

Intro to Transformers

Catherine Ning

Deep Learning Part I, Module 4 of 15.S60 IAP Software Tools 2026

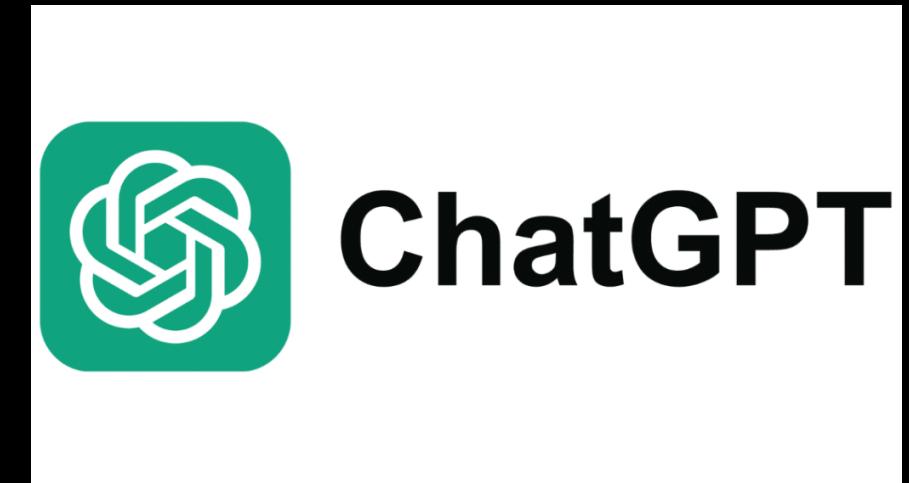
What is “GPT”?

GPT = **Generative Pre-Trained Transformer**

- **generative** means the model generates new text
- **pre-trained** means the model was trained on large amounts of data
- **Transformer** = special kind of neural network used for text-to-text, voice-to-text, text-to-voice, text-to-image, machine translation, etc.

3 common **Transformer** families:

- Encoder-only (BERT-like): representations for classification/retrieval
- Decoder-only (GPT-like): generate text / predict next word
- Encoder--decoder (T5-like): seq2seq tasks (translation, summarization)



A GPT is trained to predict what comes next: a probability distribution over tokens.



Generation loop (sampling)

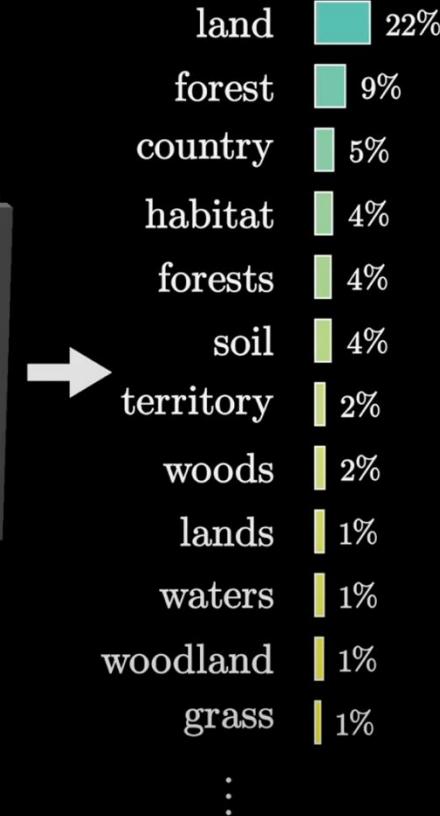
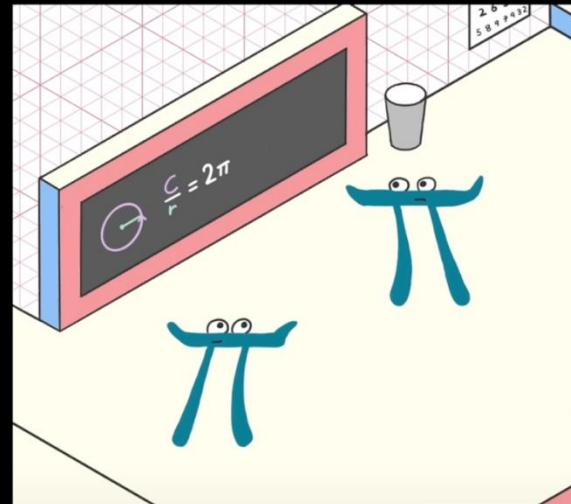
- 1) Feed prompt → distribution over next token
- 2) Sample (or pick max)
- 3) Append token and repeat

This simple loop turns “predict next token” into “generate long text.”

