Fundamentals of Large Language Models

Embeddings are all you need

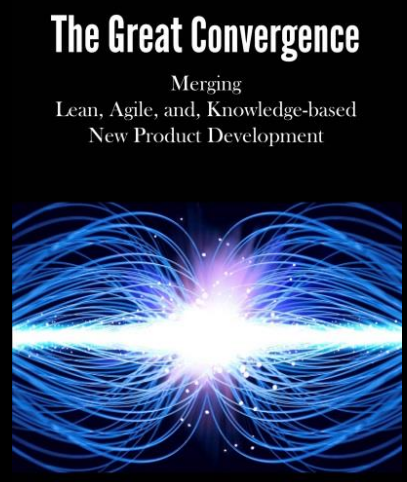
Steve Zielinski, CSEP

Currently consulting on neuromodulation device development.

Previously

Vice President, Product Development, Bioelectronic Therapies at Orchestra BioMed

BA University of St. Thomas
BEE University of Minnesota
MSDD University of St. Thomas



Goals

You should walk away with:

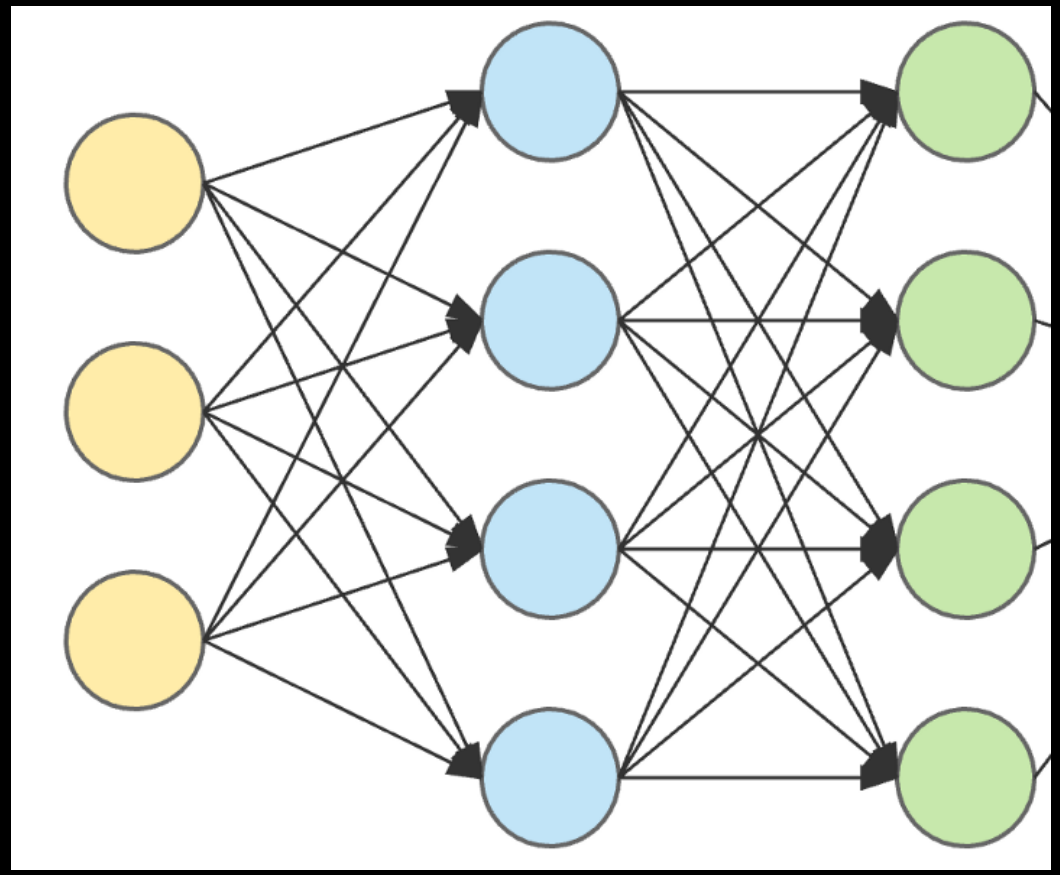
1. A deeper understanding of how ChatGPT works
2. Knowing what a Transformer is
3. A little scared/excited about the future.

Natural Language Processing

Starting Goal: Word Prediction

I want a glass of orange _____.

I want a glass of apple_____.



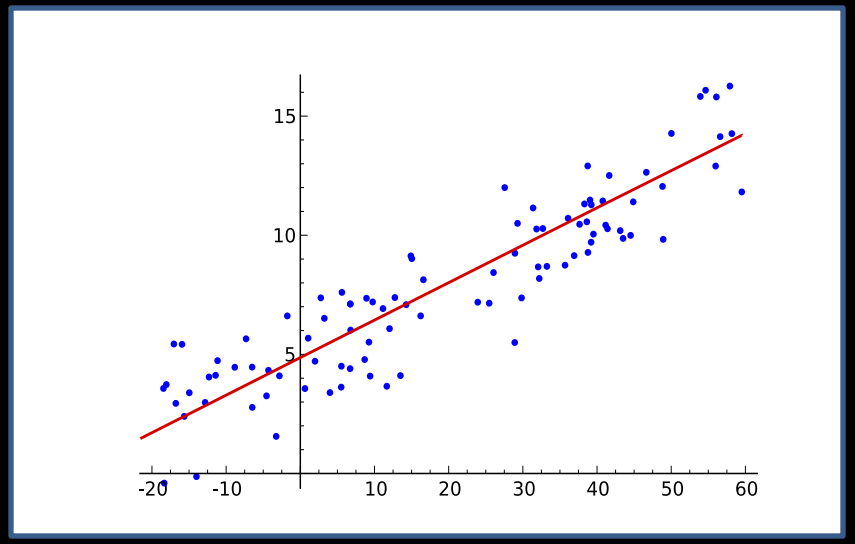
0.001

0.06

.5

.03

The NN is making a prediction about the most probable next word.



The Big Problem:

Computers don’t operate on words. They operate on numbers. How do you represent words as numbers?

Start with a vocabulary

$$V = [a, aaron, \ldots, zulu]$$

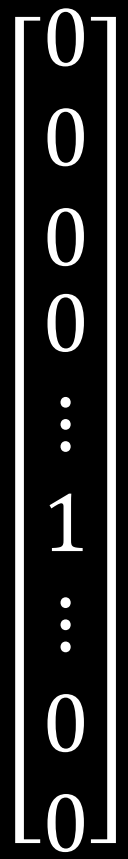
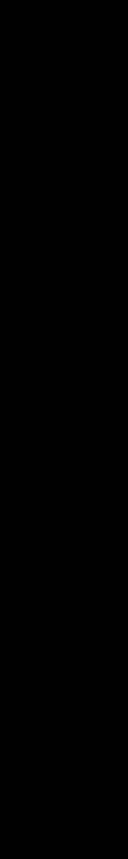
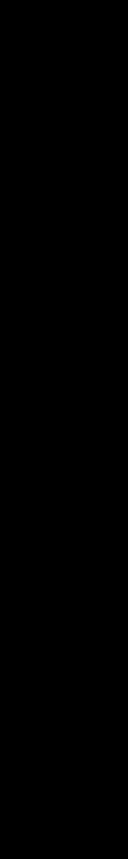
These vocabularies can be anywhere from 10,000 to 100,000 (or more) words.

You'll frequently hear numbers in the 30,000 – 50,000 word range.

GPT-3 used 50,257 words

(Actually 50,257 tokens)

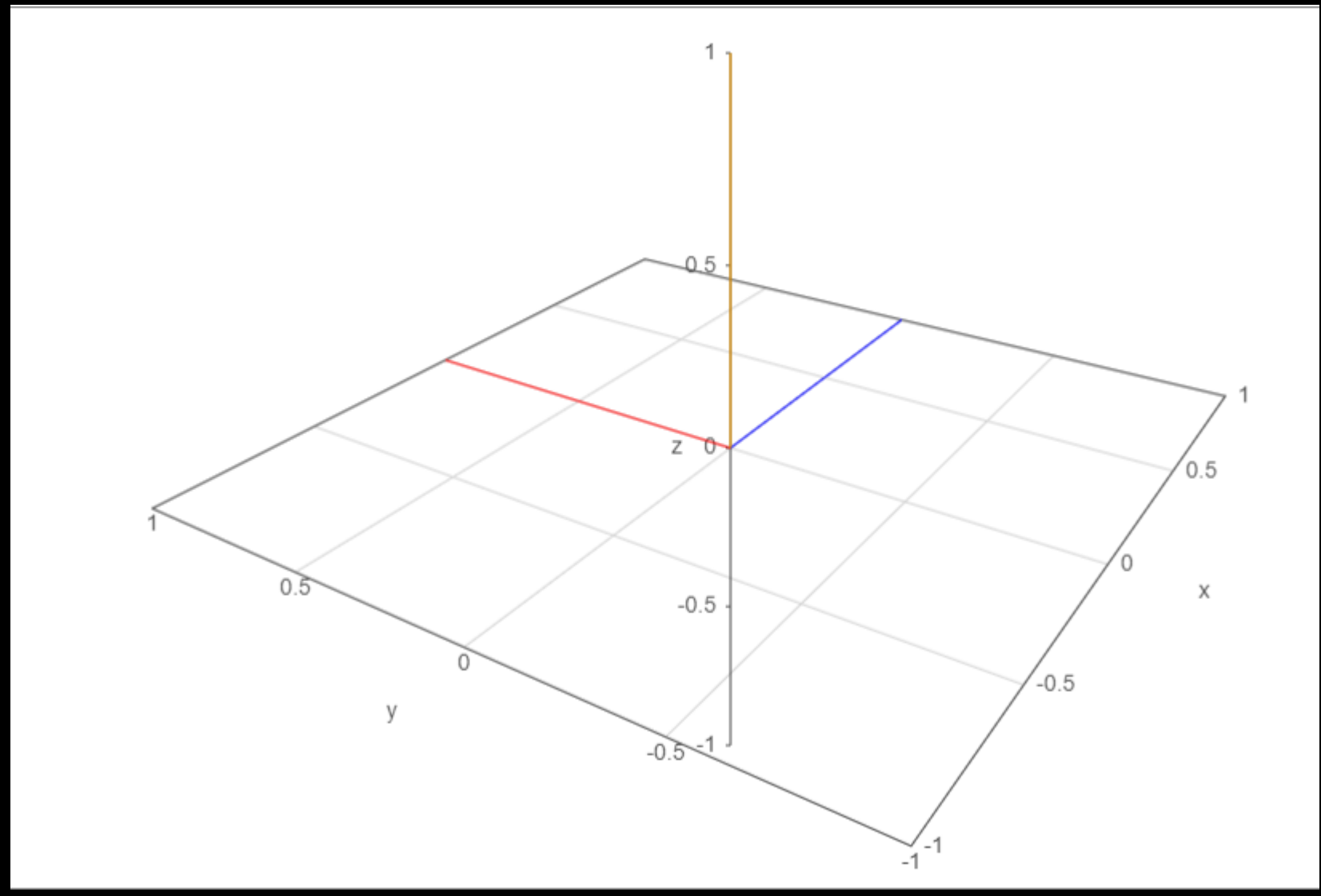
1-hot representation

man (5391)	woman (9853)	king (4914)	orange (6257)
			

One-hot doesn’t work very well

Every word is orthogonal to every other word.

