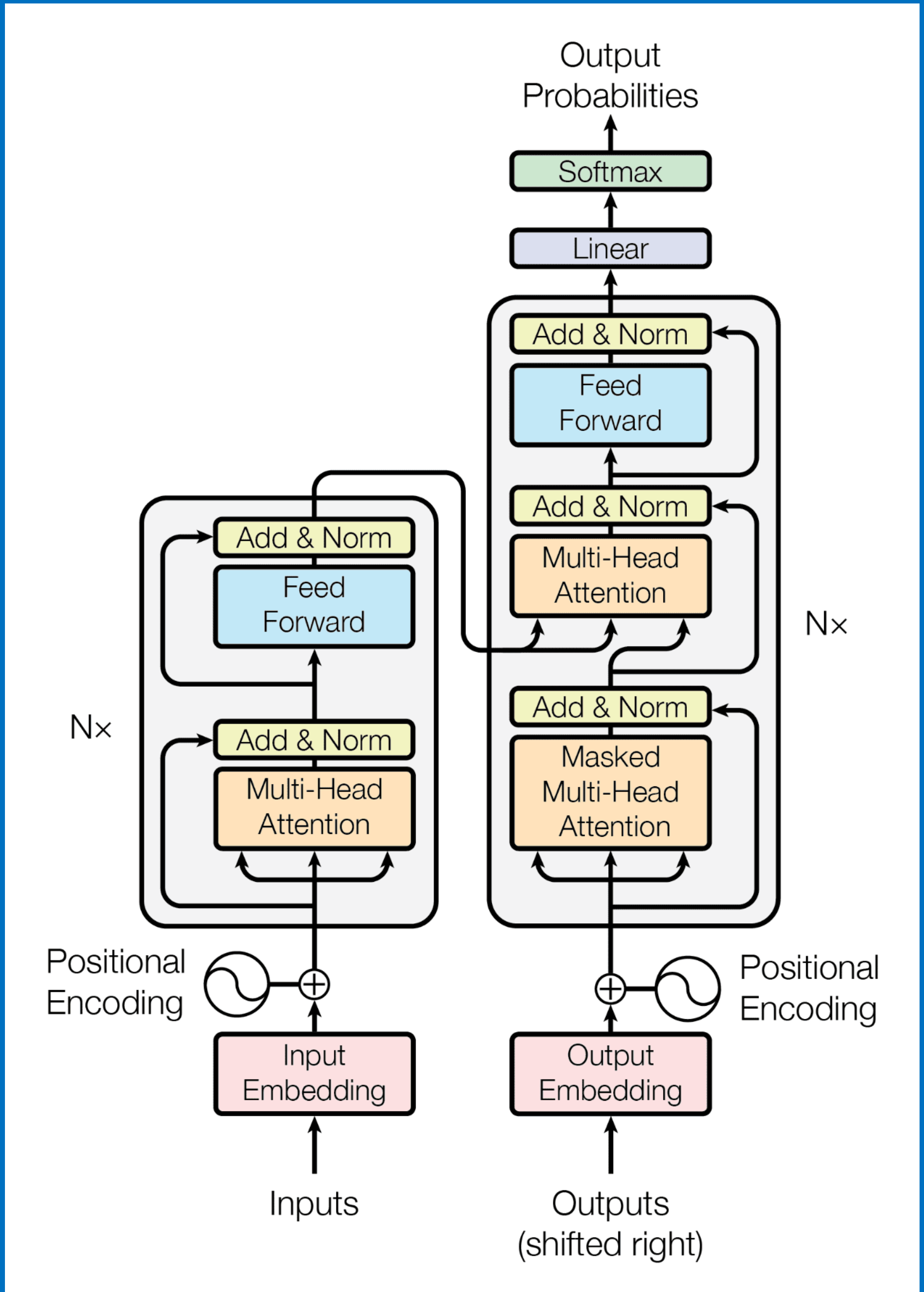


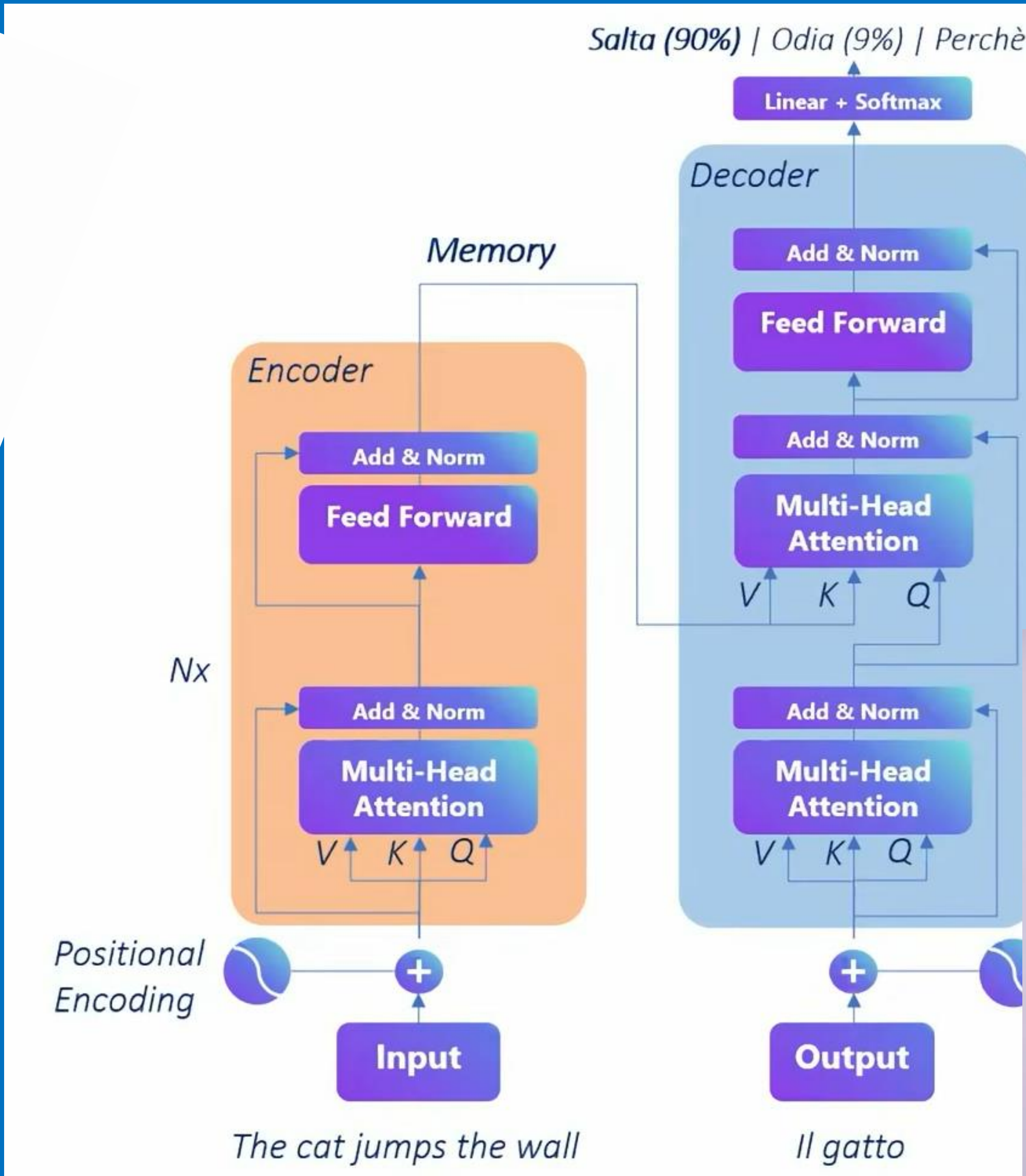
Agenda for Today:

- RNNs – the heart of all transformers
- A brief summary on Transformers
- Embedding layer, aka convert indices into dimensions
- Attention layer, single or multi-head
- ViT - Transformers in Computer Vision
- DIY – How to code your own ViT
- Should I code transformers?