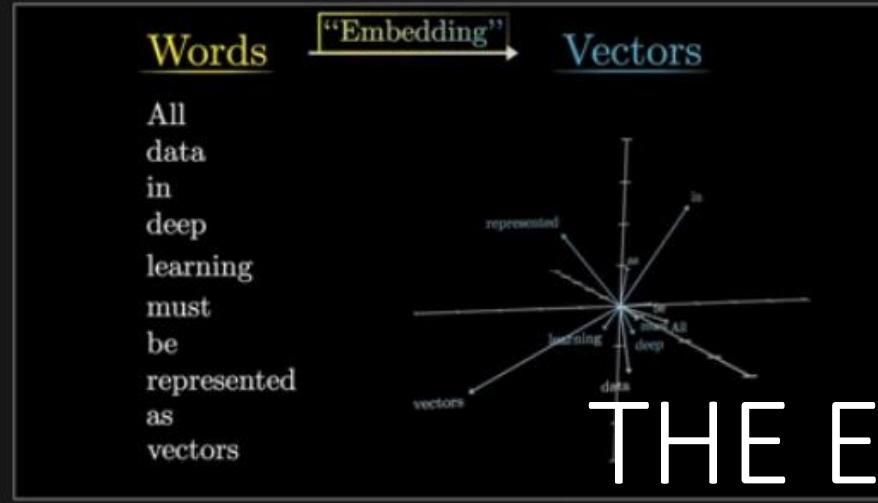
Key idea

The model is a learned function with many parameters (weights).
At inference time, the weights are fixed; only the data flowing through changes with the prompt.

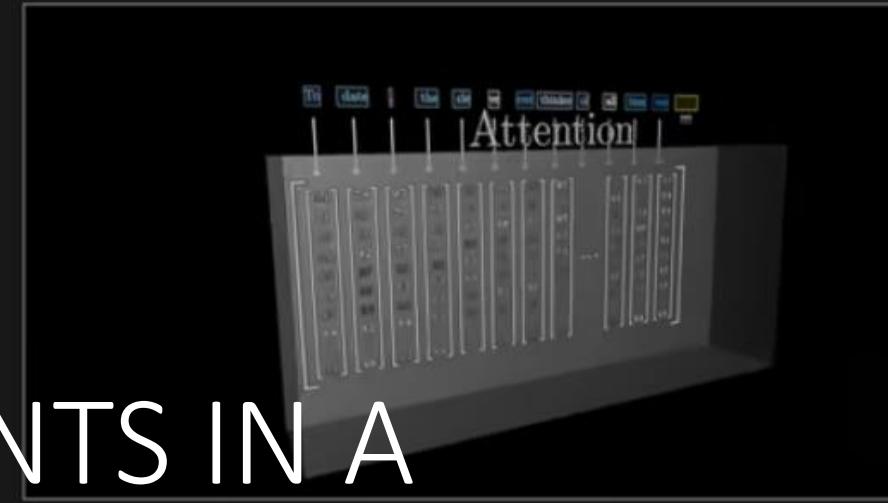
Behold, a wild pi creature,
foraging in its native _____