The “distance” between each word is the same.



Featurized representation: word embedding

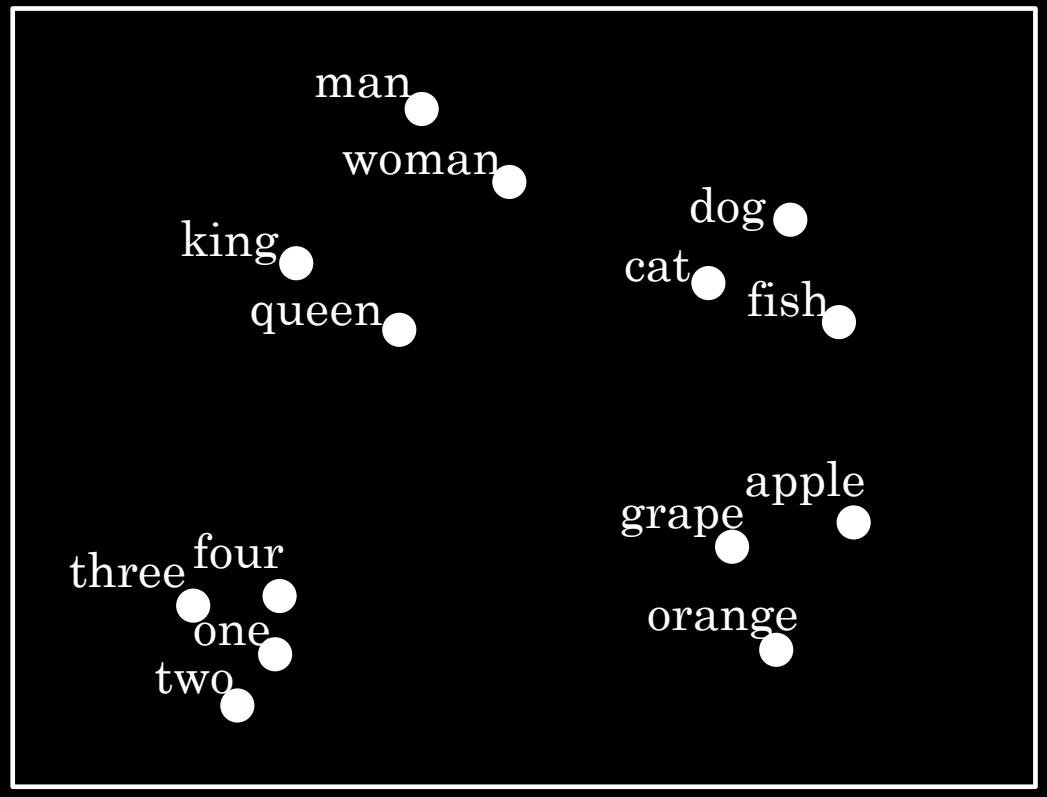
	Man (5391)	Woman (9853)	King (4914)	Queen (7157)	Apple (456)	Orange (6257)
Gender	-1	1	-0.95	0.97	0.00	0.01
Royal	0.01	0.02	0.93	0.95	-0.01	0.00
Age	0.03		0.7	0.69	0.03	-0.02
Food			0.02	0.01	0.95	0.97

Embedding values are not assigned like this picture might imply.

They are learned from large amounts of text. Similar items congregate in the same area of space.

	Man (5391)	Woman (9853)	King (4914)	Queen (7157)	Apple (456)	Orange (6257)
Gender	-1	1	-0.95	0.97	0.00	0.01
Royal	0.01	0.02	0.93	0.95	-0.01	0.00
Age	0.03		0.7	0.69	0.03	-0.02
Food			0.02	0.01	0.95	0.97

Visualizing word embeddings



[van der Maaten and Hinton., 2008. Visualizing data using t-SNE]

GPT-3 uses word embeddings with 12,288 characteristics

12,288-dimensional space

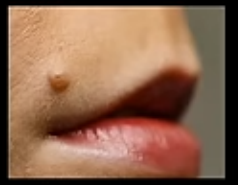
The goal of a transformer is to move the location of the starting embedding vector with information found in other parts of the sentence or document.