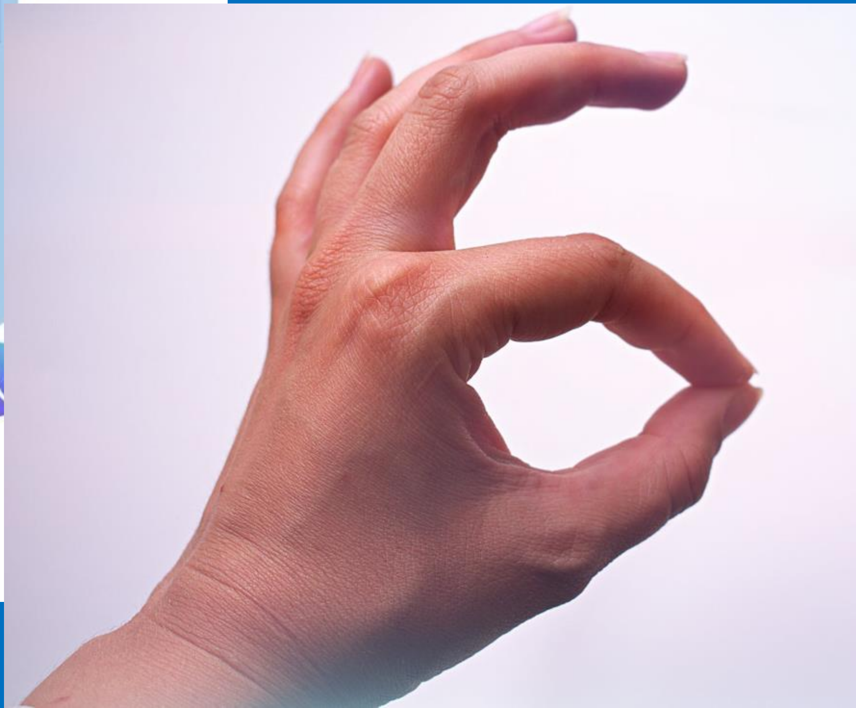

The
cat
jumps
the
wall
hstart
E
E
E
E
E
h0
h1
h2
h3
h4
<s>
D
h5
D
h6
D
h7
D
h8
D
h9
D
Il
gatto
salta
il
muro
<end>

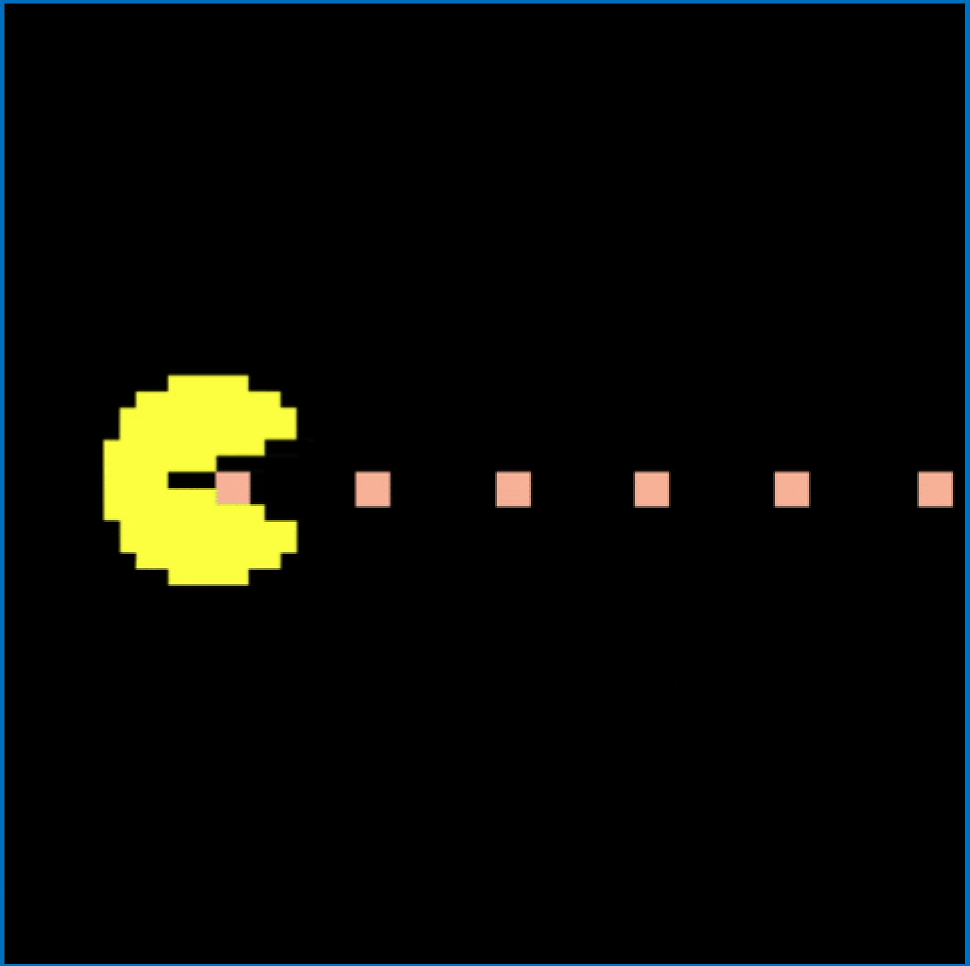
Who has seen this picture?

