UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

> Reinventa el mundo <

Deep Learning

Week 15 : Transformers

TRANSFORMATEC

2017

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkorei Google Researcl usz@google.co

Llion Jones* Google Research llion@google.com

Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

Łukasz Kaiser* Google Brain lukaszkaiser@google.com

Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

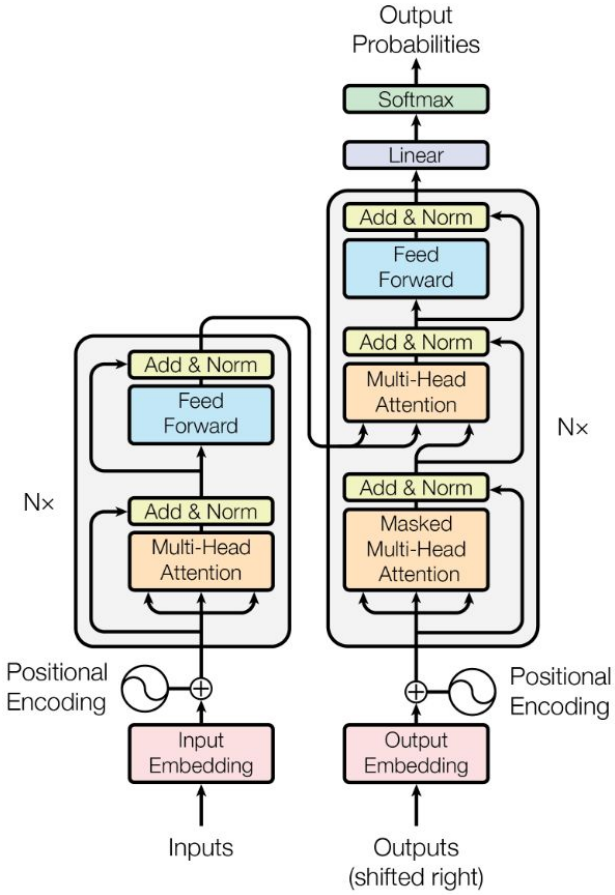
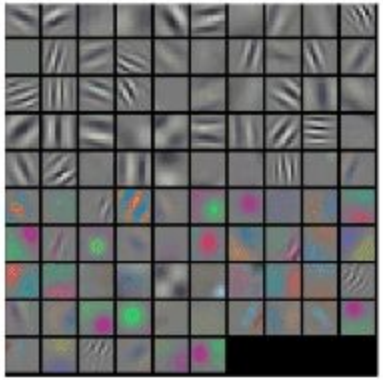


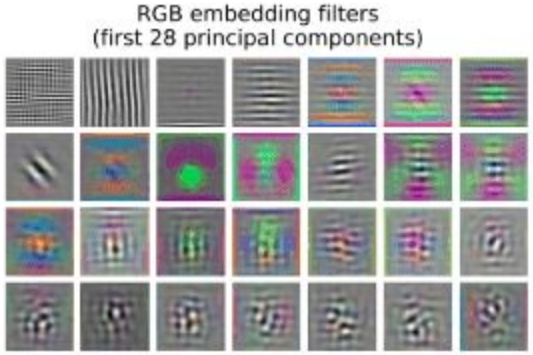
Figure 1: The Transformer - model architecture.

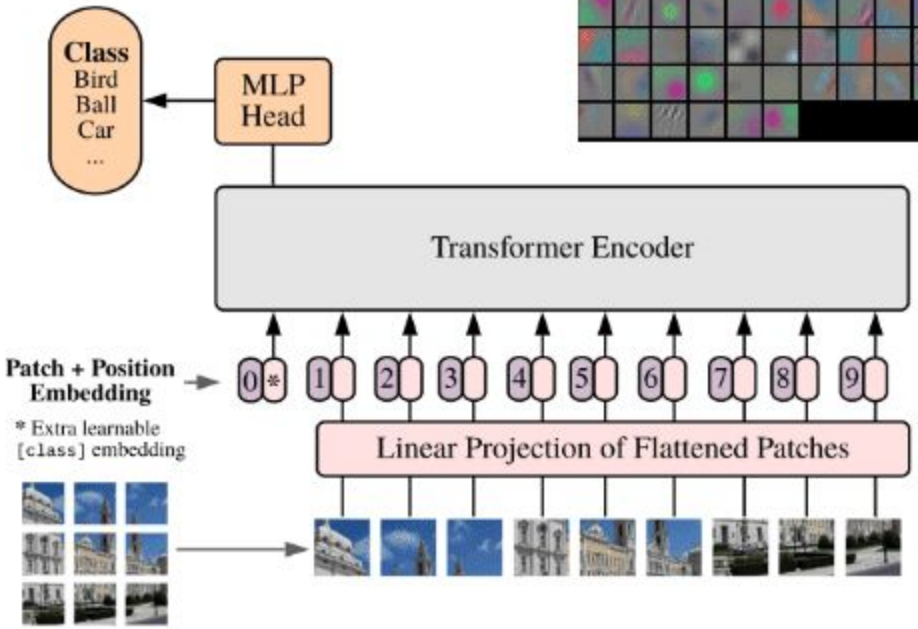
2021

Alexnet 1st conv filters



ViT 1st linear embedding filters





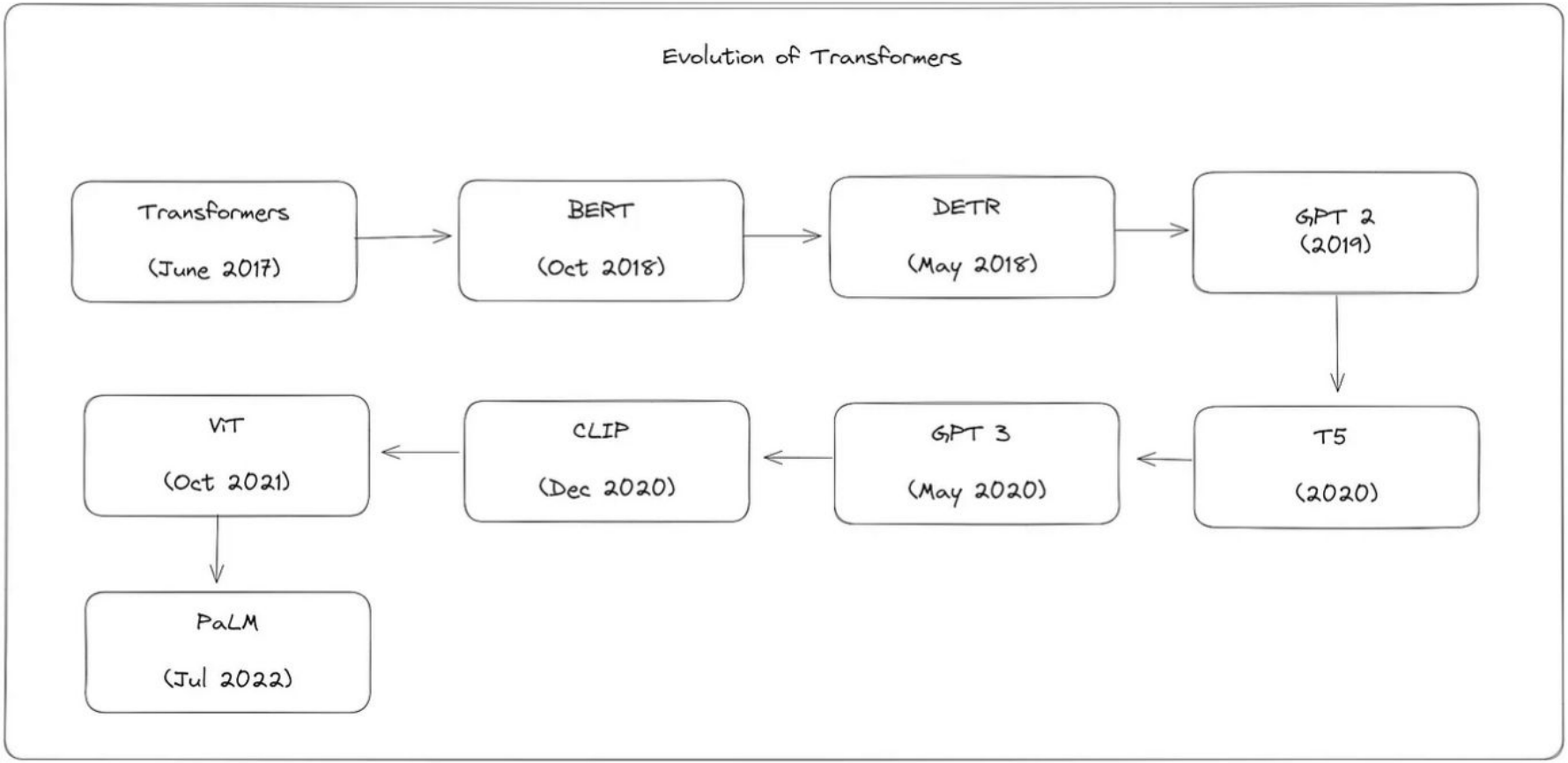
AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE

Alexey Dosovitskiy[*,†], **Lucas Beyer**[*], **Alexander Kolesnikov**[*], **Dirk Weissenborn**[*], **Xiaohua Zhai**[*], **Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby**[*,†]

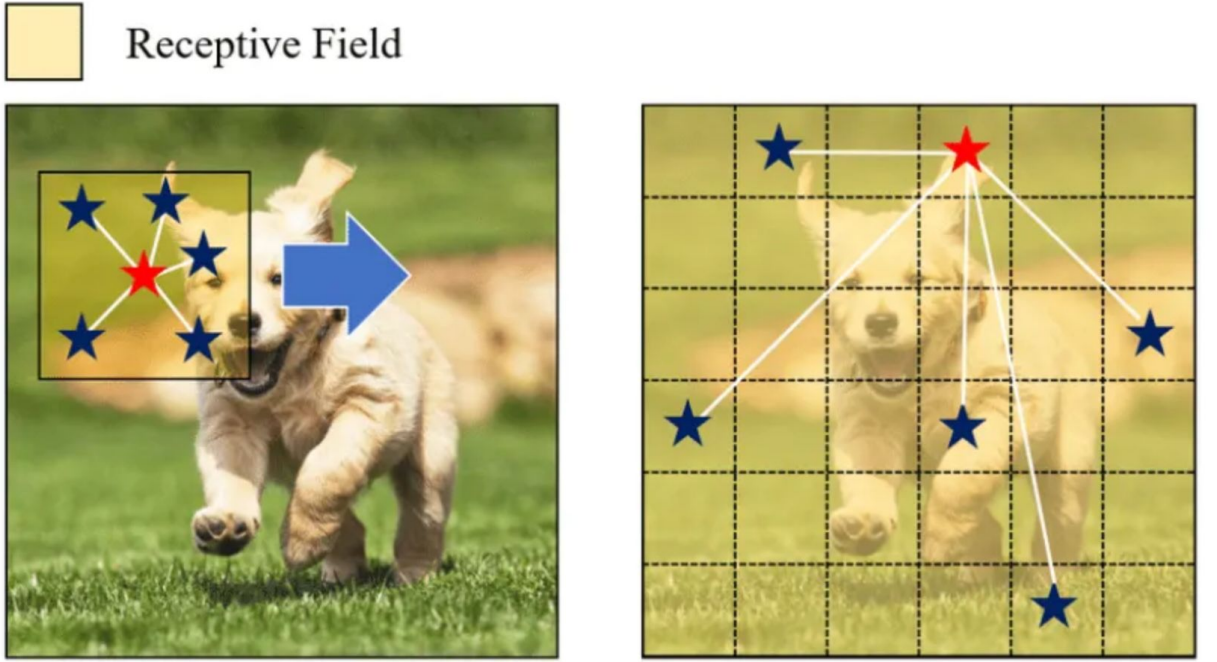
[*]equal technical contribution, [†]equal advising

Google Research, Brain Team

{adosovitskiy, neilhoulsby}@google.com


Evolution of Transformers
Transformers
(June 2017)
BERT
(Oct 2018)
DETR
(May 2018)
GPT 2
(2019)
T5
(2020)
GPT 3
(May 2020)
CLIP
(Dec 2020)
ViT
(Oct 2021)
PaLM
(Jul 2022)

Why Transformers? (Shortcomings of CNNs)



Convolution of CNN

Attention of Vision Transformer

Algunos Drawbacks de Transformers

A comparative study between vision transformers and CNNs in digital pathology

Luca Deininger[1] deininger.luca@gmail.com
Bernhard Stimpel[1] bernhard.stimpel@roche.com
Anil Yuce[1] anil.yuce@roche.com
Samaneh Abbasi-Sureshjani[1] samaneh.abbasi@roche.com
Simon Schönenberger[1] simon.schoenenberger@gmail.com
Paolo Ocampo[2] ocampo.paolo-santiago@gene.com
Konstanty Korski[1] konstanty.korski@roche.com
Fabien Gaire[1] fabien.gaire@roche.com

[1] F. Hoffmann-La Roche AG, Grenzacherstrasse 124, 4070 Basel, Switzerland
[2] Genentech, Inc., 1 DNA Way, South San Francisco, CA 94080, USA

Table 1. Model test PR AUC and accuracy (ACC). For datasets with multiple test sets (LUAD and breast), we show the unweighted mean per dataset.

Model	FW	Metric	CRC9	SLN	DLBCL	LUAD	Breast
ResNet18	×	PR AUC	**0.999**	0.885	**0.976**	0.913	0.809
		ACC	**0.995**	0.981	**0.88**	0.858	0.915
DeiT-Tiny	×	PR AUC	0.998	**0.917**	0.97	**0.94**	0.817
		ACC	0.982	**0.988**	0.874	0.88	0.913
PathNet	×	PR AUC	**0.999**	0.908	0.97	0.92	0.818
		ACC	**0.995**	0.979	0.866	**0.885**	0.92
DINO	×	PR AUC	**0.999**	0.912	0.958	0.933	**0.828**
		ACC	0.991	0.984	0.874	0.871	**0.924**

ter for the remaining tasks. The aggregated predictions of both models on slide level were correlated, indicating that the models captured similar imaging features. All together, the vision transformer models performed on par with the ResNet18 while requiring more effort to train. In order to surpass the performance of convolutional neural networks, vision transformers might require more challenging tasks to benefit from their weak inductive bias.

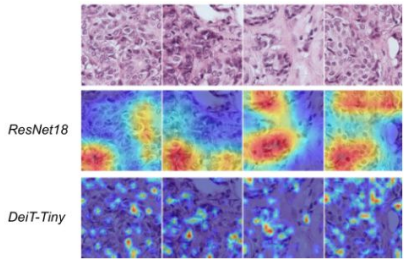


Figure 2. Comparison of *ResNet18* and *DeiT-Tiny Grad-CAM* heatmaps for randomly selected SLN tumor patches. Both models classified all shown patches correctly with a probability of > 0.9.

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

Query

Keys

Values

Normalization Term

QUERY

"So I wanna watch an underdog action movie"

Movie Database

Die Hard

A NYPD cop battles terrorists who have taken hostages in a skyscraper during a Christmas party.

Forrest Gump

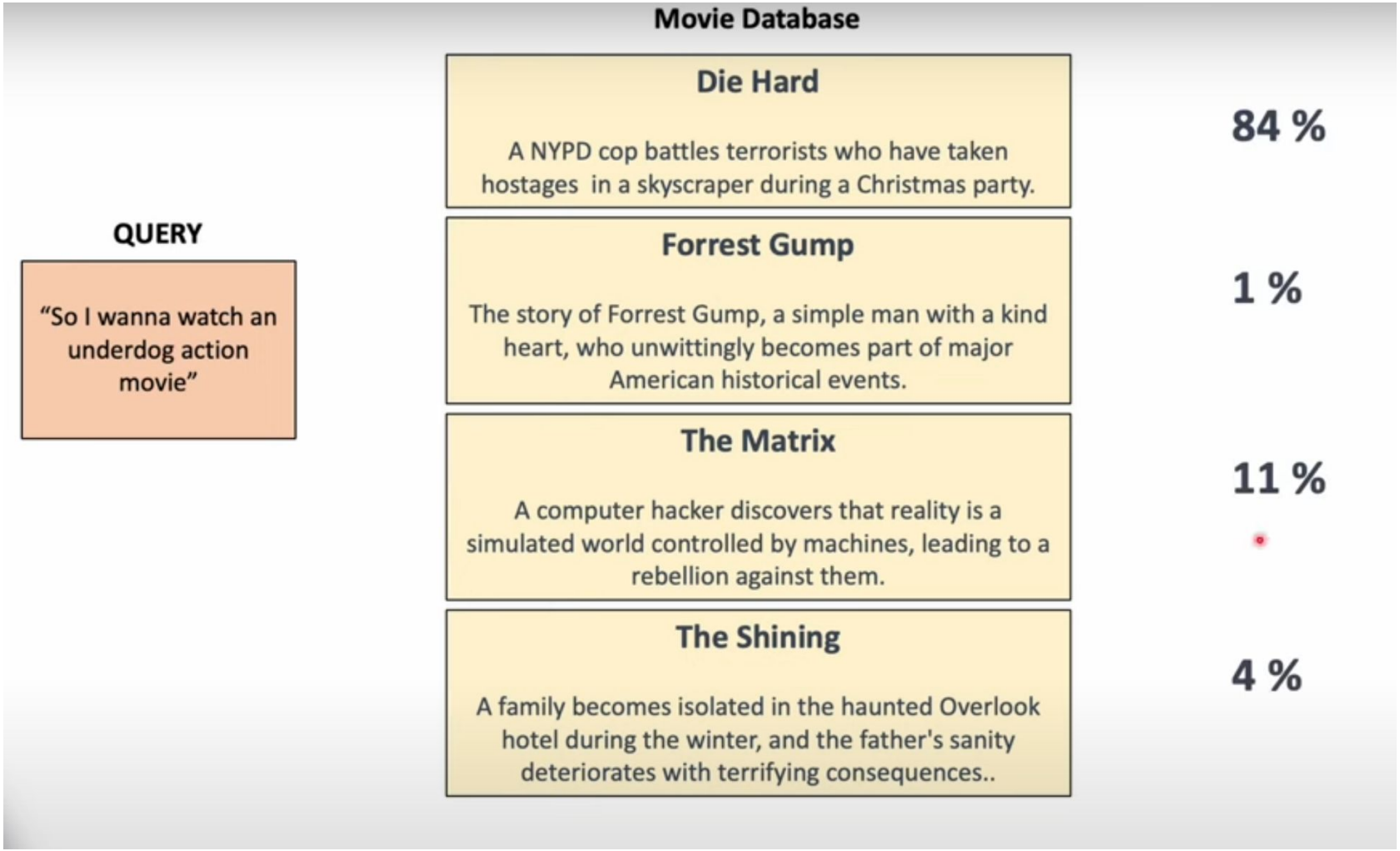
The story of Forrest Gump, a simple man with a kind heart, who unwittingly becomes part of major American historical events.

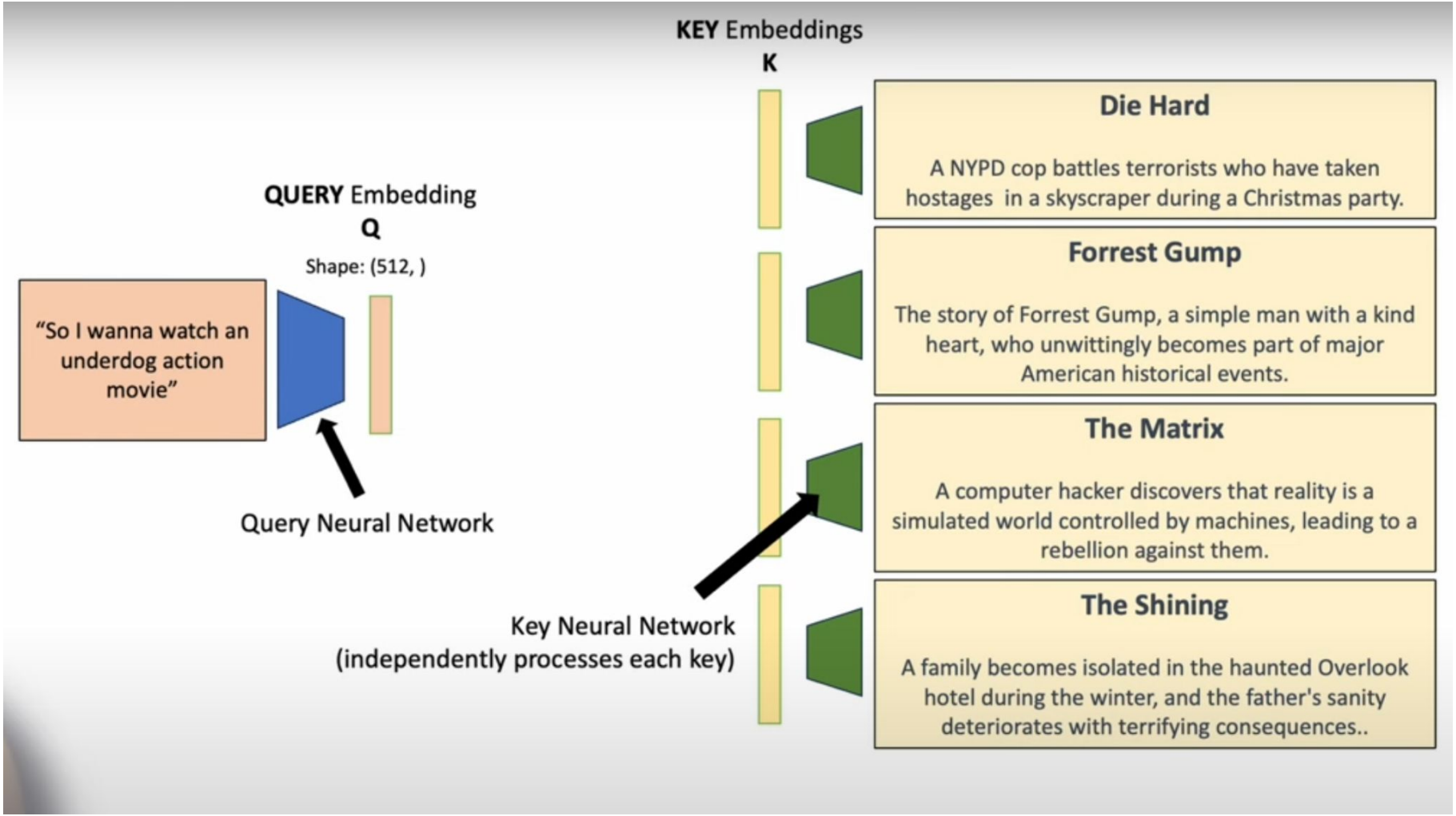
The Matrix

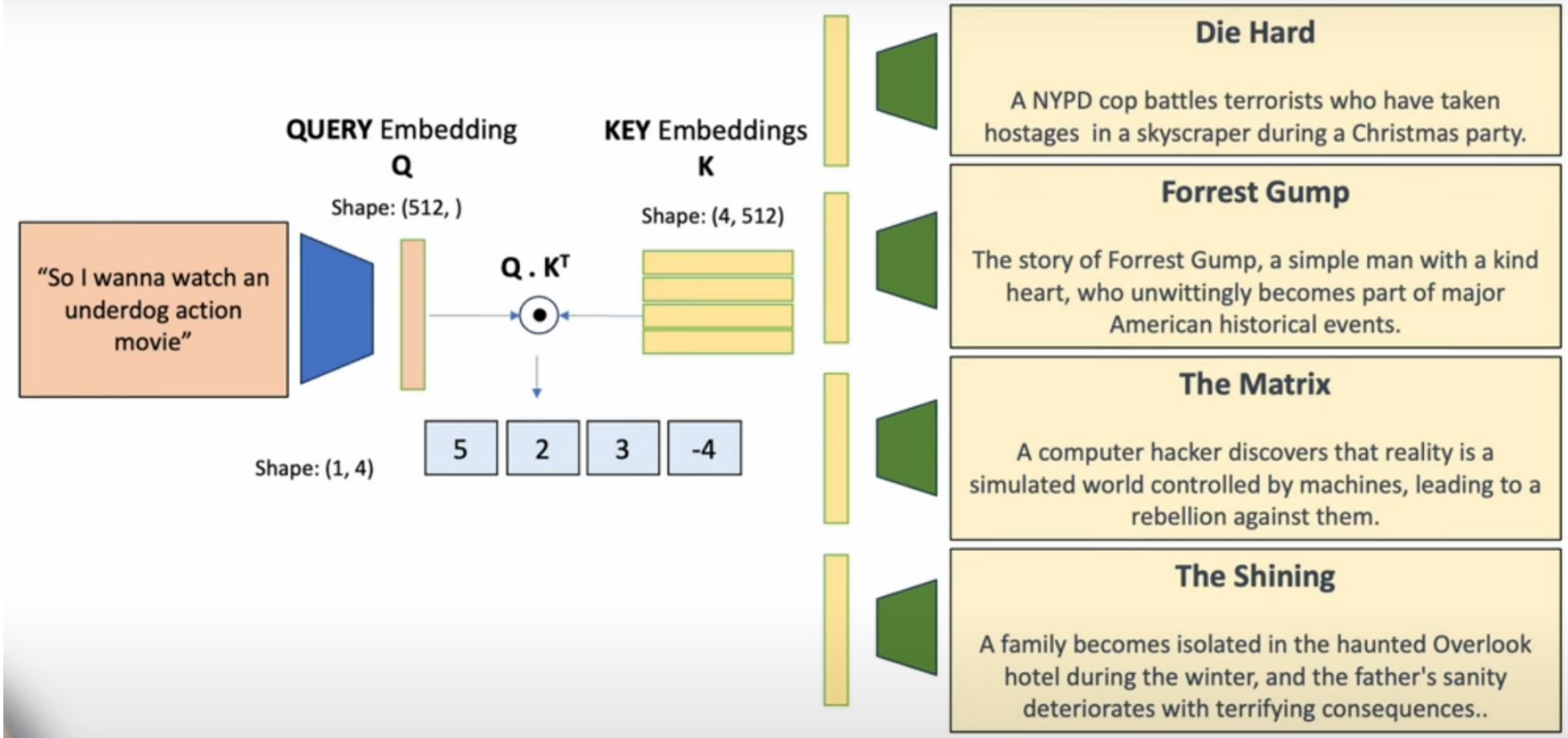
A computer hacker discovers that reality is a simulated world controlled by machines, leading to a rebellion against them.

The Shining

A family becomes isolated in the haunted Overlook hotel during the winter, and the father's sanity deteriorates with terrifying consequences..