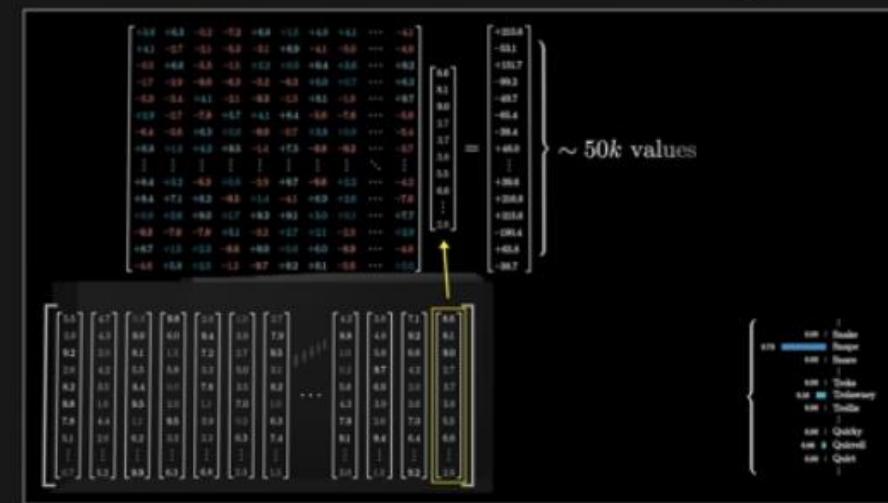
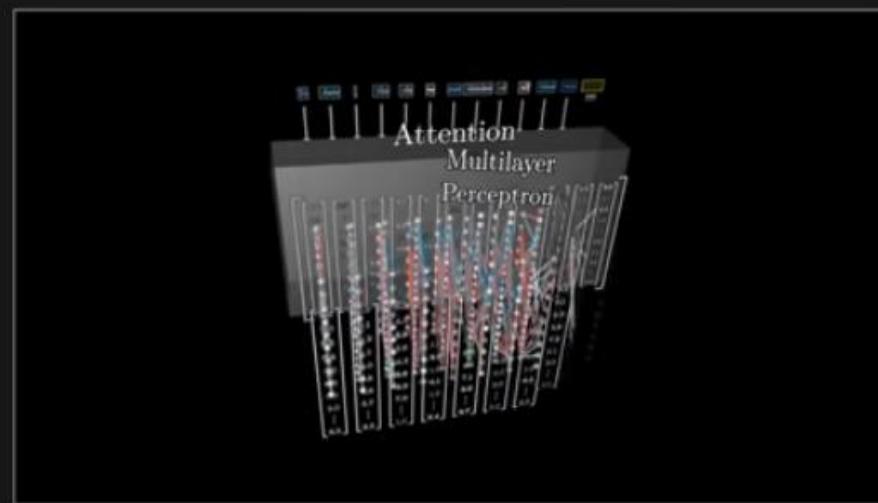
Embedding



Attention



THE ELEMENTS IN A TRANSFORMER



To date, the cleverest thinker of all time was ...

To | date | , | the | cle | ve | rest | thinker | of | all | time | was ...

- Tokens are chunks of text (often subwords + punctuation).
- Tokenization is a pre-processing step, before the transformer blocks.
- The model predicts the next token, not necessarily the next “word.”

Embedding matrix as a learned lookup table

Vocabulary size V ($\approx 50k$)

Embedding dim d ($\approx 12k$ in GPT-3)

Each token index selects a column vector. The entries start random and are tuned during training.

Parameter count (GPT-3 example)

$V = 50,257$ tokens

$d = 12,288$

$W_E = V \times d = 617,558,016$ weights

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

All words, $\sim 50k$																											
aah	aardvark	aardwolf	aargh	ab	aback	abacterial	abacus	abalone	abandon	...	zygoid	zygomatic	zygomorphic	zygosis	zygote	zygotic	zyme	zymogen	zymosis
-8.6	-1.5	-4.8	+6.9	-9.2	+9.1	-2.9	-2.8	-9.6	-6.2	...	-2.0	+8.5	-7.9	+8.8	+7.3	-0.9	-3.4	-5.3	+2.3	-9.2
-9.6	-1.4	-8.5	-4.9	-5.5	-4.9	-7.3	-9.7	-7.6	+2.3	...	+9.4	+9.7	-1.8	-6.7	+2.7	-0.2	+9.7	-8.6	+5.6	-4.2
-5.1	+3.2	-5.0	+3.3	+0.3	-1.5	+1.1	-4.2	+4.1	-1.7	...	-2.8	+6.5	+8.4	-9.0	-5.3	-3.0	+6.2	+9.6	+9.3	+8.0
-4.0	+9.7	-5.0	-7.8	+8.9	-5.3	+3.8	-8.7	+4.6	+7.6	...	-4.5	-2.4	-2.5	+4.9	-5.2	-6.5	-1.0	-3.9	+6.7	-5.2
+0.0	+8.8	+2.7	+7.3	+8.7	+5.0	+3.9	+9.3	+9.8	-0.9	...	-8.5	-4.1	-6.9	-1.6	-7.3	+2.1	-2.3	+7.8	+9.3	+0.9
-4.5	+1.8	+7.8	-1.9	+1.0	-4.5	-0.9	-1.9	-5.0	+0.1	...	-3.8	-2.5	+0.5	+5.0	-3.3	+8.4	+7.2	-8.9	-4.9	-1.1
-7.8	-3.0	+4.7	+3.6	+2.4	+4.2	-5.8	-3.1	+3.5	+7.5	...	+0.9	-4.3	-9.3	+4.2	-9.7	-2.5	+0.6	+8.3	-8.1	-1.9
-9.4	-3.1	+2.4	-4.4	-5.7	-7.6	+1.5	+3.9	+3.4	+8.9	...	-9.8	+2.9	+2.0	+1.8	+9.2	-9.6	+3.9	+6.2	+0.2	-3.3
...
+5.8	-8.0	-1.1	+0.4	+3.8	-8.1	-5.4	-1.8	+2.4	+7.7	...	+2.4	-7.3	+9.5	+7.4	+0.1	+8.4	+0.8	+8.4	+6.5	+9.3