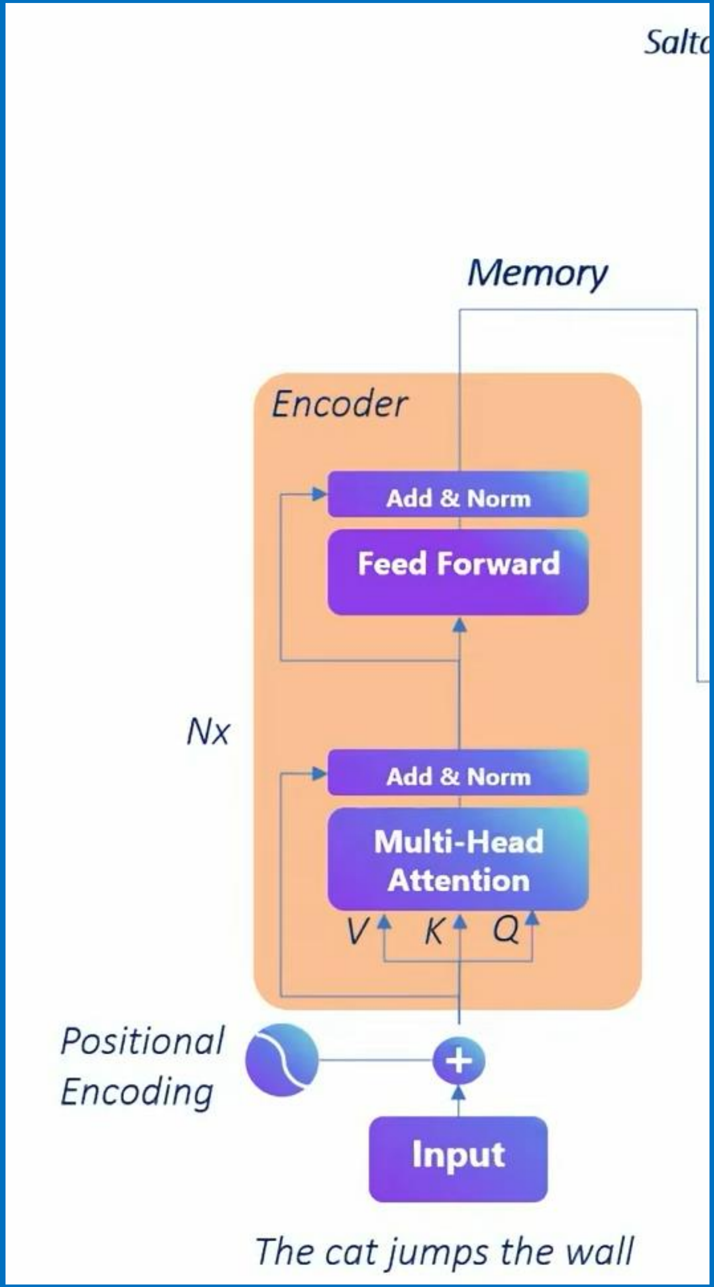


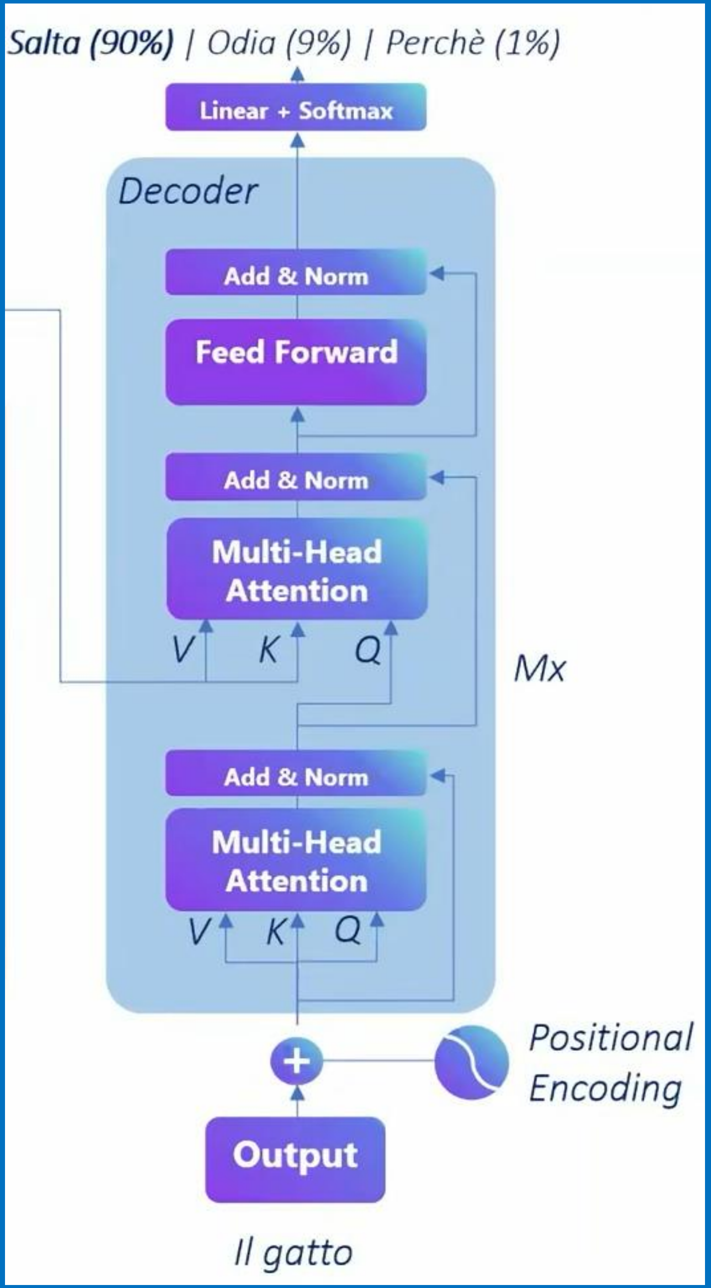


Salta (90%) | Odia (9%) | Perchè (1%)
Linear + Softmax
Decoder
Memory
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
V
K
Q
Mx
Add & Norm
Multi-Head Attention
V
K
Q
Encoder
Add & Norm
Feed Forward
Nx
Add & Norm
Multi-Head Attention
V
K
Q
Positional Encoding
+
+
Input
Output
The cat jumps the wall
Il gatto



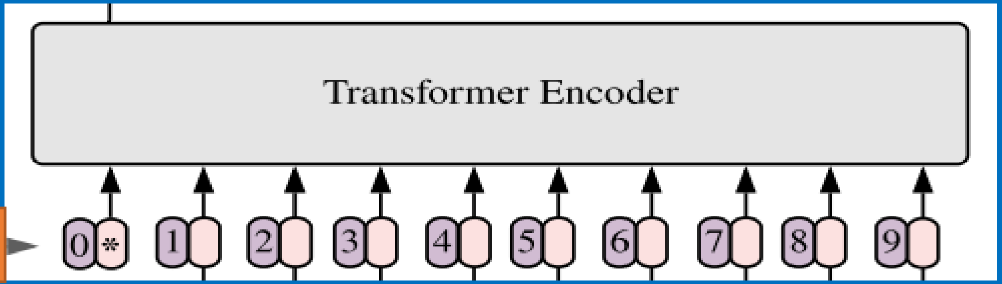



Salt
Memory
Encoder
Add & Norm
Feed Forward
Nx
Add & Norm
Multi-Head Attention
V
K
Q
Positional Encoding
Input
The cat jumps the wall

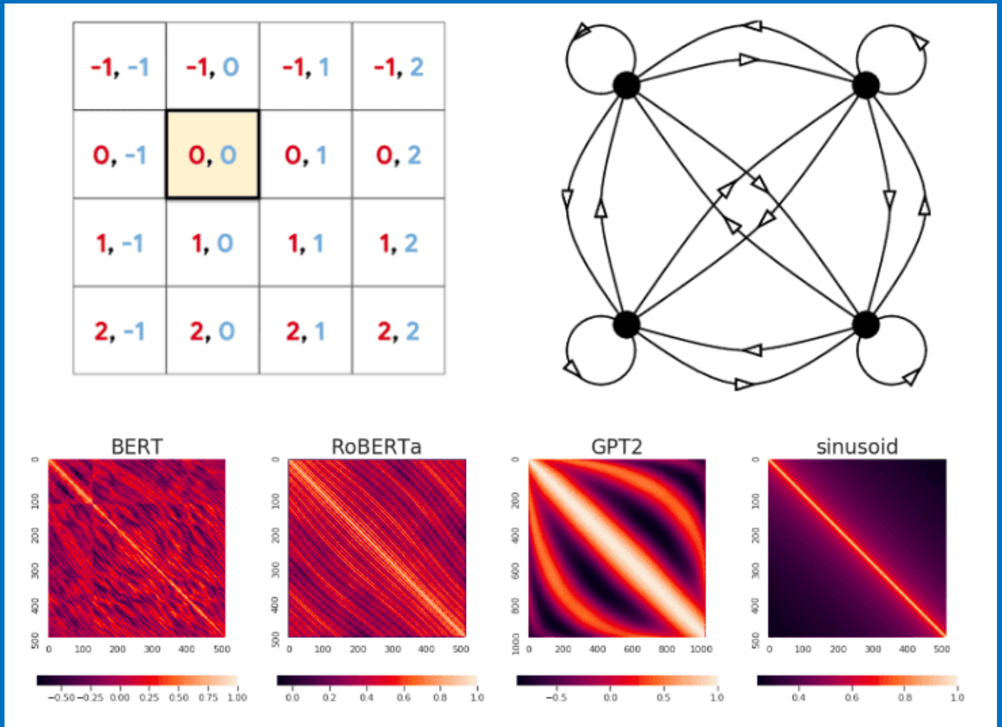

Salta (90%) | Odia (9%) | Perchè (1%)
Linear + Softmax
Decoder
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
V
K
Q
Mx
Add & Norm
Multi-Head Attention
V
K
Q
Positional Encoding
Output
Il gatto