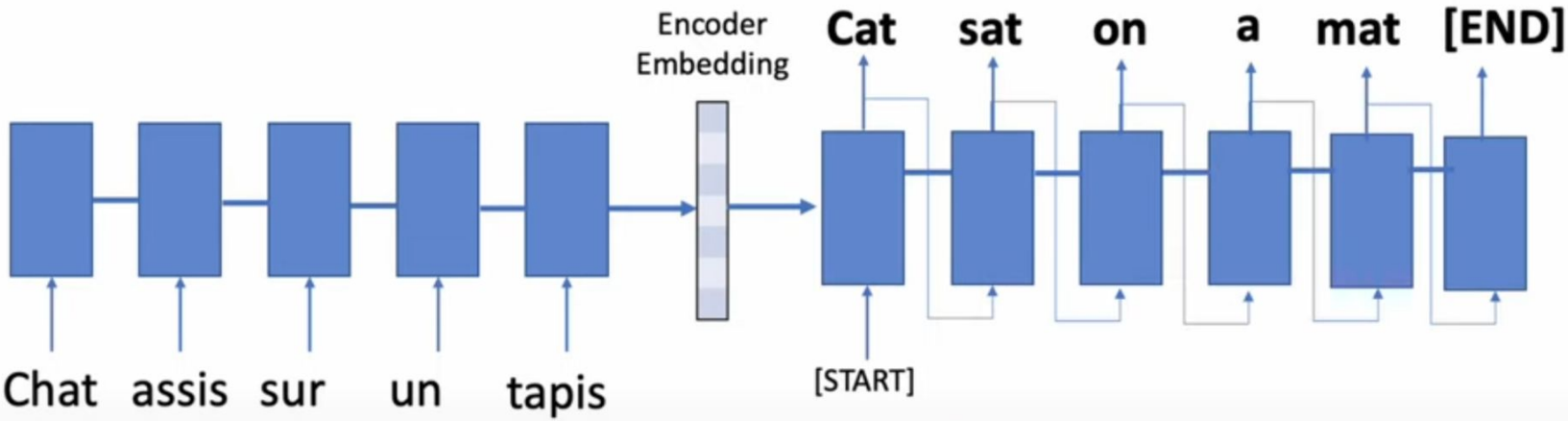

Movie Database
QUERY
"So I wanna watch an underdog action movie"
Die Hard
A NYPD cop battles terrorists who have taken hostages in a skyscraper during a Christmas party.
84 %
Forrest Gump
The story of Forrest Gump, a simple man with a kind heart, who unwittingly becomes part of major American historical events.
1 %
The Matrix
A computer hacker discovers that reality is a simulated world controlled by machines, leading to a rebellion against them.
11 %
The Shining
A family becomes isolated in the haunted Overlook hotel during the winter, and the father's sanity deteriorates with terrifying consequences..
4 %


KEY Embeddings
K
Die Hard
A NYPD cop battles terrorists who have taken hostages in a skyscraper during a Christmas party.
Forrest Gump
The story of Forrest Gump, a simple man with a kind heart, who unwittingly becomes part of major American historical events.
The Matrix
A computer hacker discovers that reality is a simulated world controlled by machines, leading to a rebellion against them.
The Shining
A family becomes isolated in the haunted Overlook hotel during the winter, and the father's sanity deteriorates with terrifying consequences..
QUERY Embedding
Q
Shape: (512,)
"So I wanna watch an underdog action movie"
Query Neural Network
Key Neural Network
(independently processes each key)


QUERY Embedding
Q
Shape: (512,)
KEY Embeddings
K
Shape: (4, 512)
"So I wanna watch an underdog action movie"
Q . K^T
Shape: (1, 4)
5
2
3
-4
Die Hard
A NYPD cop battles terrorists who have taken hostages in a skyscraper during a Christmas party.
Forrest Gump
The story of Forrest Gump, a simple man with a kind heart, who unwittingly becomes part of major American historical events.
The Matrix
A computer hacker discovers that reality is a simulated world controlled by machines, leading to a rebellion against them.
The Shining
A family becomes isolated in the haunted Overlook hotel during the winter, and the father's sanity deteriorates with terrifying consequences..

Encoder Decoder Architecture

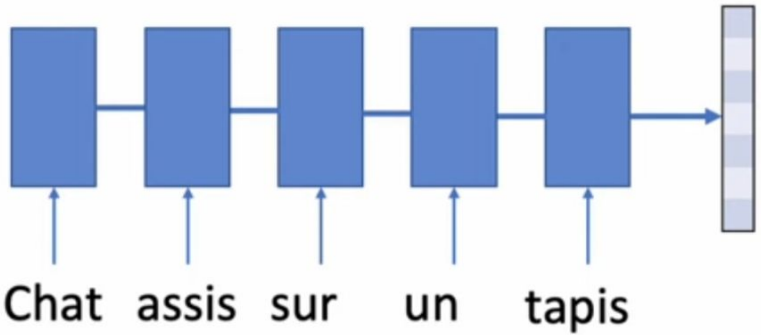
- Architecture to train models for Sequence to Sequence tasks



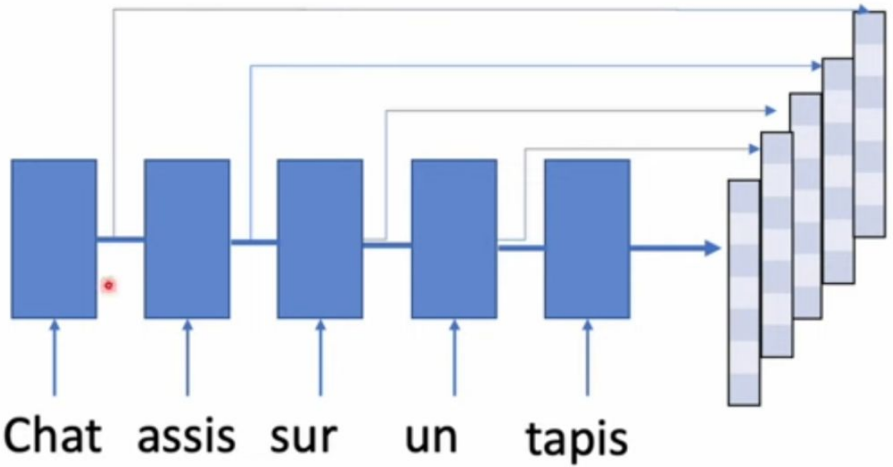
ENCODER
processes input sequence

DECODER
generates output sequence

Encoder Decoder Attention - Encoder

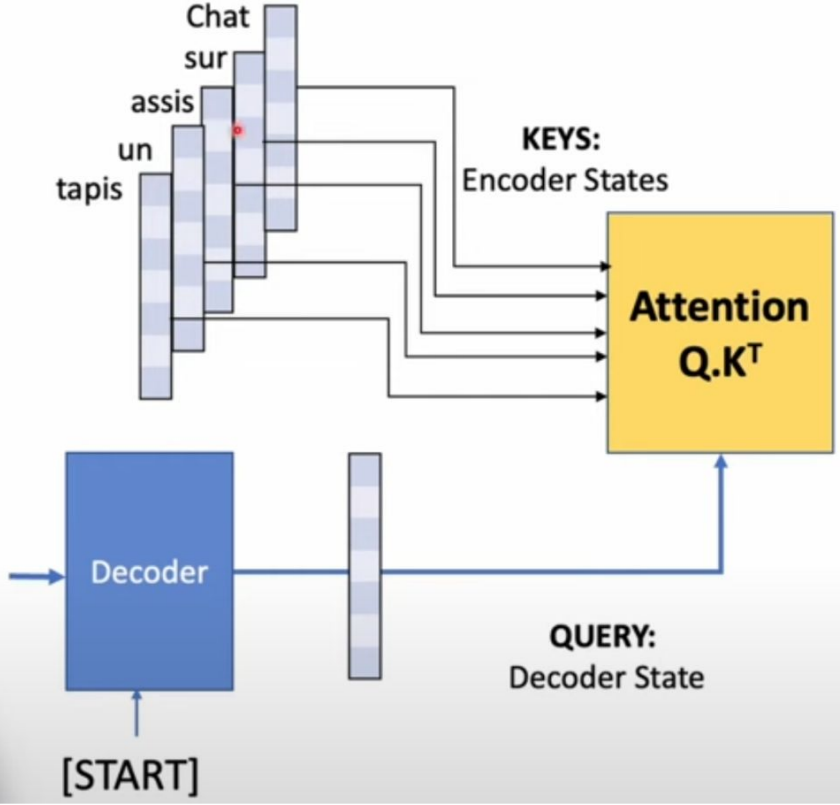


Before
Only final state of ENCODER is used

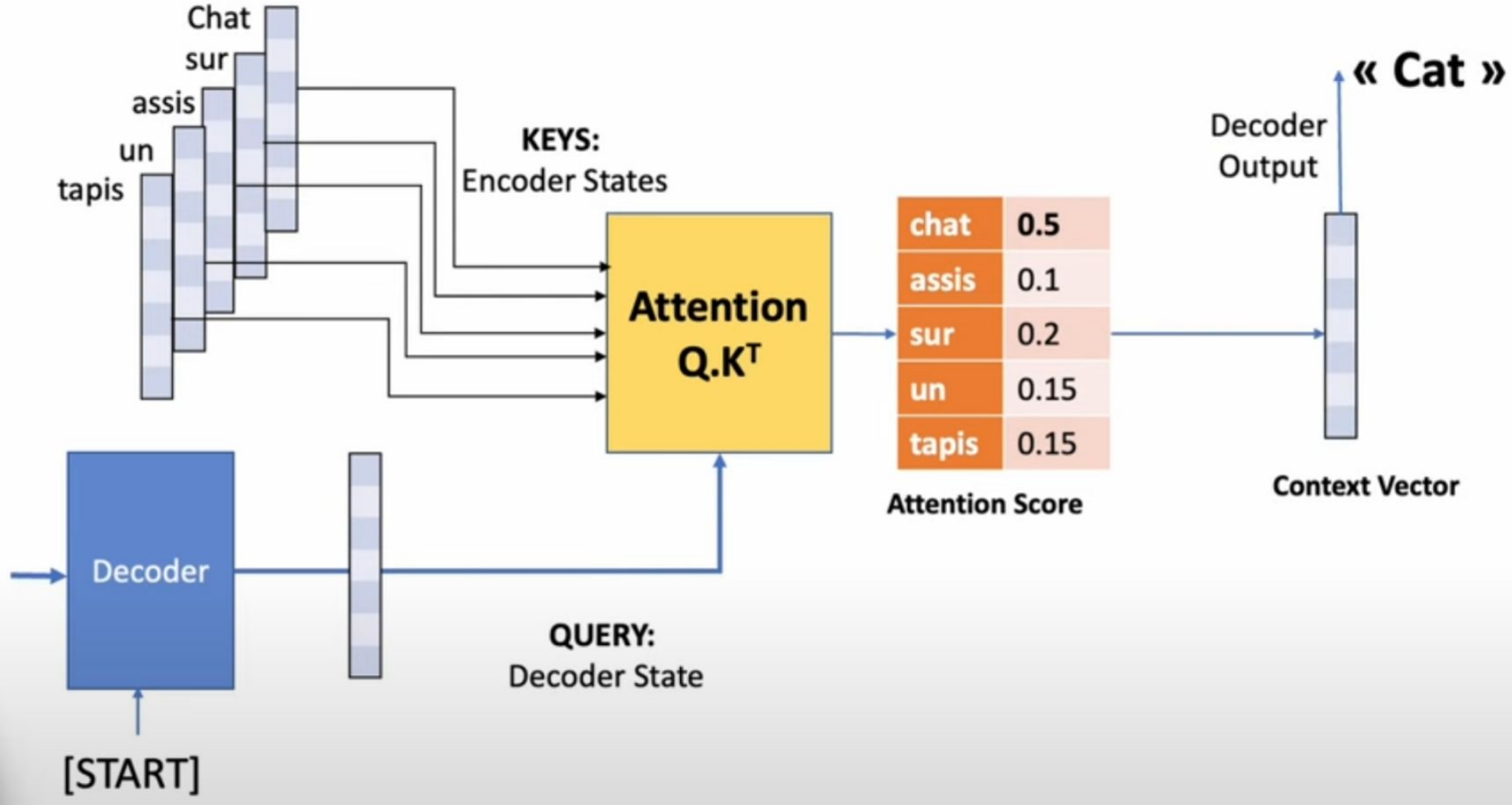


Intermediate States are KEYS

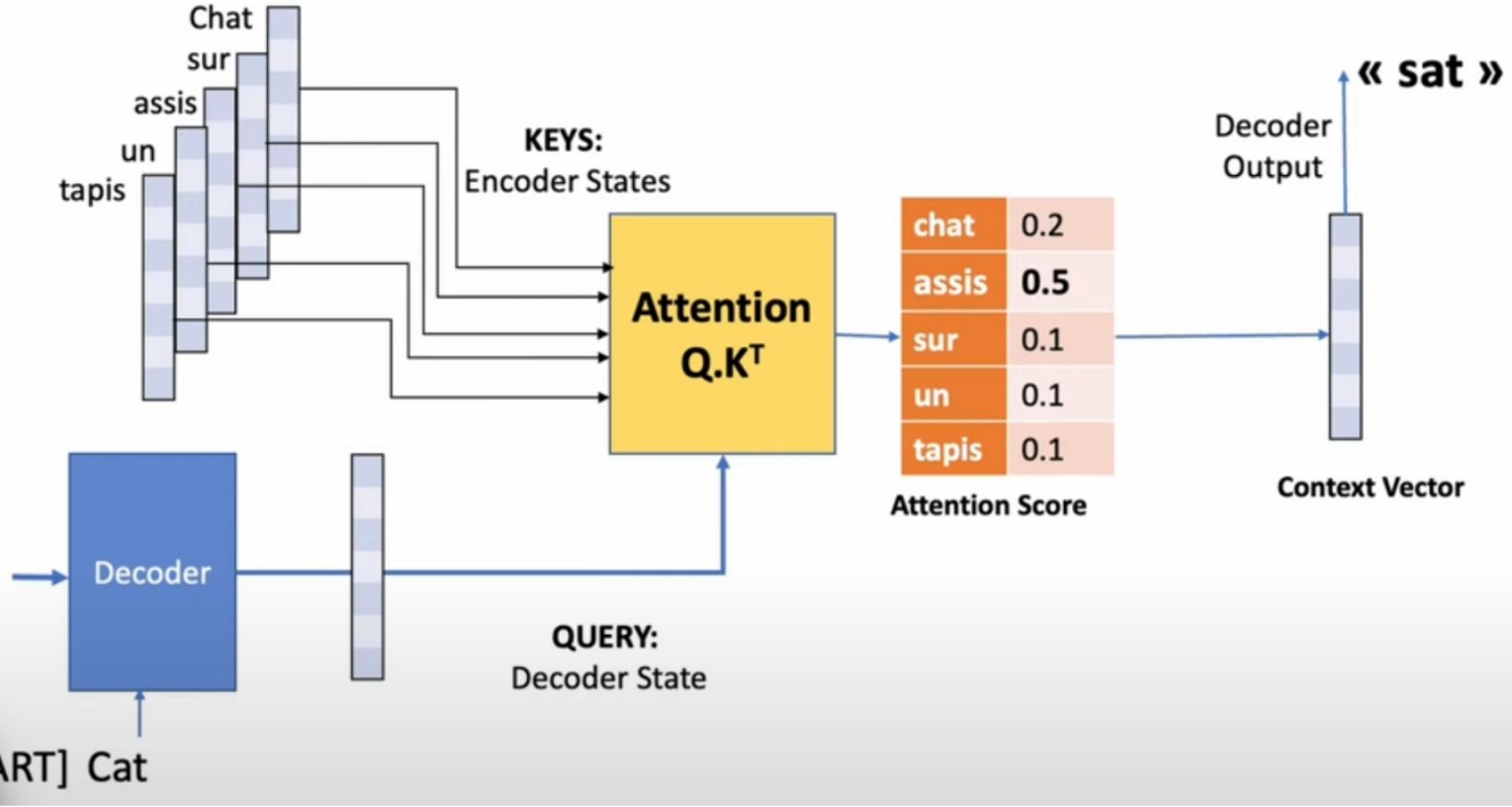
Attention to translate one word at a time



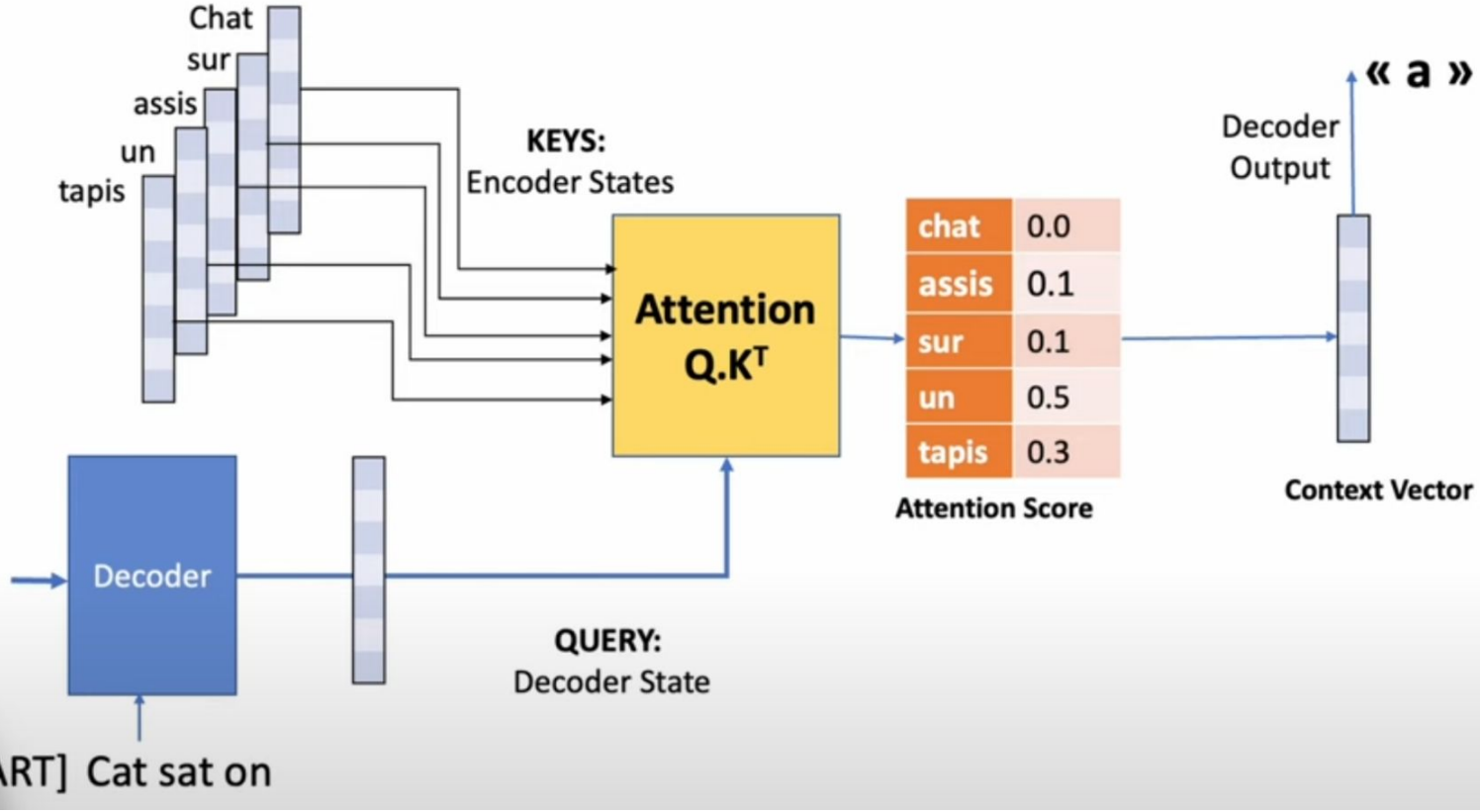
Attention to translate one word at a time



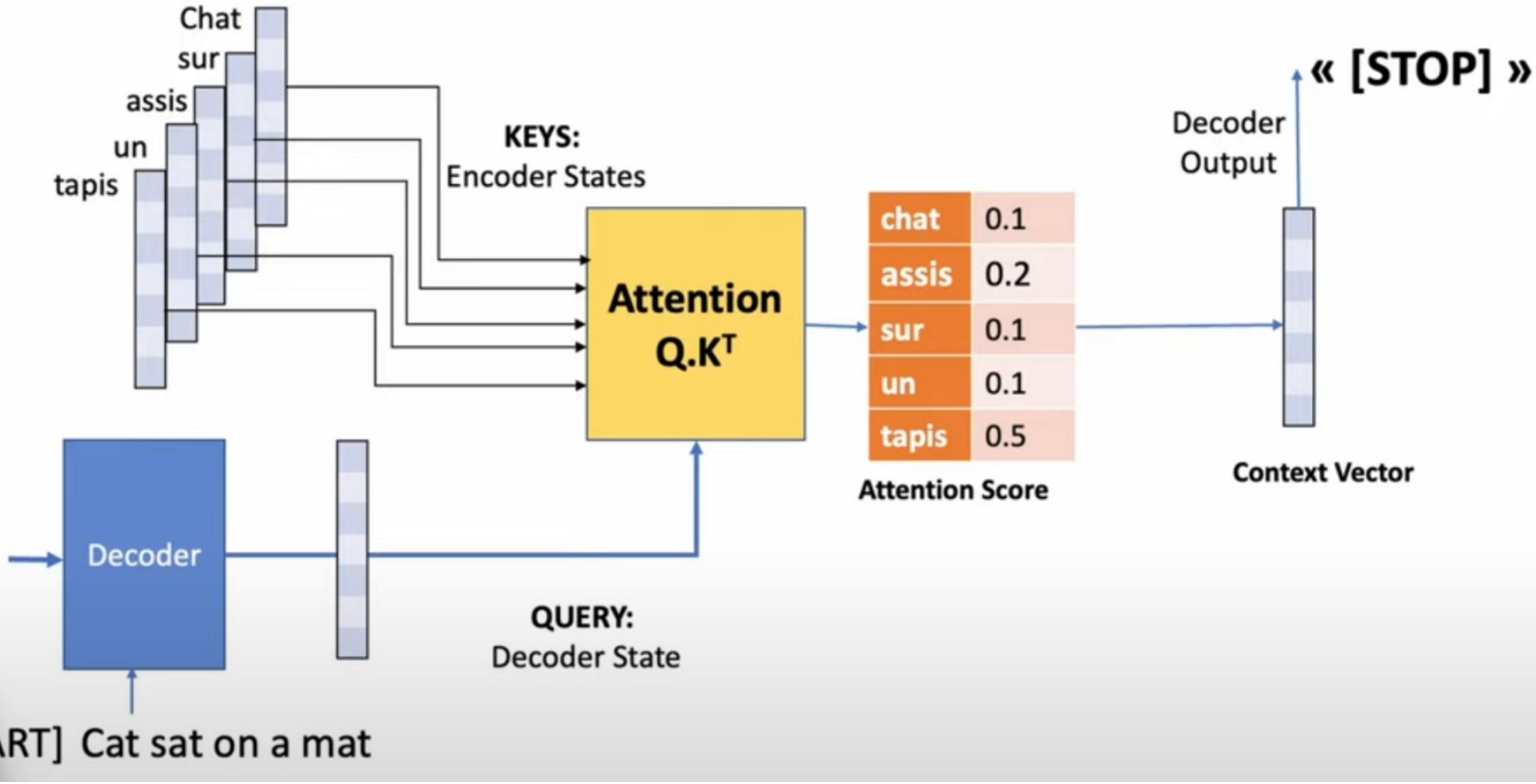
Attention to translate one word at a time



Attention to translate one word at a time

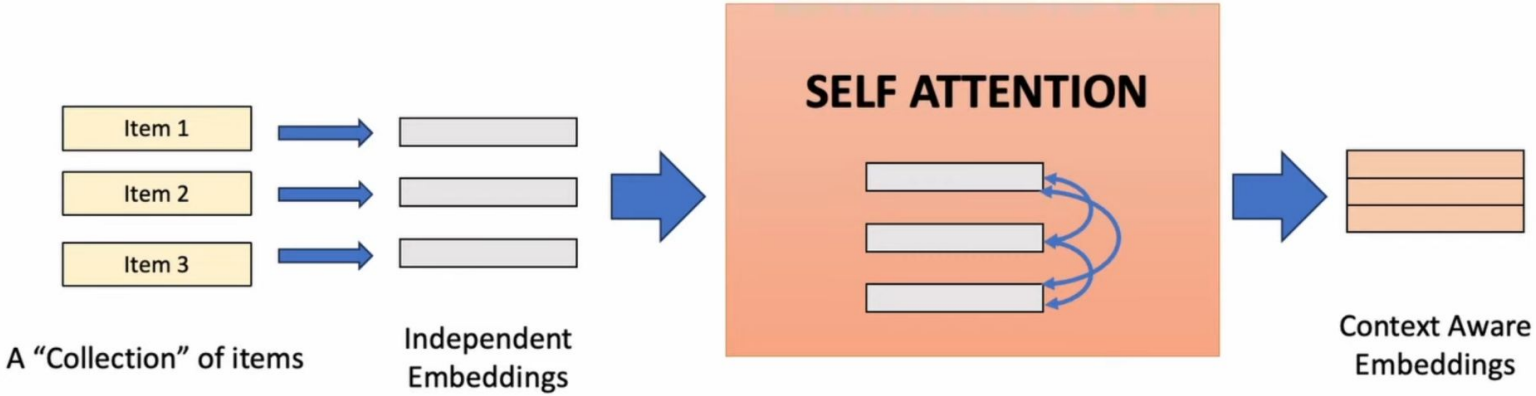


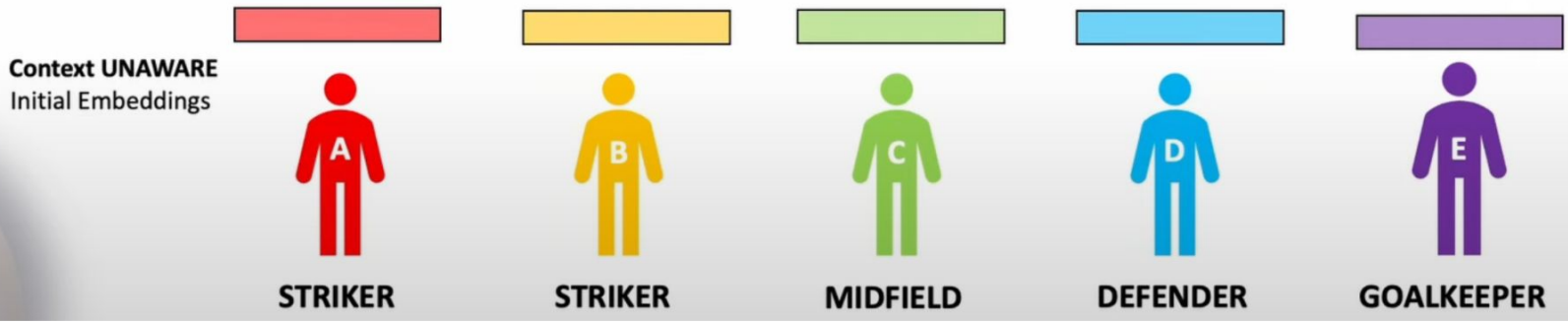
Attention to translate one word at a time



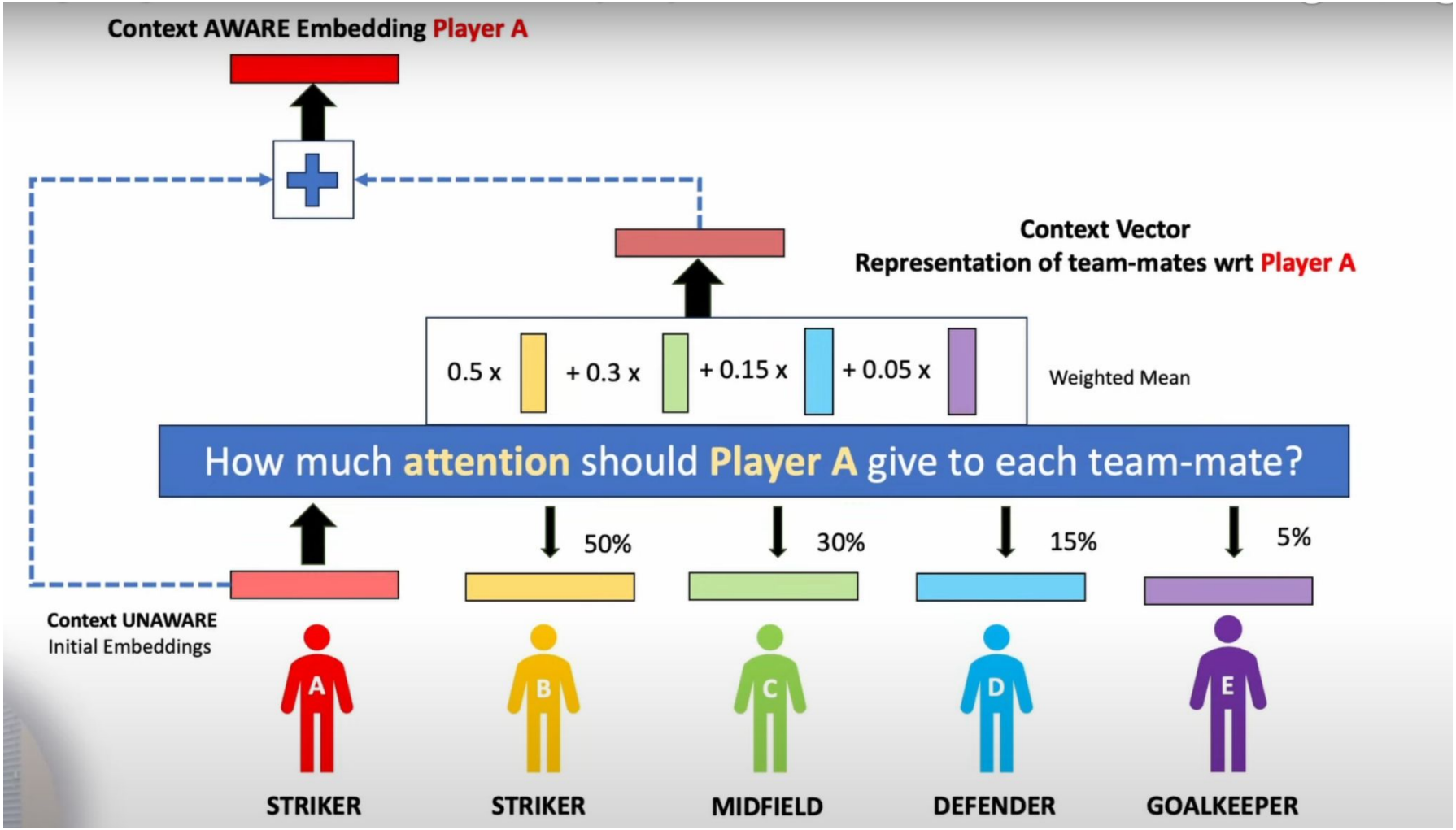
Self-Attention in Transformer Blocks [!]