Context changes meaning

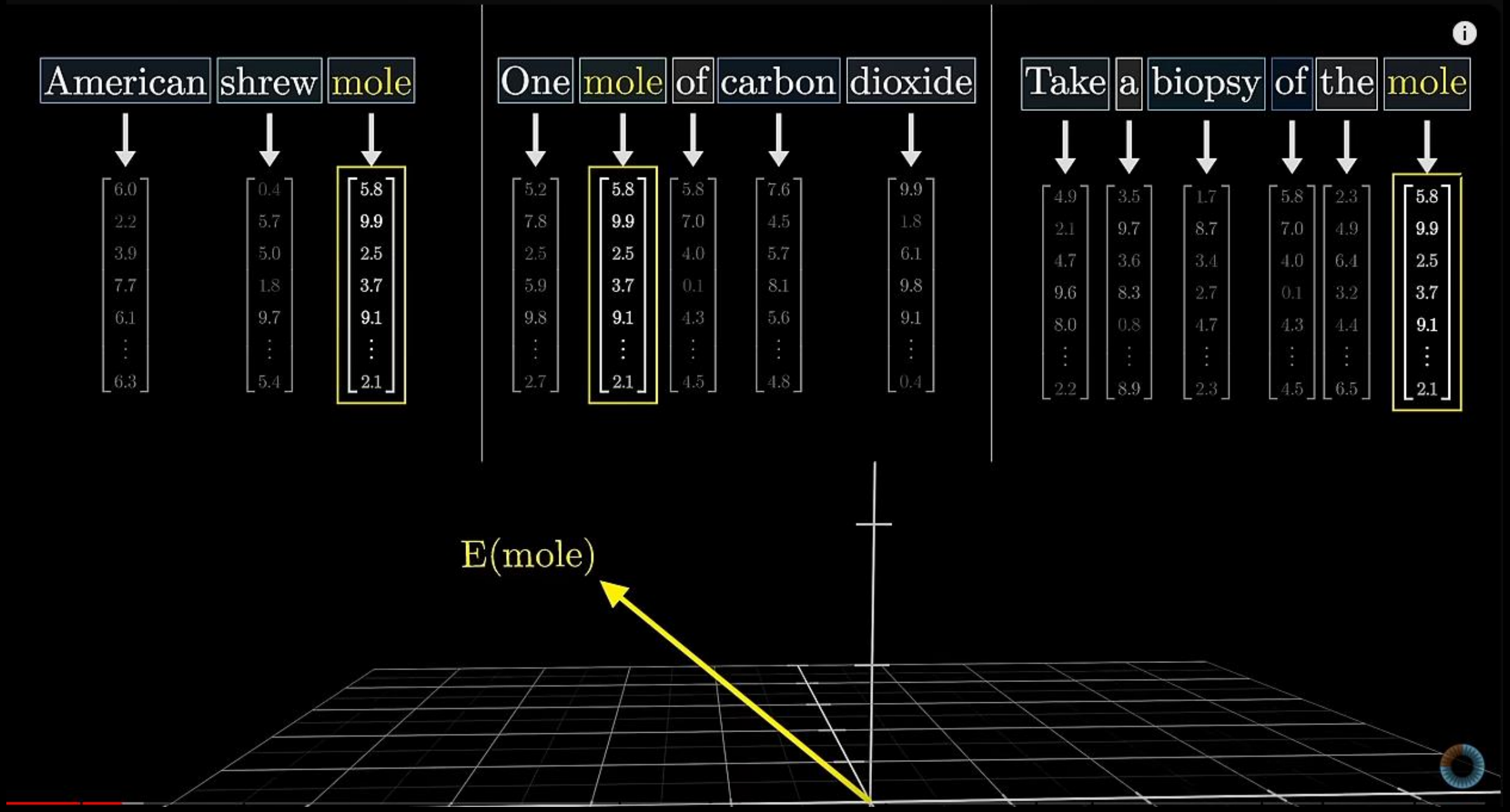
American shrew mole

6.02×10^{23}

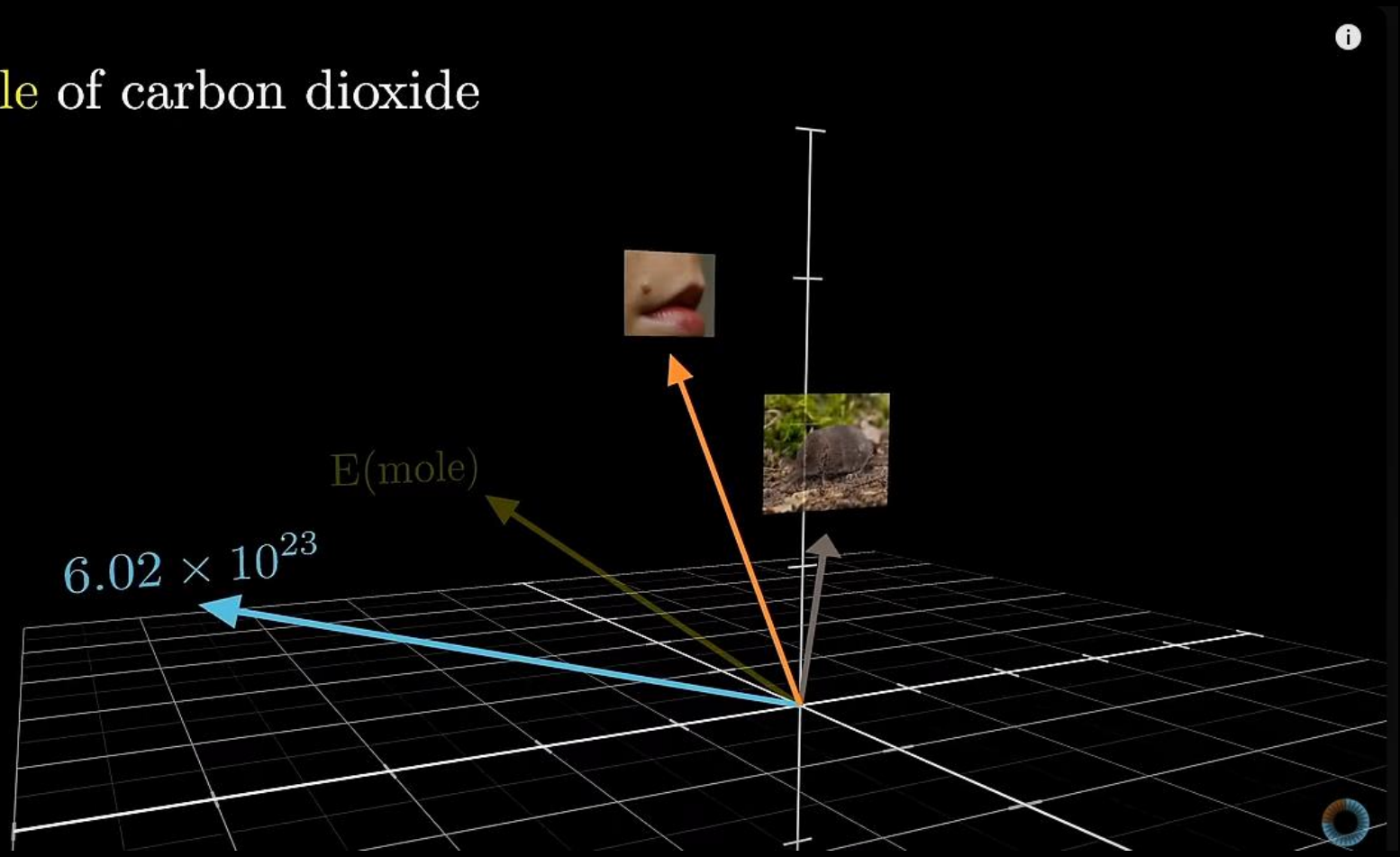
One mole of carbon dioxide



Take a biopsy of the mole


American shrew mole
6.0 2.2 3.9 7.7 6.1 ⋮ 6.3
0.4 5.7 5.0 1.8 9.7 ⋮ 5.4
5.8 9.9 2.5 3.7 9.1 ⋮ 2.1
One mole of carbon dioxide
5.2 7.8 2.5 5.9 9.8 ⋮ 2.7
5.8 9.9 2.5 3.7 9.1 ⋮ 2.1
5.8 7.0 4.0 0.1 4.3 ⋮ 4.5
7.6 4.5 5.7 8.1 5.6 ⋮ 4.8
9.9 1.8 6.1 9.8 9.1 ⋮ 0.4
Take a biopsy of the mole
4.9 2.1 4.7 9.6 8.0 ⋮ 2.2
3.5 9.7 3.6 8.3 0.8 ⋮ 8.9
1.7 8.7 3.4 2.7 4.7 ⋮ 2.3
5.8 7.0 4.0 0.1 4.3 ⋮ 4.5
2.3 4.9 6.4 3.2 4.4 ⋮ 6.5
5.8 9.9 2.5 3.7 9.1 ⋮ 2.1
E(mole)

One mole of carbon dioxide



The attention block “moves” the vector for the word to different locations in the embedding space.