DJ
designed by vexels

Positional Encoding

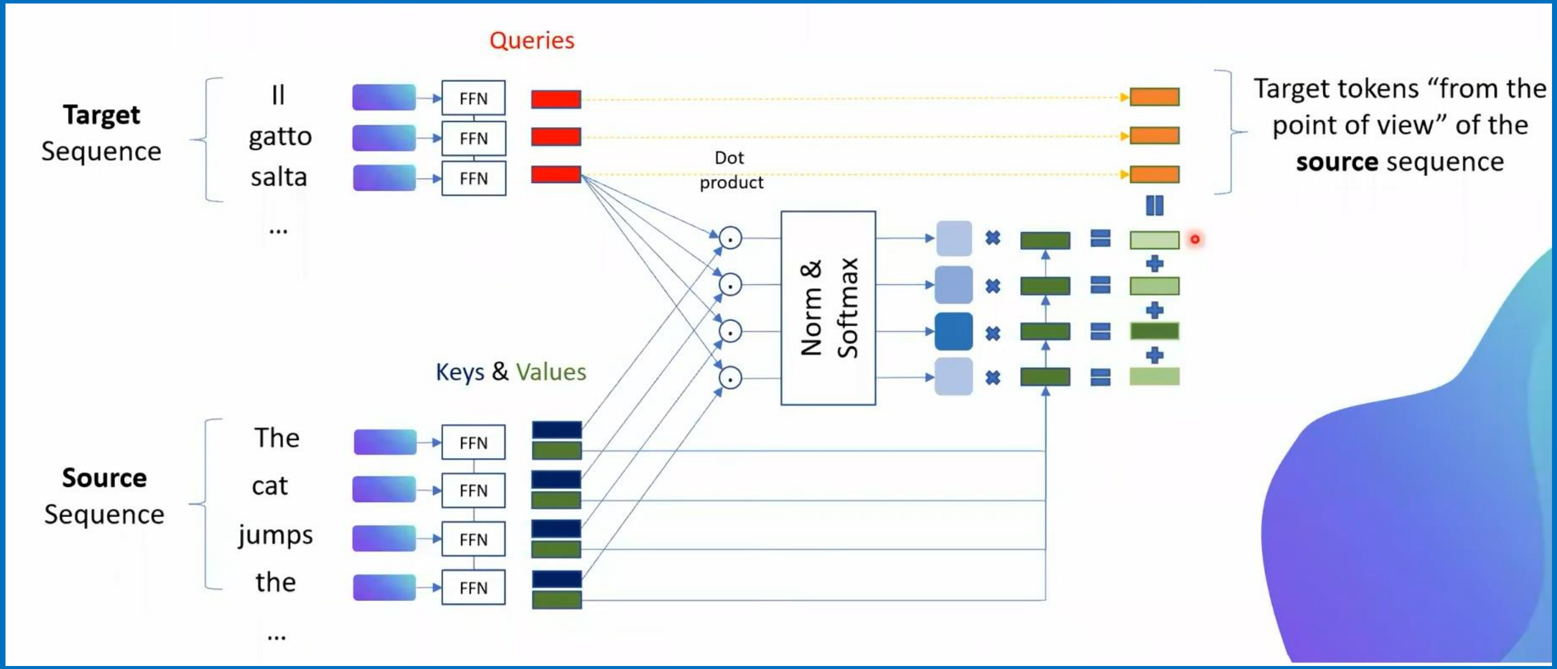


[0]
|
V
[0.25, 0.49, ...]

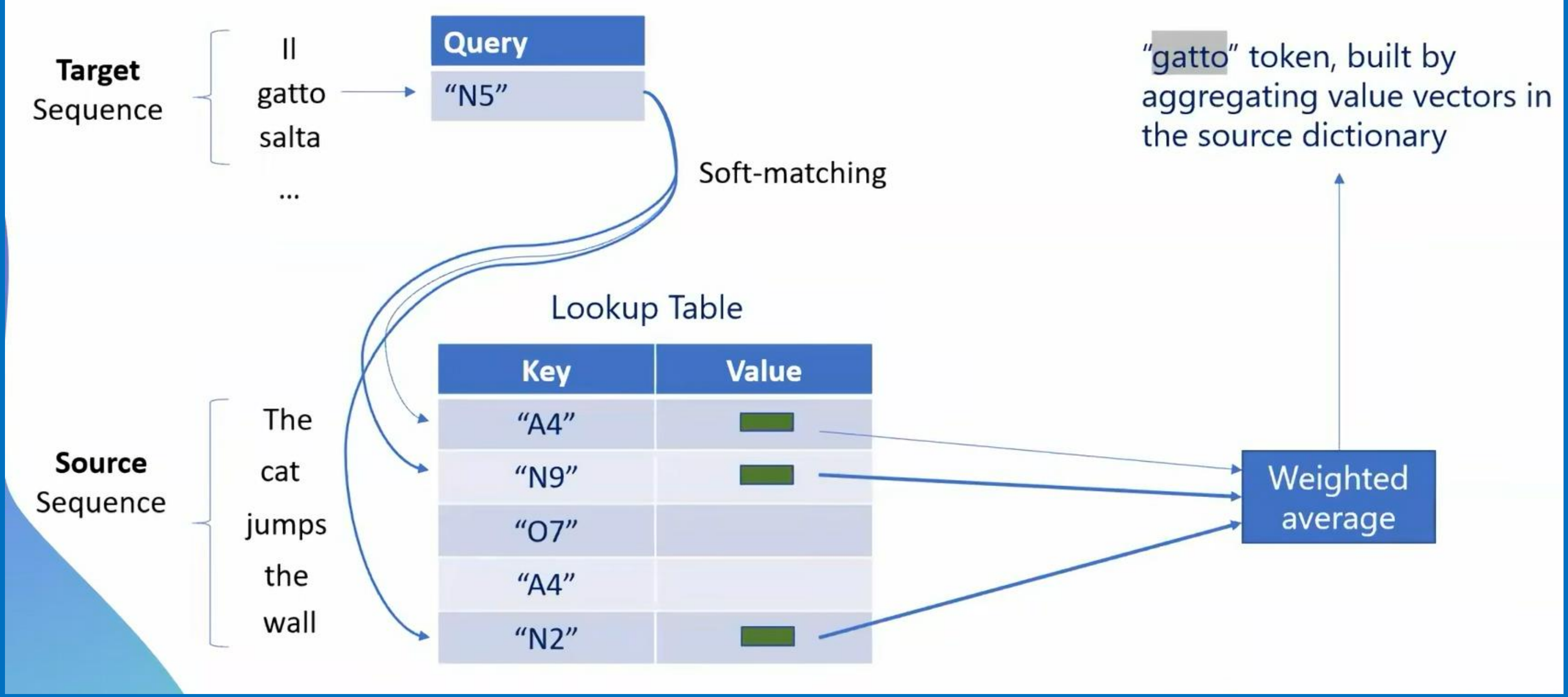


-1, -1	-1, 0	-1, 1	-1, 2
0, -1	0, 0	0, 1	0, 2
1, -1	1, 0	1, 1	1, 2
2, -1	2, 0	2, 1	2, 2

Attention


Queries
Target Sequence
Il
gatto
salta
...
FFN
FFN
FFN
Dot product
Norm & Softmax
Keys & Values
Source Sequence
The
cat
jumps
the
...
FFN
FFN
FFN
FFN
Target tokens "from the point of view" of the **source** sequence