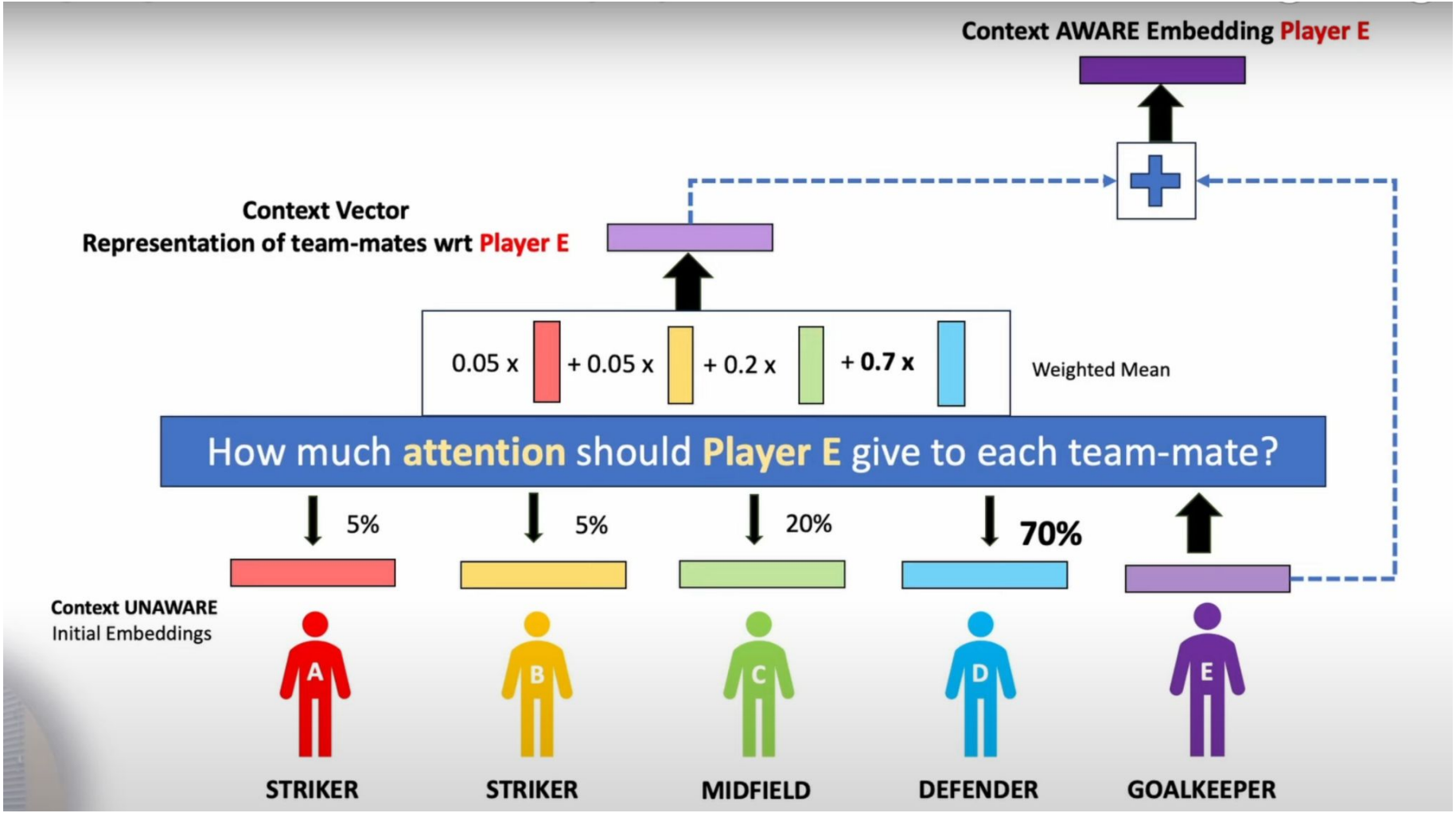
What SELF ATTENTION achieves




Context UNAWARE
Initial Embeddings
A
B
C
D
E
STRIKER
STRIKER
MIDFIELD
DEFENDER
GOALKEEPER


How much attention should Player A give to each team-mate?
50%
30%
15%
5%
Context UNAWARE
Initial Embeddings
A
B
C
D
E
STRIKER
STRIKER
MIDFIELD
DEFENDER
GOALKEEPER

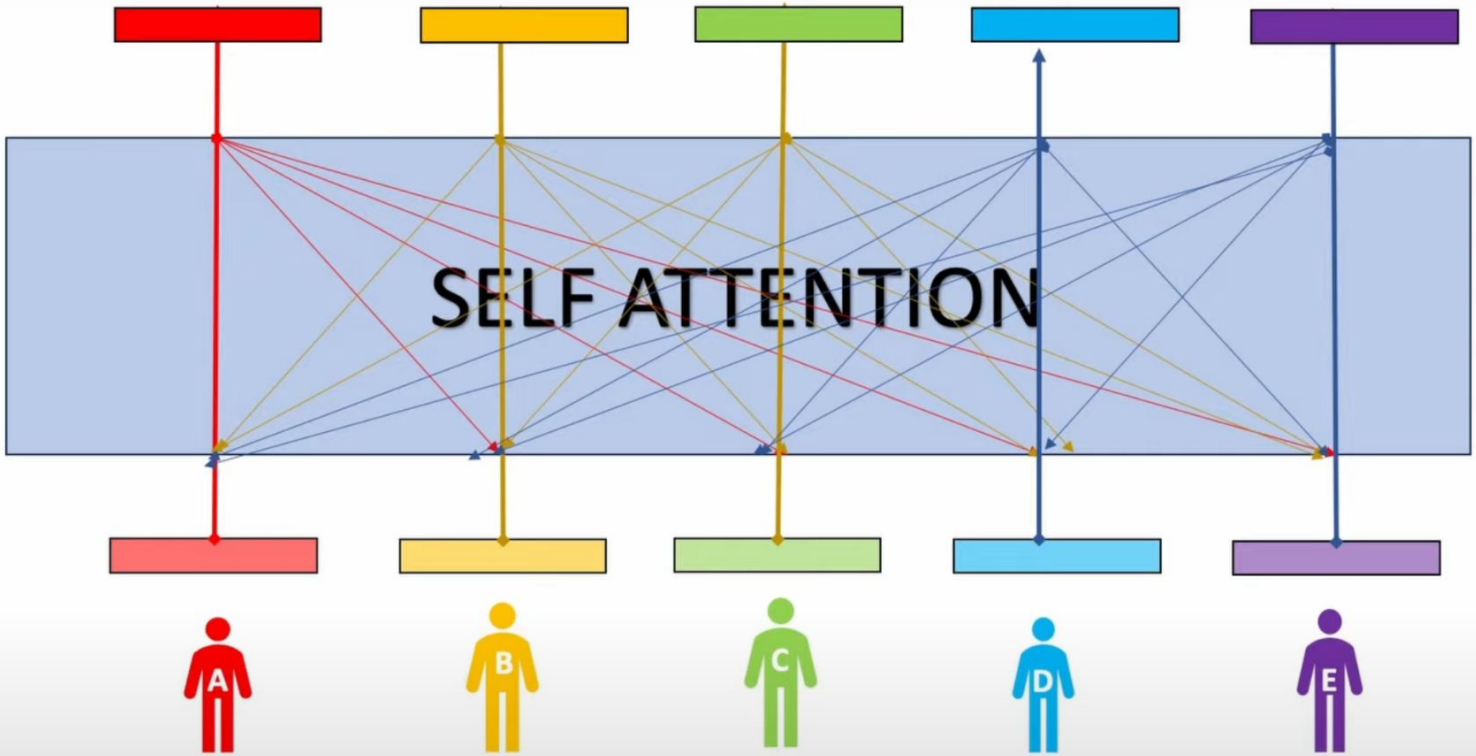

Context AWARE Embedding Player A
Context Vector
Representation of team-mates wrt Player A
0.5 x
+ 0.3 x
+ 0.15 x
+ 0.05 x
Weighted Mean
How much attention should Player A give to each team-mate?
50%
30%
15%
5%
Context UNAWARE
Initial Embeddings
A
B
C
D
E
STRIKER
STRIKER
MIDFIELD
DEFENDER
GOALKEEPER


Context AWARE Embedding Player E
Context Vector
Representation of team-mates wrt Player E
0.05 x
+ 0.05 x
+ 0.2 x
+ 0.7 x
Weighted Mean
How much attention should Player E give to each team-mate?
5%
5%
20%
70%
Context UNAWARE
Initial Embeddings
A
B
C
D
E
STRIKER
STRIKER
MIDFIELD
DEFENDER
GOALKEEPER

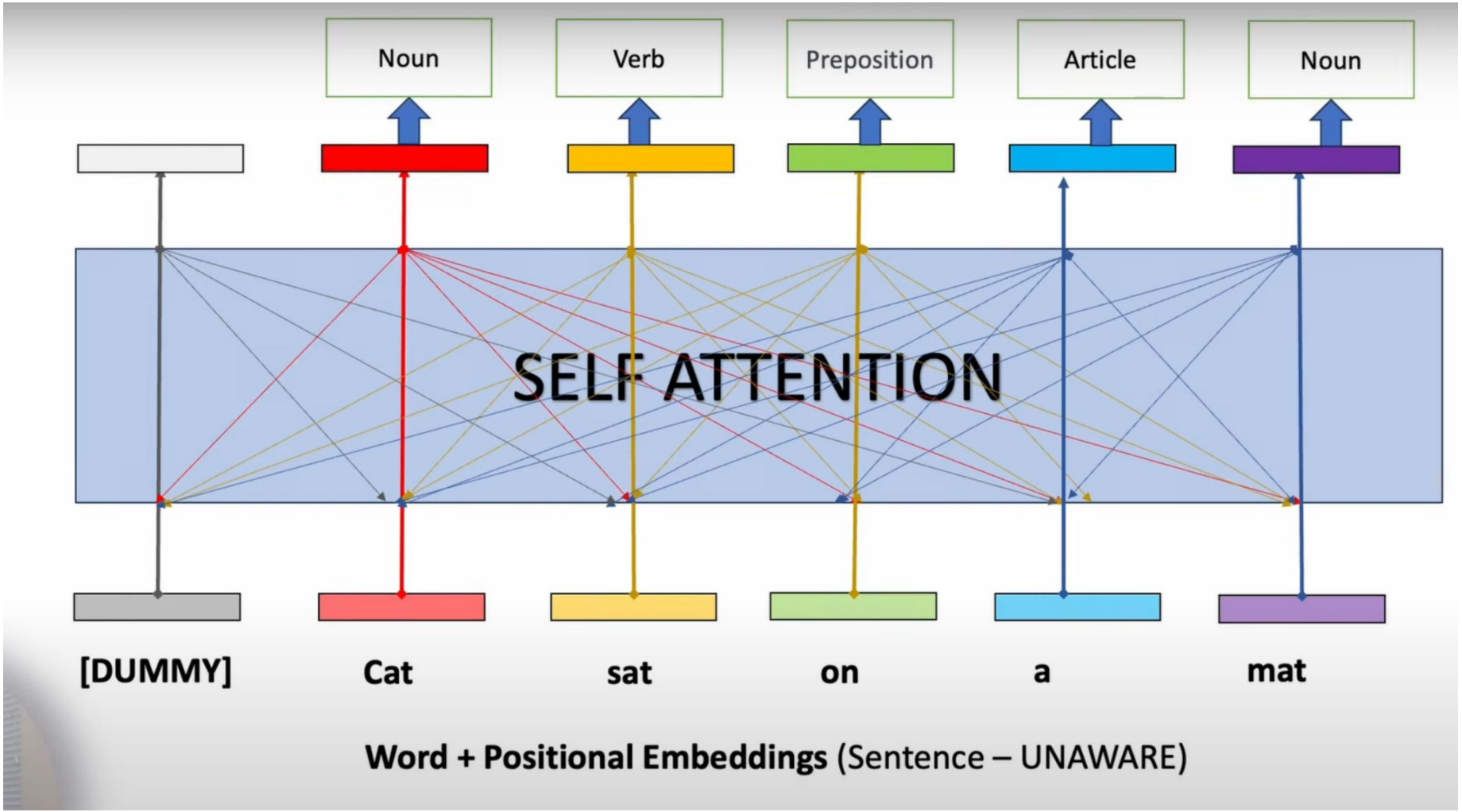
-	0.5	0.3	0.15	0.05
0.5	-	0.3	0.15	0.05
0.3	0.3	-	0.25	0.15
0.1	0.1	0.4	-	0.4
0.05	0.05	0.2	0.7	-

The Attention Matrix tells us how much attention each player gives to it's team mates.

TEAM AWARE PLAYER EMBEDDINGS



Independent Player Embeddings (Team – UNAWARE)


Noun
Verb
Preposition
Article
Noun
SELF ATTENTION
[DUMMY]
Cat
sat
on
a
mat
Word + Positional Embeddings (Sentence – UNAWARE)

Self-Attention: keys, queries, values from the same sequence

Let $\boldsymbol{w}_{1:n}$ be a sequence of words in vocabulary V, like *Zuko made his uncle tea*.

For each $\boldsymbol{w}_i$, let $\boldsymbol{x}_i = E\boldsymbol{w}_i$, where $E \in \mathbb{R}^{d \times |V|}$ is an embedding matrix.

1. Transform each word embedding with weight matrices Q, K, V, each in $\mathbb{R}^{d \times d}$

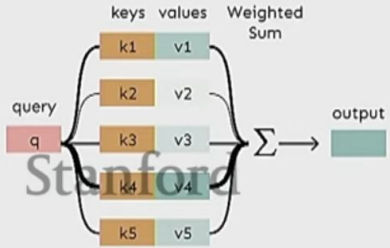
$\boldsymbol{q}_i = Q\boldsymbol{x}_i$ (queries) $\quad \boldsymbol{k}_i = K\boldsymbol{x}_i$ (keys) $\quad \boldsymbol{v}_i = V\boldsymbol{x}_i$ (values)

2. Compute pairwise similarities between keys and queries; normalize with softmax

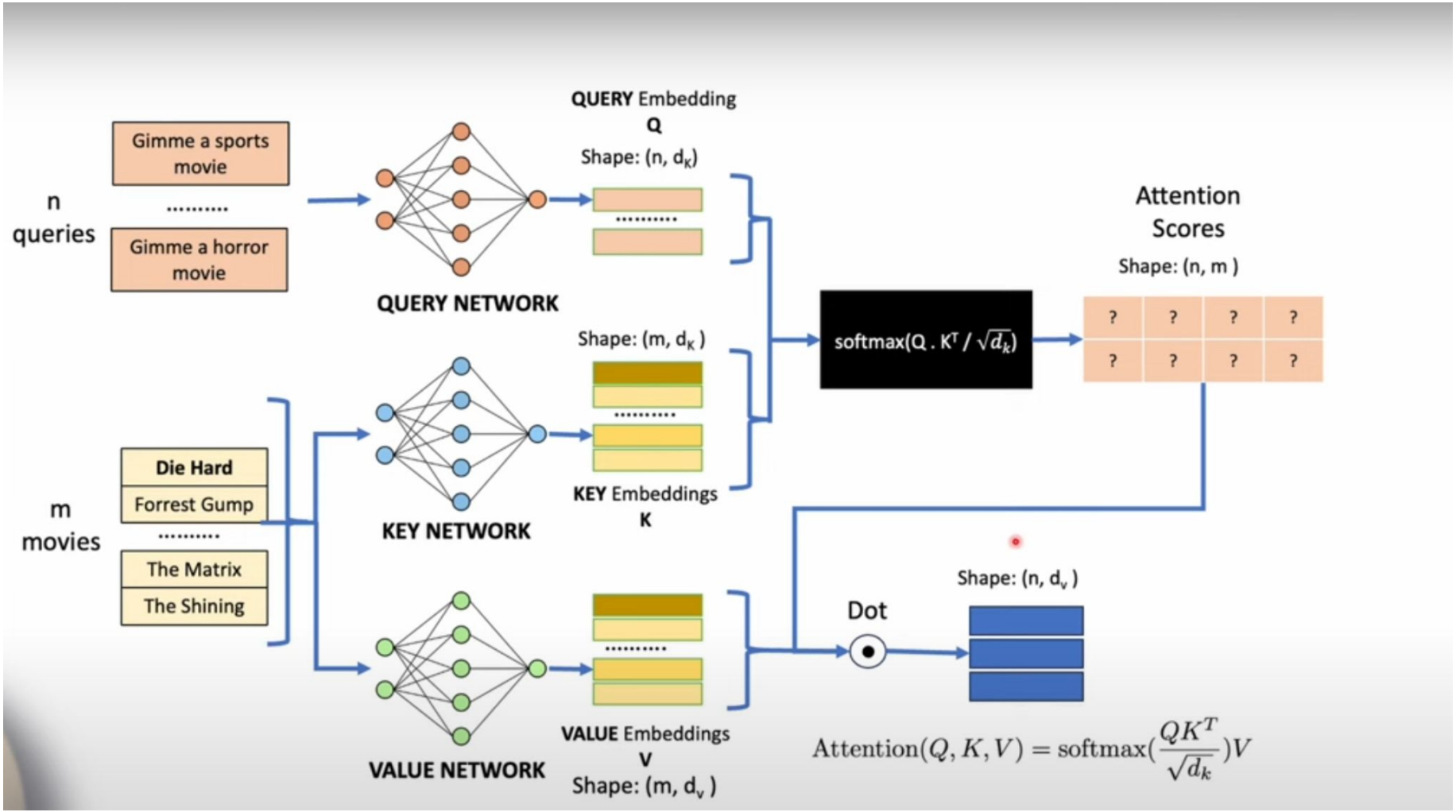
$$\boldsymbol{e}_{ij} = \boldsymbol{q}_i^\top \boldsymbol{k}_j \qquad \boldsymbol{\alpha}_{ij} = \frac{\exp(\boldsymbol{e}_{ij})}{\sum_{j'} \exp(\boldsymbol{e}_{ij'})}$$

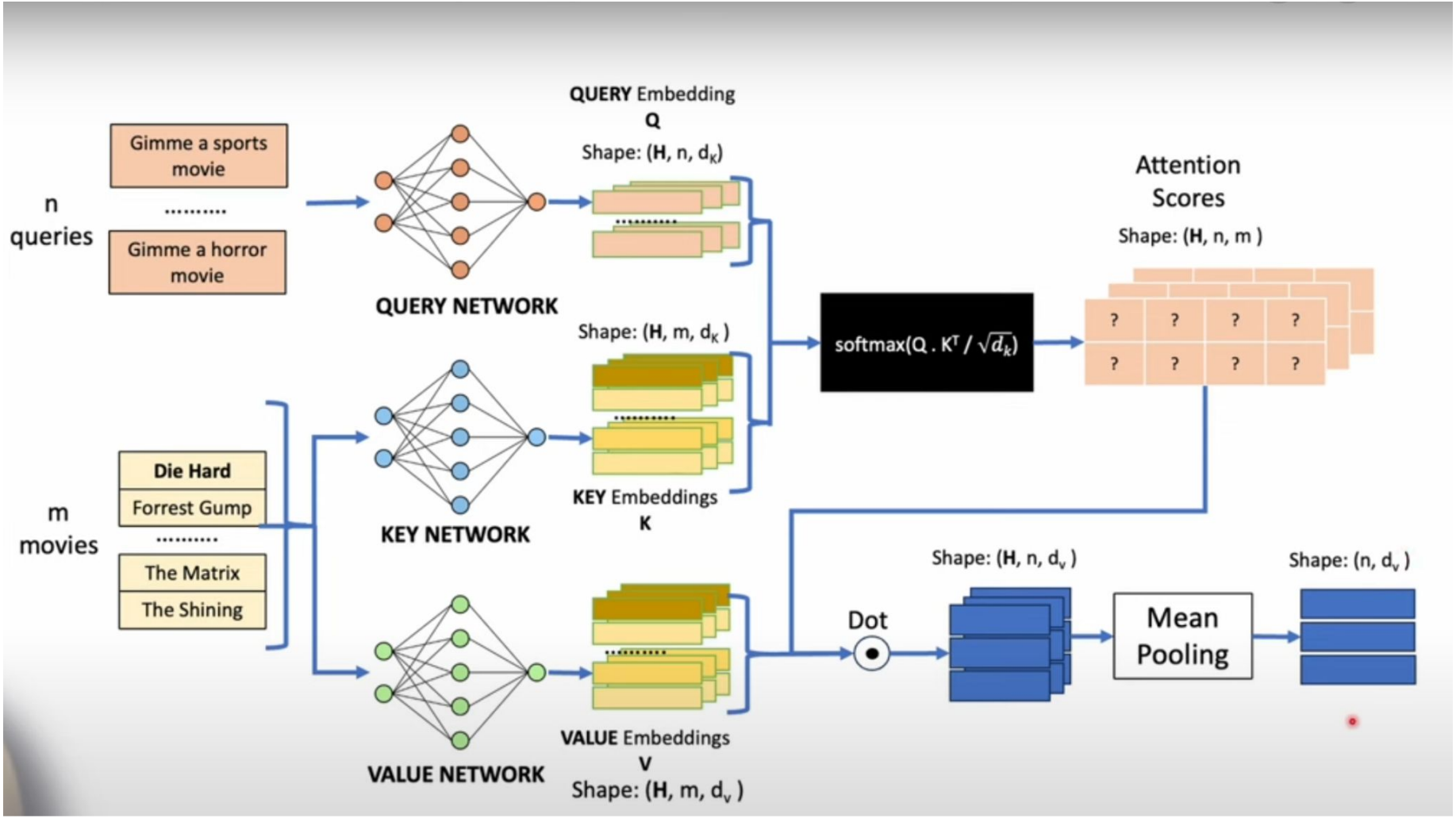
3. Compute output for each word as weighted sum of values

$$\boldsymbol{o}_i = \sum_j \boldsymbol{\alpha}_{ij} \boldsymbol{v}_j$$



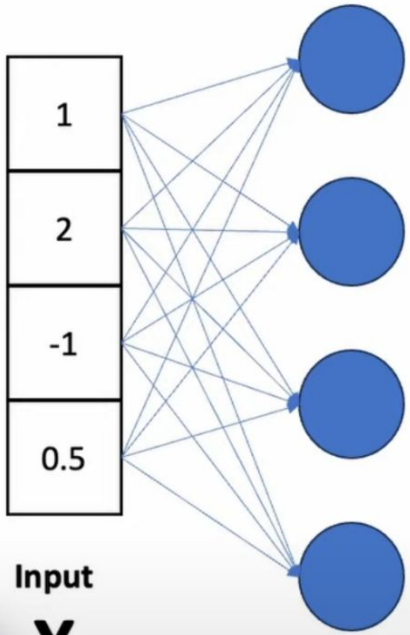
Now an example but that uses multiple heads but that is not Self-Attention!


n queries
Gimme a sports movie
.....
Gimme a horror movie
QUERY NETWORK
QUERY Embedding
Q
Shape: (n, d_k)
m movies
Die Hard
Forrest Gump
.....
The Matrix
The Shining
KEY NETWORK
Shape: (m, d_K)
KEY Embeddings
K
VALUE NETWORK
VALUE Embeddings
V
Shape: (m, d_v)
softmax(Q . K^T / $\sqrt{d_k}$)
Attention Scores
Shape: (n, m)
? ? ? ?
? ? ? ?
Dot
Shape: (n, d_v)
$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$


n
queries
Gimme a sports movie
.....
Gimme a horror movie
QUERY NETWORK
QUERY Embedding
Q
Shape: (H, n, d_K)
m
movies
Die Hard
Forrest Gump
.....
The Matrix
The Shining
KEY NETWORK
Shape: (H, m, d_K)
KEY Embeddings
K
VALUE NETWORK
VALUE Embeddings
V
Shape: (H, m, d_v)
Attention
Scores
Shape: (H, n, m)
softmax(Q . K^T / $\sqrt{d_k}$)
? ? ? ?
? ? ? ?
Dot
Shape: (H, n, d_v)
Mean
Pooling
Shape: (n, d_v)

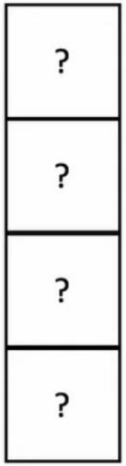
Parallelism between MLP + Transformer

Good old Fully Connected Layer



Input

X



Output

Y

X

1
2
-1
0.5

*

W (Weight Matrix)

0.1	0.2	-1	-0.6
2	1	0.5	-1
0	1	0.5	-1
2	-1	-2	1

\+

B (Bias)

-1
0
0.5
1

=

Y

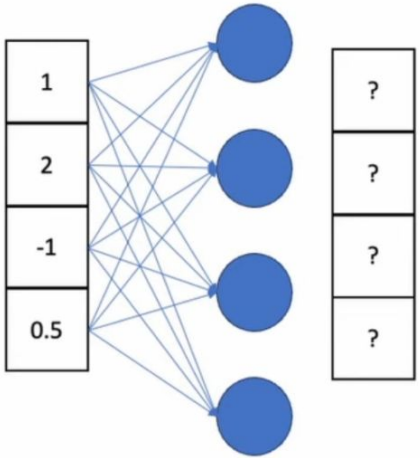
0.2
2.9
1.5
3.5

$$Y = WX + B$$

These are just random weights and biases for illustration

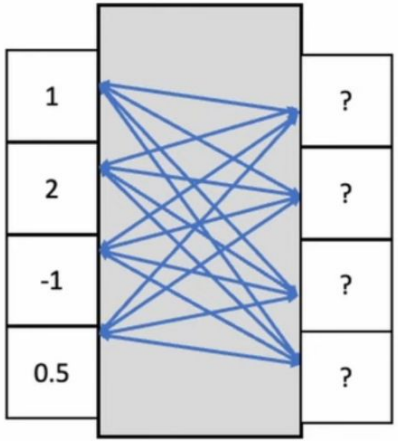
What is Self Attention doing with all of the extra operations?