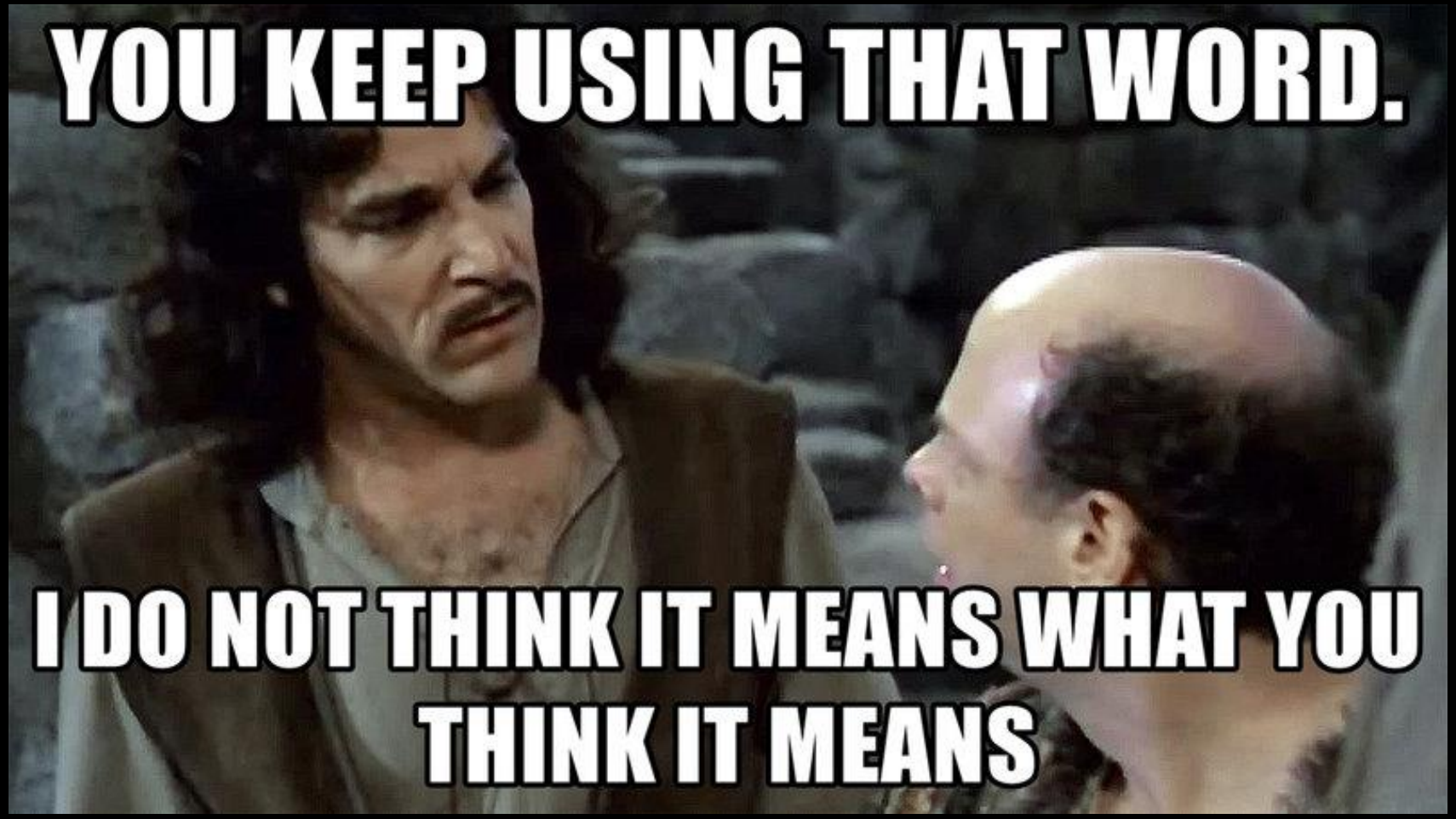
Transformer

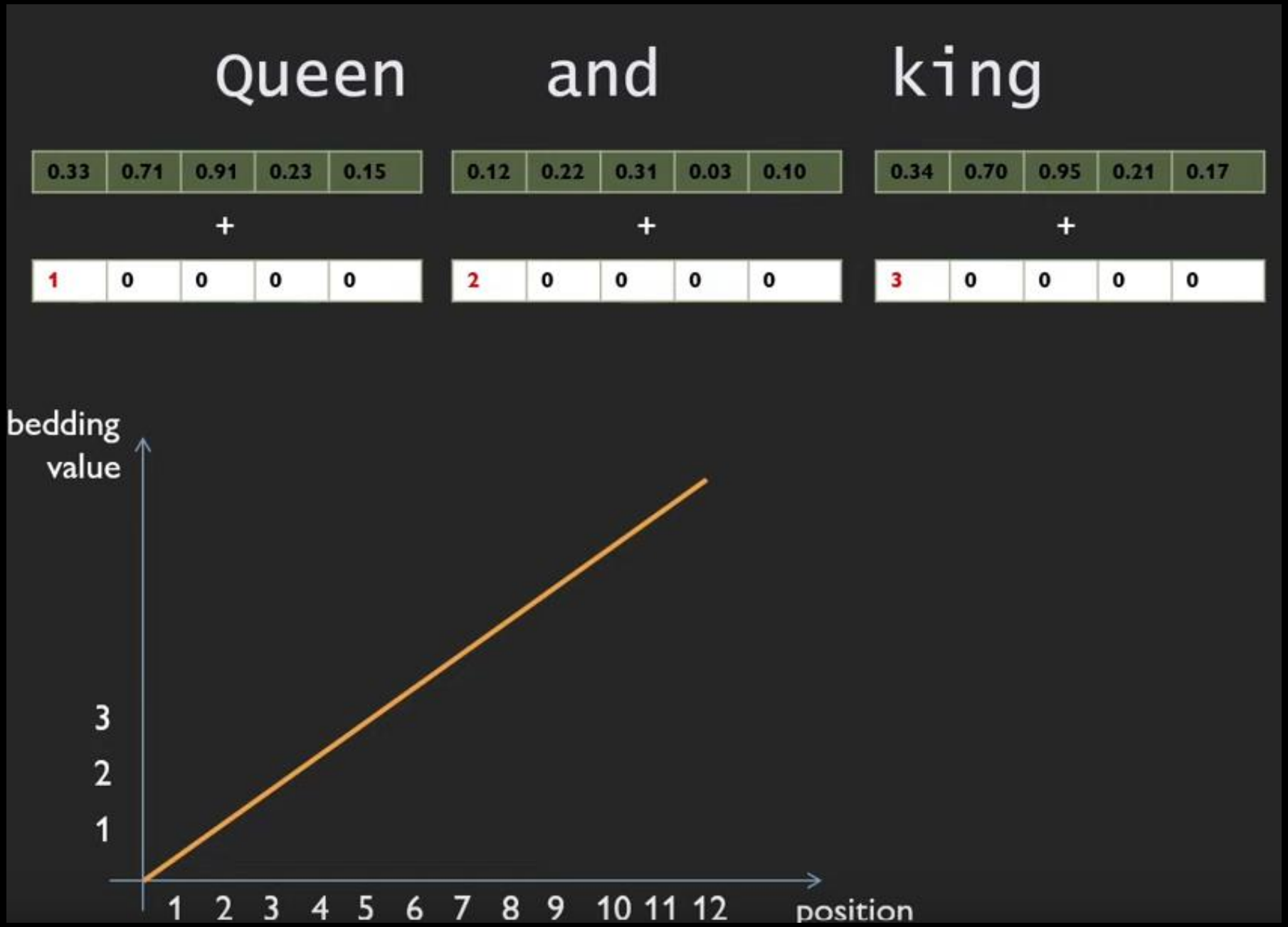
- Innovation #1

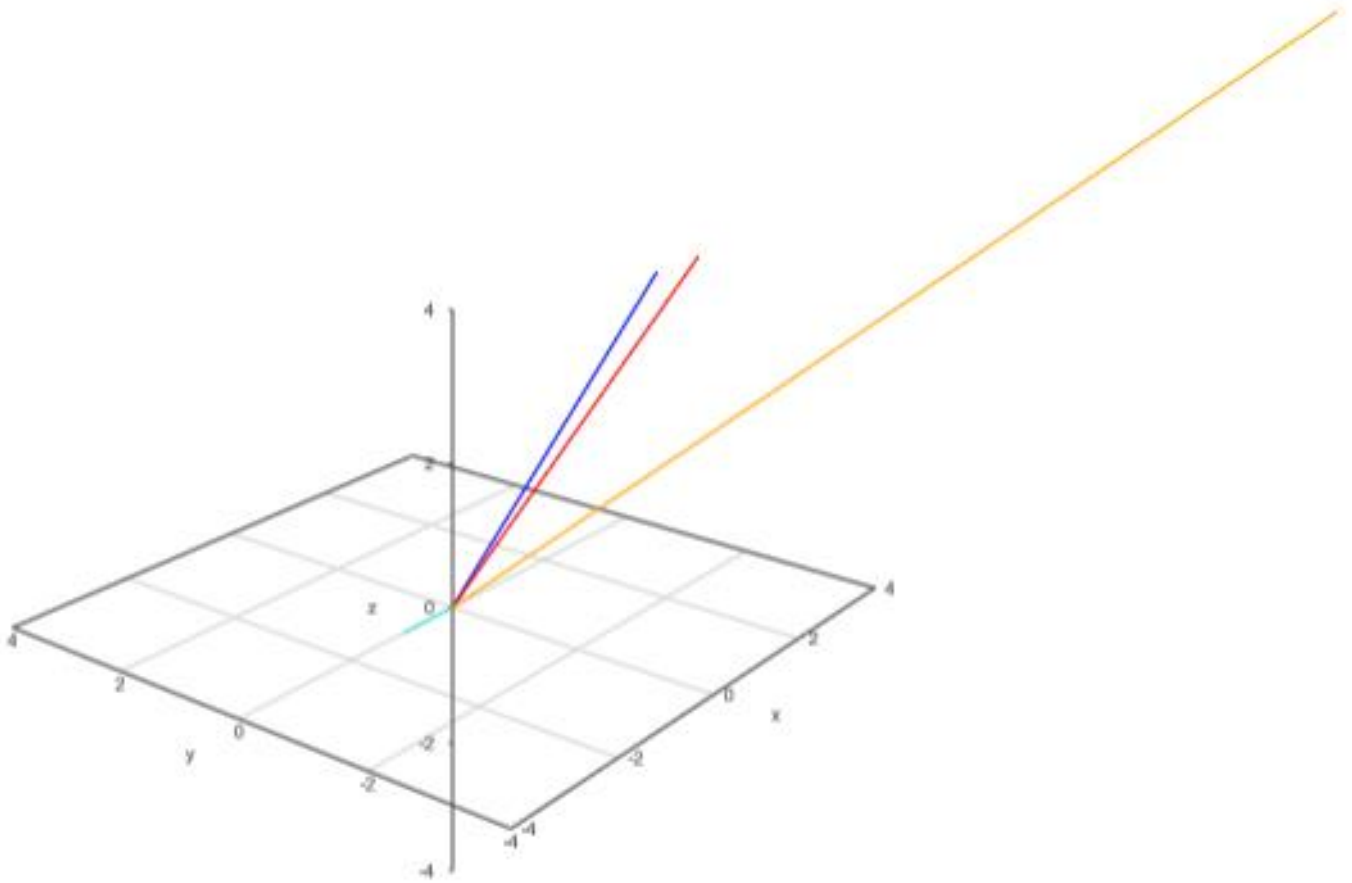
	One	mole	of	carbon	dioxide
0	0		-0.95	0.97	0.00
0.01	0.02		0.93	0.95	-0.01
0.03			0.7	0.69	0.03
			0.02	0.01	0.95

Position

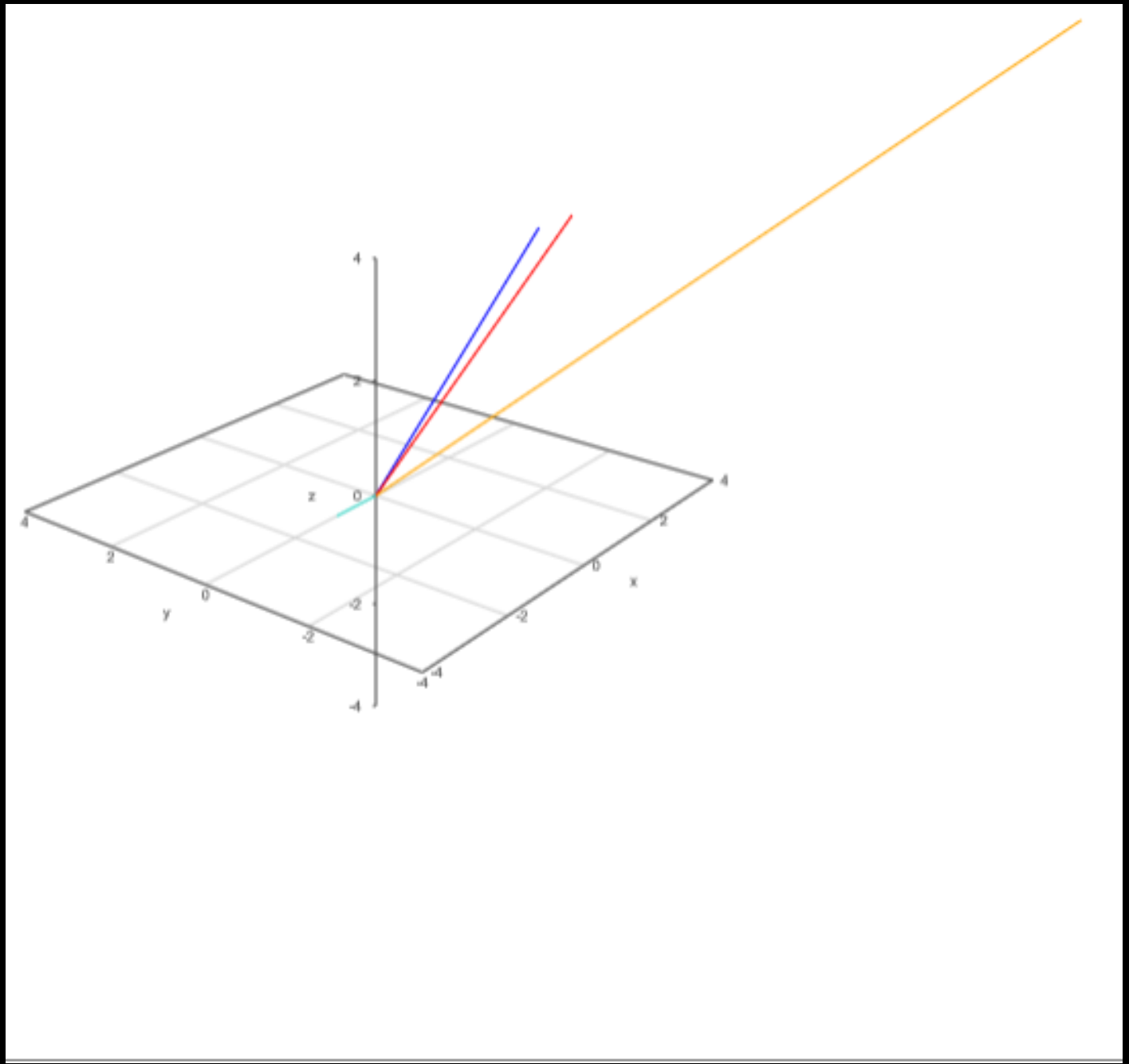
Position


YOU KEEP USING THAT WORD.
I DO NOT THINK IT MEANS WHAT YOU THINK IT MEANS


Queen
and
king
0.33 0.71 0.91 0.23 0.15
+
1 0 0 0 0
0.12 0.22 0.31 0.03 0.10
+
2 0 0 0 0
0.34 0.70 0.95 0.21 0.17
+
3 0 0 0 0
bedding value
3
2
1
1 2 3 4 5 6 7 8 9 10 11 12
position


4
2
0
-2
-4
z
0
4
2
0
-2
-4
x
4
2
0
-2
-4
y

The word position in the sentence is overwhelming the semantics.



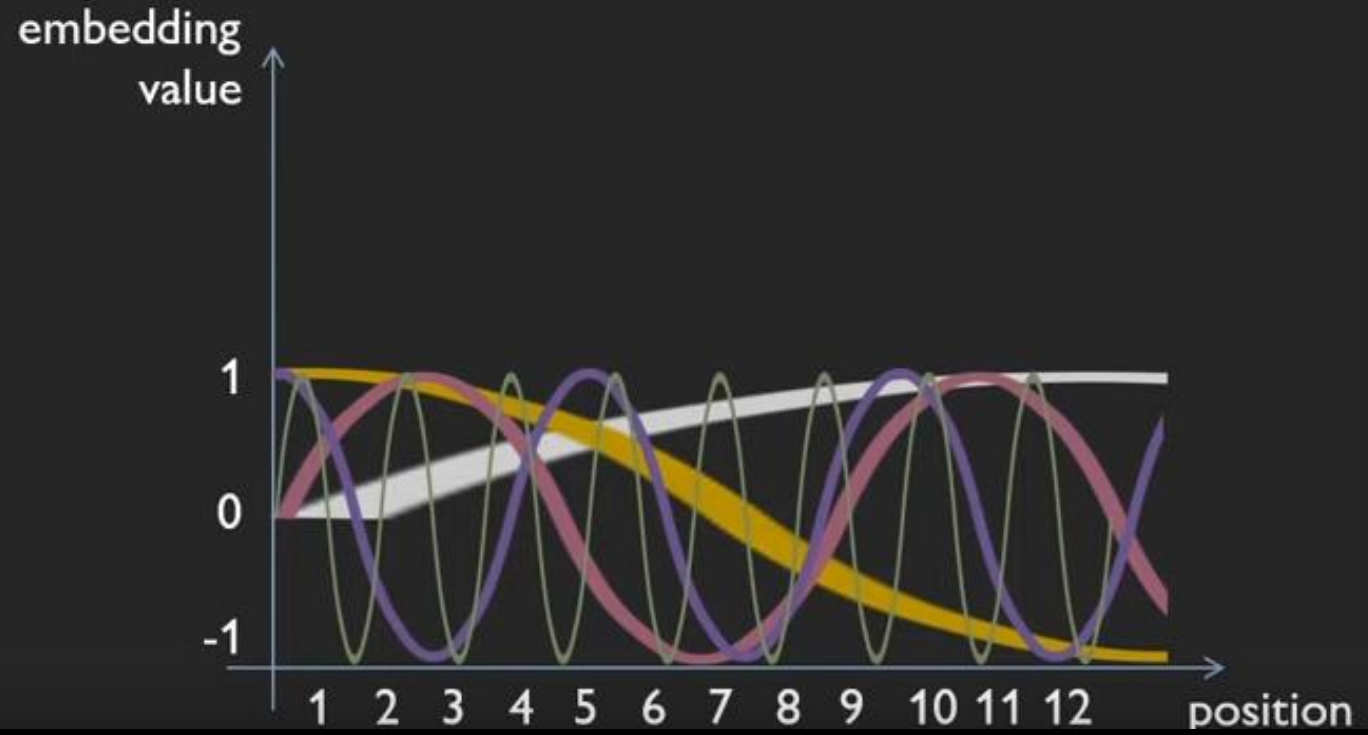
$$PE(\text{position}, 2i) = \sin\left(\frac{\text{position}}{10000^{\frac{2i}{d_{model}}}}\right)$$

$$PE(\text{position}, 2i + 1) = \cos\left(\frac{\text{position}}{10000^{\frac{2i}{d_{model}}}}\right)$$