Embedding matrix

Embedding Matrix

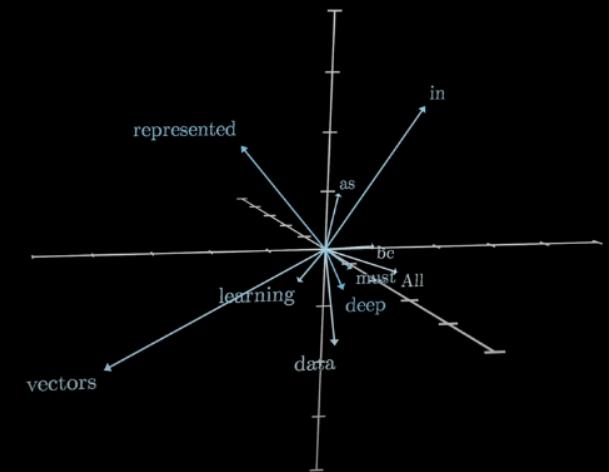
This is how search works you know!



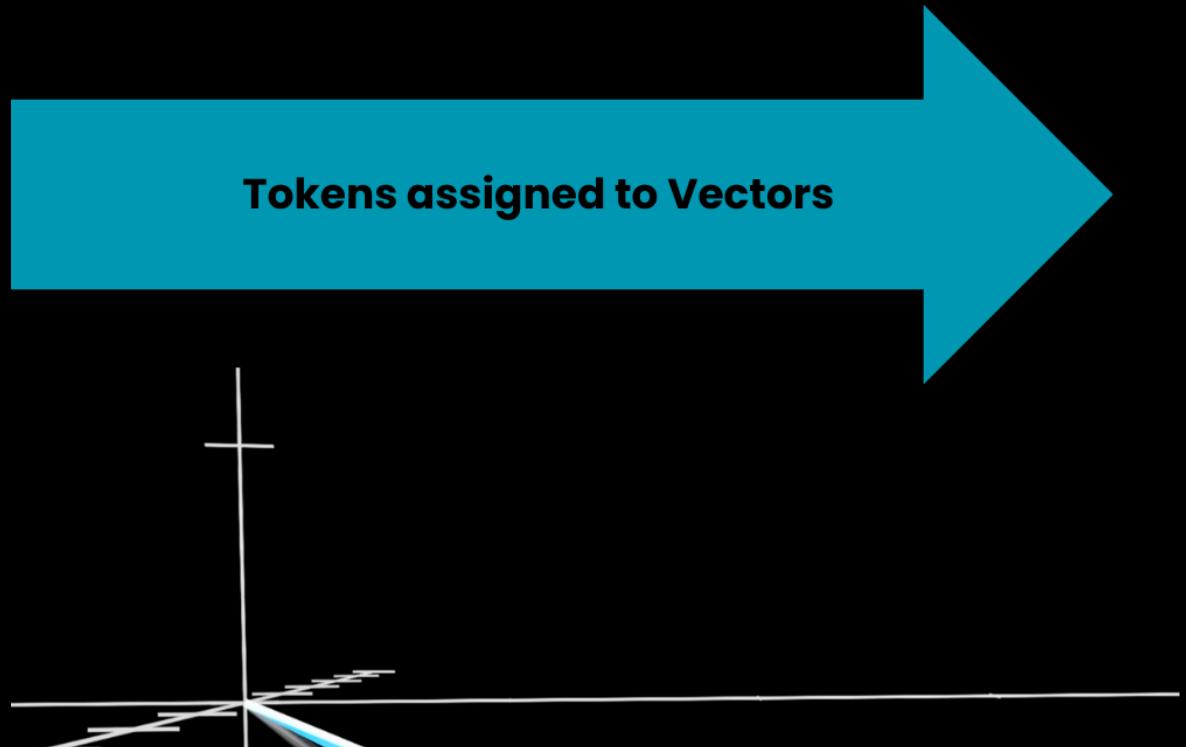
Embedding Process

Words $\xrightarrow{\text{“Embedding”}}$ Vectors

All data in deep learning must be represented as vectors



Tokenization & Embedding



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