Linear Layer

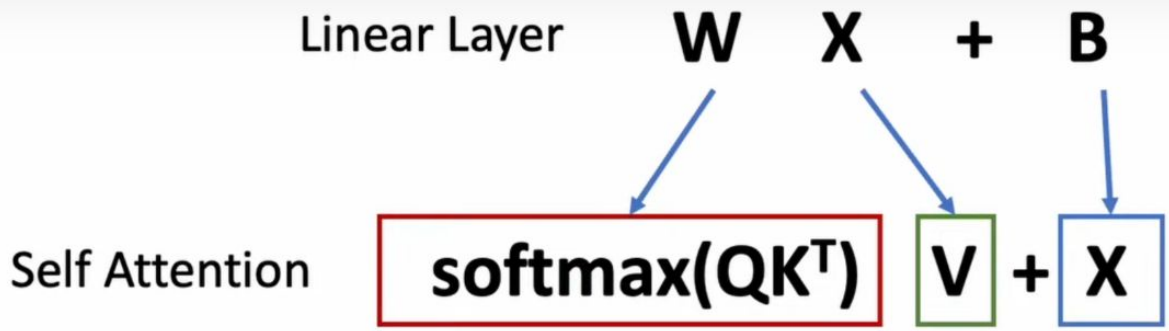


- Input and Output are shape n x d
- One Weight & Bias Matrix
- **WX + B**
- **$O(nd^2)$**

Self Attention Layer



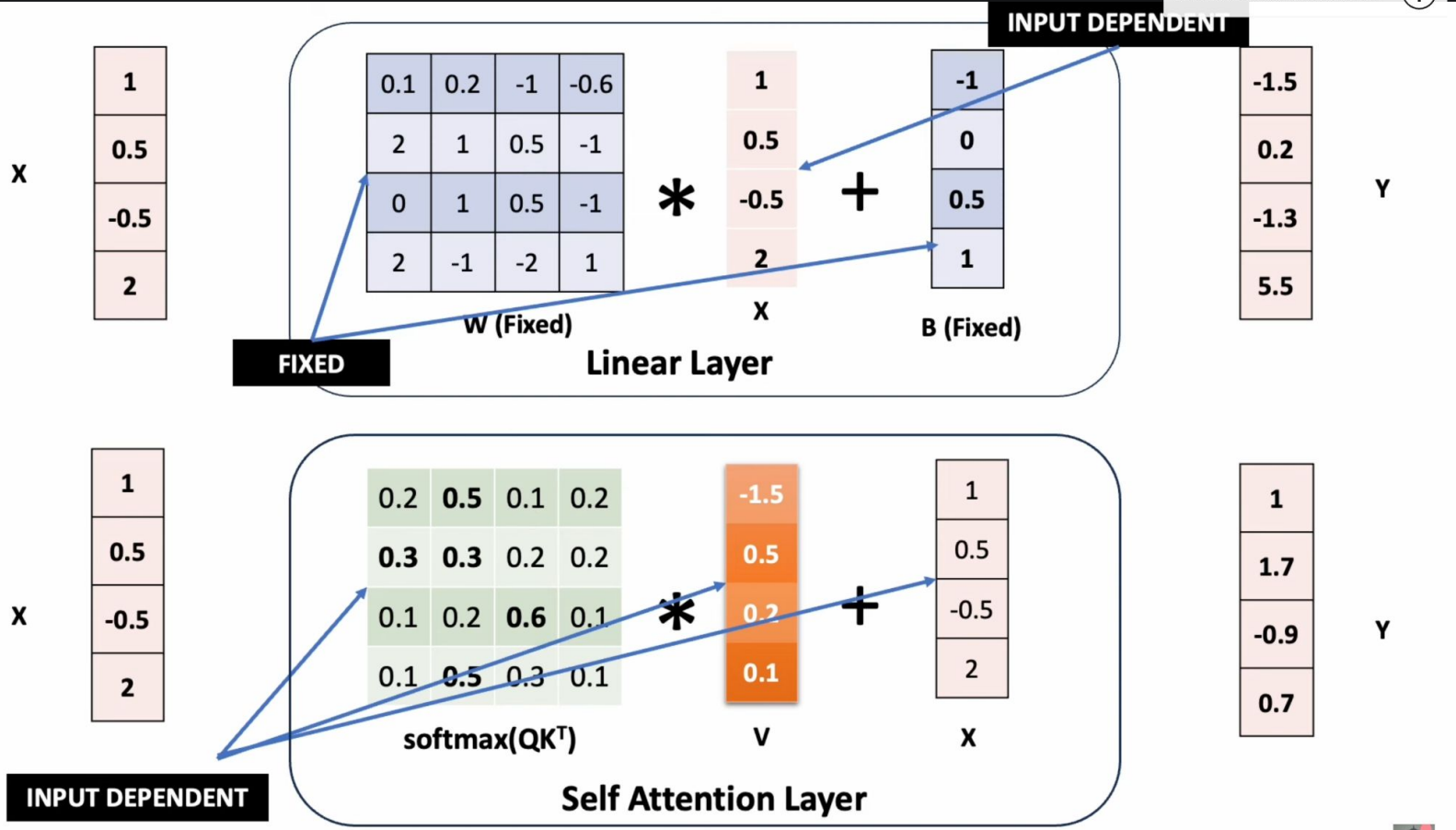
- Input and Output are shape n x d
- 3 Dense Layers (K, Q, V) & Softmax
- **$\text{softmax}(QK^T)\,V + X$**
- **$O(nd^2) + O(n^2d)$** (K/Q/V Nets + Dot Prods)

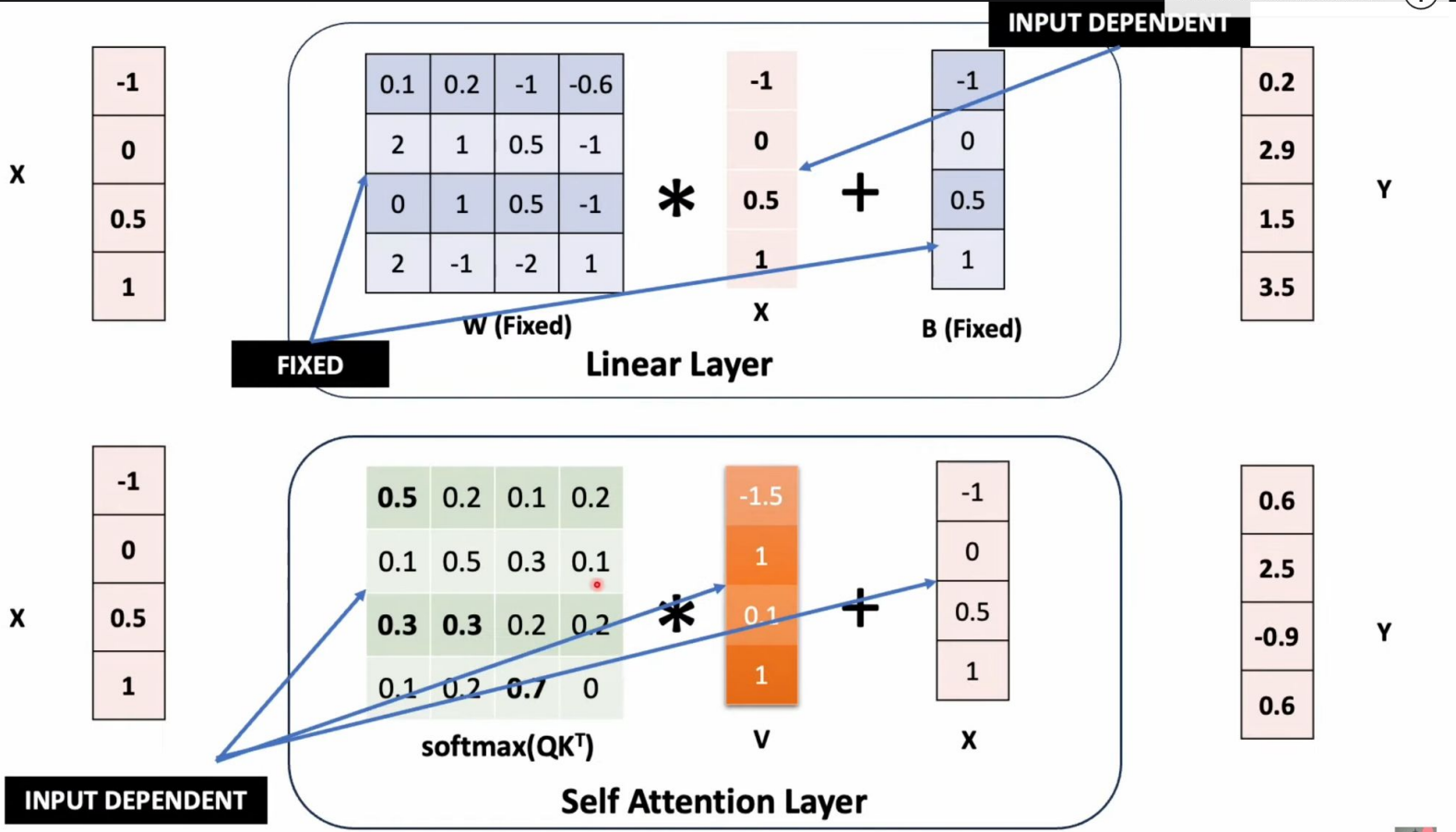


Caveat: **Q, K, and V are all obtained from X**
through the Query, Key, and Value Neural Nets

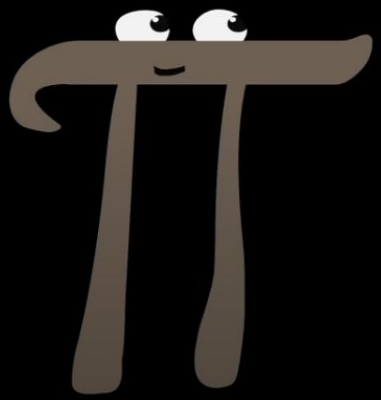
In other words, **W and B are both obtained from X**

In other words, W and B in Self Attention are Input Dependent


INPUT DEPENDENT
X
1
0.5
-0.5
2
0.1 0.2 -1 -0.6
2 1 0.5 -1
0 1 0.5 -1
2 -1 -2 1
W (Fixed)
*
1
0.5
-0.5
2
X
+
-1
0
0.5
1
B (Fixed)
-1.5
0.2
-1.3
5.5
Y
FIXED
Linear Layer
X
1
0.5
-0.5
2
0.2 0.5 0.1 0.2
0.3 0.3 0.2 0.2
0.1 0.2 0.6 0.1
0.1 0.5 0.3 0.1
softmax(QK^T)
*
-1.5
0.5
0.2
0.1
V
+
1
0.5
-0.5
2
X
1
1.7
-0.9
0.7
Y
INPUT DEPENDENT
Self Attention Layer

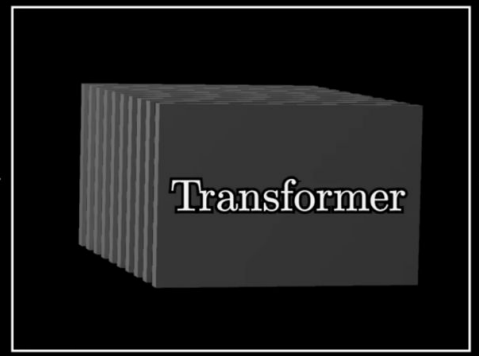

INPUT DEPENDENT
X
-1
0
0.5
1
0.1 0.2 -1 -0.6
2 1 0.5 -1
0 1 0.5 -1
2 -1 -2 1
W (Fixed)
*
-1
0
0.5
1
X
+
-1
0
0.5
1
B (Fixed)
0.2
2.9
1.5
3.5
Y
FIXED
Linear Layer
X
-1
0
0.5
1
0.5 0.2 0.1 0.2
0.1 0.5 0.3 0.1
0.3 0.3 0.2 0.2
0.1 0.2 0.7 0
softmax(QK^T)
*
-1.5
1
0.1
1
V
+
-1
0
0.5
1
X
0.6
2.5
-0.9
0.6
Y
INPUT DEPENDENT
Self Attention Layer

Generative Pre-trained Transformer



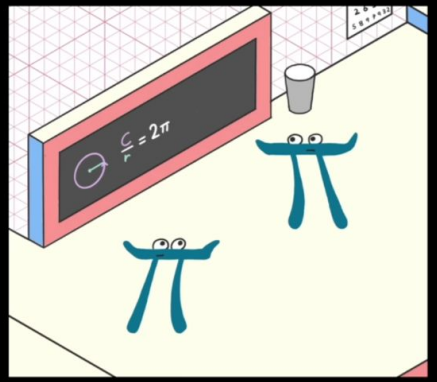
machine translation

Attention is all you need



注意力就是你所需要的一切

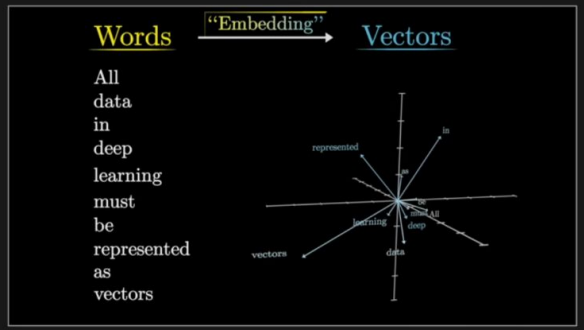
Behold, a wild pi creature,
foraging in its native _____



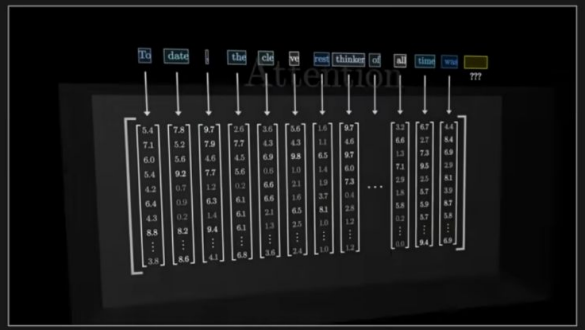
Transformer

land	22%
forest	9%
country	5%
habitat	4%
forests	4%
soil	4%
territory	2%
woods	2%
lands	1%
waters	1%
woodland	1%
grass	1%
⋮	

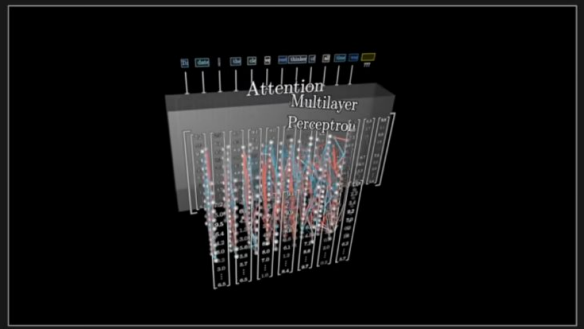
Embedding



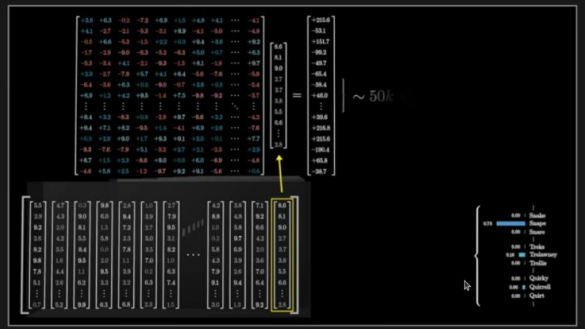
Attention



MLPs

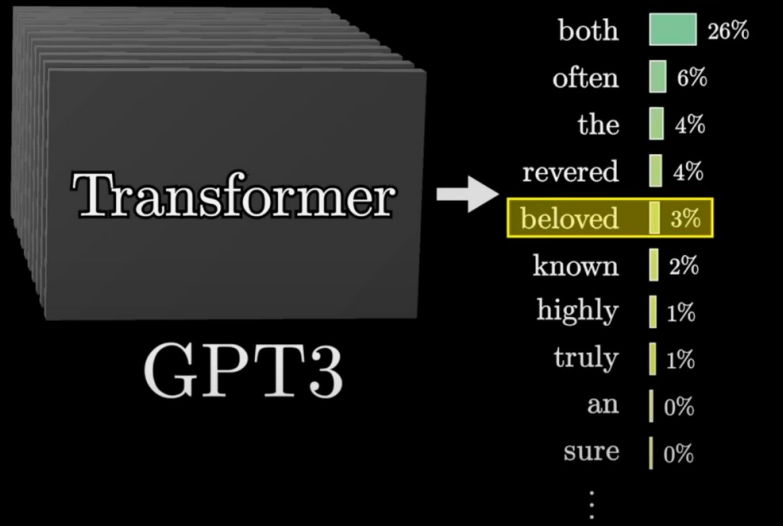


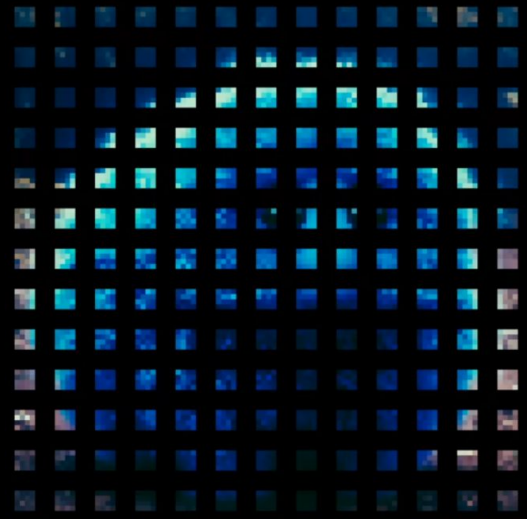
Unembedding



Embeddings

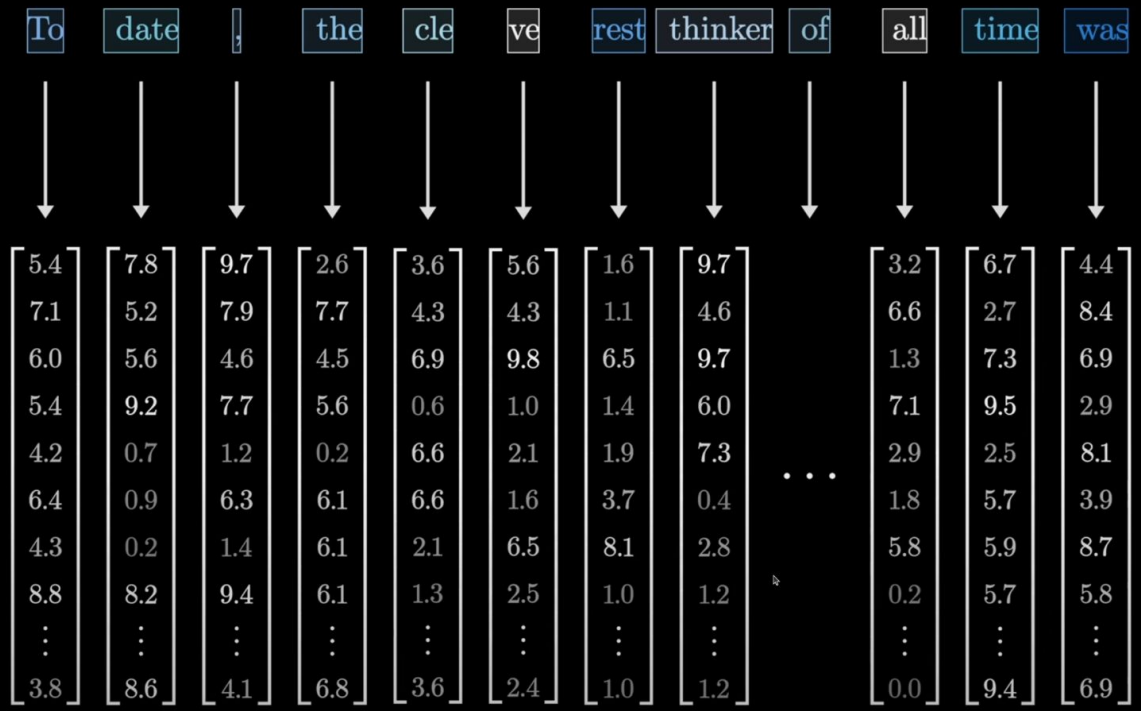
Behold, a wild pi creature, foraging in its native habitat of mathematical formulas and computer code! With its infinite digits and irrational tendencies, this strange creature is beloved

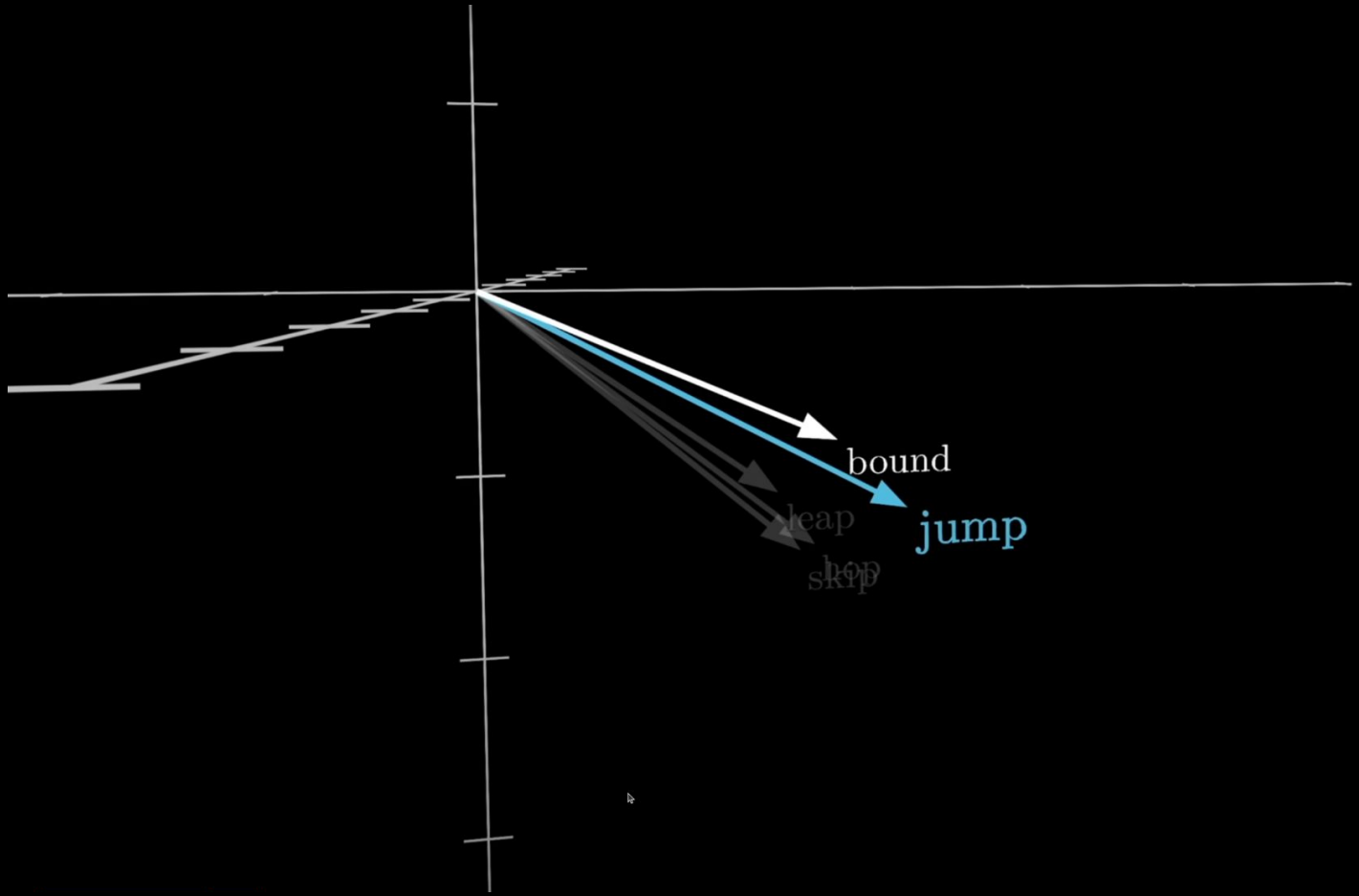


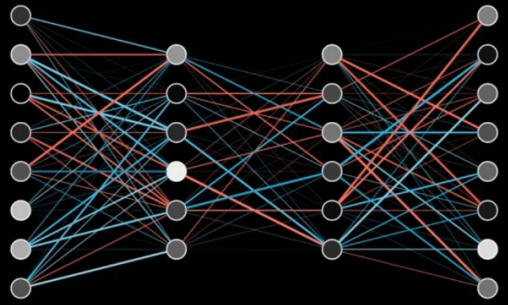


To date, the cleverest thinker of all time was

???


To
date
,
the
cle
ve
rest
thinker
of
all
time
was
???
5.4 7.1 6.0 5.4 4.2 6.4 4.3 8.8 3.8
7.8 5.2 5.6 9.2 0.7 0.9 0.2 8.2 8.6
9.7 7.9 4.6 7.7 1.2 6.3 1.4 9.4 4.1
2.6 7.7 4.5 5.6 0.2 6.1 6.1 6.1 6.8
3.6 4.3 6.9 0.6 6.6 6.6 2.1 1.3 3.6
5.6 4.3 9.8 1.0 2.1 1.6 6.5 2.5 2.4
1.6 1.1 6.5 1.4 1.9 3.7 8.1 1.0 1.0
9.7 4.6 9.7 6.0 7.3 0.4 2.8 1.2 1.2
3.2 6.6 1.3 7.1 2.9 1.8 5.8 0.2 0.0
6.7 2.7 7.3 9.5 2.5 5.7 5.9 5.7 9.4
4.4 8.4 6.9 2.9 8.1 3.9 8.7 5.8 6.9


bound
jump
leap
hop
skip



A machine learning model ...

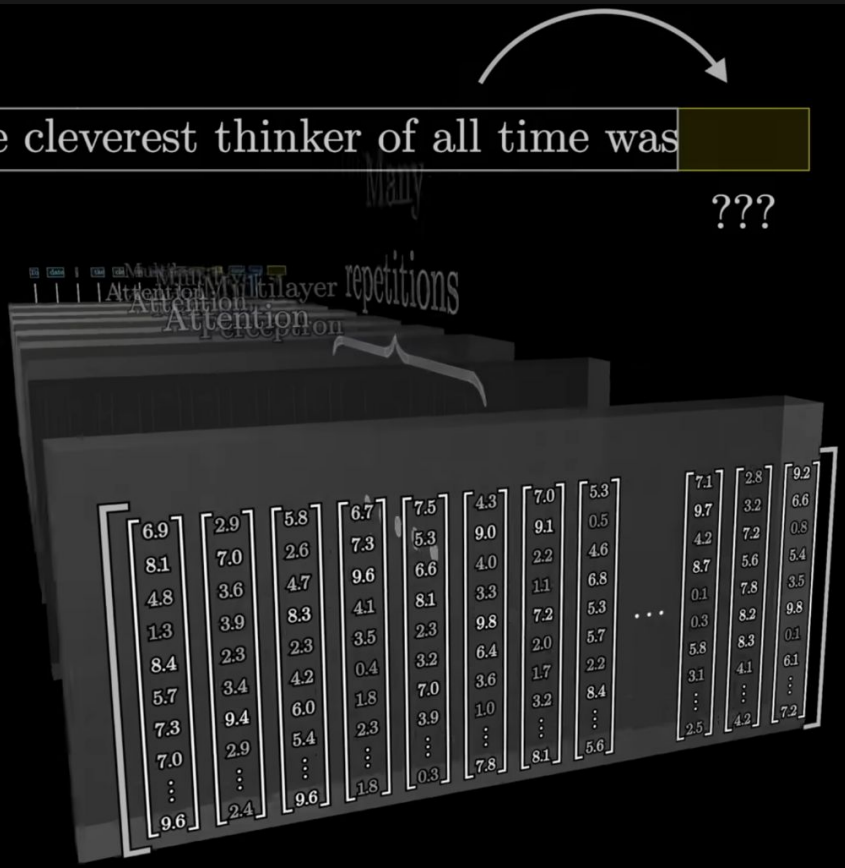
$$\begin{bmatrix} 3.6 \\ 5.6 \\ 4.3 \\ 9.8 \\ 1.0 \\ \vdots \\ 2.1 \end{bmatrix} \begin{bmatrix} 1.6 \\ 6.5 \\ 2.5 \\ 4.6 \\ 2.4 \\ \vdots \\ 1.6 \end{bmatrix} \begin{bmatrix} 1.1 \\ 6.5 \\ 1.4 \\ 1.9 \\ 3.7 \\ \vdots \\ 8.1 \end{bmatrix} \begin{bmatrix} 1.0 \\ 8.3 \\ 1.0 \\ 9.7 \\ 4.6 \\ \vdots \\ 9.7 \end{bmatrix}$$

A fashion model ...

$$\begin{bmatrix} 6.0 \\ 7.3 \\ 0.4 \\ 2.8 \\ 1.2 \\ \vdots \\ 2.9 \end{bmatrix} \begin{bmatrix} 1.2 \\ 3.1 \\ 4.1 \\ 0.6 \\ 6.9 \\ \vdots \\ 5.6 \end{bmatrix} \begin{bmatrix} 2.6 \\ 5.2 \\ 0.9 \\ 5.7 \\ 9.2 \\ \vdots \\ 3.2 \end{bmatrix}$$

To date, the cleverest thinker of all time was

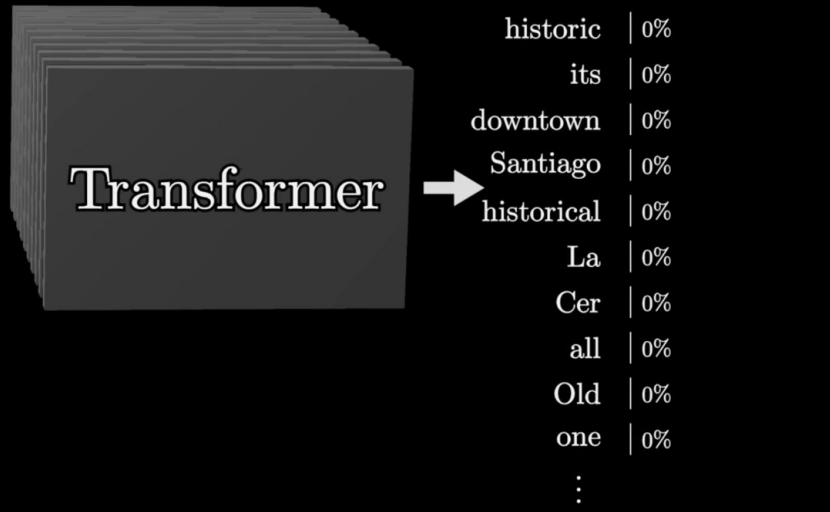
???