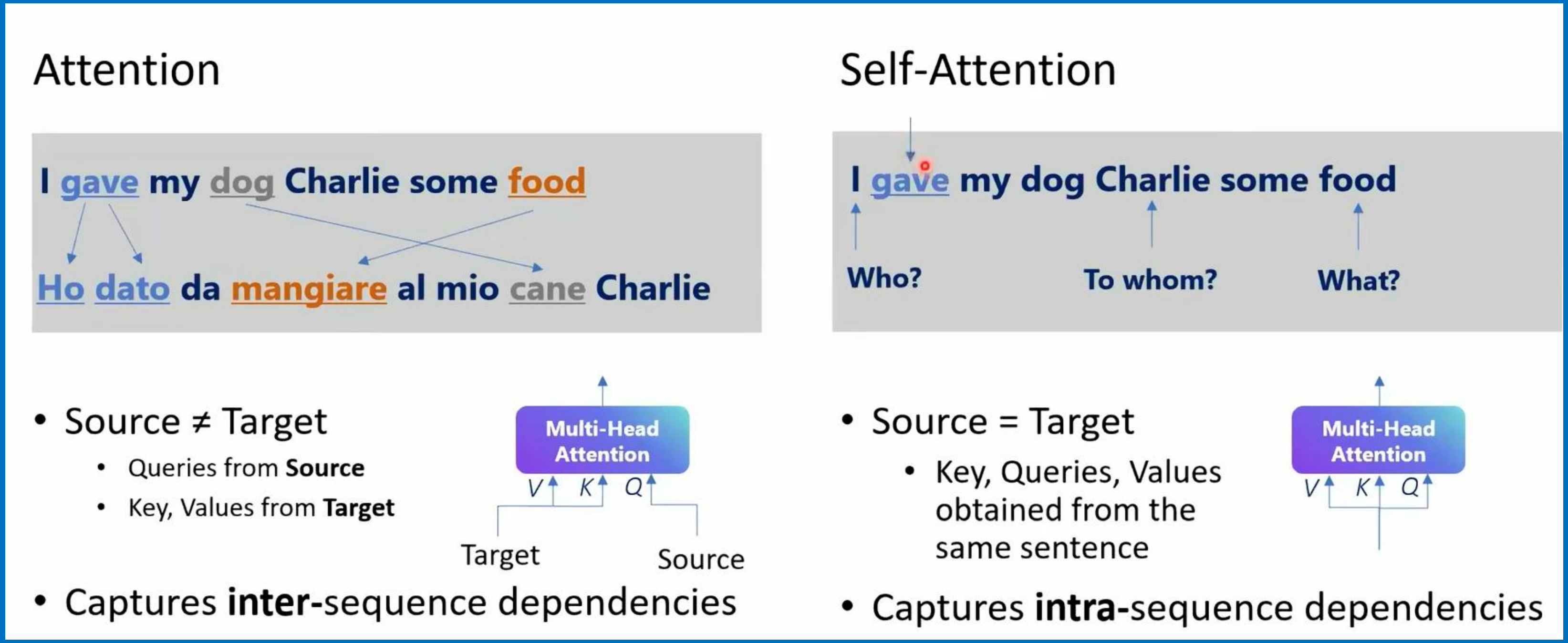
Attention... SIMPLIFIED!



Lookup Table

Key	Value
"A4"	🟩
"N9"	🟩
"O7"	
"A4"	
"N2"	🟩

Attention... SIMPLIFIED!



Attention

I gave my dog Charlie some food

Ho dato da mangiare al mio cane Charlie

- Source ≠ Target
  - Queries from **Source**
  - Key, Values from **Target**
- Captures **inter-**sequence dependencies

Self-Attention

I gave my dog Charlie some food

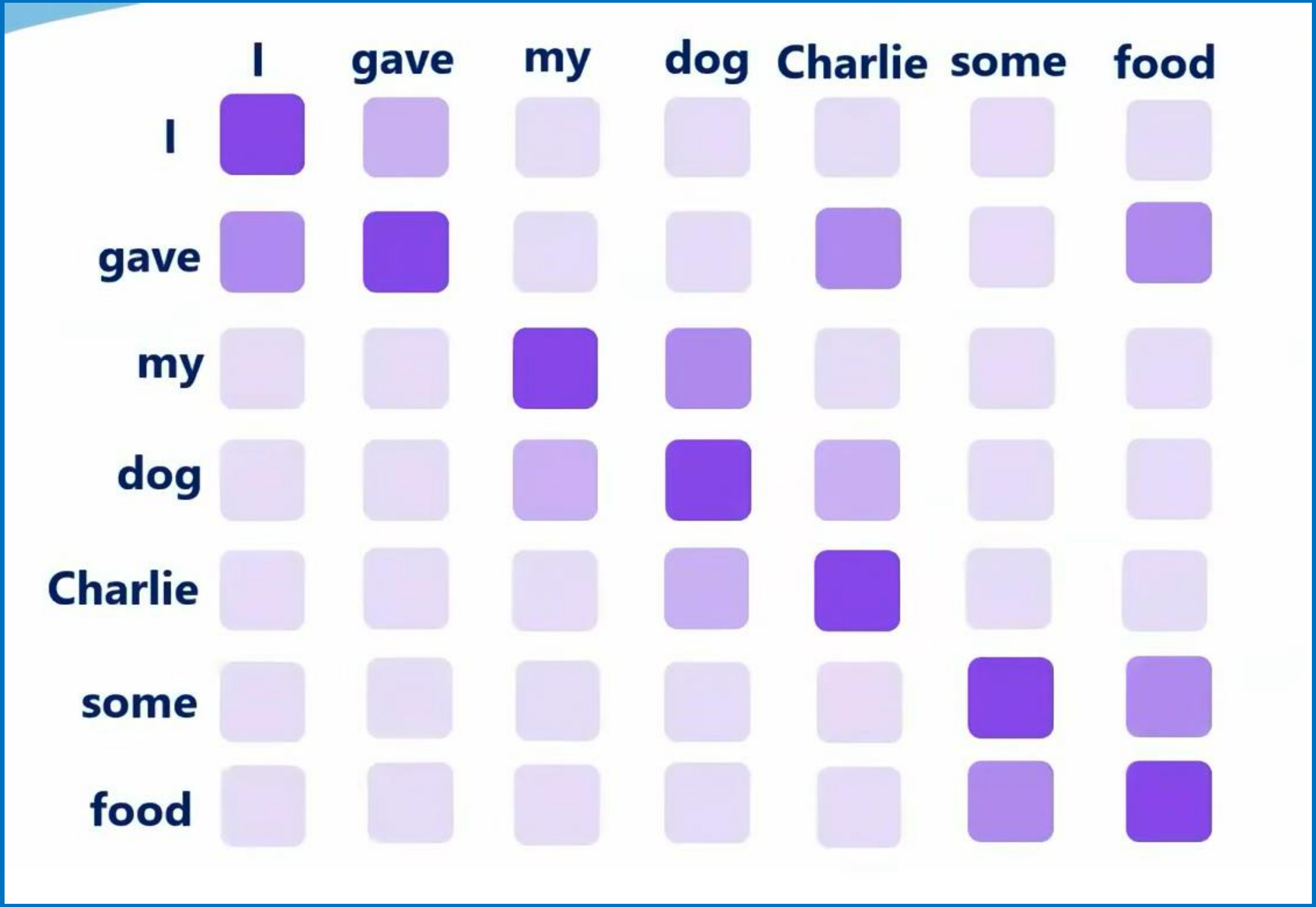
Who?

To whom?