Words with similar meanings have close vectors

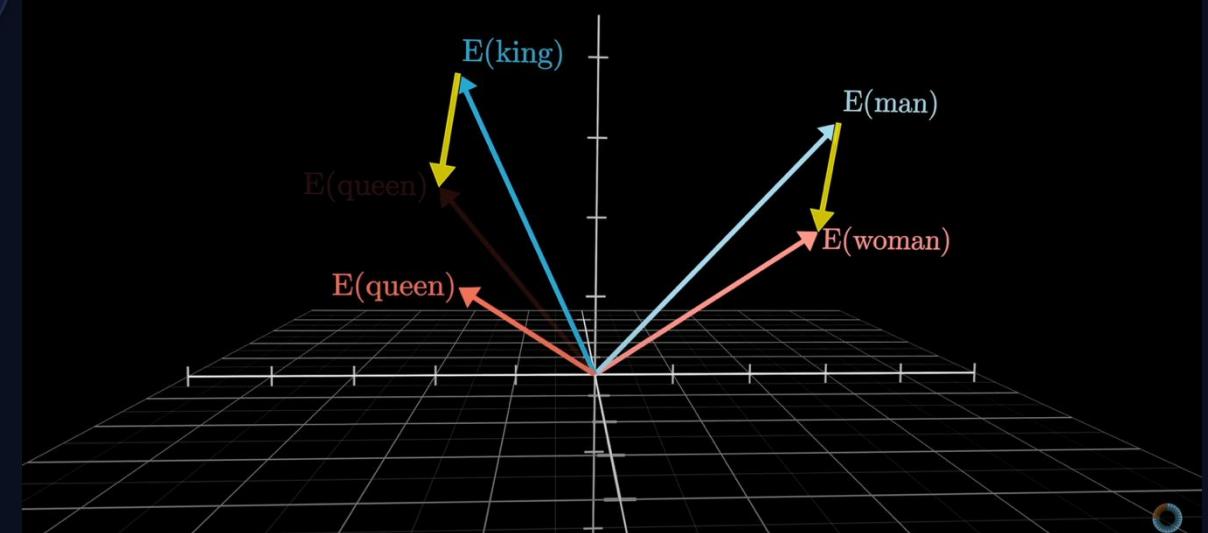
Geometric view

Think of each token vector as a point/direction in a high-dimensional space.

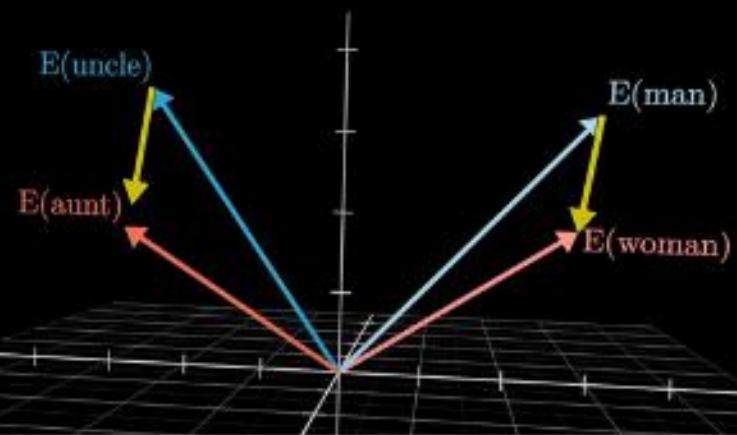
During training, models often arrange vectors so that ***directions*** correlate with attributes (gender, plurality, geography, etc.).