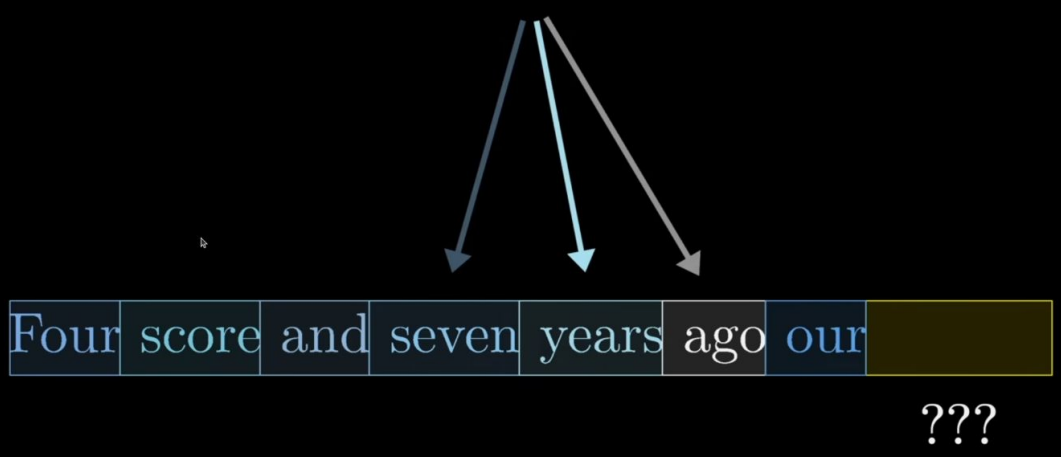


What follows is a conversation between a user and a helpful, very knowledgeable AI assistant.

User: Give me some ideas for what to do when visiting Santiago.

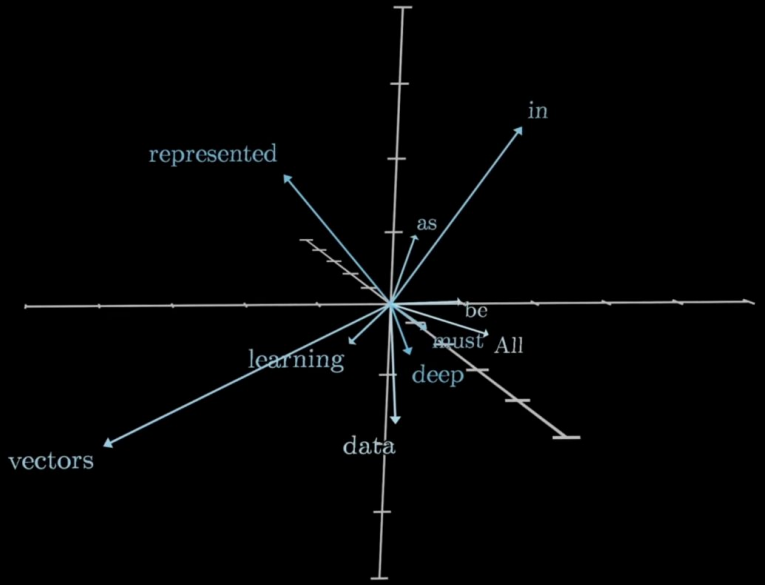
AI Assistant: Sure, there are plenty of things to do in Santiago! One option could be to take a walking tour of the _____




Tokens
Four
score
and
seven
years
ago
our
???

Words → "Embedding" → Vectors

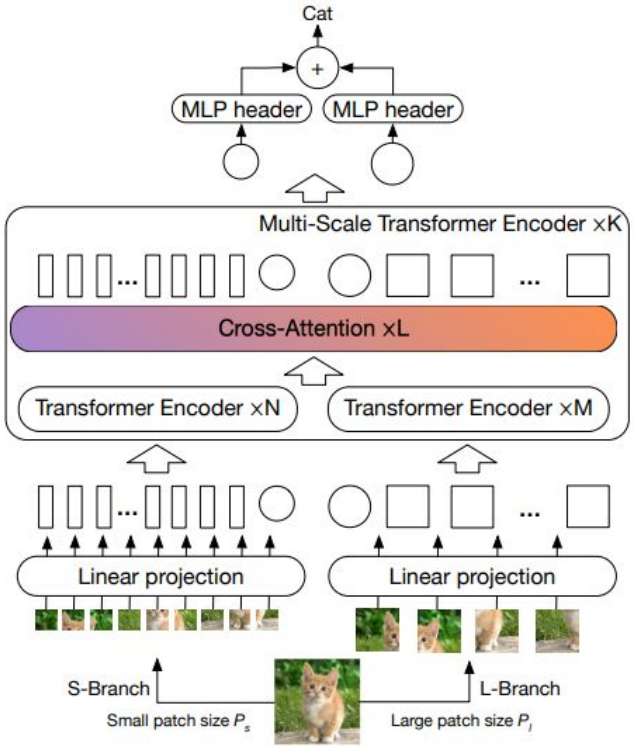
All
data
in
deep
learning
must
be
represented
as
vectors



Parenthesis : What are Tokens in Images ??

Parenthesis : What are Tokens in Images ??

Dual-branch vision transformer to extract multi-scale feature representations + Gramian-like local texture computation



CrossViT

Chen et. al., 2021

Attention

American shrew mole

One mole of carbon dioxide

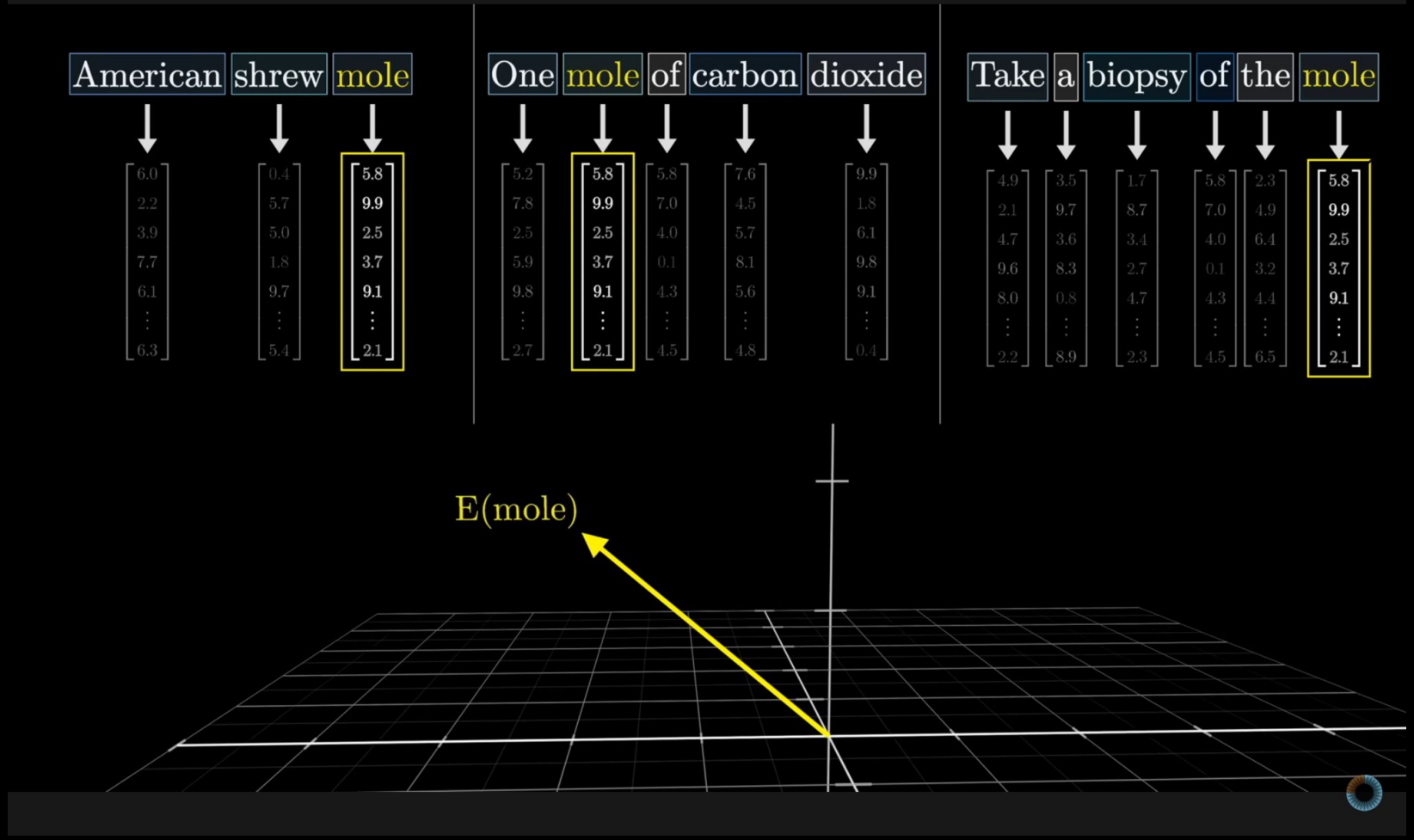
Take a biopsy of the mole

American shrew mole

6.02×10^{23}

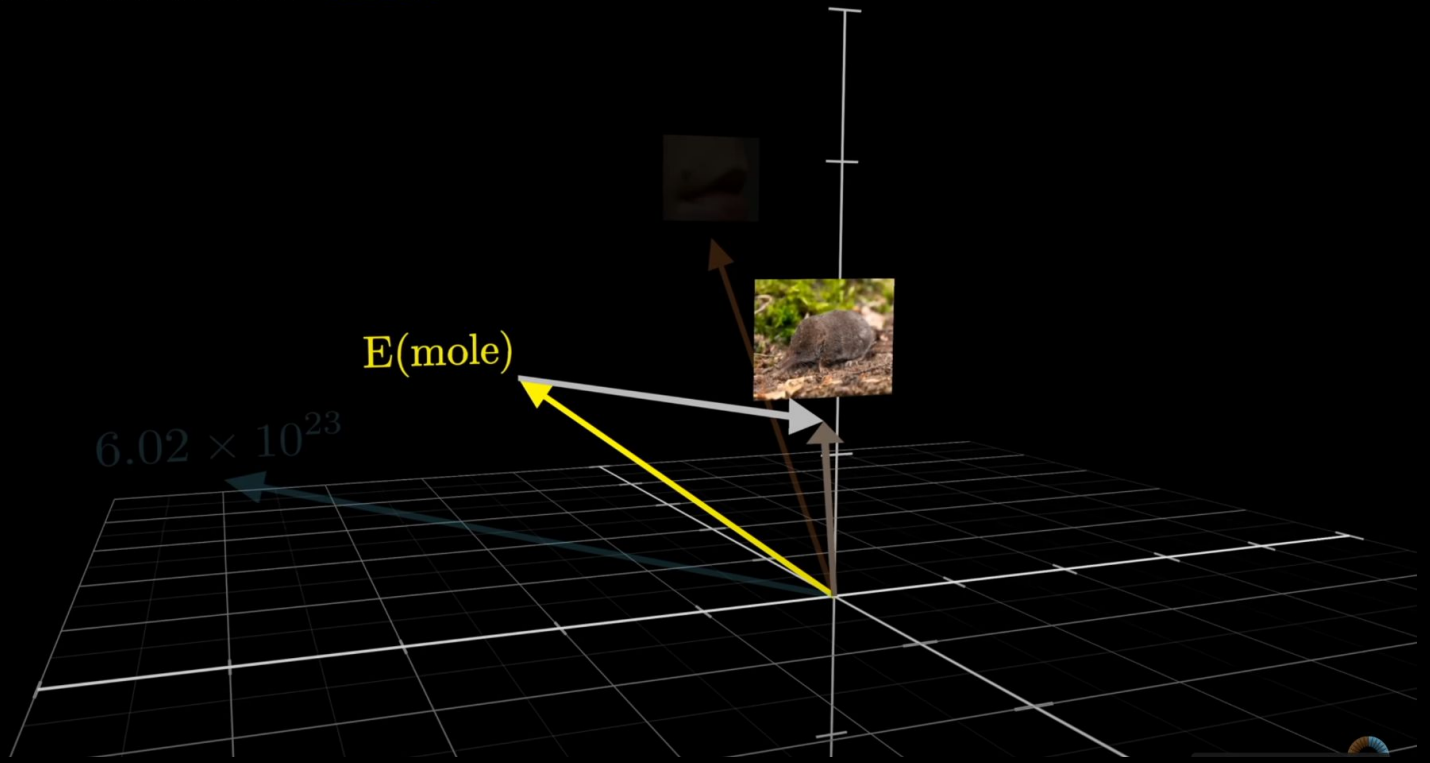
One mole of carbon dioxide

Take a biopsy of the mole

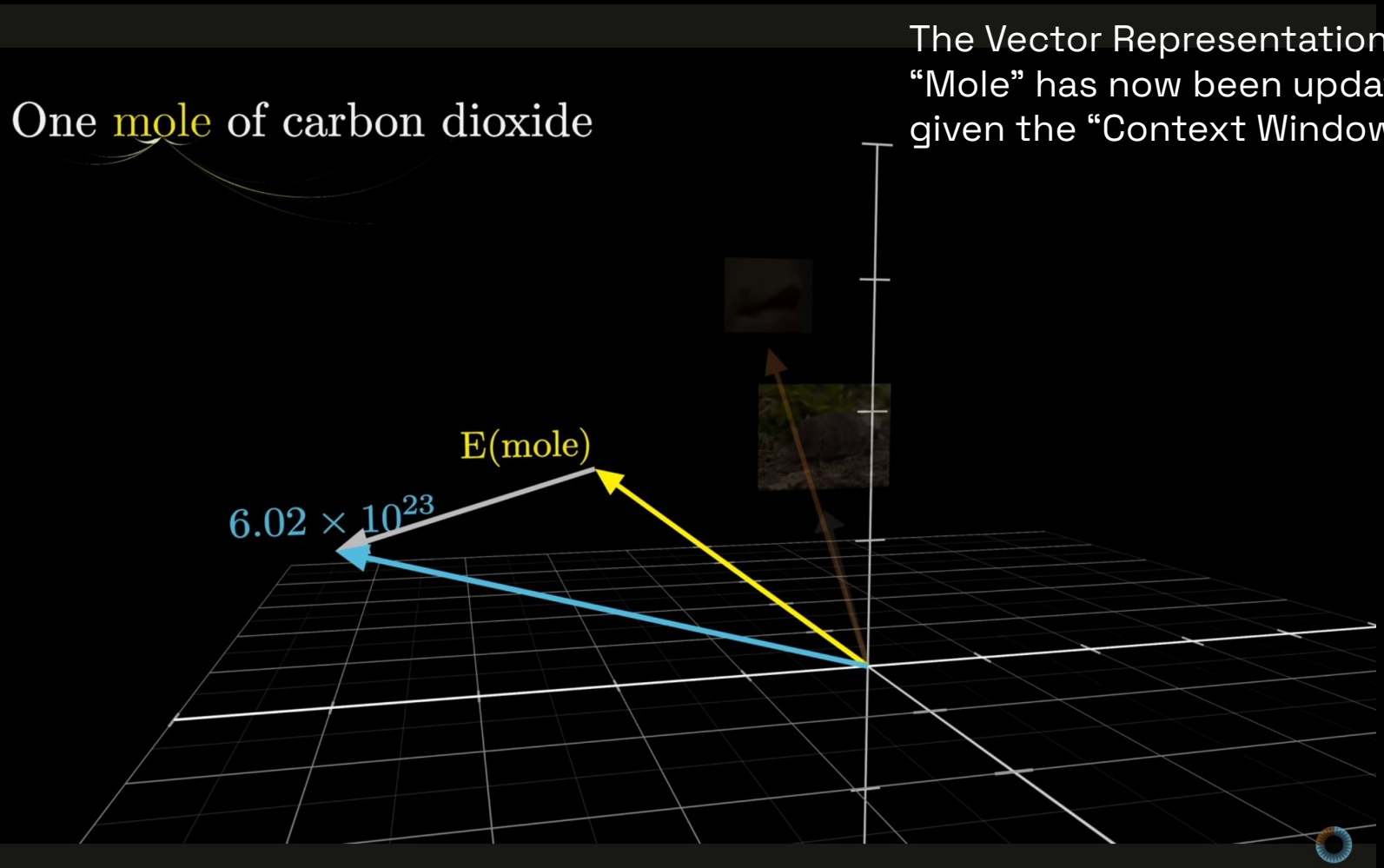

American shrew mole
One mole of carbon dioxide
Take a biopsy of the mole
E(mole)

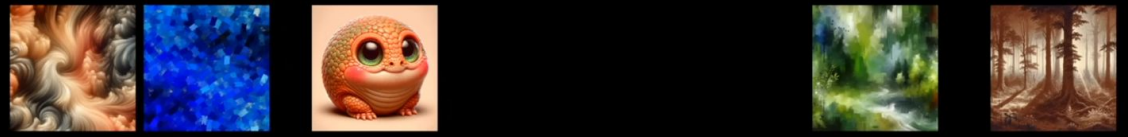
The Vector Representation of “Mole” has now been updated, given the “Context Window”

American shrew mole



The Vector Representation of “Mole” has now been updated, given the “Context Window”



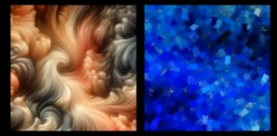


a	fluffy	blue	creature	roamed	the	verdant	forest
1	2	3	4	5	6	7	8

Position

12,288

5.7	3.2	9.2	8.2	3.6	5.6	5.7	8.7
3.9	4.1	3.2	0.0	4.3	4.3	2.2	5.8
4.6	0.7	6.6	6.7	6.9	9.8	9.4	8.7
5.9	6.9	1.3	2.7	0.6	1.0	4.4	6.9
3.6	5.6	7.1	7.3	6.6	2.1	8.4	7.2
6.3	2.6	2.9	9.5	6.6	1.6	6.9	5.0
4.3	5.2	1.8	2.5	2.1	6.5	2.9	9.5
5.4	0.9	5.8	5.7	1.3	2.5	8.1	6.4
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
3.4	5.7	0.2	5.9	3.6	2.4	3.9	4.2

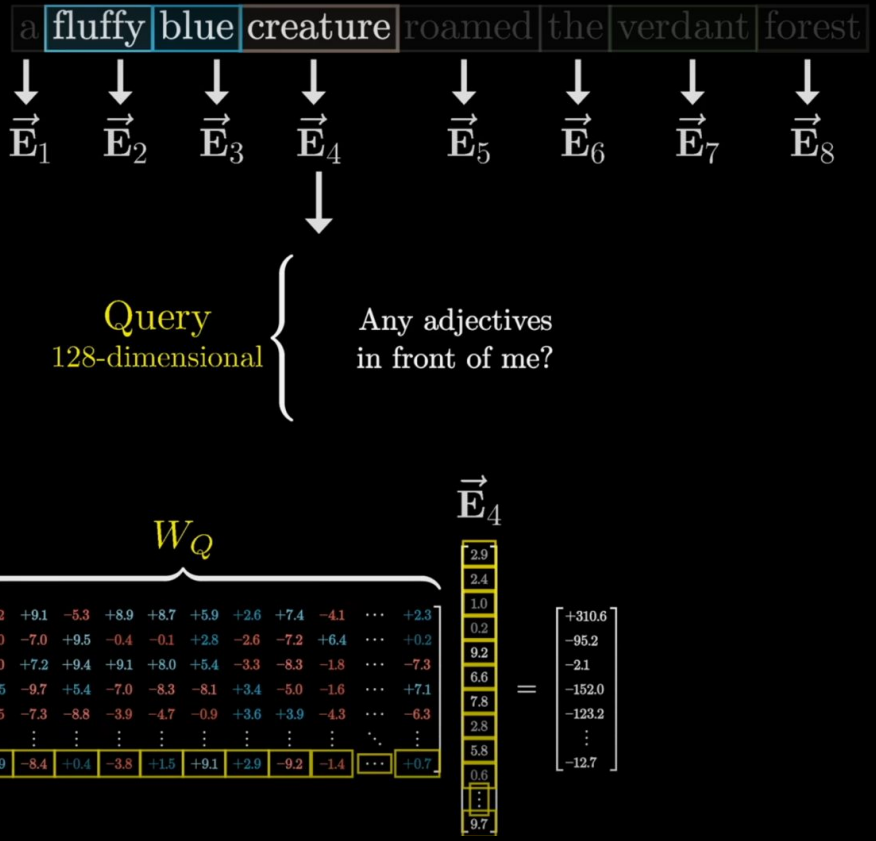


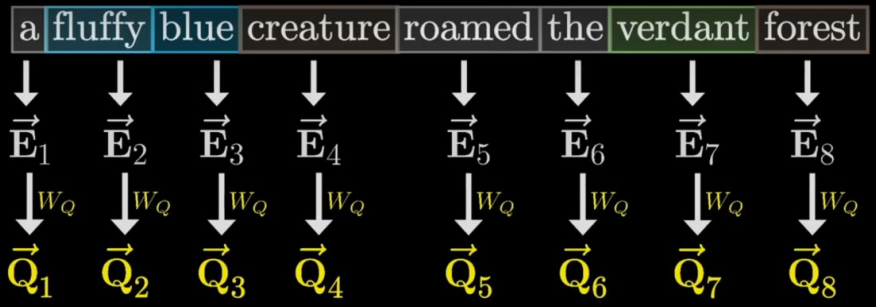


a	fluffy	blue	creature	roamed	the	verdant	forest

$\vec{E}_1$ $\vec{E}_2$ $\vec{E}_3$ $\vec{E}_4$ $\vec{E}_5$ $\vec{E}_6$ $\vec{E}_7$ $\vec{E}_8$

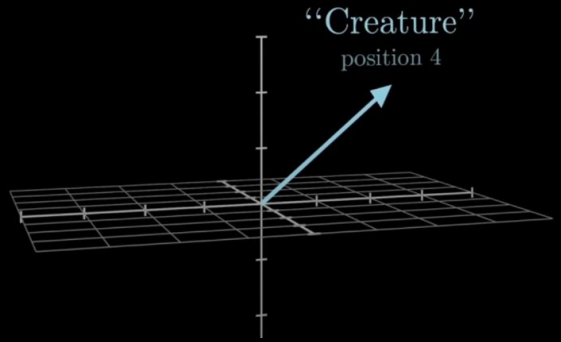
$\vec{E}'_1$ $\vec{E}'_2$ $\vec{E}'_3$ $\vec{E}'_4$ $\vec{E}'_5$ $\vec{E}'_6$ $\vec{E}'_7$ $\vec{E}'_8$

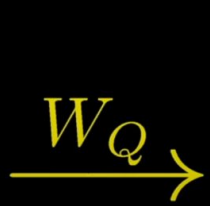


a fluffy blue creature roamed the verdant forest
$\vec{E}_1$ $\vec{E}_2$ $\vec{E}_3$ $\vec{E}_4$ $\vec{E}_5$ $\vec{E}_6$ $\vec{E}_7$ $\vec{E}_8$
Query
128-dimensional
Any adjectives in front of me?
W_Q
$\vec{E}_4$
+7.5 −3.2 +9.1 −5.3 +8.9 +8.7 +5.9 +2.6 +7.4 −4.1 ⋯ +2.3
−9.6 −3.0 −7.0 +9.5 −0.4 −0.1 +2.8 −2.6 −7.2 +6.4 ⋯ +0.2
−5.5 −8.0 +7.2 +9.4 +9.1 +8.0 +5.4 −3.3 −8.3 −1.8 ⋯ −7.3
−8.8 +4.5 −9.7 +5.4 −7.0 −8.3 −8.1 +3.4 −5.0 −1.6 ⋯ +7.1
+4.5 −4.5 −7.3 −8.8 −3.9 −4.7 −0.9 +3.6 +3.9 −4.3 ⋯ −6.3
−9.0 +5.9 −8.4 +0.4 −3.8 +1.5 +9.1 +2.9 −9.2 −1.4 ⋯ +0.7
2.9 2.4 1.0 0.2 9.2 6.6 7.8 2.8 5.8 0.6 9.7
=
+310.6 −95.2 −2.1 −152.0 −123.2 −12.7