Queen				
0.33	0.71	0.91	0.23	0.15
+				
0.2	0.7	0.1	0.99	0.01

and				
0.12	0.22	0.31	0.03	0.10
+				
0.01	-0.8	0.9	0.80	0.02

king				
0.34	0.70	0.95	0.21	0.17
+				
-0.9	-0.7	0.8	0.70	0.03




Queen
and
king
0.33 0.71 0.91 0.23 0.15
0.12 0.22 0.31 0.03 0.10
0.34 0.70 0.95 0.21 0.17
+
+
+
0.01 0.01 0.01 0.01 0.01
0.02 0.21 0.33 0.43 0.98
0.03 0.32 0.85 0.91 0.24


Prince
[PAD]
[PAD]
0.23 0.72 0.62 0.21 0.17
0.00 0.00 0.00 0.00 0.00
0.00 0.00 0.00 0.00 0.00
+
+
+
0.01 0.01 0.01 0.01 0.01
0.02 0.21 0.33 0.43 0.98
0.03 0.32 0.85 0.91 0.24

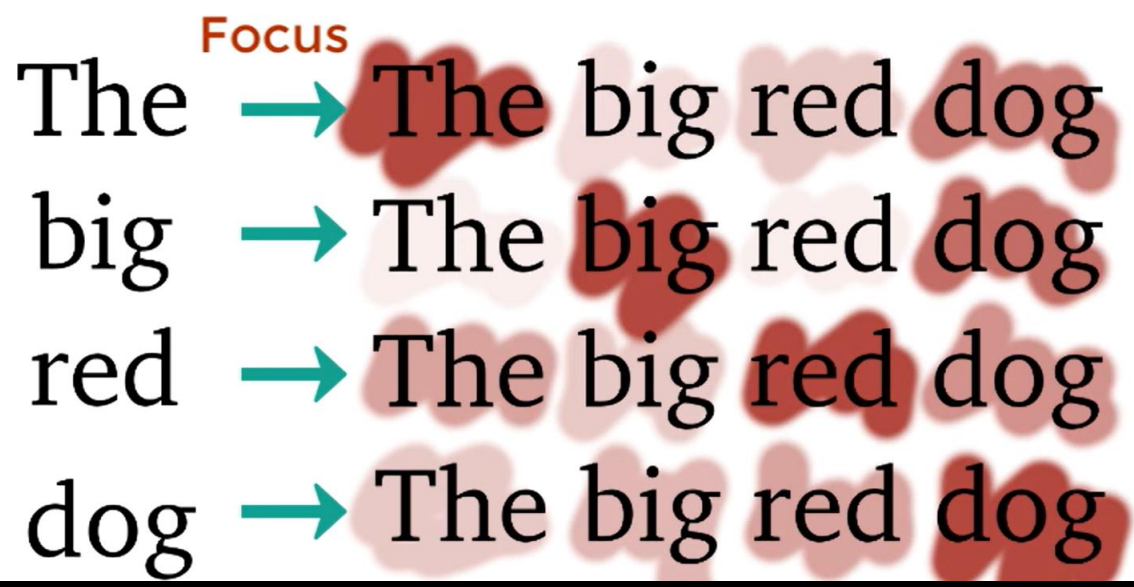
Parallel computations

- Because word position is now embedded in the vector, I can operate on all the input words in parallel!

- This is why GPUs are conquering the world!

Transformer Components

Attention : What part of the input should we focus?



Attention Vectors

$[0.71 \quad 0.04 \quad 0.07 \quad 0.18]^T$

$[0.01 \quad 0.84 \quad 0.02 \quad 0.13]^T$

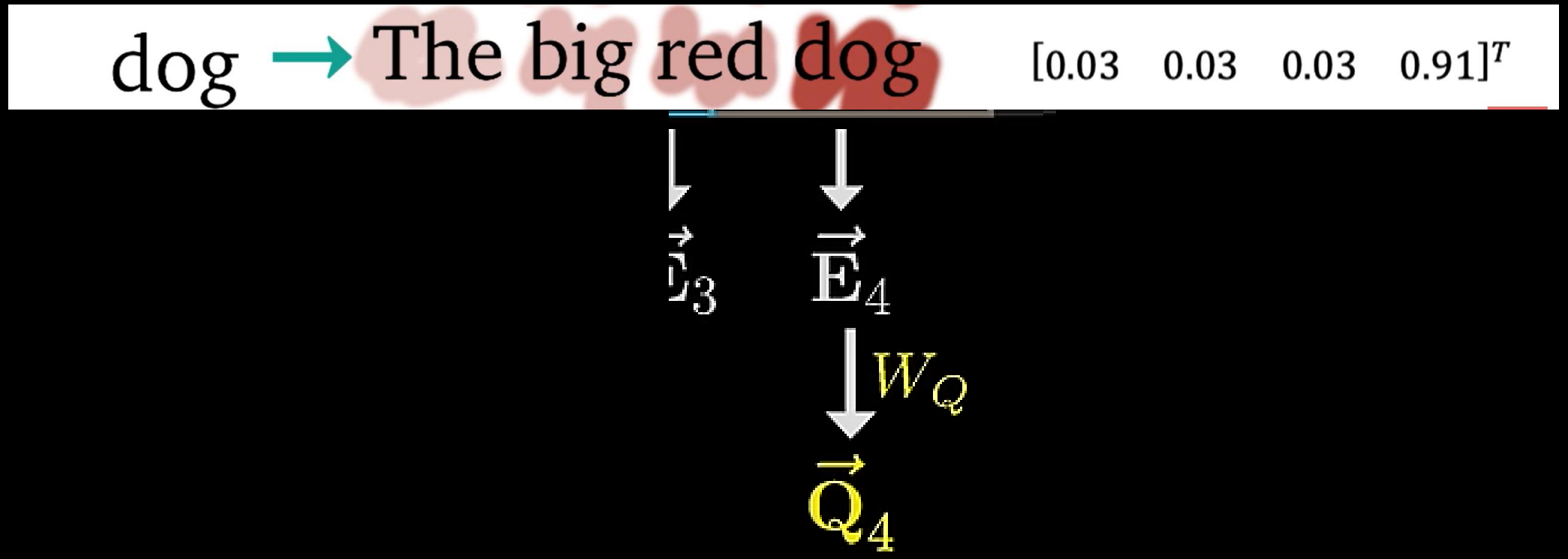
$[0.09 \quad 0.05 \quad 0.62 \quad 0.24]^T$

$[0.03 \quad 0.03 \quad 0.03 \quad 0.91]^T$

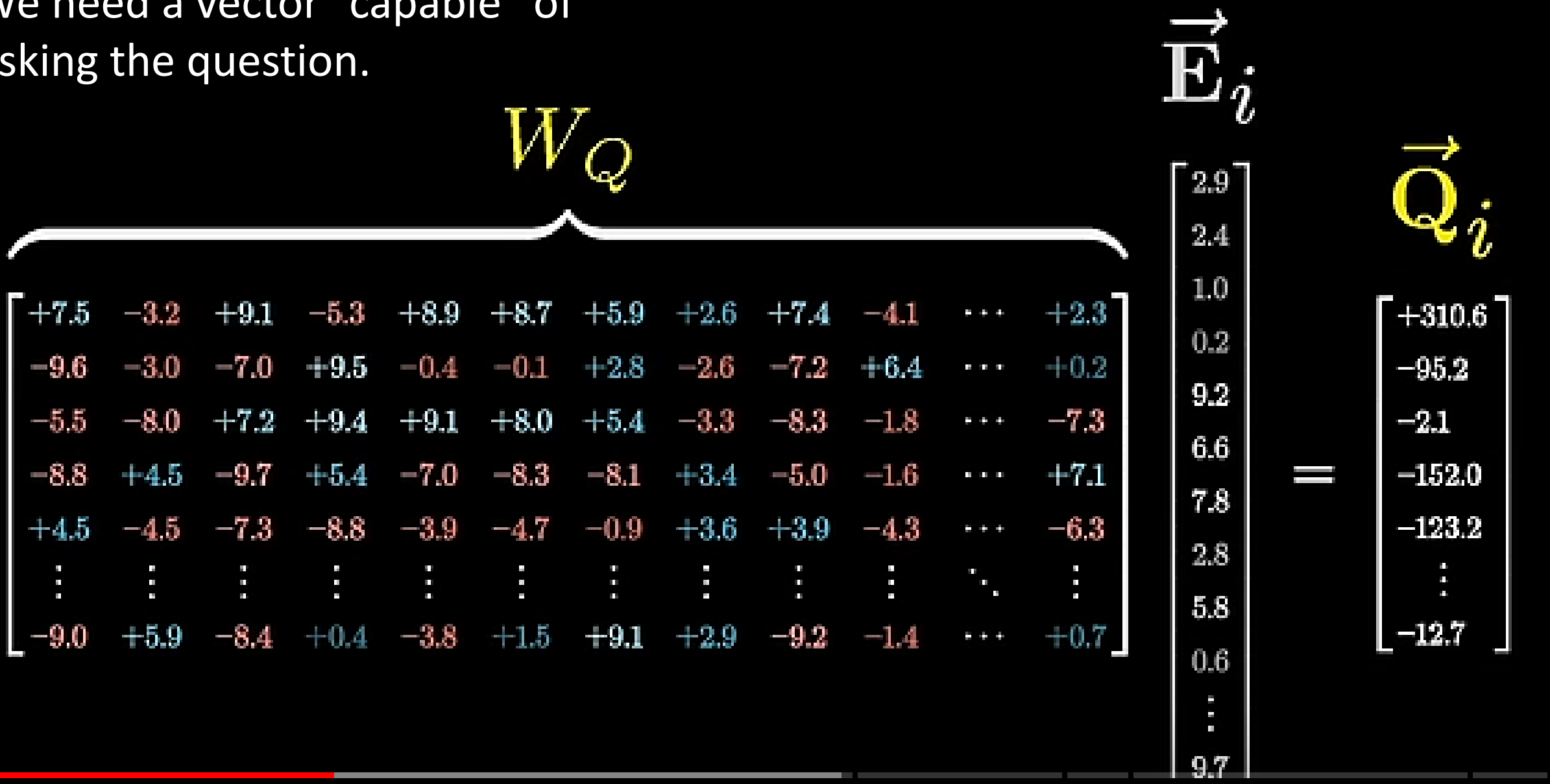
The attention vectors establish a relationship between words.

QUERIES, KEYS, AND VALUES (OH, MY!)