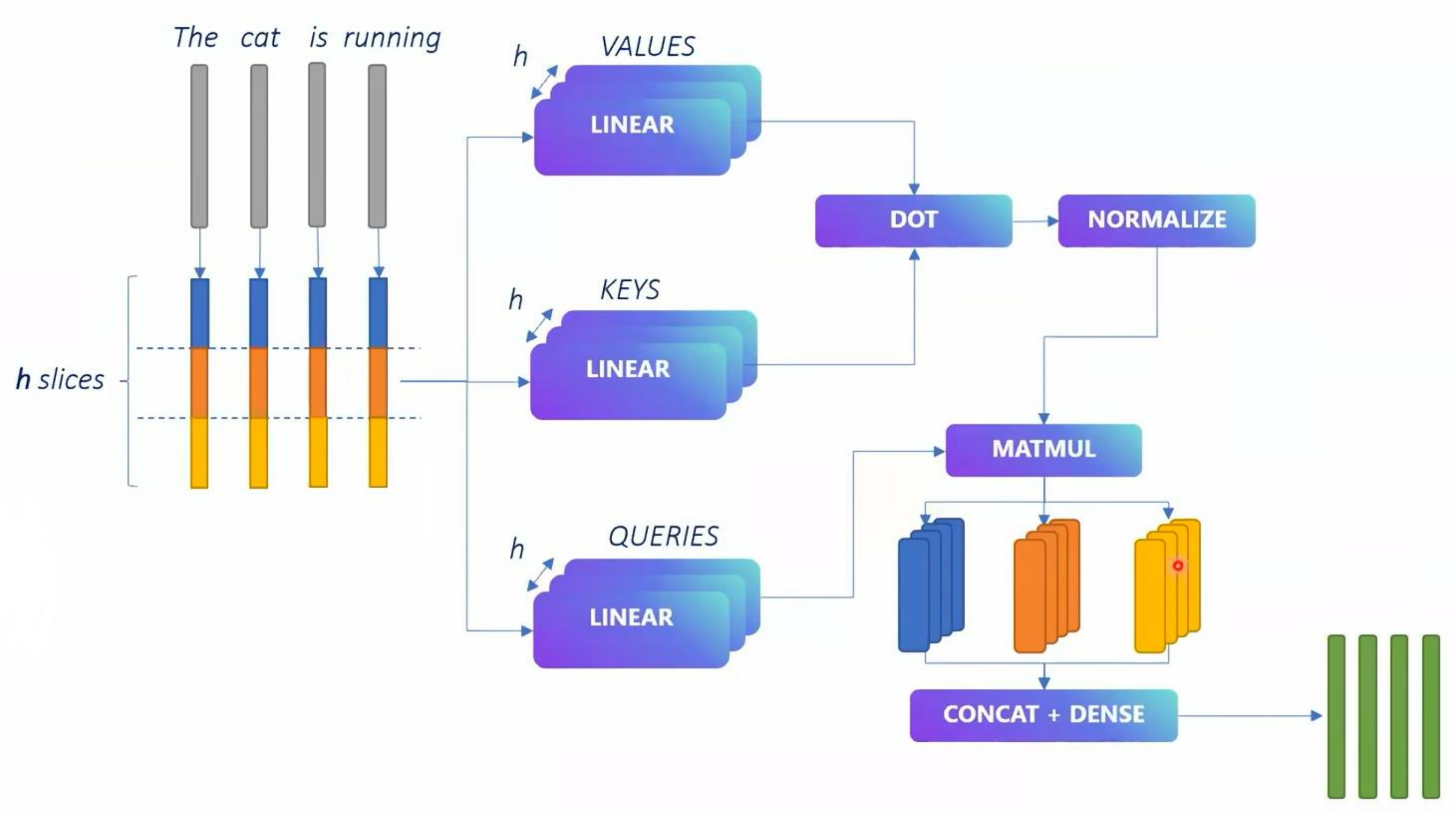
What?

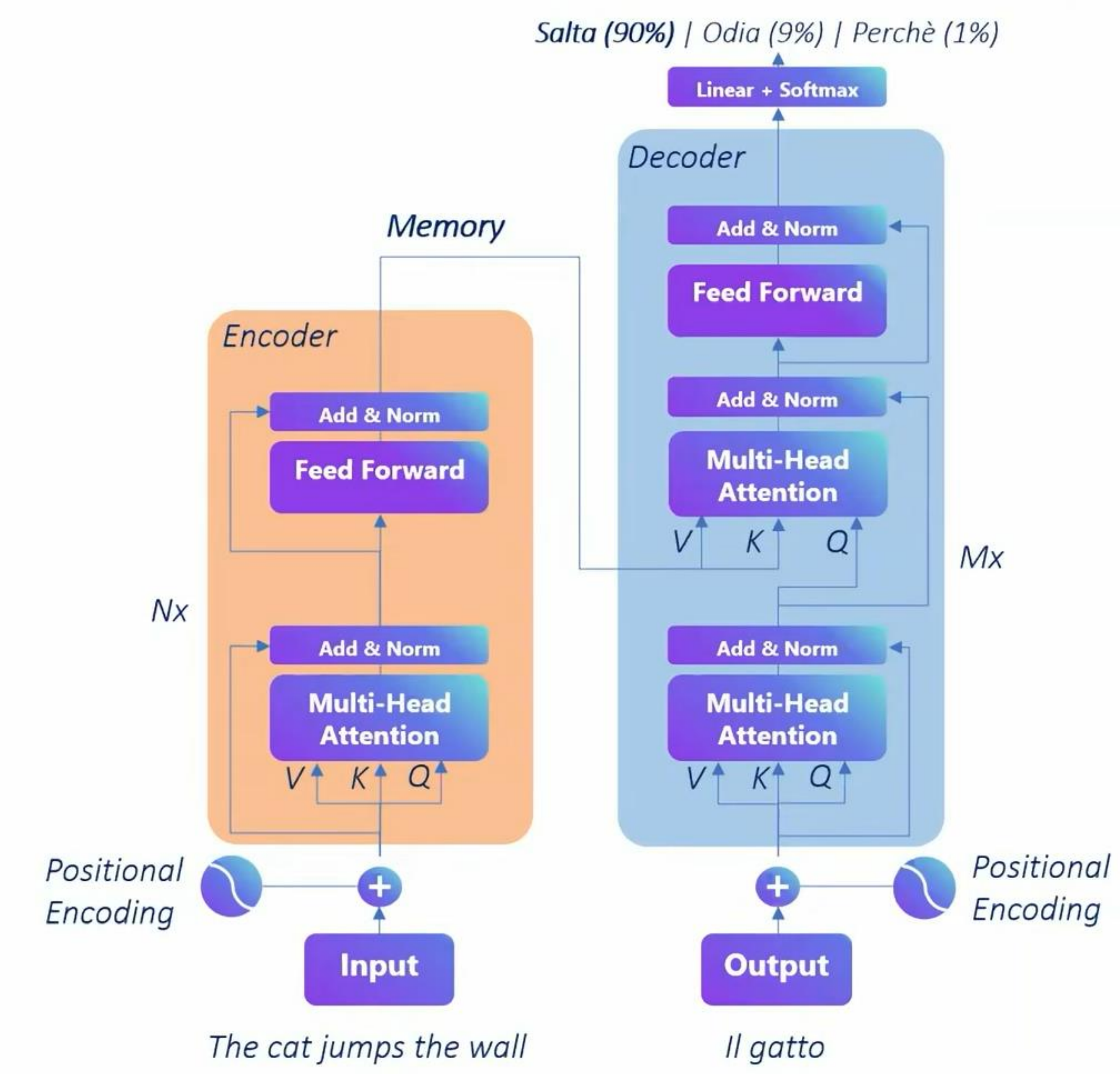
- Source = Target
  - Key, Queries, Values obtained from the same sentence
- Captures **intra-**sequence dependencies