Embedding space

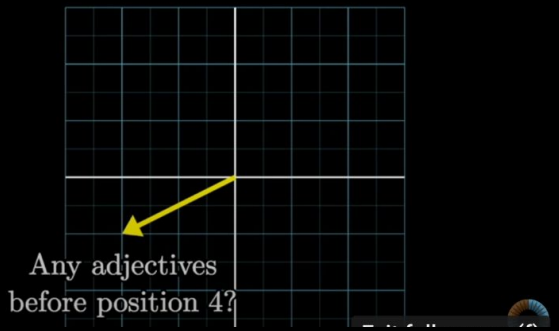
12,288-dimensional

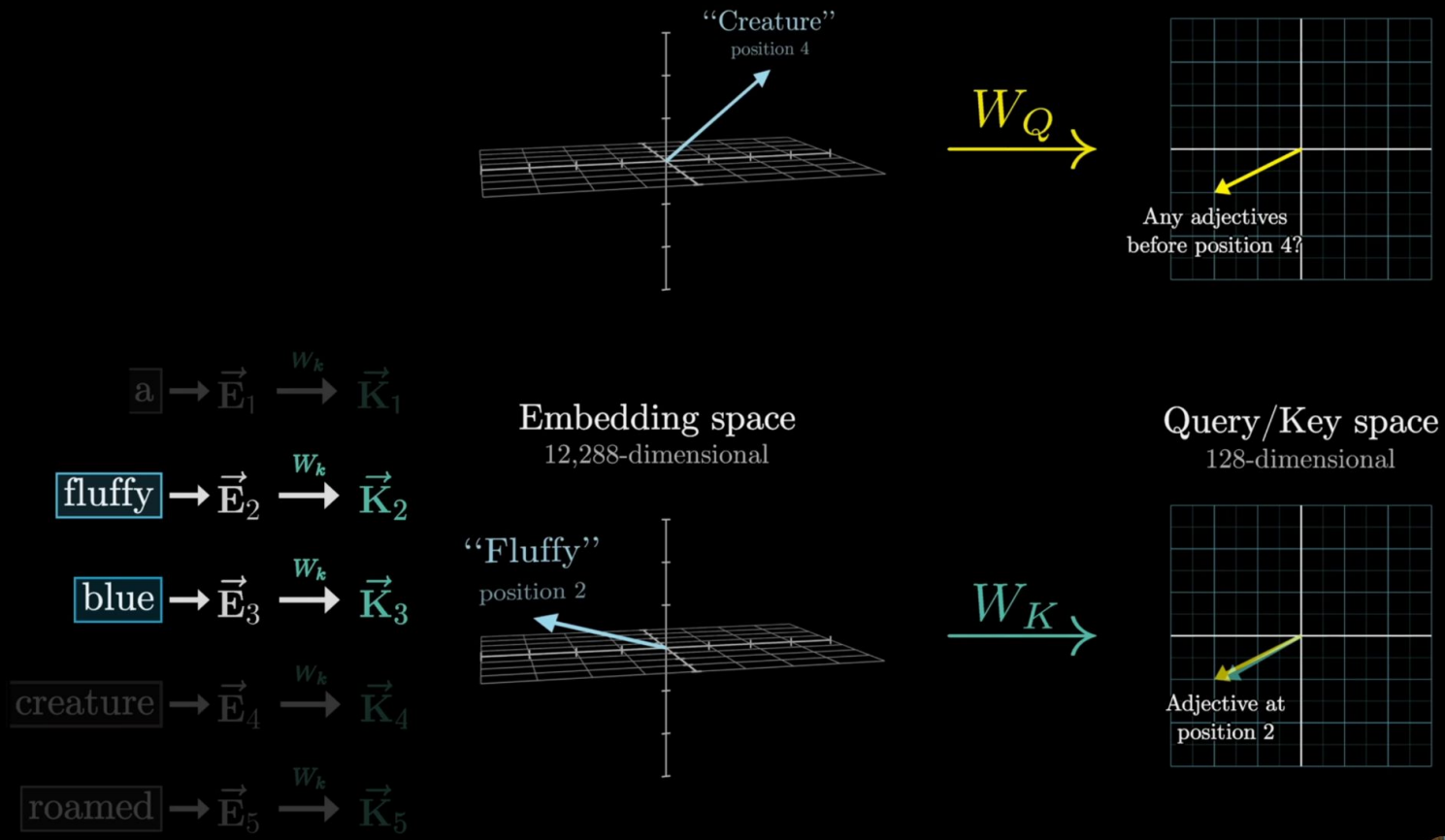




Query/Key space

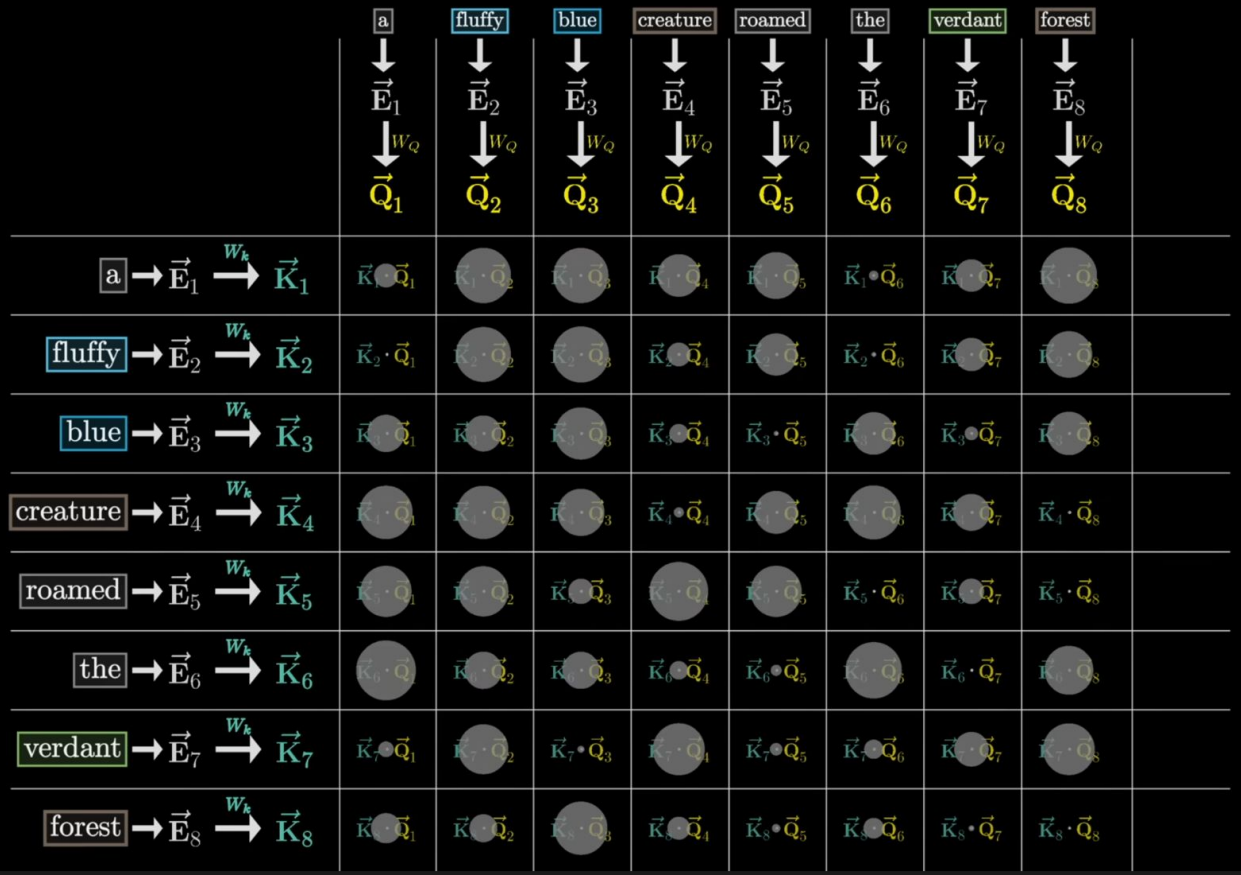
128-dimensional

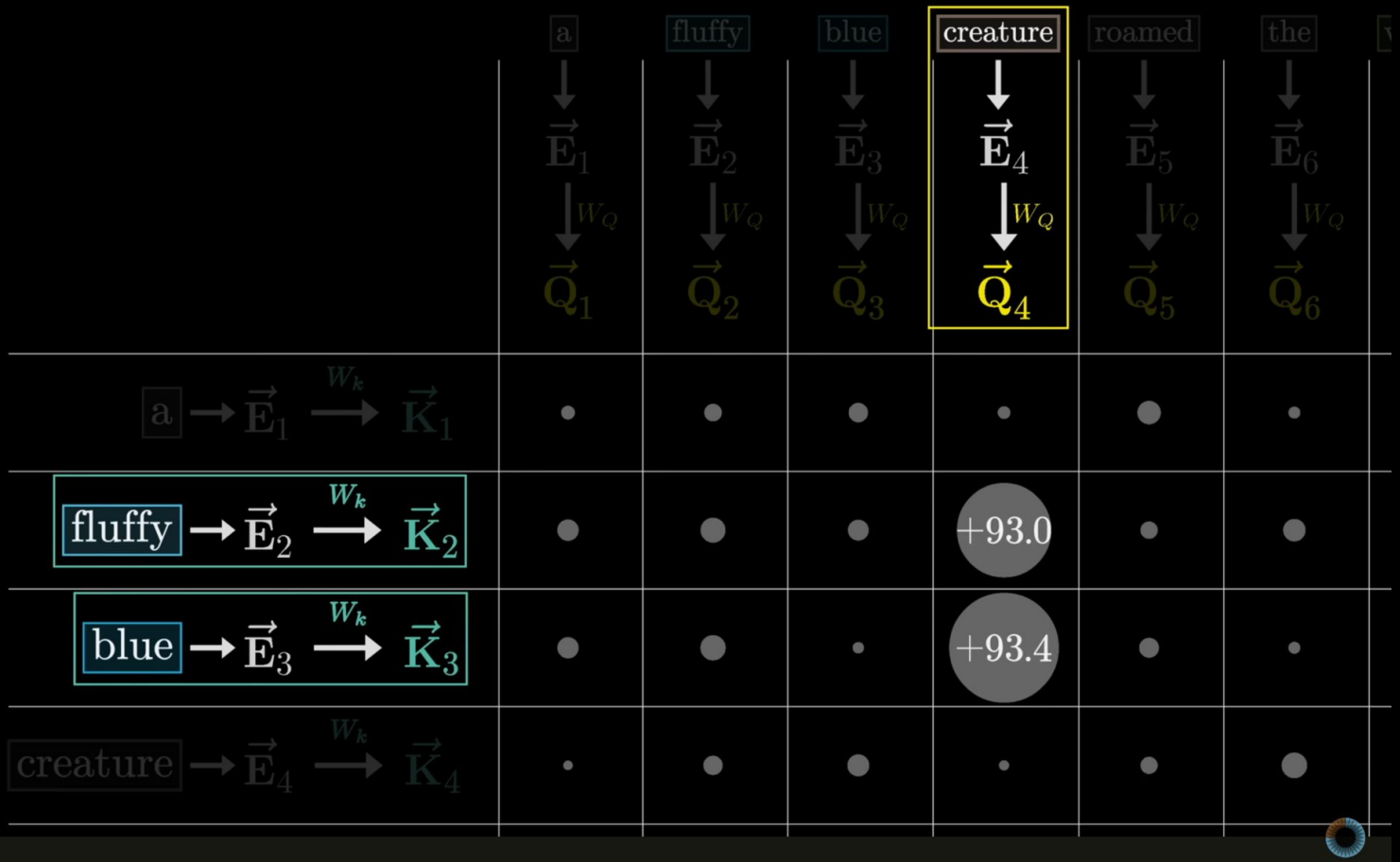


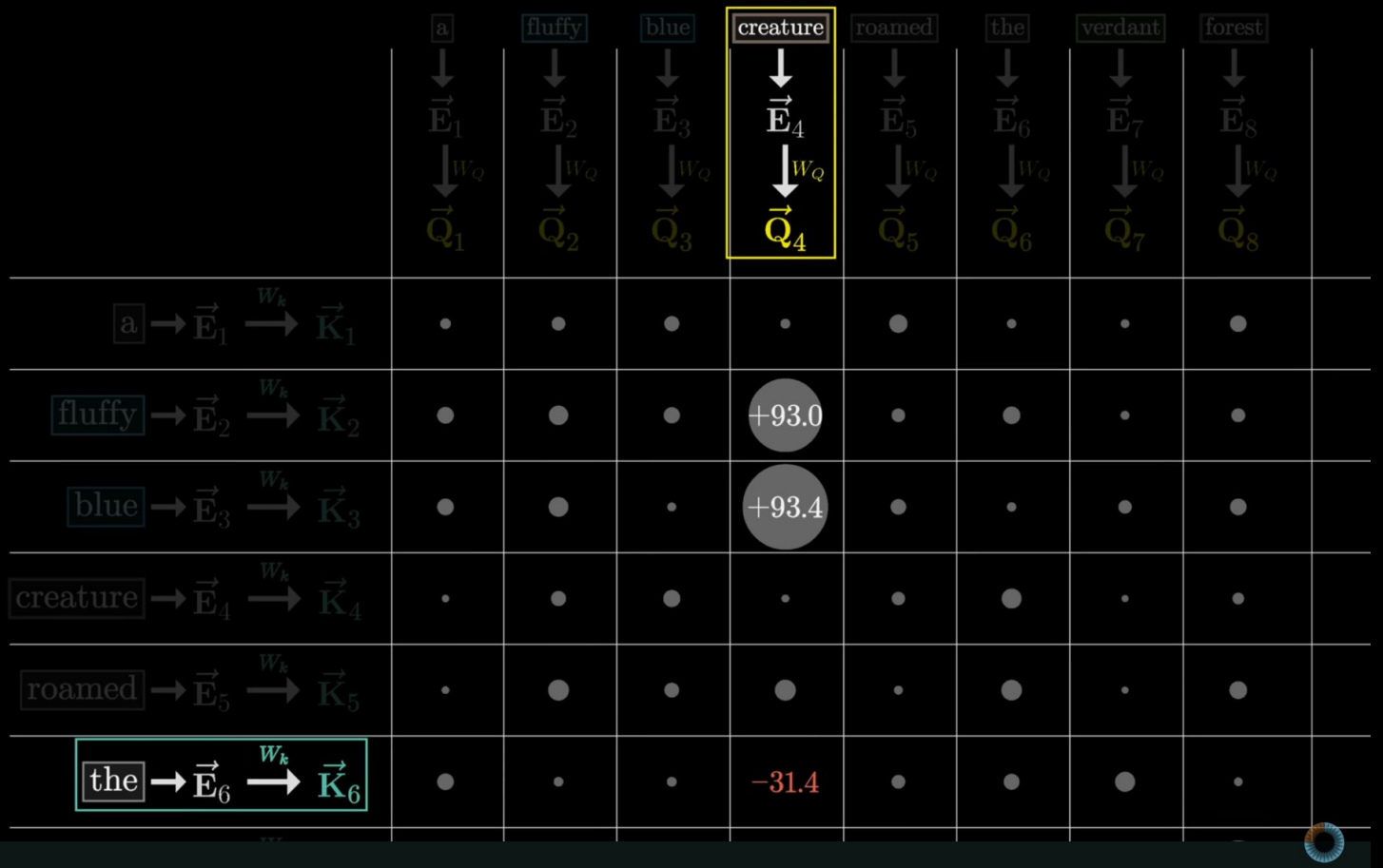

"Creature"
position 4
W_Q
Any adjectives
before position 4?
a → $\vec{E}_1$ → $\vec{K}_1$
fluffy → $\vec{E}_2$ → $\vec{K}_2$
blue → $\vec{E}_3$ → $\vec{K}_3$
creature → $\vec{E}_4$ → $\vec{K}_4$
roamed → $\vec{E}_5$ → $\vec{K}_5$
W_k
Embedding space
12,288-dimensional
"Fluffy"
position 2
W_K
Query/Key space
128-dimensional
Adjective at
position 2


a
fluffy
blue
creature
roamed
the
verdant
forest
$\vec{E}_1$ $\vec{E}_2$ $\vec{E}_3$ $\vec{E}_4$ $\vec{E}_5$ $\vec{E}_6$ $\vec{E}_7$ $\vec{E}_8$
W_Q
$\vec{Q}_1$ $\vec{Q}_2$ $\vec{Q}_3$ $\vec{Q}_4$ $\vec{Q}_5$ $\vec{Q}_6$ $\vec{Q}_7$ $\vec{Q}_8$
a → $\vec{E}_1$ W_k → $\vec{K}_1$
fluffy → $\vec{E}_2$ W_k → $\vec{K}_2$
blue → $\vec{E}_3$ W_k → $\vec{K}_3$
creature → $\vec{E}_4$ W_k → $\vec{K}_4$
roamed → $\vec{E}_5$ W_k → $\vec{K}_5$
the → $\vec{E}_6$ W_k → $\vec{K}_6$
verdant → $\vec{E}_7$ W_k → $\vec{K}_7$
forest → $\vec{E}_8$ W_k → $\vec{K}_8$

	a $\vec{E}_1 \xrightarrow{W_Q} \vec{Q}_1$	fluffy $\vec{E}_2 \xrightarrow{W_Q} \vec{Q}_2$	blue $\vec{E}_3 \xrightarrow{W_Q} \vec{Q}_3$	creature $\vec{E}_4 \xrightarrow{W_Q} \vec{Q}_4$	roamed $\vec{E}_5 \xrightarrow{W_Q} \vec{Q}_5$	the $\vec{E}_6 \xrightarrow{W_Q} \vec{Q}_6$	verdant $\vec{E}_7 \xrightarrow{W_Q} \vec{Q}_7$	forest $\vec{E}_8 \xrightarrow{W_Q} \vec{Q}_8$
a $\vec{E}_1 \xrightarrow{W_K} \vec{K}_1$	$\vec{K}_1 \cdot \vec{Q}_1$	$\vec{K}_1 \cdot \vec{Q}_2$	$\vec{K}_1 \cdot \vec{Q}_3$	$\vec{K}_1 \cdot \vec{Q}_4$	$\vec{K}_1 \cdot \vec{Q}_5$	$\vec{K}_1 \cdot \vec{Q}_6$	$\vec{K}_1 \cdot \vec{Q}_7$	$\vec{K}_1 \cdot \vec{Q}_8$
fluffy $\vec{E}_2 \xrightarrow{W_K} \vec{K}_2$	$\vec{K}_2 \cdot \vec{Q}_1$	$\vec{K}_2 \cdot \vec{Q}_2$	$\vec{K}_2 \cdot \vec{Q}_3$	$\vec{K}_2 \cdot \vec{Q}_4$	$\vec{K}_2 \cdot \vec{Q}_5$	$\vec{K}_2 \cdot \vec{Q}_6$	$\vec{K}_2 \cdot \vec{Q}_7$	$\vec{K}_2 \cdot \vec{Q}_8$
blue $\vec{E}_3 \xrightarrow{W_K} \vec{K}_3$	$\vec{K}_3 \cdot \vec{Q}_1$	$\vec{K}_3 \cdot \vec{Q}_2$	$\vec{K}_3 \cdot \vec{Q}_3$	$\vec{K}_3 \cdot \vec{Q}_4$	$\vec{K}_3 \cdot \vec{Q}_5$	$\vec{K}_3 \cdot \vec{Q}_6$	$\vec{K}_3 \cdot \vec{Q}_7$	$\vec{K}_3 \cdot \vec{Q}_8$
creature $\vec{E}_4 \xrightarrow{W_K} \vec{K}_4$	$\vec{K}_4 \cdot \vec{Q}_1$	$\vec{K}_4 \cdot \vec{Q}_2$	$\vec{K}_4 \cdot \vec{Q}_3$	$\vec{K}_4 \cdot \vec{Q}_4$	$\vec{K}_4 \cdot \vec{Q}_5$	$\vec{K}_4 \cdot \vec{Q}_6$	$\vec{K}_4 \cdot \vec{Q}_7$	$\vec{K}_4 \cdot \vec{Q}_8$
roamed $\vec{E}_5 \xrightarrow{W_K} \vec{K}_5$	$\vec{K}_5 \cdot \vec{Q}_1$	$\vec{K}_5 \cdot \vec{Q}_2$	$\vec{K}_5 \cdot \vec{Q}_3$	$\vec{K}_5 \cdot \vec{Q}_4$	$\vec{K}_5 \cdot \vec{Q}_5$	$\vec{K}_5 \cdot \vec{Q}_6$	$\vec{K}_5 \cdot \vec{Q}_7$	$\vec{K}_5 \cdot \vec{Q}_8$
the $\vec{E}_6 \xrightarrow{W_K} \vec{K}_6$	$\vec{K}_6 \cdot \vec{Q}_1$	$\vec{K}_6 \cdot \vec{Q}_2$	$\vec{K}_6 \cdot \vec{Q}_3$	$\vec{K}_6 \cdot \vec{Q}_4$	$\vec{K}_6 \cdot \vec{Q}_5$	$\vec{K}_6 \cdot \vec{Q}_6$	$\vec{K}_6 \cdot \vec{Q}_7$	$\vec{K}_6 \cdot \vec{Q}_8$
verdant $\vec{E}_7 \xrightarrow{W_K} \vec{K}_7$	$\vec{K}_7 \cdot \vec{Q}_1$	$\vec{K}_7 \cdot \vec{Q}_2$	$\vec{K}_7 \cdot \vec{Q}_3$	$\vec{K}_7 \cdot \vec{Q}_4$	$\vec{K}_7 \cdot \vec{Q}_5$	$\vec{K}_7 \cdot \vec{Q}_6$	$\vec{K}_7 \cdot \vec{Q}_7$	$\vec{K}_7 \cdot \vec{Q}_8$
forest $\vec{E}_8 \xrightarrow{W_K} \vec{K}_8$	$\vec{K}_8 \cdot \vec{Q}_1$	$\vec{K}_8 \cdot \vec{Q}_2$	$\vec{K}_8 \cdot \vec{Q}_3$	$\vec{K}_8 \cdot \vec{Q}_4$	$\vec{K}_8 \cdot \vec{Q}_5$	$\vec{K}_8 \cdot \vec{Q}_6$	$\vec{K}_8 \cdot \vec{Q}_7$	$\vec{K}_8 \cdot \vec{Q}_8$

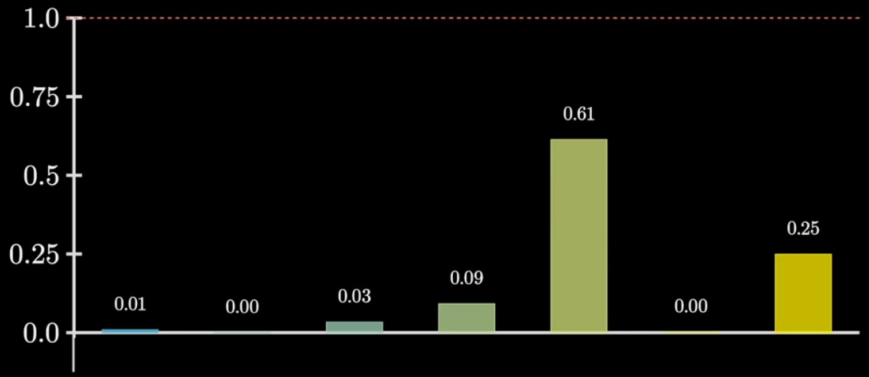

a
fluffy
blue
creature
roamed
the
verdant
forest
$\vec{E}_1$ $\vec{E}_2$ $\vec{E}_3$ $\vec{E}_4$ $\vec{E}_5$ $\vec{E}_6$ $\vec{E}_7$ $\vec{E}_8$
W_Q
$\vec{Q}_1$ $\vec{Q}_2$ $\vec{Q}_3$ $\vec{Q}_4$ $\vec{Q}_5$ $\vec{Q}_6$ $\vec{Q}_7$ $\vec{Q}_8$
a → $\vec{E}_1$ W_K → $\vec{K}_1$
fluffy → $\vec{E}_2$ W_K → $\vec{K}_2$
blue → $\vec{E}_3$ W_K → $\vec{K}_3$
creature → $\vec{E}_4$ W_K → $\vec{K}_4$
roamed → $\vec{E}_5$ W_K → $\vec{K}_5$
the → $\vec{E}_6$ W_K → $\vec{K}_6$
verdant → $\vec{E}_7$ W_K → $\vec{K}_7$
forest → $\vec{E}_8$ W_K → $\vec{K}_8$


a
fluffy
blue
creature
roamed
the
$\vec{E}_1$
$\vec{E}_2$
$\vec{E}_3$
$\vec{E}_4$
$\vec{E}_5$
$\vec{E}_6$
W_Q
$\vec{Q}_1$
$\vec{Q}_2$
$\vec{Q}_3$
$\vec{Q}_4$
$\vec{Q}_5$
$\vec{Q}_6$
a → $\vec{E}_1$ W_k $\vec{K}_1$
fluffy → $\vec{E}_2$ W_k $\vec{K}_2$
blue → $\vec{E}_3$ W_k $\vec{K}_3$
creature → $\vec{E}_4$ W_k $\vec{K}_4$
+93.0
+93.4


a
fluffy
blue
creature
roamed
the
verdant
forest
$\vec{E}_1$ $\vec{E}_2$ $\vec{E}_3$ $\vec{E}_4$ $\vec{E}_5$ $\vec{E}_6$ $\vec{E}_7$ $\vec{E}_8$
W_Q
$\vec{Q}_1$ $\vec{Q}_2$ $\vec{Q}_3$ $\vec{Q}_4$ $\vec{Q}_5$ $\vec{Q}_6$ $\vec{Q}_7$ $\vec{Q}_8$
a → $\vec{E}_1$ W_K → $\vec{K}_1$
fluffy → $\vec{E}_2$ W_K → $\vec{K}_2$
blue → $\vec{E}_3$ W_K → $\vec{K}_3$
creature → $\vec{E}_4$ W_K → $\vec{K}_4$
roamed → $\vec{E}_5$ W_K → $\vec{K}_5$
the → $\vec{E}_6$ W_K → $\vec{K}_6$
+93.0
+93.4
−31.4

creature	roamed	the	verdant	forest
$\vec{E}_4$	$\vec{E}_5$	$\vec{E}_6$	$\vec{E}_7$	$\vec{E}_8$
W_Q	W_Q	W_Q	W_Q	W_Q
$\vec{Q}_4$	$\vec{Q}_5$	$\vec{Q}_6$	$\vec{Q}_7$	$\vec{Q}_8$
−27.8	−5.2	−89.3	−45.2	−36.1
+93.0				
+93.4				
+4.9	−32.4	−92.3	−9.5	−28.1
−55.4	+0.6	−64.7	−96.7	−18.9
−31.4	+5.5	+0.6	−4.6	−96.8
−59.8	−79.0	−97.9	+3.7	+93.8
−90.6	−75.6	−89.0	−70.8	+4.7