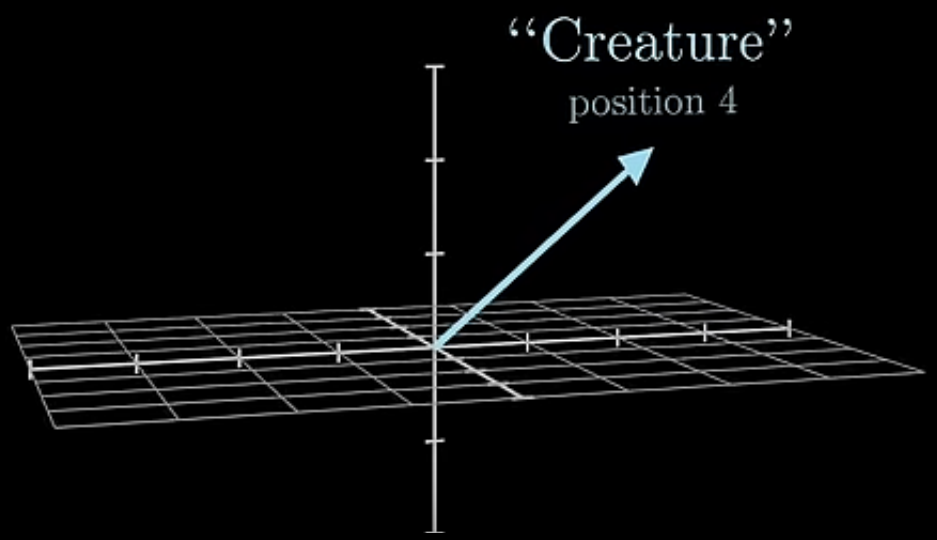
Query: A word “asks”: “What is in front of me?”



We need a vector “capable” of asking the question.

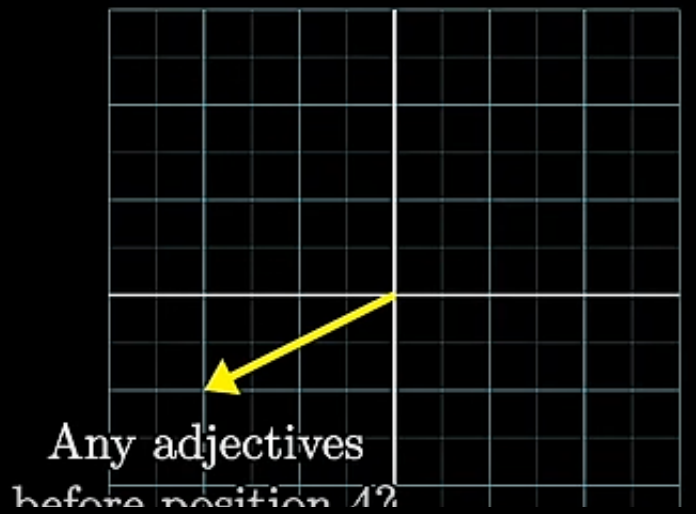


Embedding space
12,288-dimensional

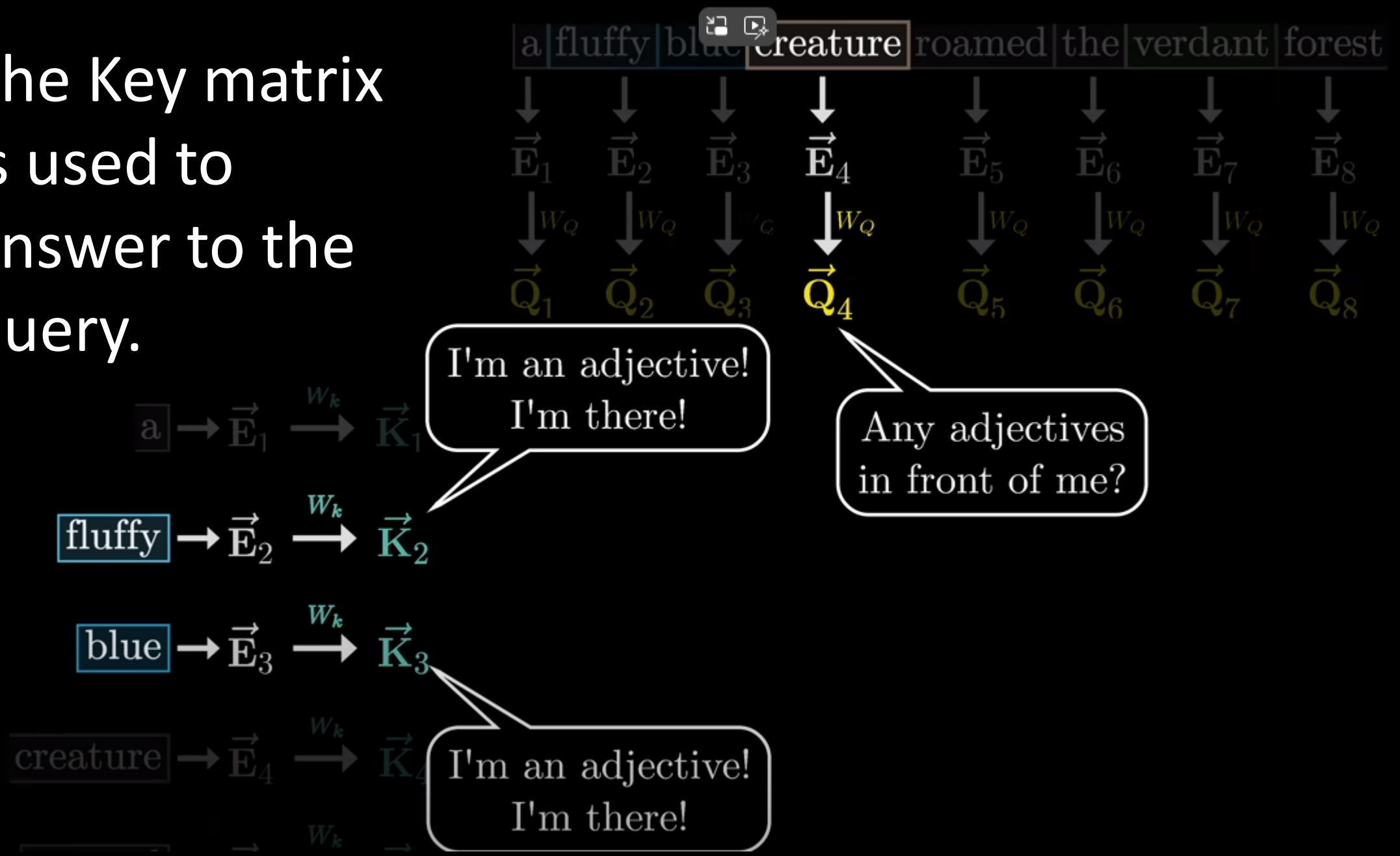


W_Q

Query/Key space
128-dimensional



The Key matrix is used to answer to the query.



a → $\vec{E}_1$ —W_k→ $\vec{K}_1$

fluffy → $\vec{E}_2$ —W_k→ $\vec{K}_2$

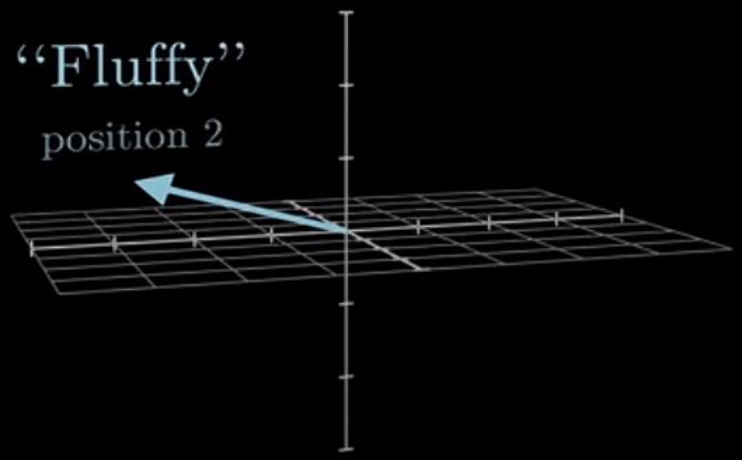
blue → $\vec{E}_3$ —W_k→ $\vec{K}_3$

creature → $\vec{E}_4$ —W_k→ $\vec{K}_4$

W_k

Embedding space

12,288-dimensional



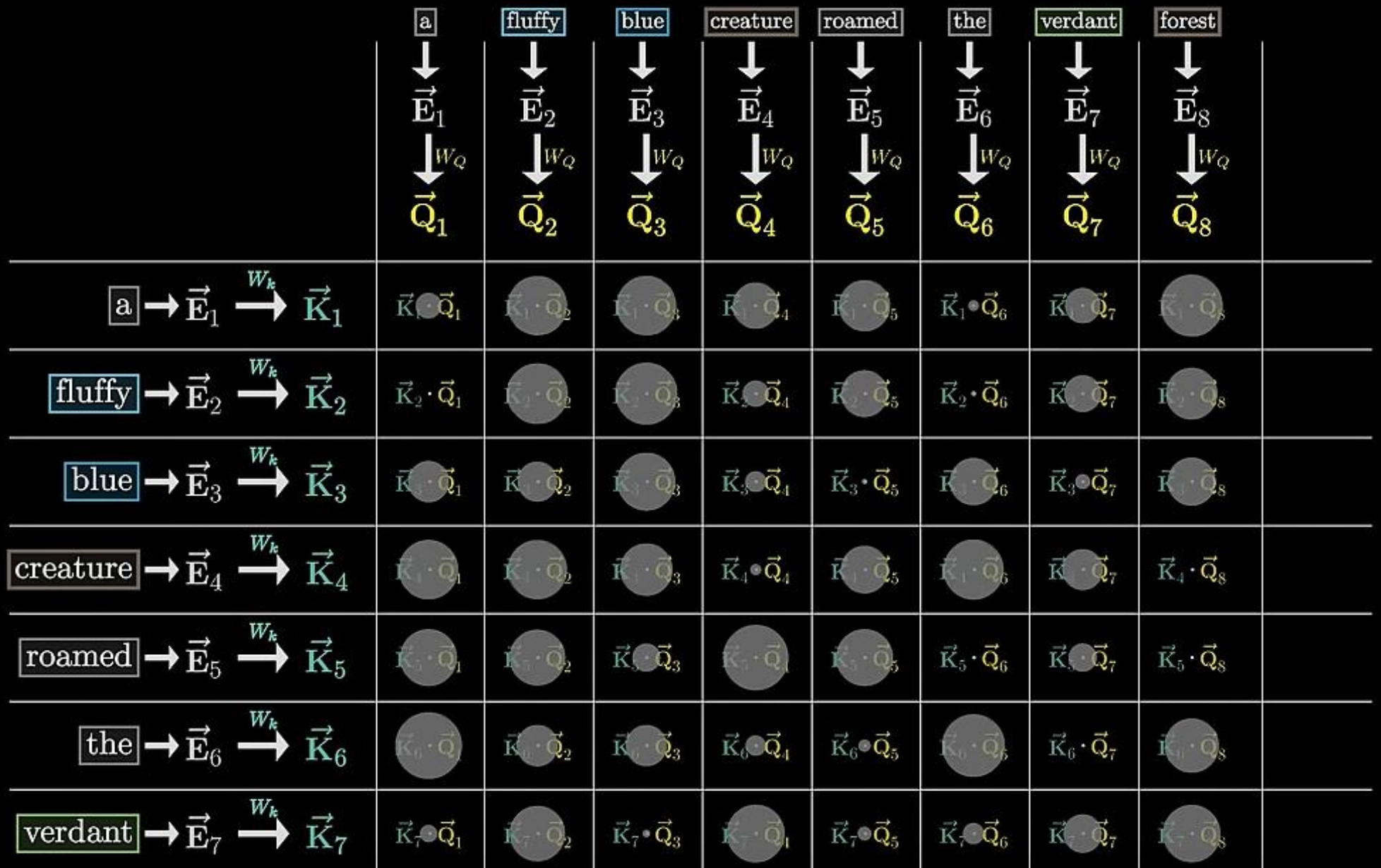
W_K

Query/Key space

128-dimensional

Whoa! Hold on. Where did we get these Query and Key matrices?