Self - Attention

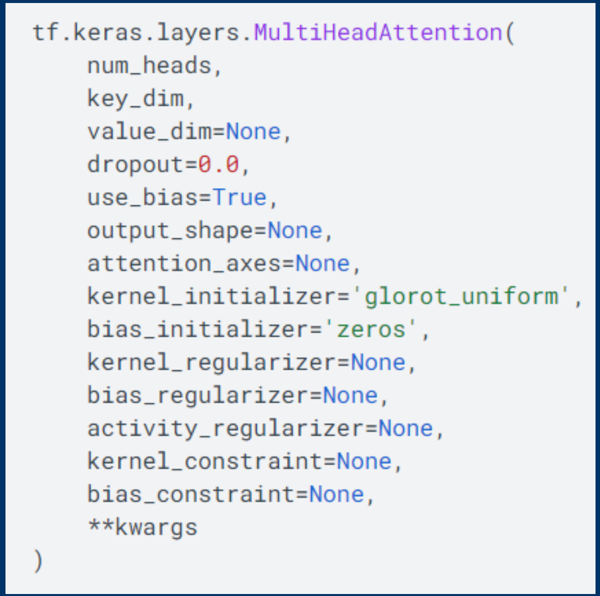


Multi – Head Attention

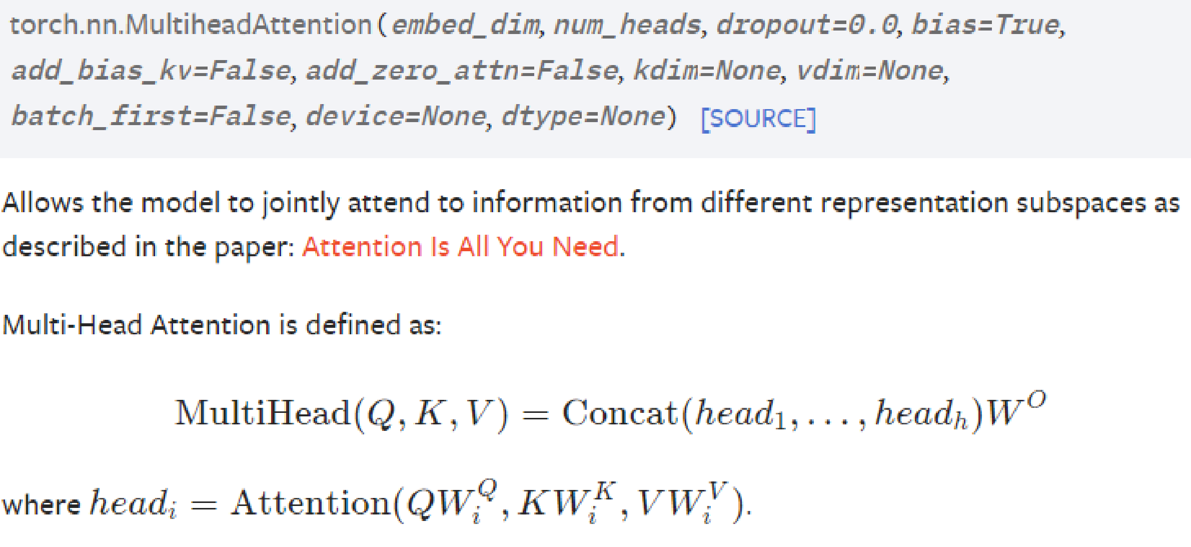



Salta (90%) | Odia (9%) | Perchè (1%)
Linear + Softmax
Decoder
Memory
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
V
K
Q
Mx
Add & Norm
Multi-Head Attention
V
K
Q
Encoder
Add & Norm
Feed Forward
Nx
Add & Norm
Multi-Head Attention
V
K
Q
Positional Encoding
Positional Encoding
Input
Output
The cat jumps the wall
Il gatto

But I Can’t Code Complex Stuffs Like These ☹



```
tf.keras.layers.MultiHeadAttention(
    num_heads,
    key_dim,
    value_dim=None,
    dropout=0.0,
    use_bias=True,
    output_shape=None,
    attention_axes=None,
    kernel_initializer='glorot_uniform',
    bias_initializer='zeros',
    kernel_regularizer=None,
    bias_regularizer=None,
    activity_regularizer=None,
    kernel_constraint=None,
    bias_constraint=None,
    **kwargs
)
```



torch.nn.MultiheadAttention(*embed_dim*, *num_heads*, *dropout=0.0*, *bias=True*, *add_bias_kv=False*, *add_zero_attn=False*, *kdim=None*, *vdim=None*, *batch_first=False*, *device=None*, *dtype=None*) [SOURCE]

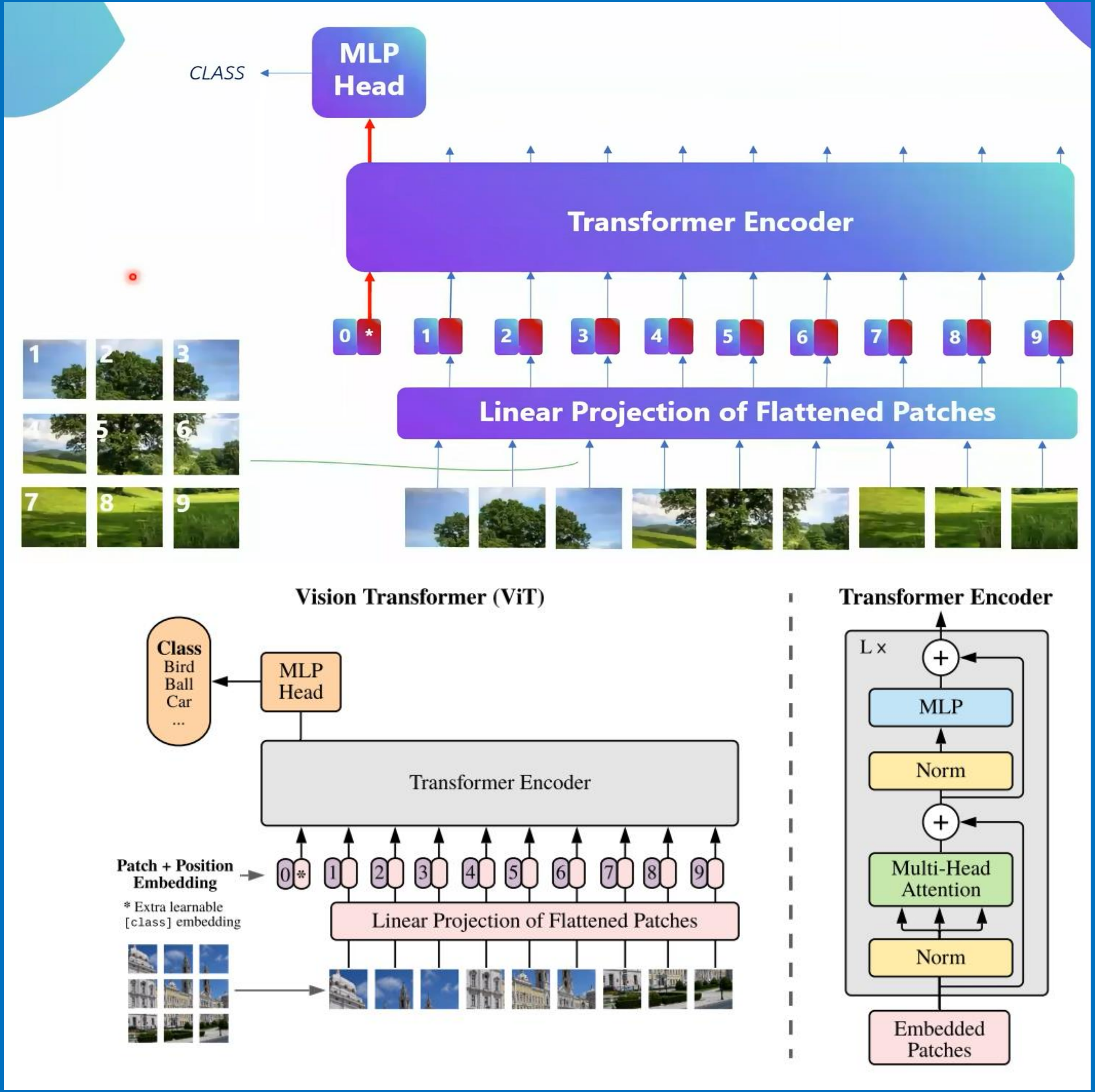
Allows the model to jointly attend to information from different representation subspaces as described in the paper: Attention Is All You Need.