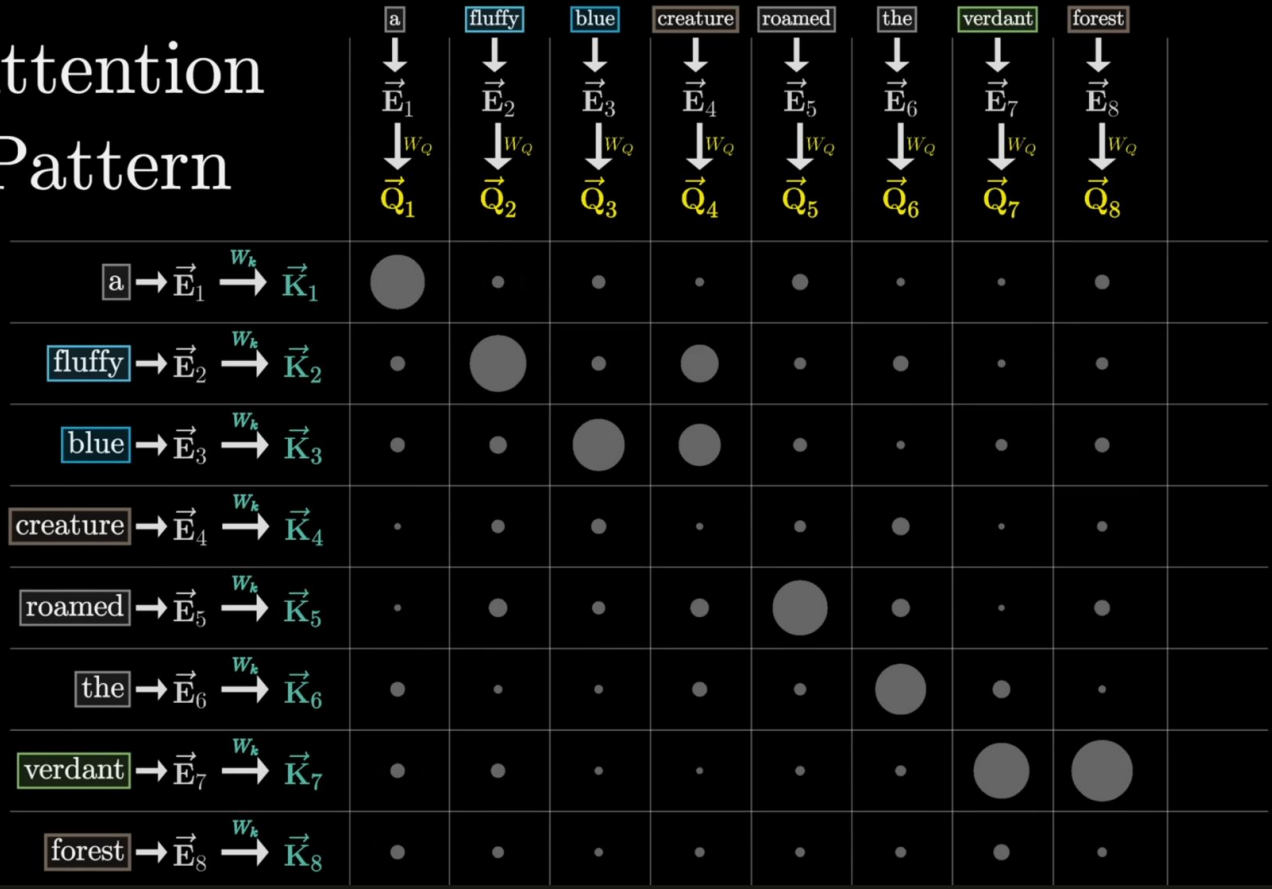
We want these to act like weights

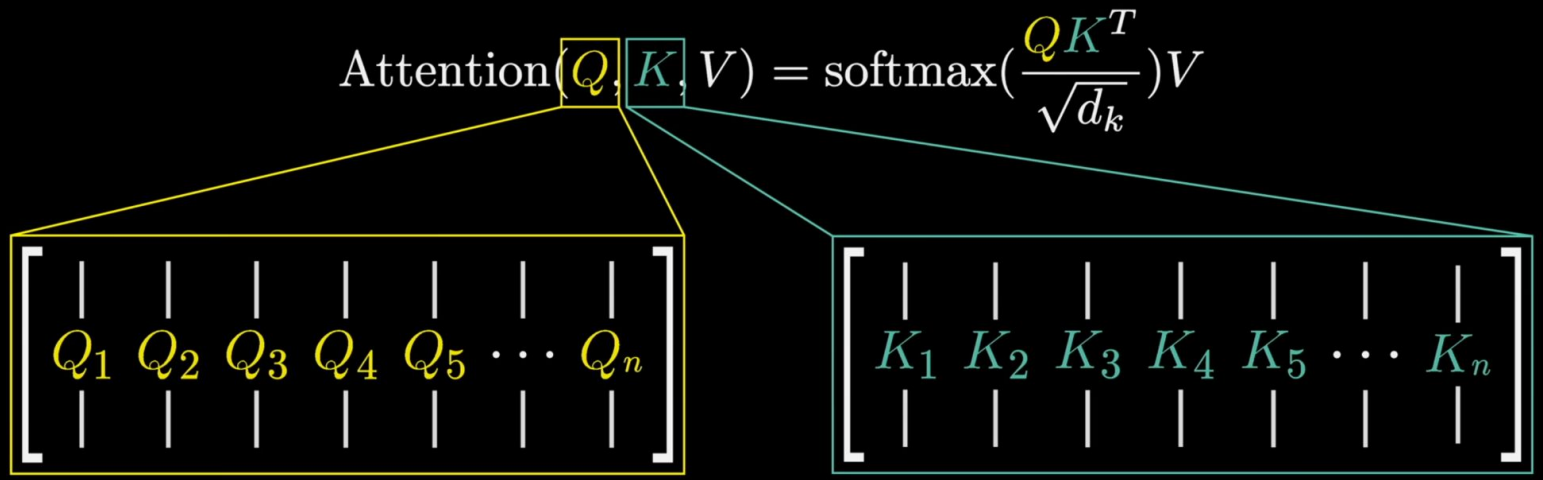



a
fluffy
blue
creature
roamed
the
verdant
forest
softmax
0.00
0.00
0.00
0.00
0.01
0.99
0.00
0.00

	a $\vec{E}_1 \xrightarrow{W_Q} \vec{Q}_1$	fluffy $\vec{E}_2 \xrightarrow{W_Q} \vec{Q}_2$	blue $\vec{E}_3 \xrightarrow{W_Q} \vec{Q}_3$	creature $\vec{E}_4 \xrightarrow{W_Q} \vec{Q}_4$	roamed $\vec{E}_5 \xrightarrow{W_Q} \vec{Q}_5$	the $\vec{E}_6 \xrightarrow{W_Q} \vec{Q}_6$	verdant $\vec{E}_7 \xrightarrow{W_Q} \vec{Q}_7$	forest $\vec{E}_8 \xrightarrow{W_Q} \vec{Q}_8$
a $\vec{E}_1 \xrightarrow{W_K} \vec{K}_1$	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
fluffy $\vec{E}_2 \xrightarrow{W_K} \vec{K}_2$	0.00	1.00	0.00	0.42	0.00	0.00	0.00	0.00
blue $\vec{E}_3 \xrightarrow{W_K} \vec{K}_3$	0.00	0.00	1.00	0.58	0.00	0.00	0.00	0.00
creature $\vec{E}_4 \xrightarrow{W_K} \vec{K}_4$	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
roamed $\vec{E}_5 \xrightarrow{W_K} \vec{K}_5$	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00
the $\vec{E}_6 \xrightarrow{W_K} \vec{K}_6$	0.00	0.00	0.00	0.00	0.99	1.00	0.00	0.00
verdant $\vec{E}_7 \xrightarrow{W_K} \vec{K}_7$	0.00	0.00	0.00	0.00	0.00	0.00	1.00	1.00
forest $\vec{E}_8 \xrightarrow{W_K} \vec{K}_8$	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Attention Pattern



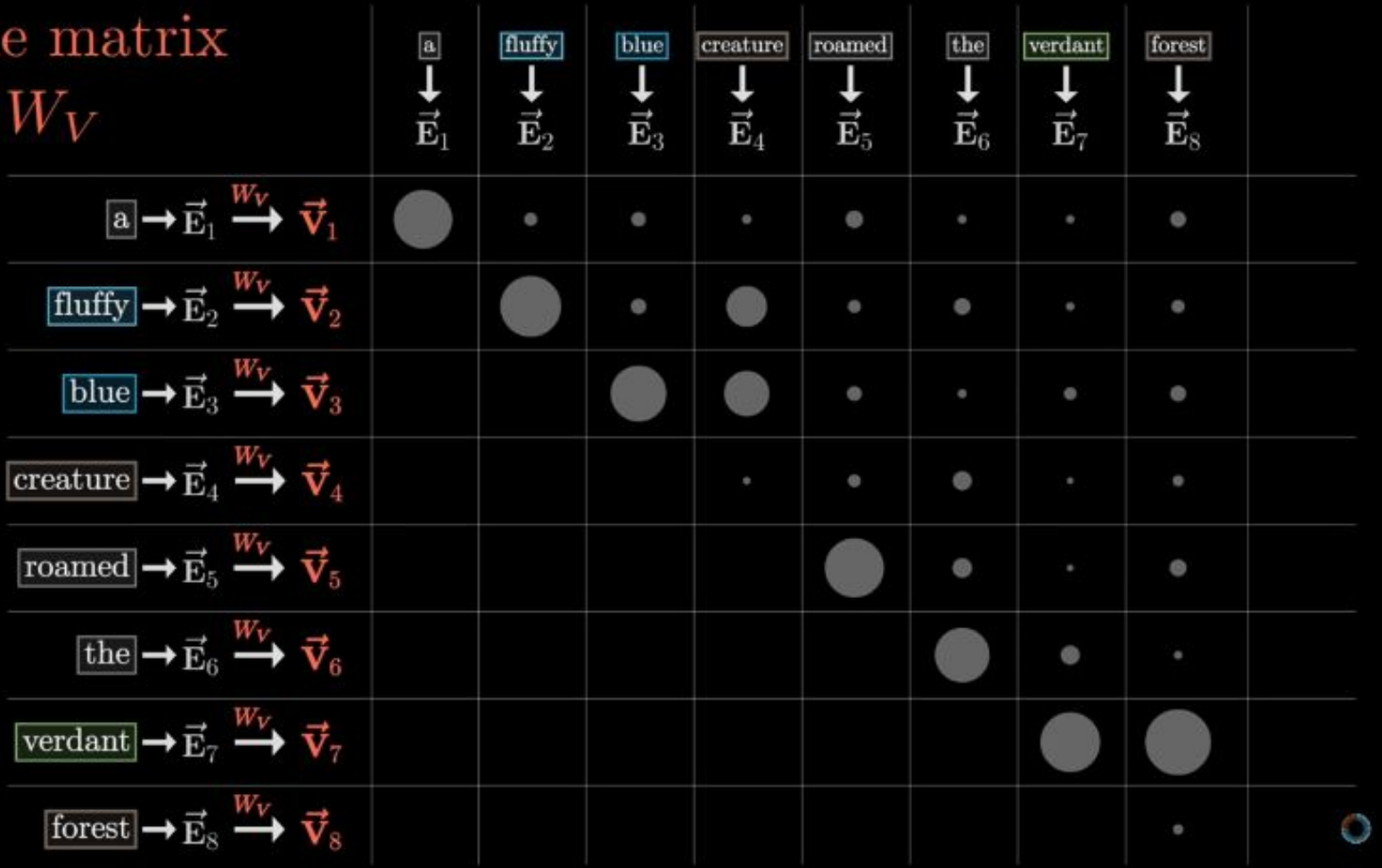

$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$
$Q_1\ Q_2\ Q_3\ Q_4\ Q_5 \cdots Q_n$
$K_1\ K_2\ K_3\ K_4\ K_5 \cdots K_n$

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

	Q_1	Q_2	Q_3	Q_4	Q_5	$\cdots$	Q_n	
K_1	$Q_1 \cdot K_1$	$Q_2 \cdot K_1$	$Q_3 \cdot K_1$	$Q_4 \cdot K_1$	$Q_5 \cdot K_1$	$\cdots$	$Q_n \cdot K_1$	
K_2	$Q_1 \cdot K_2$	$Q_2 \cdot K_2$	$Q_3 \cdot K_2$	$Q_4 \cdot K_2$	$Q_5 \cdot K_2$	$\cdots$	$Q_n \cdot K_2$	
K_3	$Q_1 \cdot K_3$	$Q_2 \cdot K_3$	$Q_3 \cdot K_3$	$Q_4 \cdot K_3$	$Q_5 \cdot K_3$	$\cdots$	$Q_n \cdot K_3$	
K_4	$Q_1 \cdot K_4$	$Q_2 \cdot K_4$	$Q_3 \cdot K_4$	$Q_4 \cdot K_4$	$Q_5 \cdot K_4$	$\cdots$	$Q_n \cdot K_4$	
K_5	$Q_1 \cdot K_5$	$Q_2 \cdot K_5$	$Q_3 \cdot K_5$	$Q_4 \cdot K_5$	$Q_5 \cdot K_5$	$\cdots$	$Q_n \cdot K_5$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	

Value matrix

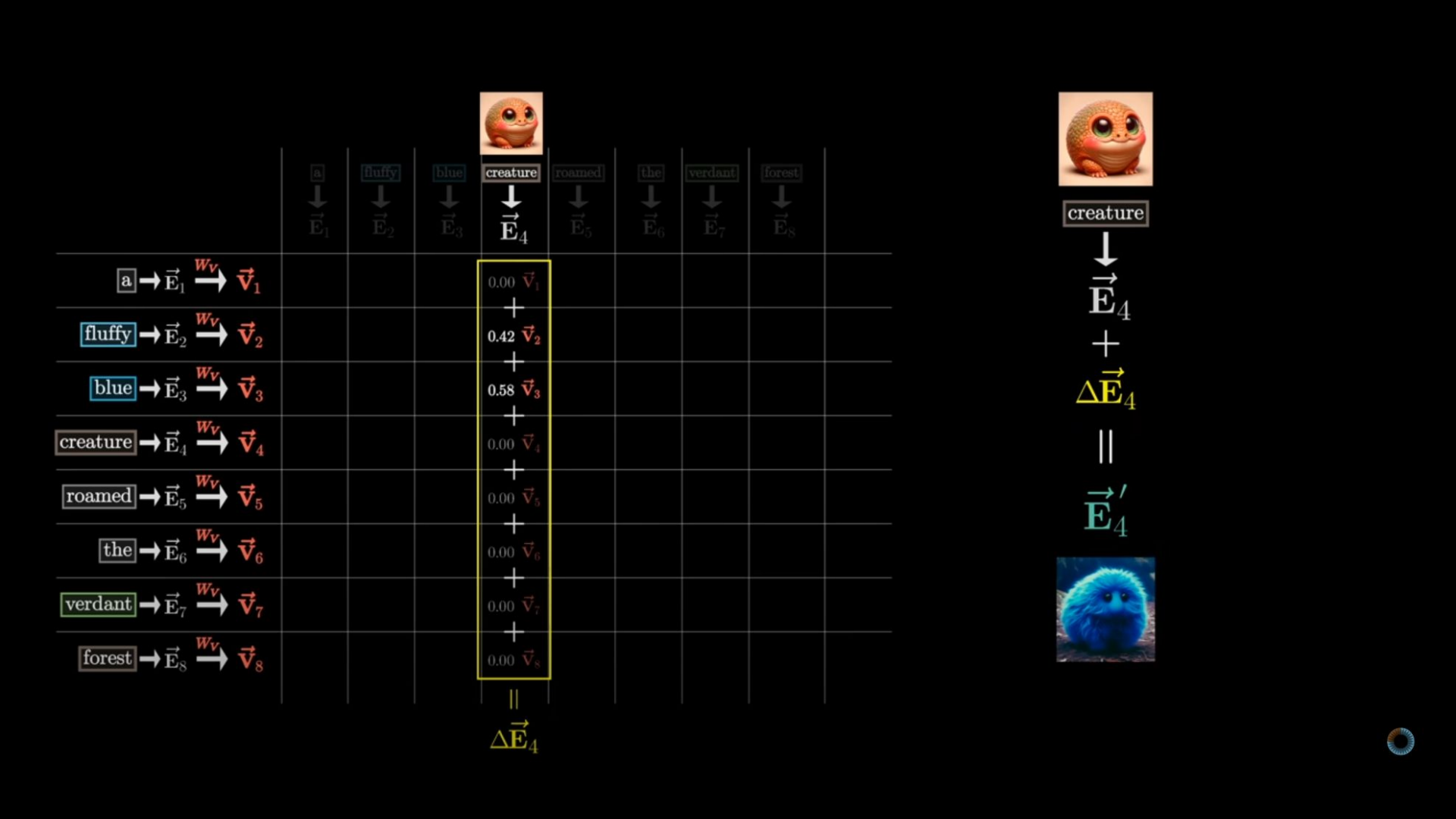
W_V



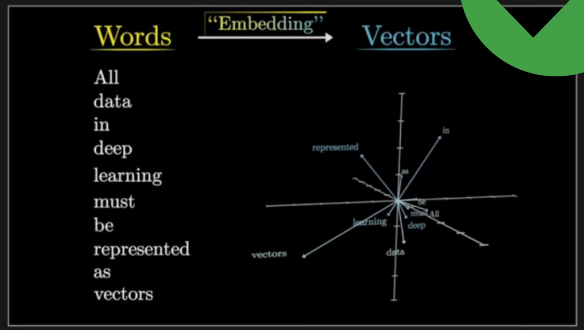
Value matrix

W_V

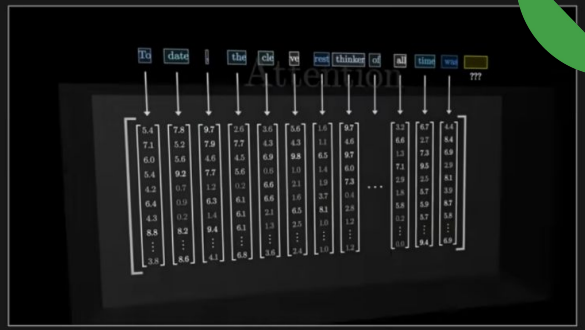
	a $\downarrow \vec{E}_1$	fluffy $\downarrow \vec{E}_2$	blue $\downarrow \vec{E}_3$	creature $\downarrow \vec{E}_4$	roamed $\downarrow \vec{E}_5$	the $\downarrow \vec{E}_6$	verdant $\downarrow \vec{E}_7$	forest $\downarrow \vec{E}_8$
a $\rightarrow \vec{E}_1 \xrightarrow{W_V} \vec{V}_1$				0.00 $\vec{V}_1$				
fluffy $\rightarrow \vec{E}_2 \xrightarrow{W_V} \vec{V}_2$				0.42 $\vec{V}_2$				
blue $\rightarrow \vec{E}_3 \xrightarrow{W_V} \vec{V}_3$				0.58 $\vec{V}_3$				
creature $\rightarrow \vec{E}_4 \xrightarrow{W_V} \vec{V}_4$				0.00 $\vec{V}_4$				
roamed $\rightarrow \vec{E}_5 \xrightarrow{W_V} \vec{V}_5$				0.00 $\vec{V}_5$				
the $\rightarrow \vec{E}_6 \xrightarrow{W_V} \vec{V}_6$				0.00 $\vec{V}_6$				
verdant $\rightarrow \vec{E}_7 \xrightarrow{W_V} \vec{V}_7$				0.00 $\vec{V}_7$				
forest $\rightarrow \vec{E}_8 \xrightarrow{W_V} \vec{V}_8$				0.00 $\vec{V}_8$				


a
fluffy
blue
creature
roamed
the
verdant
forest
$\vec{E}_1$ $\vec{E}_2$ $\vec{E}_3$ $\vec{E}_4$ $\vec{E}_5$ $\vec{E}_6$ $\vec{E}_7$ $\vec{E}_8$
a → $\vec{E}_1$ W_V → $\vec{V}_1$
fluffy → $\vec{E}_2$ W_V → $\vec{V}_2$
blue → $\vec{E}_3$ W_V → $\vec{V}_3$
creature → $\vec{E}_4$ W_V → $\vec{V}_4$
roamed → $\vec{E}_5$ W_V → $\vec{V}_5$
the → $\vec{E}_6$ W_V → $\vec{V}_6$
verdant → $\vec{E}_7$ W_V → $\vec{V}_7$
forest → $\vec{E}_8$ W_V → $\vec{V}_8$
0.00 $\vec{V}_1$
+
0.42 $\vec{V}_2$
+
0.58 $\vec{V}_3$
+
0.00 $\vec{V}_4$
+
0.00 $\vec{V}_5$
+
0.00 $\vec{V}_6$
+
0.00 $\vec{V}_7$
+
0.00 $\vec{V}_8$
||
$\Delta\vec{E}_4$
creature
$\vec{E}_4$
+
$\Delta\vec{E}_4$
||
$\vec{E}'_4$

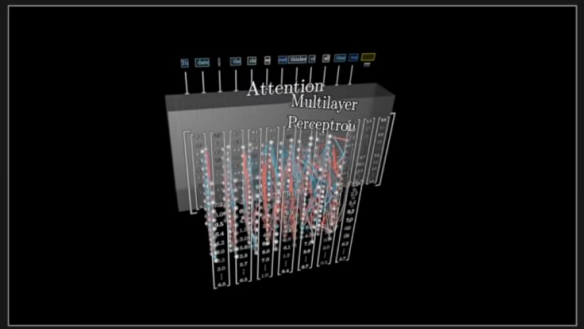
Embedding



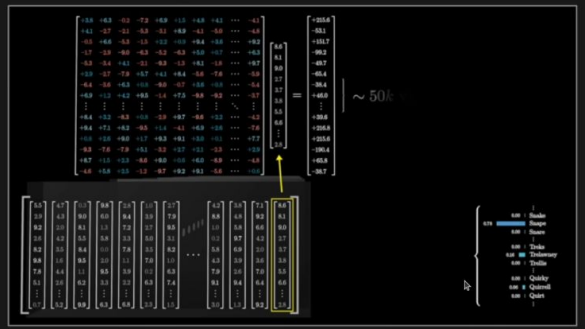
Attention



MLPs



Unembedding



Scaling Vision Transformers to 22 Billion Parameters

Mostafa Dehghani* **Josip Djolonga*** **Basil Mustafa*** **Piotr Padlewski*** **Jonathan Heek***
Justin Gilmer **Andreas Steiner** **Mathilde Caron** **Robert Geirhos** **Ibrahim Alabdulmohsin**
Rodolphe Jenatton **Lucas Beyer** **Michael Tschannen** **Anurag Arnab** **Xiao Wang**
Carlos Riquelme **Matthias Minderer** **Joan Puigcerver** **Utku Evci** **Manoj Kumar**
Sjoerd van Steenkiste **Gamaleldin F. Elsayed** **Aravindh Mahendran** **Fisher Yu**
Avital Oliver **Fantine Huot** **Jasmijn Bastings** **Mark Patrick Collier** **Alexey A. Gritsenko**
Vighnesh Birodkar **Cristina Vasconcelos** **Yi Tay** **Thomas Mensink** **Alexander Kolesnikov**
Filip Pavetić **Dustin Tran** **Thomas Kipf** **Mario Lučić** **Xiaohua Zhai** **Daniel Keysers**
Jeremiah Harmsen **Neil Houlsby***
Google Research

Abstract

The scaling of Transformers has driven breakthrough capabilities for language models. At present, the largest large language models (LLMs) contain upwards of 100B parameters. Vision Transformers (ViT) have introduced the same architecture to image and video modelling, but these have not yet been successfully scaled to nearly the same degree; the largest dense ViT contains 4B parameters (Chen et al., 2022). We present a recipe for highly efficient and stable training of a 22B-parameter ViT (ViT-22B) and perform a wide variety of experiments on the resulting model. When evaluated on downstream tasks (often with a lightweight linear model on frozen features), ViT-22B demonstrates increasing performance with scale. We further observe other interesting benefits of scale, including an improved tradeoff between fairness and performance, state-of-the-art alignment to human visual perception in terms of shape/texture bias, and improved robustness. ViT-22B demonstrates the potential for "LLM-like" scaling in vision, and provides key steps towards getting there.

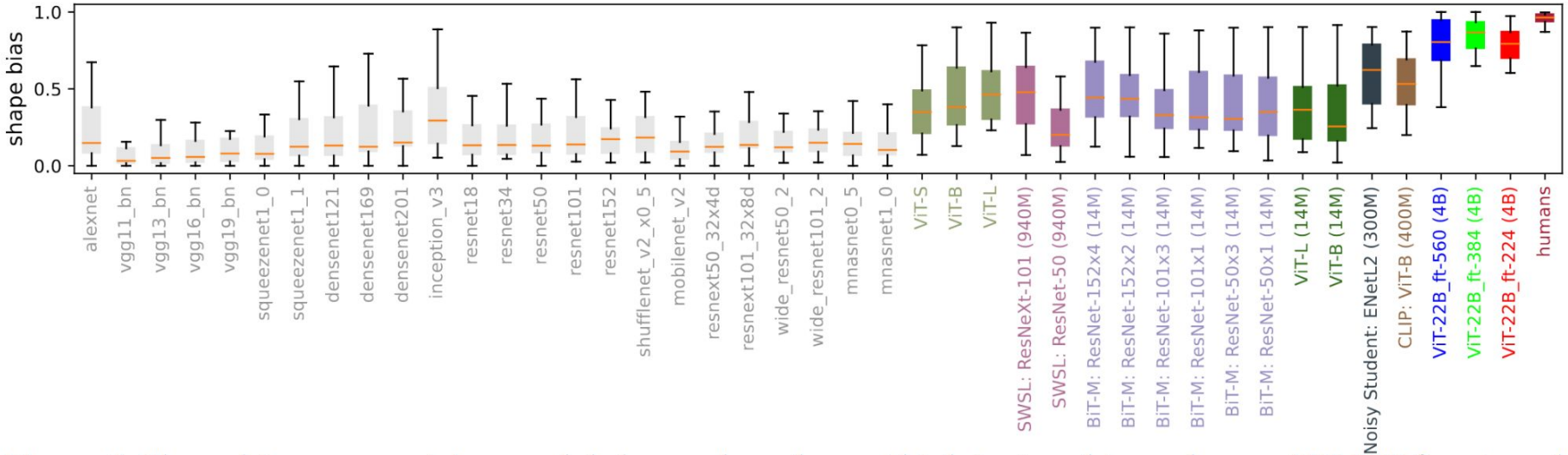
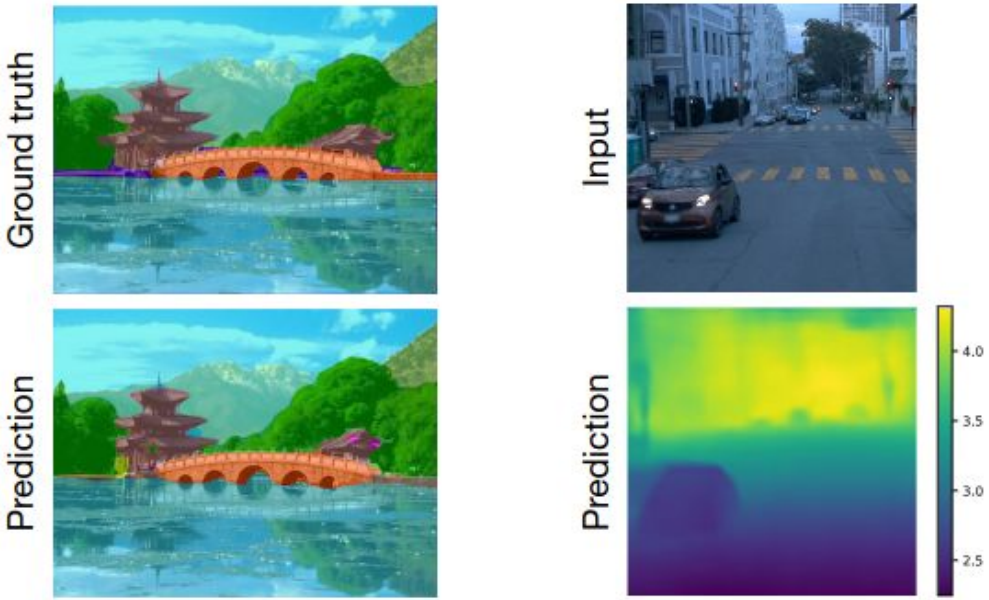


Figure 8: Shape bias: many vision models have a low shape / high texture bias, whereas ViT-22B fine-tuned on ImageNet (red, green, blue trained on 4B images as indicated by brackets after model names, unless trained on ImageNet only) have the highest shape bias recorded in a ML model to date, bringing them closer towards a human-like shape bias.

Fraction of ADE20k train data	1/16	1/8	1/4	1/2	1
ViT-L (Touvron et al., 2022)	36.1	41.3	45.6	48.4	51.9
ViT-G (Zhai et al., 2022a)	42.4	47.0	50.2	52.4	**55.6**
ViT-22B (Ours)	**44.7**	**47.2**	**50.6**	**52.5**	54.9



(a) Semantic segmentation (b) Depth estimation

Figure 6: Dense prediction from frozen ViT-22B features.