- But, we haven’t answered how “good” an answer the key is to the query.
- For that we use a dot product.
- This is referred to as the “similarity” between the words.

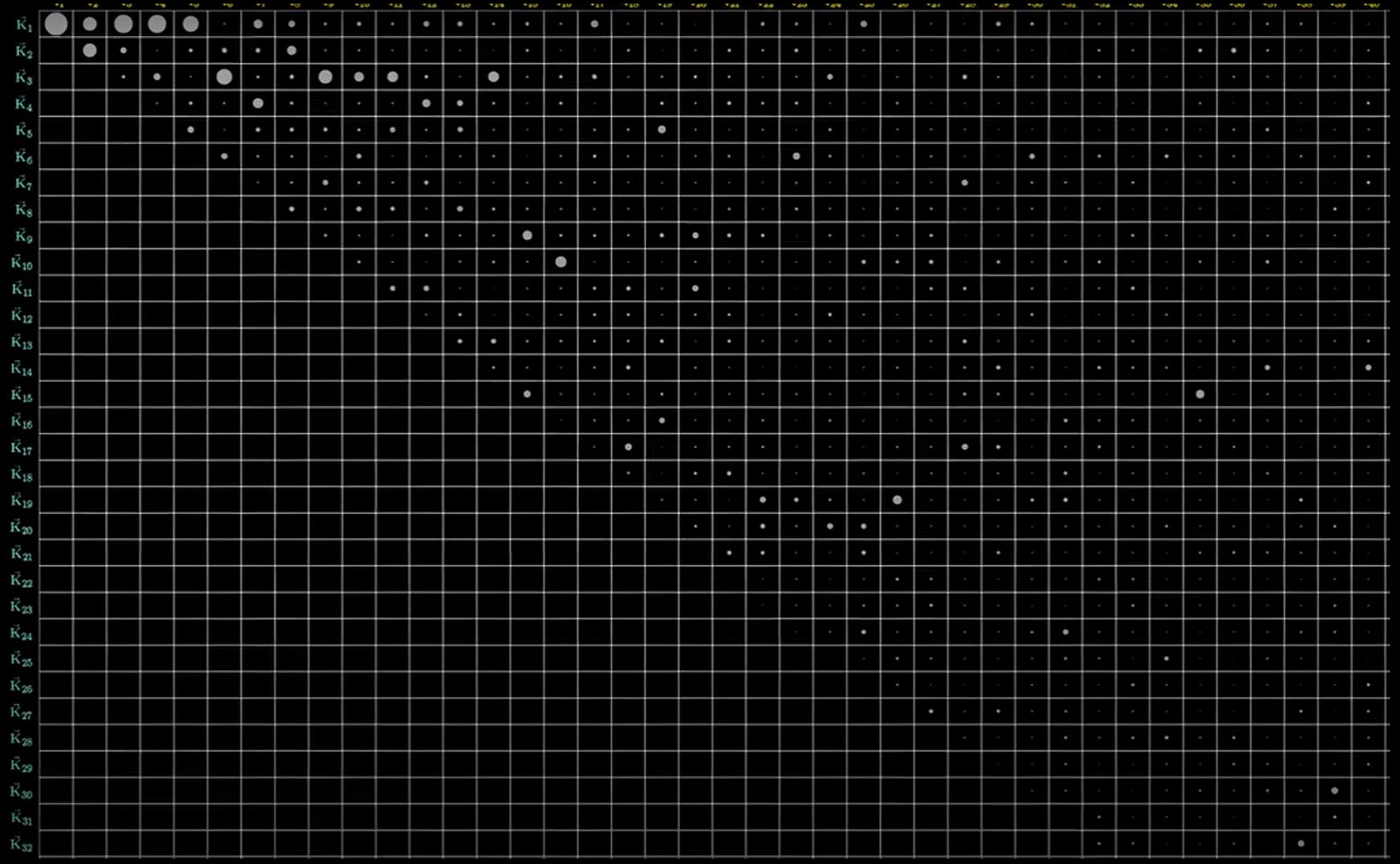

a
fluffy
blue
creature
roamed
the
verdant
forest
$\vec{E}_1$ $\vec{E}_2$ $\vec{E}_3$ $\vec{E}_4$ $\vec{E}_5$ $\vec{E}_6$ $\vec{E}_7$ $\vec{E}_8$
W_Q
$\vec{Q}_1$ $\vec{Q}_2$ $\vec{Q}_3$ $\vec{Q}_4$ $\vec{Q}_5$ $\vec{Q}_6$ $\vec{Q}_7$ $\vec{Q}_8$
a → $\vec{E}_1$ W_k → $\vec{K}_1$
fluffy → $\vec{E}_2$ W_k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

	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 $\vec{E}_1 \xrightarrow{W_K} \vec{K}_1$	+0.7	−83.7	−24.7	−27.8	−5.2	−89.3	−45.2	−36.1
fluffy $\vec{E}_2 \xrightarrow{W_K} \vec{K}_2$	−73.4	+2.9	−5.4	+93.0	−48.2	−87.3	−49.7	+7.8
blue $\vec{E}_3 \xrightarrow{W_K} \vec{K}_3$	−53.4	−5.7	+1.8	+93.4	−55.6	−56.0	−26.1	−62.1
creature $\vec{E}_4 \xrightarrow{W_K} \vec{K}_4$	−21.5	−29.7	−56.1	+4.9	−32.4	−92.3	−9.5	−28.1
roamed $\vec{E}_5 \xrightarrow{W_K} \vec{K}_5$	−20.1	−40.9	−87.8	−55.4	+0.6	−64.7	−96.7	−18.9
the $\vec{E}_6 \xrightarrow{W_K} \vec{K}_6$	−87.9	−33.3	−22.6	−31.4	+5.5	+0.6	−4.6	−96.8
verdant $\vec{E}_7 \xrightarrow{W_K} \vec{K}_7$	−41.2	−55.5	−42.3	−59.8	−79.0	−97.9	+3.7	+93.8

The the columns get normalized.

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

Attention matrix size is N^2 of the context window!

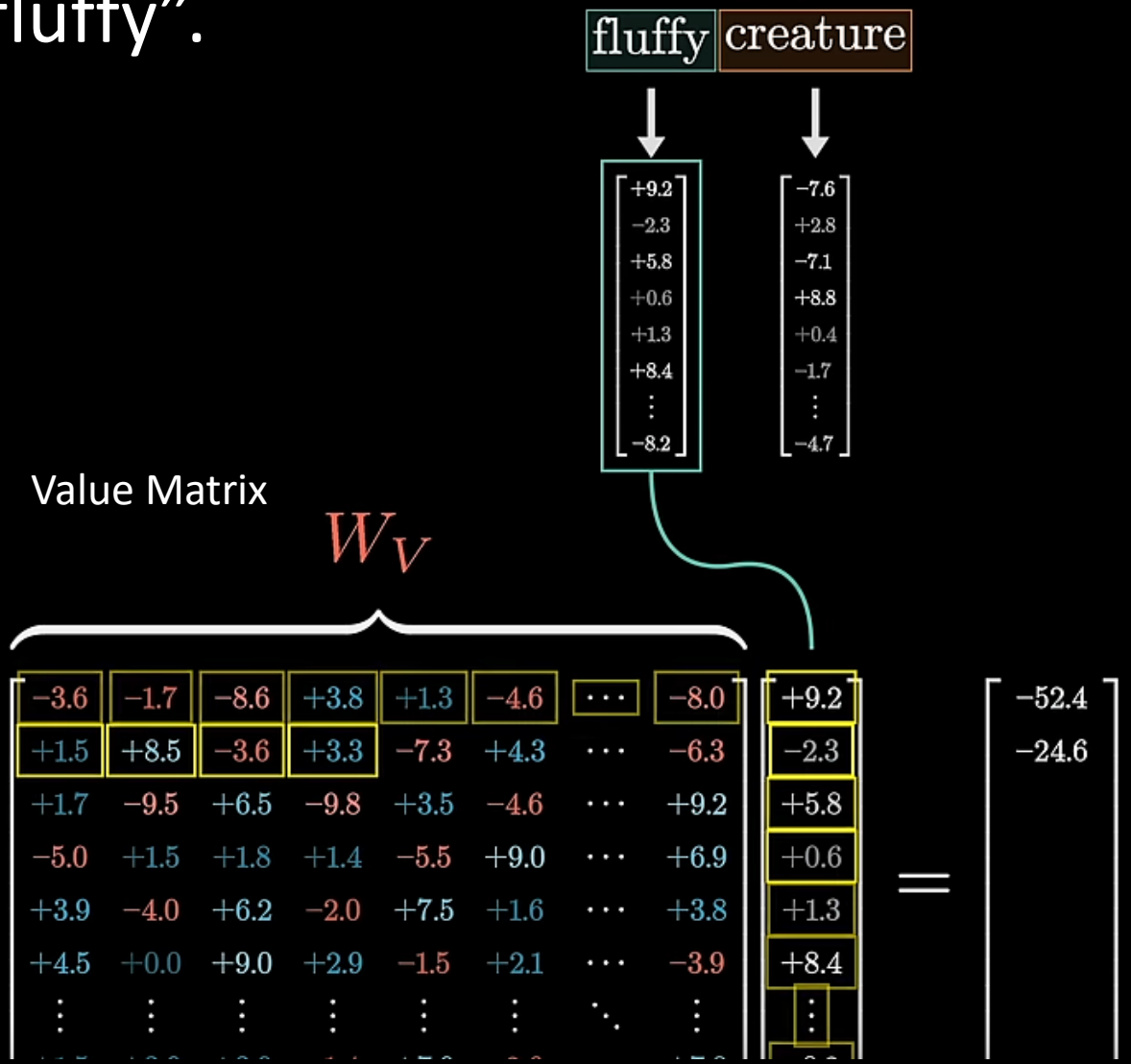


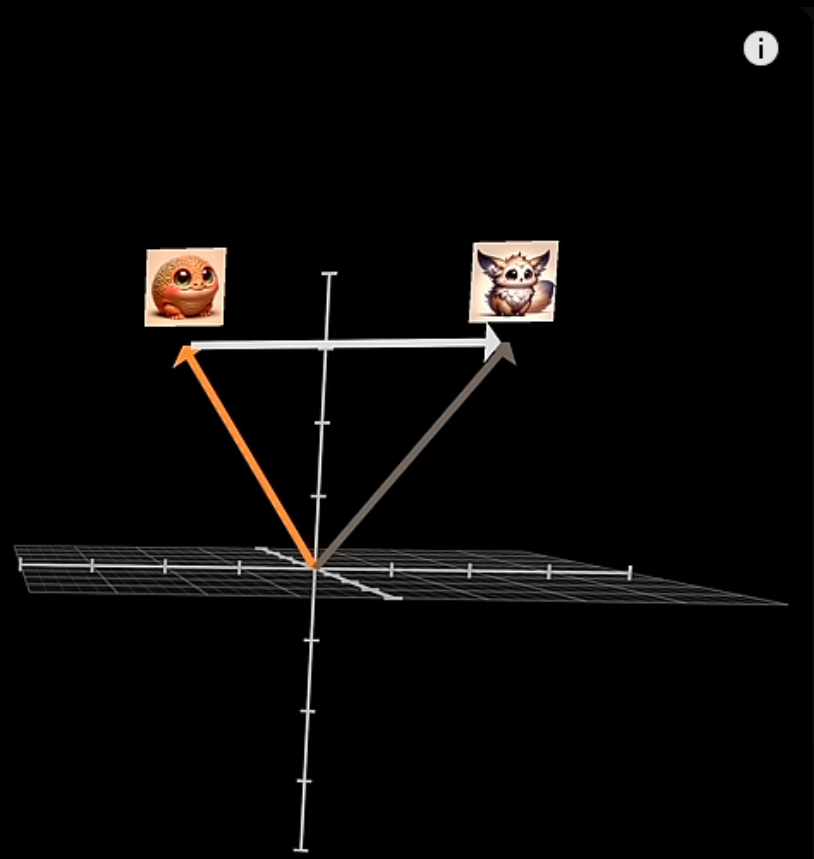
At this moment, I have a column of numbers telling me how important another word is to the word of interest (WOI).