Multi-Head Attention is defined as:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(head_1, \dots, head_h)W^O$$

where $head_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$.

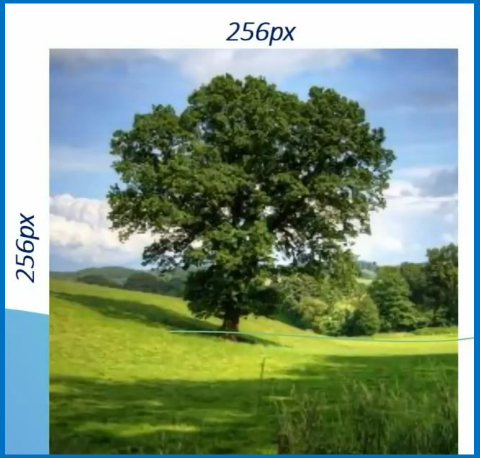
Transformers in Computer Vision



Attention for Images

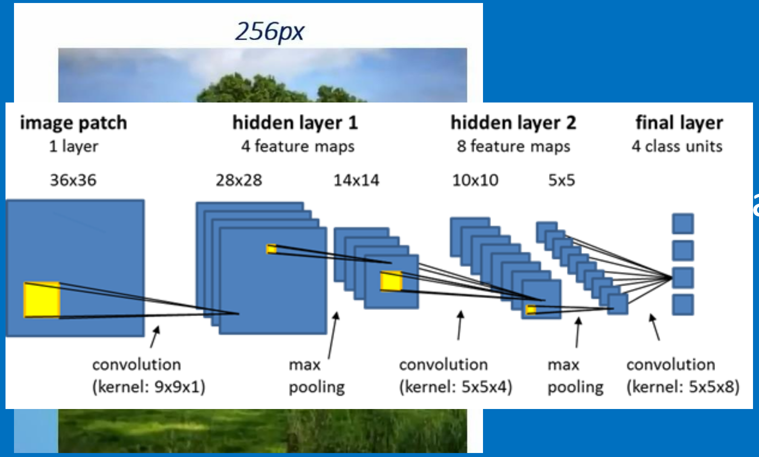
Positional Encoding & Improving Attention Performance

Every Pixel



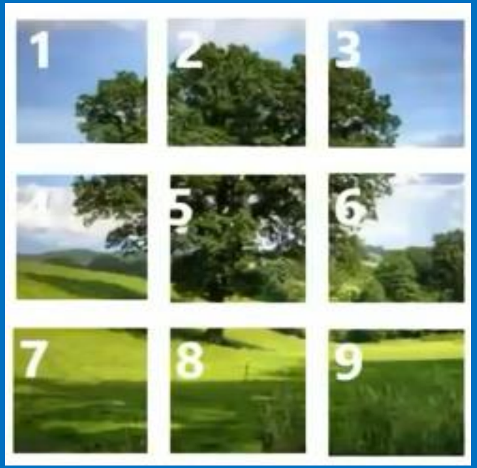
256^2 relations = (256^2)^2 operations

‚Square' Attention



are_side//2)^2 – still a lot

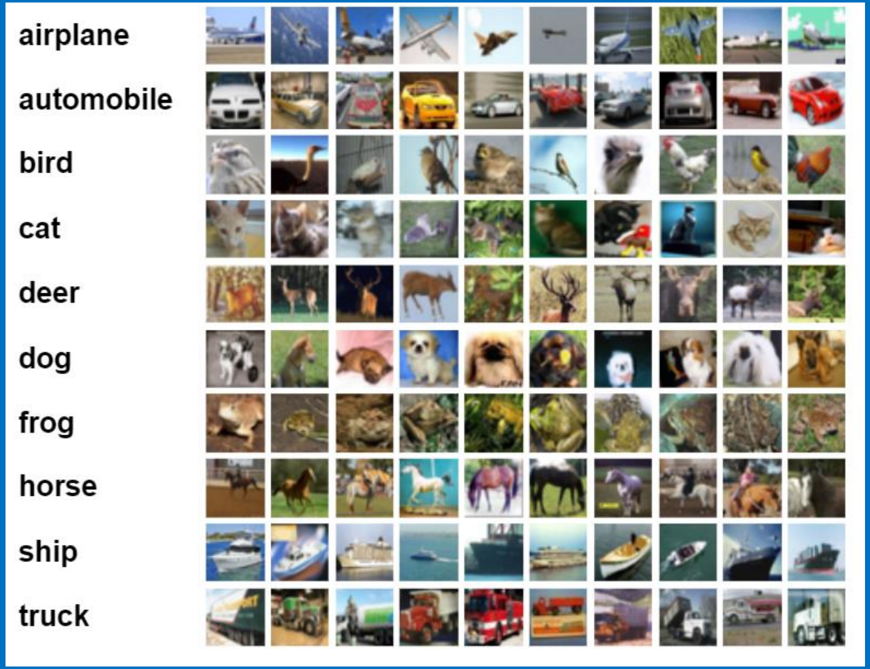
Chunks



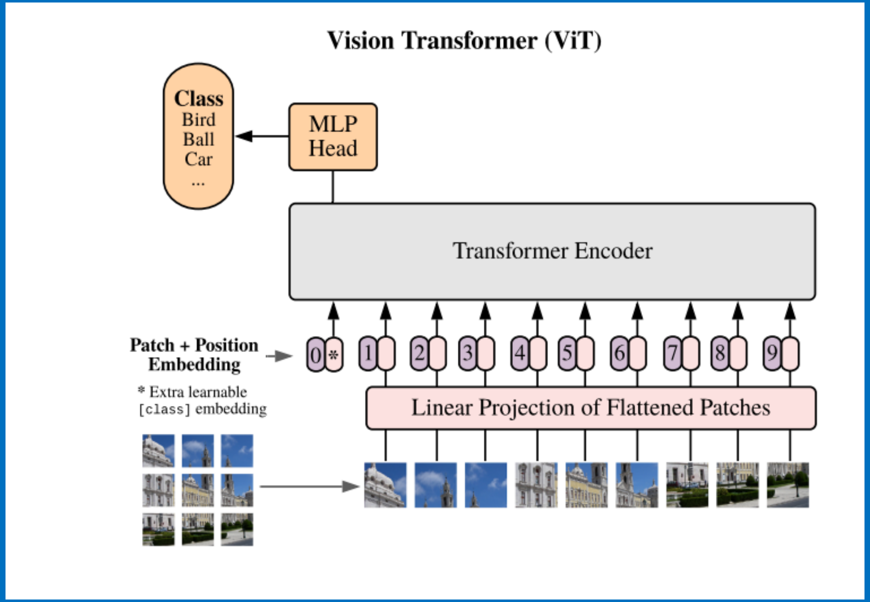
Best performance, but reduced Attention benefits

DIY
Time to Code It!

Dataset:
CIFAR100



ViT Model from „Transformers for Image Recognition at Scale" paper


Vision Transformer (ViT)
Class
Bird
Ball
Car
...
MLP Head
Transformer Encoder
Patch + Position Embedding
* Extra learnable [class] embedding
0*
1
2
3
4
5
6
7
8
9
Linear Projection of Flattened Patches

Should You Try It?

- Basis of ChatGPT – „GPT” is „Generative Pre-Trained Transformer”
- ChatGPT Chief Scientist Salary - ~2 000 000 $/yr
- ViTs are expected to be the next „big boom”

References:

- Attention is All You Need
  https://arxiv.org/pdf/1706.03762.pdf

- An Image Is Worth 16x16 Words: Transformers For Image Rrecognition At Scale
  https://arxiv.org/pdf/2010.11929v2.pdf

Thank You!