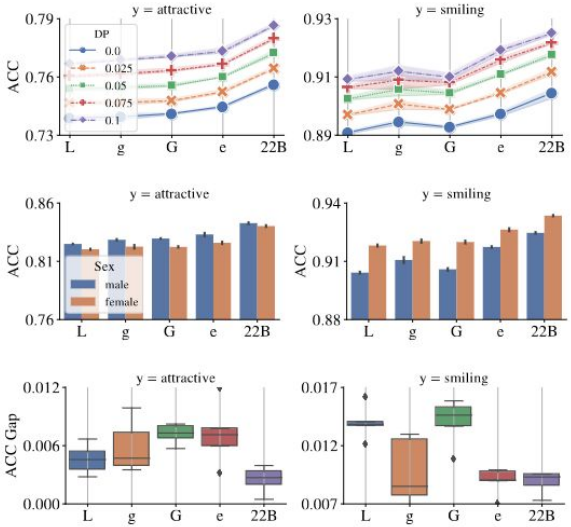
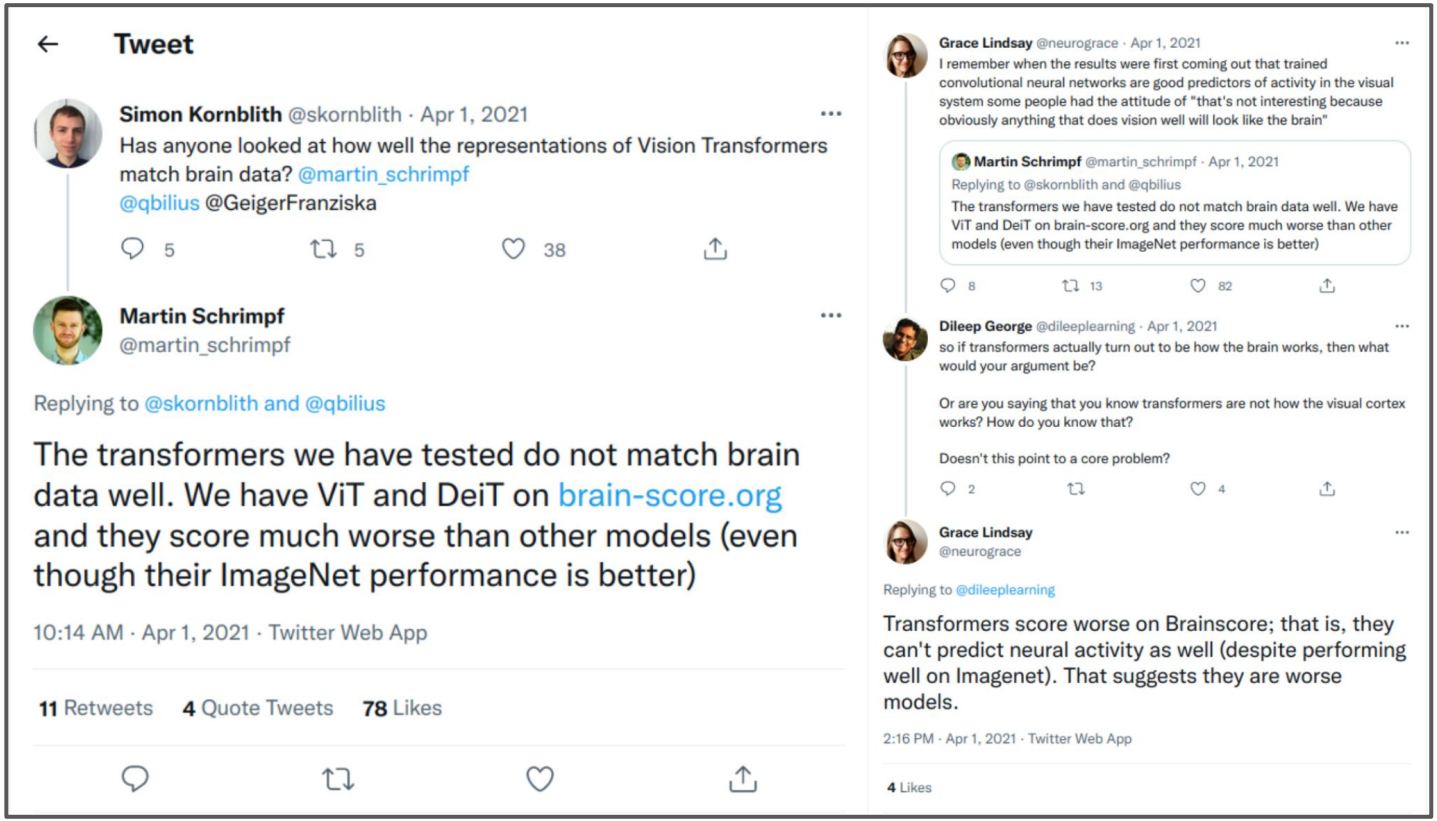


Figure 7: TOP: Accuracy (ACC) for ViT variants *after* debiasing for each DP level. MIDDLE: Accuracy for each subgroup in CelebA *prior to* debiasing. BOTTOM: y-axis is absolute difference in performance across the two subgroups: females and males. ViT-22B provides a more equitable performance, compared to smaller ViT architectures.

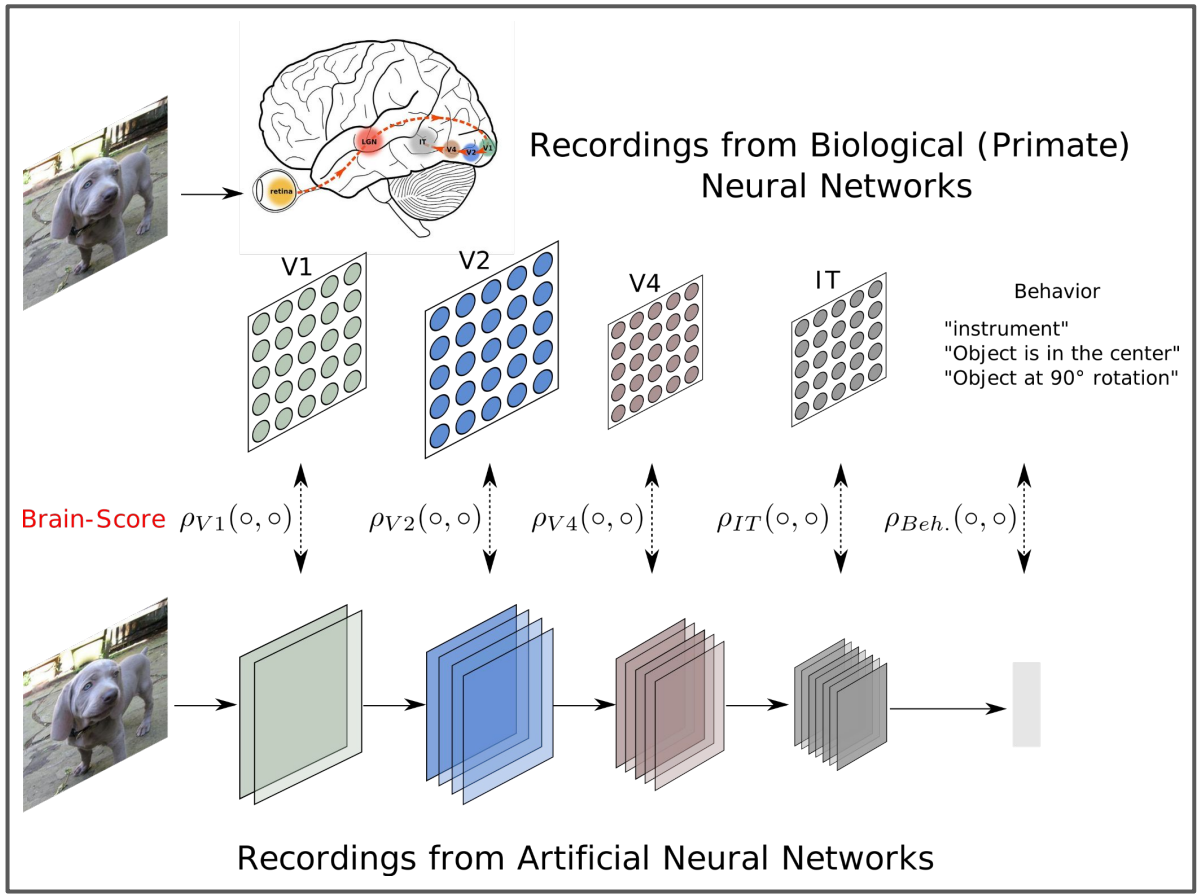
Are Transformers good models of the Human Visual Cortex?

William Berrios

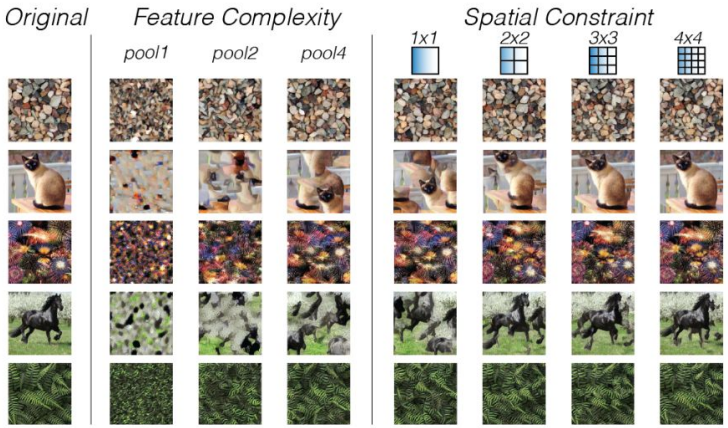


Brain-Score

www.brain-score.org



Multi-Resolution + Local Texture Computation

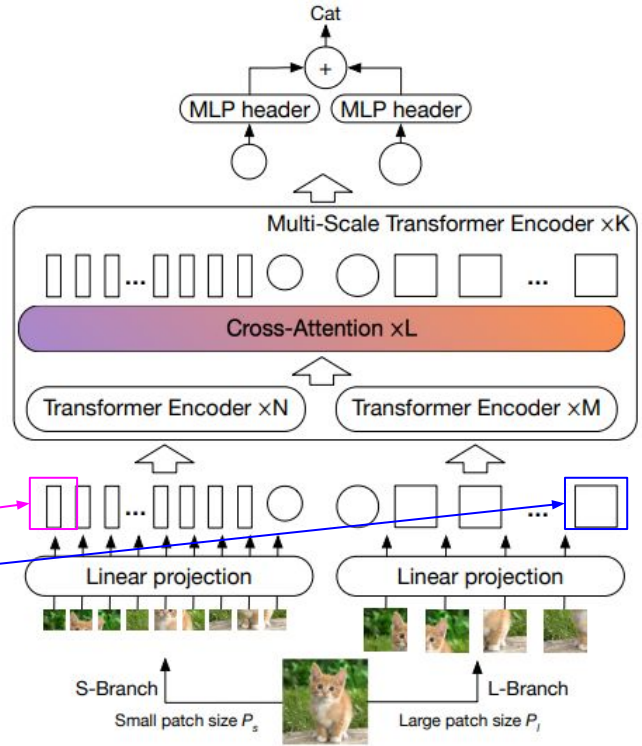


Gatys et al., 2016
Jagadeesh & Gardner, 2021

$$T_M = \begin{bmatrix} \phi_1(x)\phi_1(x) & \phi_1(x)\phi_2(x) & \dots & \phi_1(x)\phi_n(x) \\ \phi_2(x)\phi_1(x) & \phi_2(x)\phi_2(x) & \dots & \phi_2(x)\phi_n(x) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_n(x)\phi_1(x) & \phi_n(x)\phi_2(x) & \dots & \phi_n(x)\phi_n(x) \end{bmatrix}$$

Deza et al., 2020

Dual-branch vision transformer to extract multi-scale feature representations + Gramian-like local texture computation



CrossViT

Chen et. al., 2021

Our Model beyond the Competition

Rank	Model (submitted by)	average	V1 (23 benchmarks)	V2 (1 benchmark)	V4 (4 benchmarks)	IT (5 benchmarks)	behavior (1 benchmark)	engine
1	effnetb1_cutmixpatch_augmix_robust32_avge4e7_manylayers_324x288 *Alexander Riedel*	.495	.568	.360	.481	.412	.652	.297
2	voneresnet-50-robust *Tiago Marques*	.471	.531	.391	.471	.417	.545	X
3	custom_model_cv_18_dagger_408 *William Berrios*	.467	.493	.342	.514	.425	.562	.473
4	resnet50_finetune_cutmix_e3_robust_linf8255_e0_247x234 *Alexander Riedel*	.466	.584	.362	.472	.364	.549	
5	effnetb1_cutmix_augmix_sam_e1_5avg_424x377 *Alexander Riedel*	.463	.482	.291	.499	.381	.664	.033
6	resnet50_finetune_cutmix_AVGe2e3_robust_linf8255_e0_247x234 *Alexander Riedel*	.462	.584	.360	.464	.368	.536	.285
7	voneresnet-50-non_stochastic *Tiago Marques*	.461	.569	.326	.484	.398	.530	.552

	Violet Xiang							
200	deit_tiny_patch16_224_id *Violet Xiang*	.151	.091	.251	.120	.081	.211	
201	ode_net_mar15 *Agrim Sharma*	.150	.423	.152	.120	.058		
202	resnet-18-LC_untrained *Roman Pogodin*	.137	.347	.056	.221	.115	-0.055	
203	dcn_full_mar15 *Agrim Sharma*	.134	.391	.128	.103	.047		
204	dcgan *Brain-Score Team*	.112	.158	.226	.108	.043	.023	
205	dcn_ode *Agrim Sharma*	.108	.144	.208	.129	.056		
206	unet_entire *Mike Ferguson*	.100	.135	.204	.119	.048	-0.006	
207	pixels-baseline *Brain-Score Team*	.051	.158	.003	.048	.028	.020	
208	resnet-50x4_untrained *Roman Pogodin*	X	X	X	X	X	X	

Cortex-1

Brain-Score

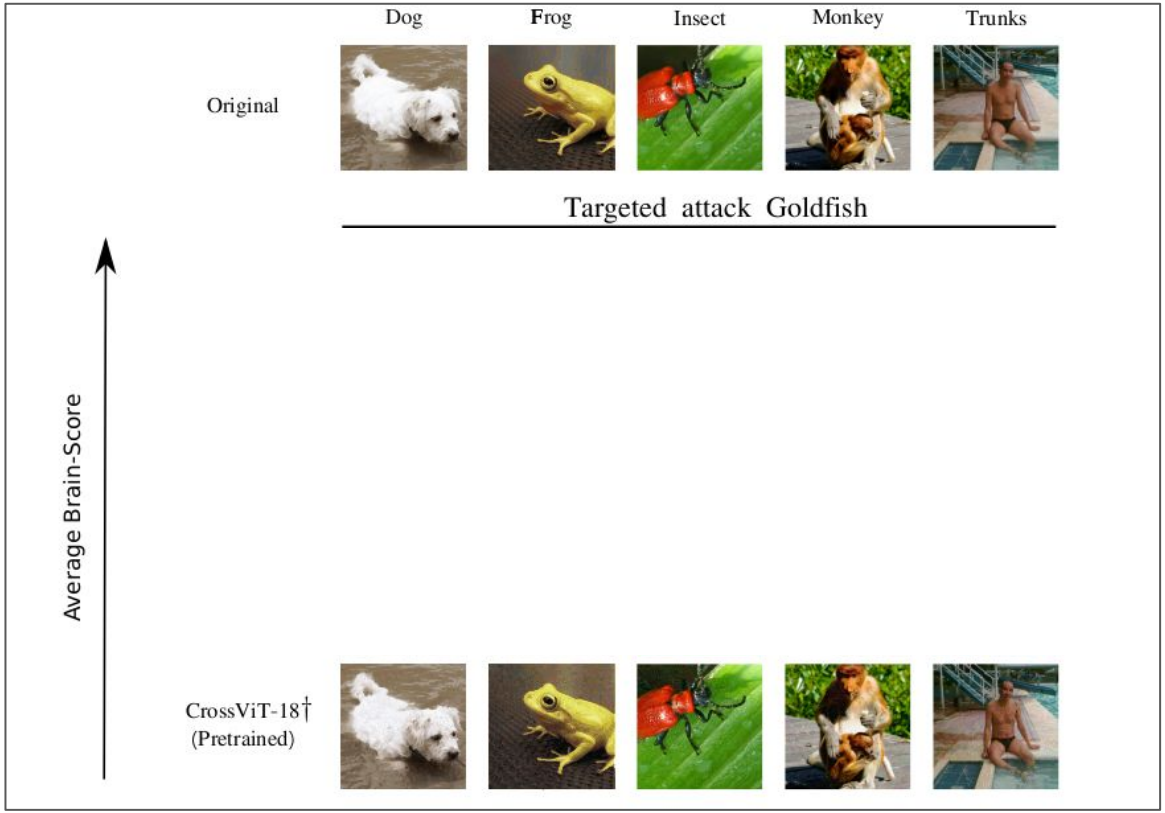
Embedding Model	Organization	Number of Training Images	Brain-Score Area V4	Global Brain-Score Model Ranking (*)	NeuroAI
Cortex (Artificio)		1 Million	**.514**	**8**	✓
BagofTricks (Ernst-Abbe University)		1 Million	.485	**1**	✓
VOneResNet (MIT)		1 Million	.484	11	✓
VITO (Google DeepMind)		1 Million	.483	33	✓
CrossVit (IBM)		1 Million	.478	46	X
ResNet50-CLIP (OpenAI)		400 Million	.484	48	X
TaherehNet (Columbia)		1 Million	**.532**	86	✓
ViT-B32 (Google Brain)		15 Million	.471	132	X

* Ranking done over 242 models across V1,V2,V4, IT, Behaviour
& Engineering benchmarks. Visit https://www.brain-score.org to see complete list.

December 10th, 2023.

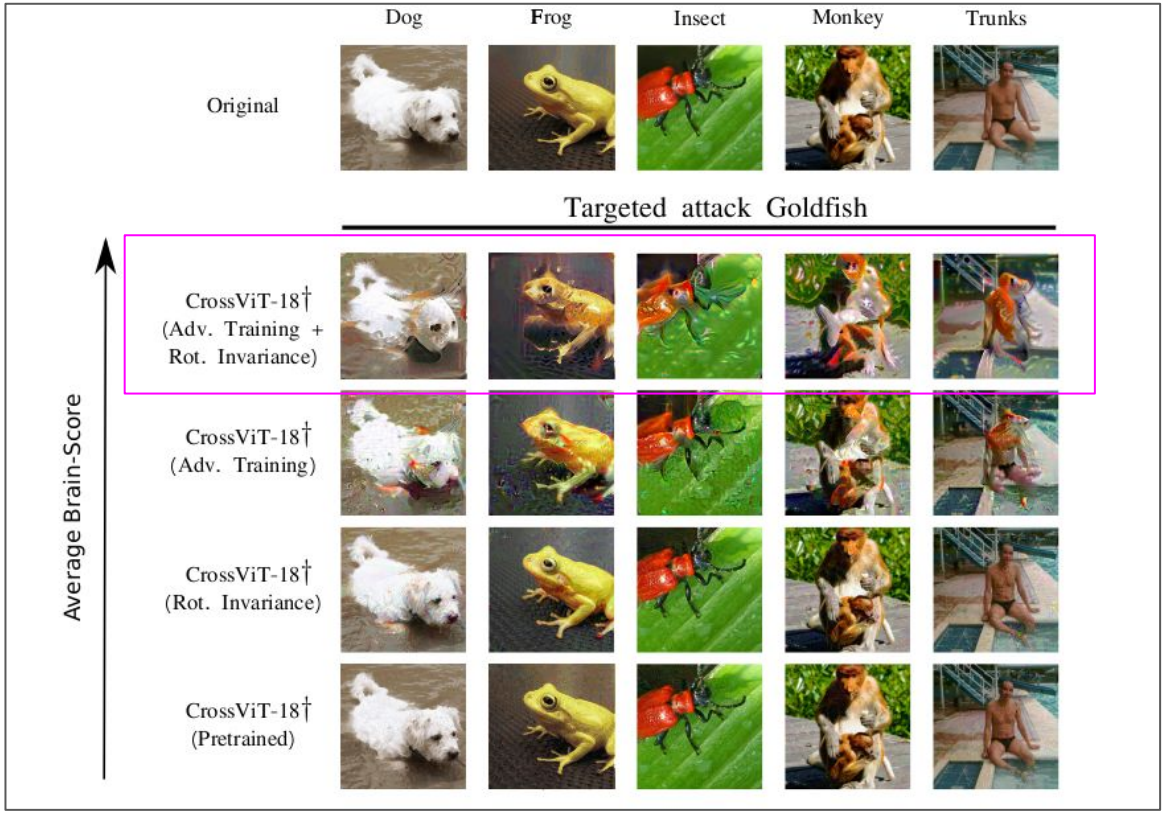
Berrios & Deza. BSW 2022

Testing Human-Machine Perceptual Alignment in Cortex-1 via Adversarial Attacks



Berrios & Deza. BSW 2022

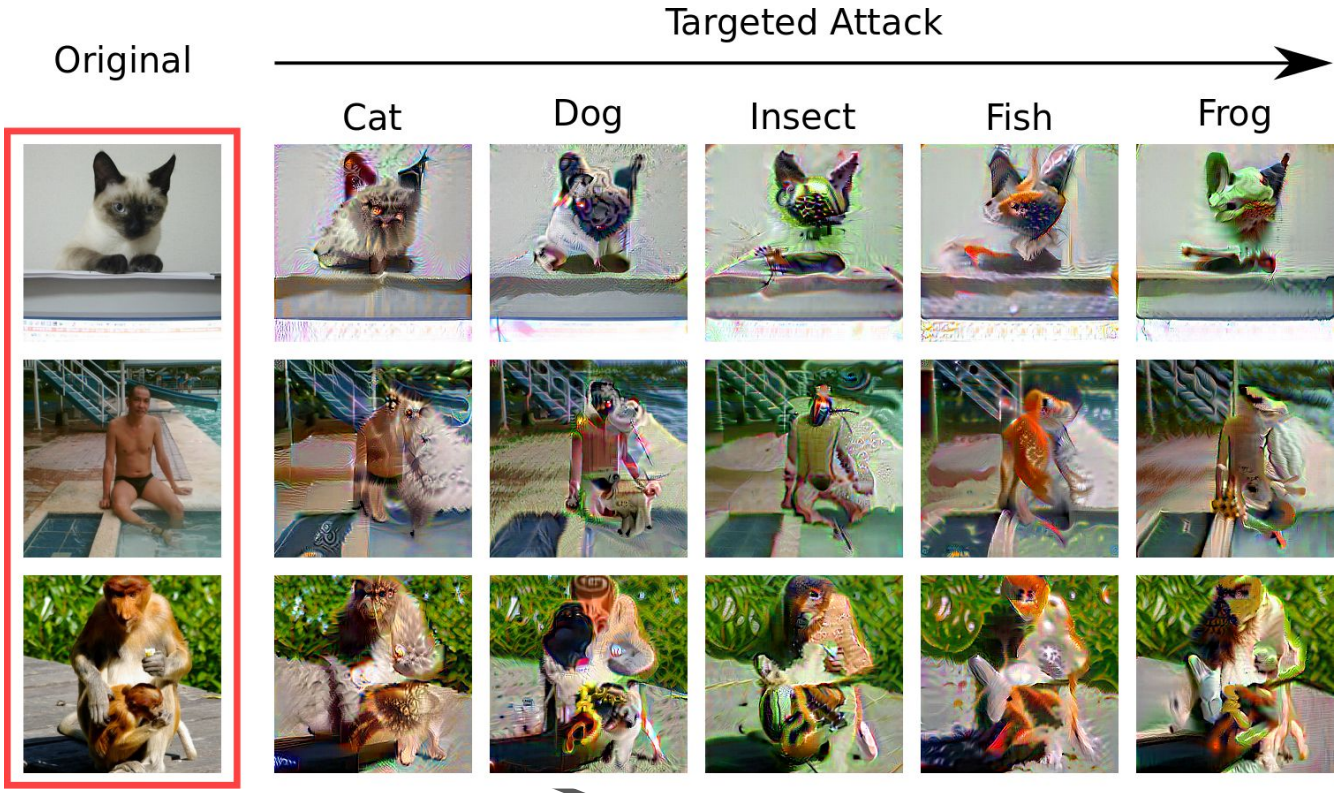
Testing Human-Machine Perceptual Alignment in Cortex-1 via Adversarial Attacks



"As the average Brain-Score increases in our system, the distortions seem to fool a human as well"

Berrios & Deza. BSW 2022

Cortex-1 Model when Adversarially attacked for other classes



Berrios & Deza. BSW 2022

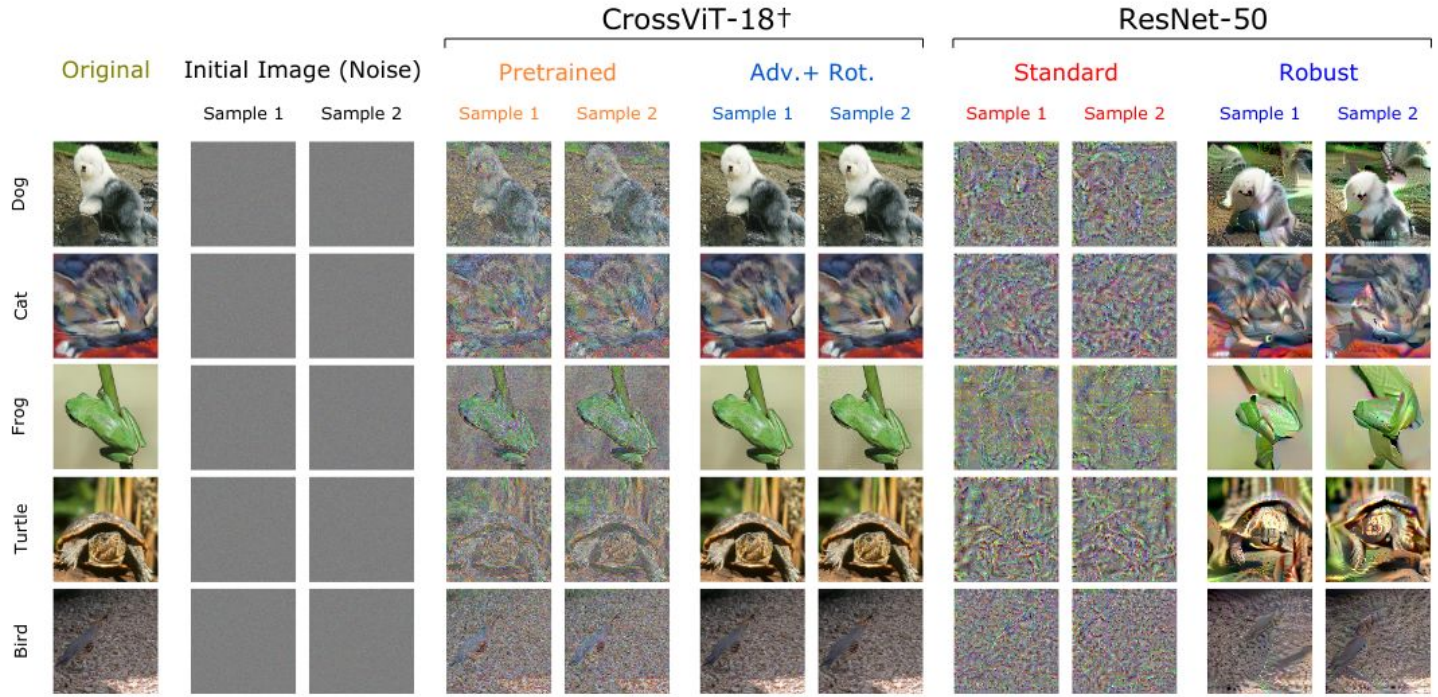


Figure 6: A summary of Feature Inversion models when applied on two different randomly samples noise images from a subset of the stimuli used in Harrington & Deza (2022). Standard and Pretrained models poorly invert the original stimuli leaving high spatial frequency artifacts. Adversarial training improves image inversion models, and this is even more evident for Transformer models. Notice that Transformer models independent of their optimization seem to preserve a higher shape bias as they recover the global structure of the original images. Extended figure can be viewed in the Appendix.

Current Limitations of Cortex-1 : O.O.D. + Adversarial Robustness Inconsistency

Network	Clean Accuracy (↑)	mce (↓)	Gauss	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contrast	Elastic	Pixel	JPEG
ResNet50-Augmix	77.53	67.1	65.5	65.1	66.4	67.7	81	63.9	65.5	71.6	70.9	66.5	57.8	60.2	76.9	59.5	68.5
CrossViT-18† (Adv + Rot)	73.53	79.5	80.7	81.6	83.2	90.2	78.7	82.4	80	77.6	74	107.9	65	100.4	74.2	57.4	58.7
CrossViT-18† (Adv)	64.60	88.8	85	85.7	86.7	96.7	88	92.1	91.3	85.8	83.6	109.3	82.2	104.9	90	70.3	80.9
CrossViT-18† (Rot)	79.22	73.1	75.4	76.7	75	75.7	85.3	72.3	79.2	68.8	70.9	64.3	54.7	67.6	78.4	75.4	76.4
CrossViT-18†	**83.05**	**51**	**46.1**	**48.8**	**46.4**	**61.2**	**72.6**	**54.4**	**65**	**44.9**	**42.1**	**37.2**	**41.5**	**37**	**67.2**	**46.8**	**54.2**

Table 4: A table showing the comparison of mean corruption errors (mce)'s across CrossViT models contingent on their training regime. A ResNet50-Augmix is shown as a reference of a particularly strong model to common corruptions. Here lower scores are indicative of better robustness to the different distortion types of Hendrycks & Dietterich (2019).