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

- What I don’t know is which direction I should move the WOI.
- Enter the Value matrix.

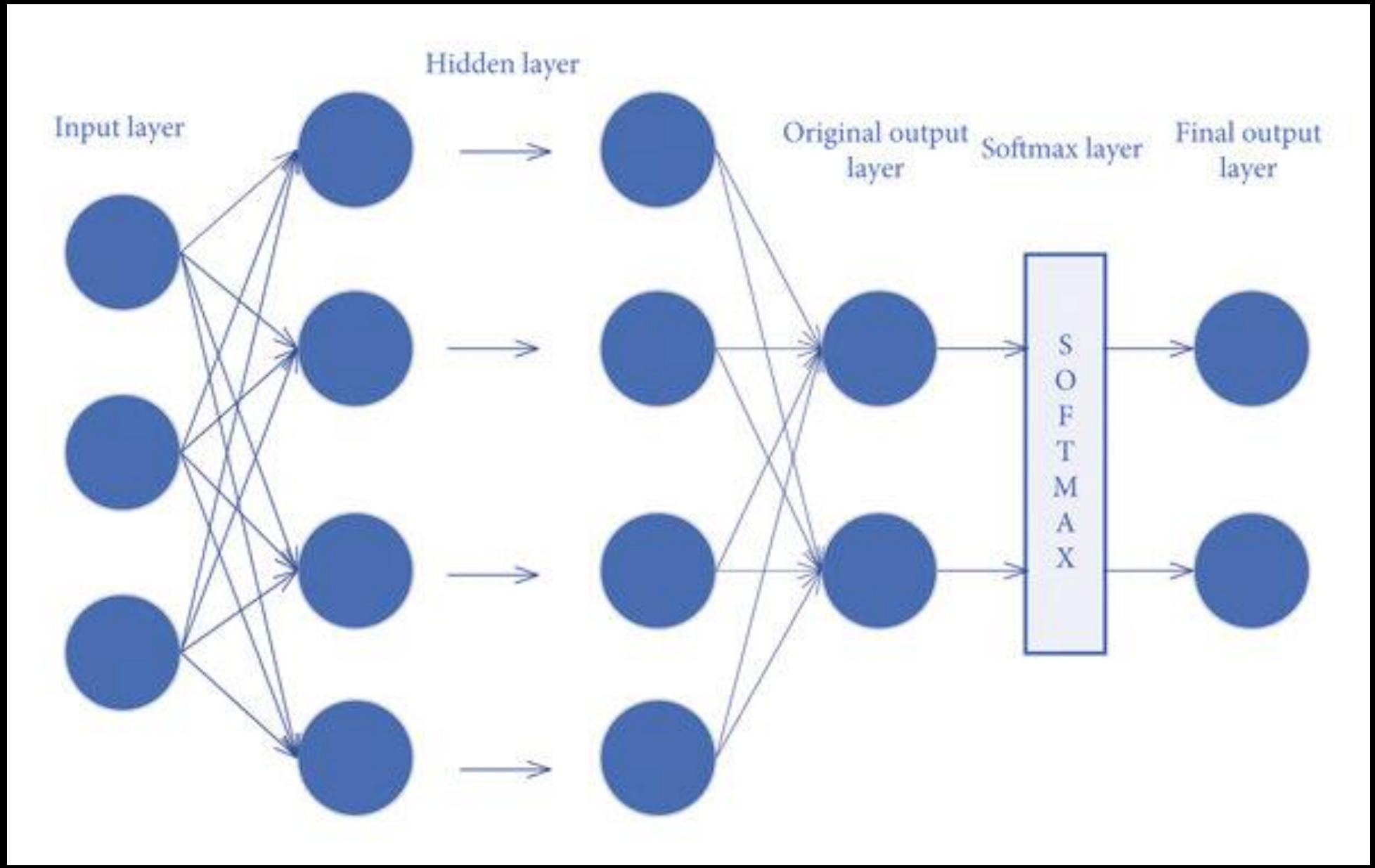
The Value Matrix for “creature” is being updated with the column from “fluffy”.

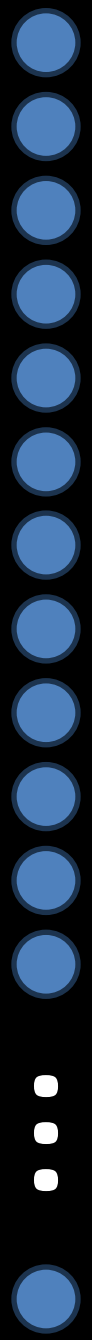




- Having “transformed” a word to its new position, we now ask “What word is most likely to follow you?”

- You give the 12,288-element vector (“transformed” word embedding) to yet another neural network that has 50,257 outputs.


Hidden layer
Input layer
Original output layer
Softmax layer
Final output layer
SOFTMAX



Each output is a number between 0..1 representing how proabable it is that the vocabulary word is the next word.

Temperature

- They don’t always take the most probable next word.

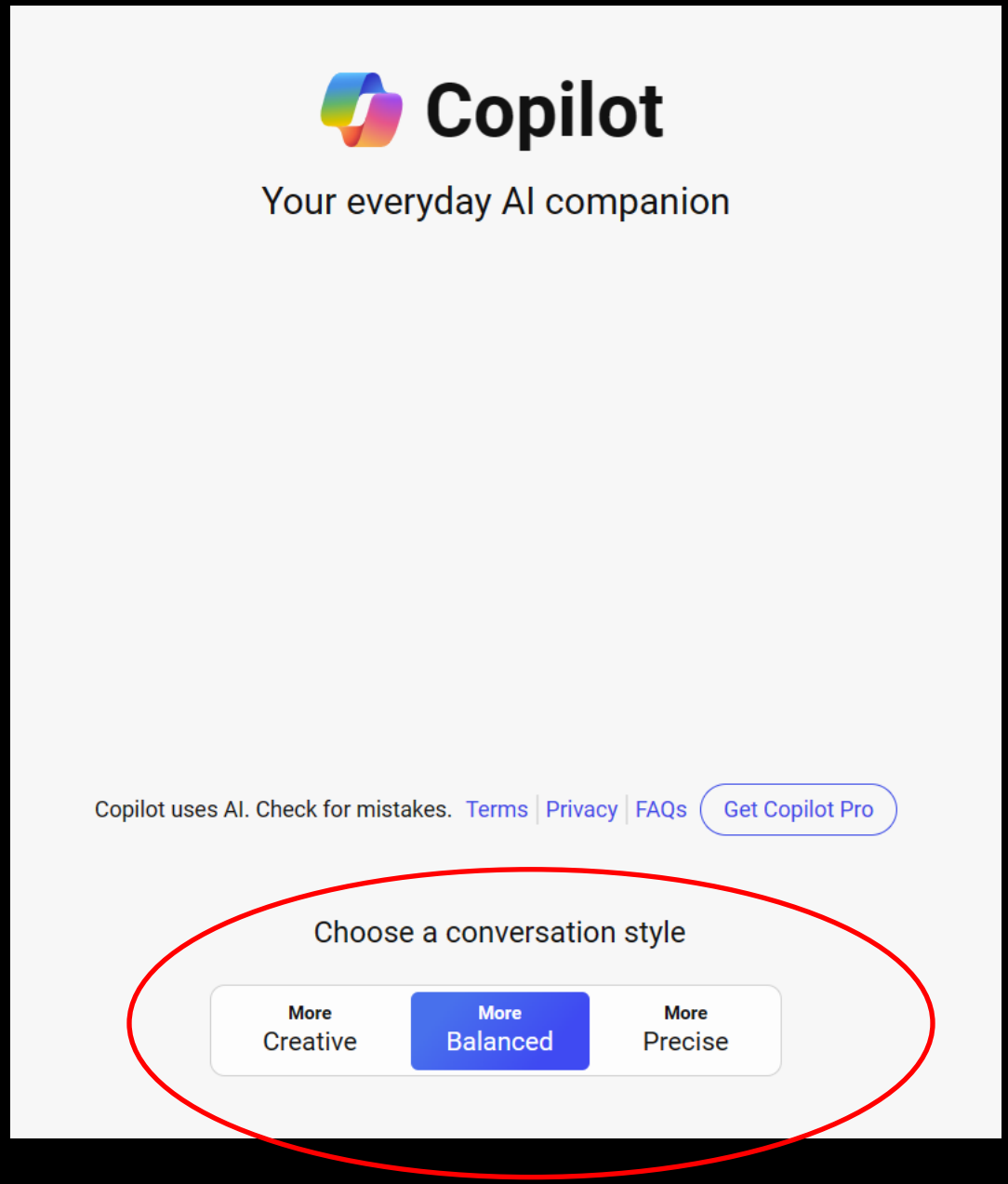


Figure 1: ML-benchmark saturation relative to human performance (black line) [Kiela et al., 2021].

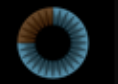
SQuAD2.0 combines the 100,000 questions in SQuAD1.1 with over 50,000 unanswerable questions written adversarially by crowdworkers to look similar to answerable ones. To do well on SQuAD2.0, systems must not only answer questions when possible, but also determine when no answer is supported by the paragraph and abstain from answering.

Figure 1: ML-benchmark saturation relative to human performance (black line) [Kiela et al., 2021].

The General Language Understanding Evaluation (GLUE) benchmark is a collection of resources for training, evaluating, and analyzing natural language understanding systems.

Switchboard: Recognition of conversational speech over telephone.

Credits

- A number of images are credit to 3Blue1Brown.com
  - Their logo on the slides helps identify images from them. 
  - Please check out their YouTube channel.