- Classic analogy structure (e.g., “king – man + woman \approx queen”) is one way to ***visualize*** meaningful directions.
- (In reality, not every analogy works perfectly)

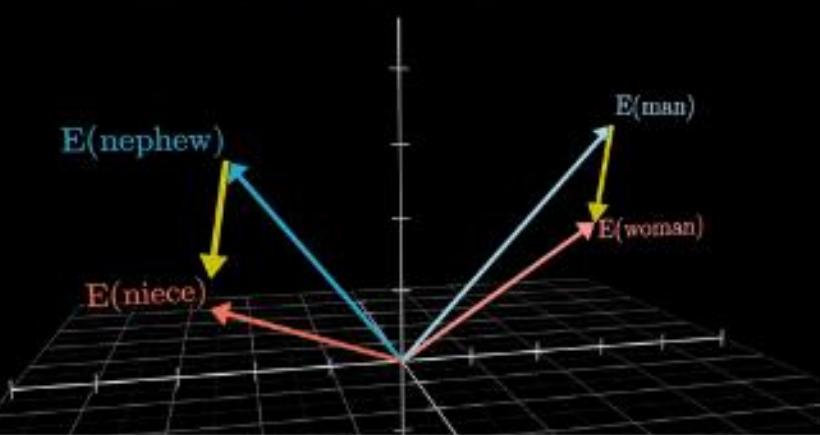
$$E(\text{queen}) \approx E(\text{king}) + E(\text{woman}) - E(\text{man})$$



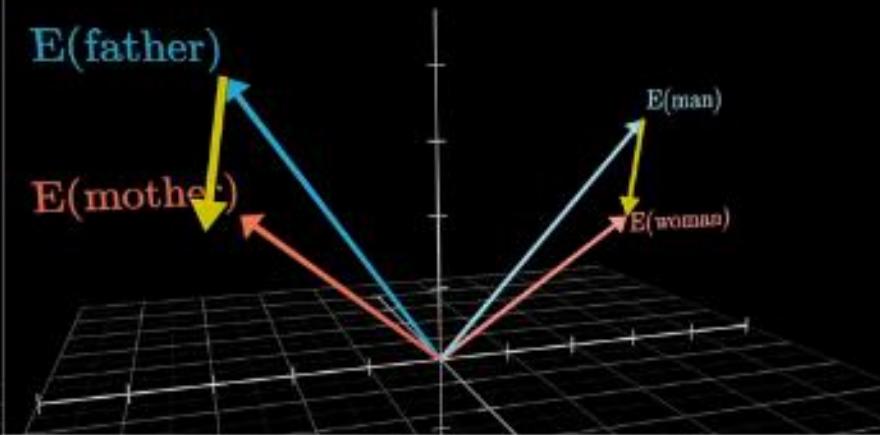
$$E(\text{aunt}) - E(\text{uncle}) \approx E(\text{woman}) - E(\text{man})$$



$$E(\text{niece}) - E(\text{nephew}) \approx E(\text{woman}) - E(\text{man})$$



$$E(\text{mother}) - E(\text{father}) \approx E(\text{woman}) - E(\text{man})$$



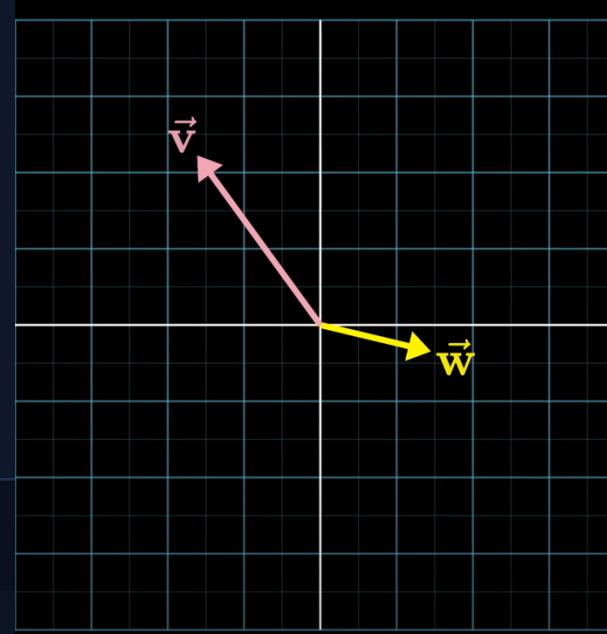
Dot product intuition

Geometrically: $\mathbf{v} \cdot \mathbf{w}$ is large when \mathbf{v} and \mathbf{w} point in similar directions, near 0 when orthogonal, negative when opposed.

Computationally: multiply component-wise and sum.

$$\mathbf{v} \cdot \mathbf{w} = \sum_i v_i w_i$$

→ used as a similarity score inside attention



$$\underbrace{\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_n \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_n \end{bmatrix}}_{\text{Dot product}} = v_1 w_1 + v_2 w_2 + v_3 w_3 + \dots + v_n w_n$$

||
-3.11

Each layer alternates two operations:

Attention: tokens “talk”

Vectors update based on *context*

Feed-forward (MLP): tokens “think”

Same learned function applied to every position independently.

At the start, each token is just a lookup from the embedding matrix.

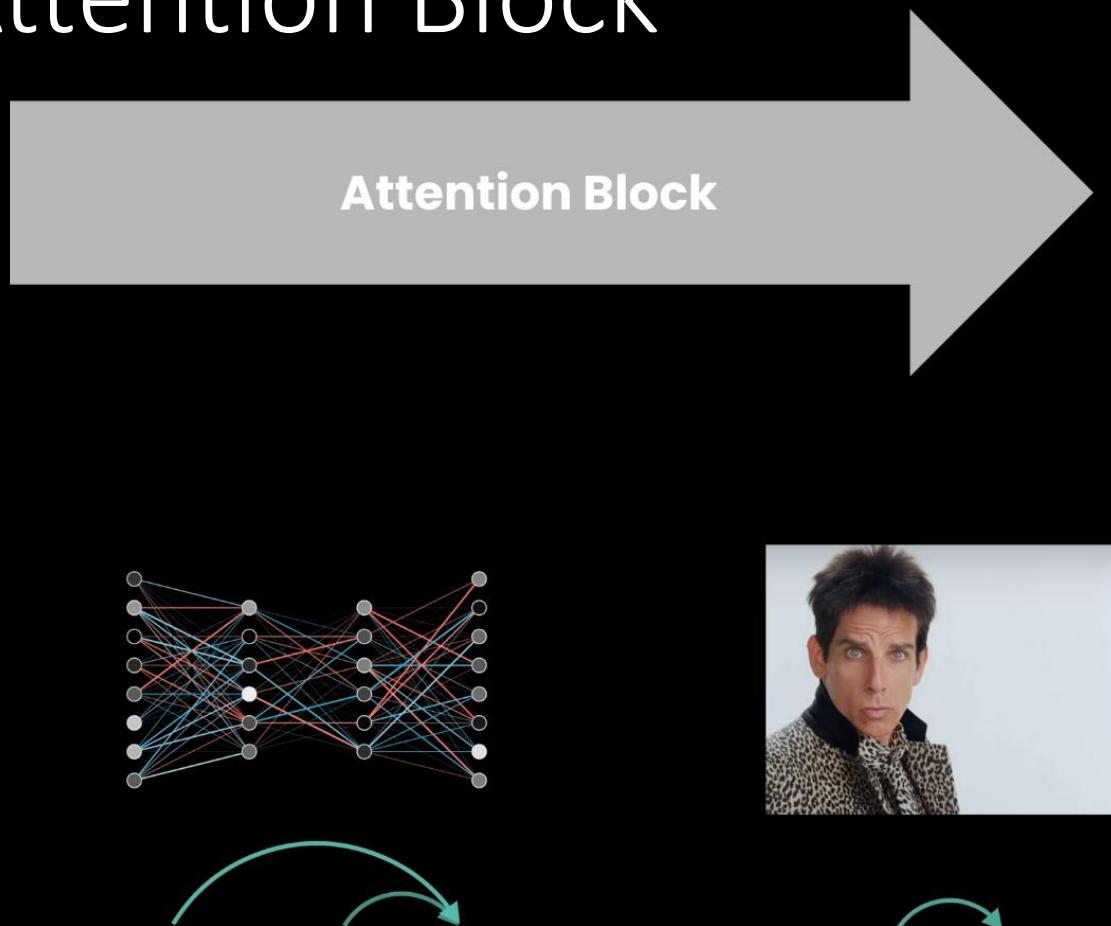
As many layers apply attention + MLP, the vector can be “tugged” into a nuanced direction representing meaning-in-context.

Deep = many layers



Step 3: transformer blocks

Attention Block



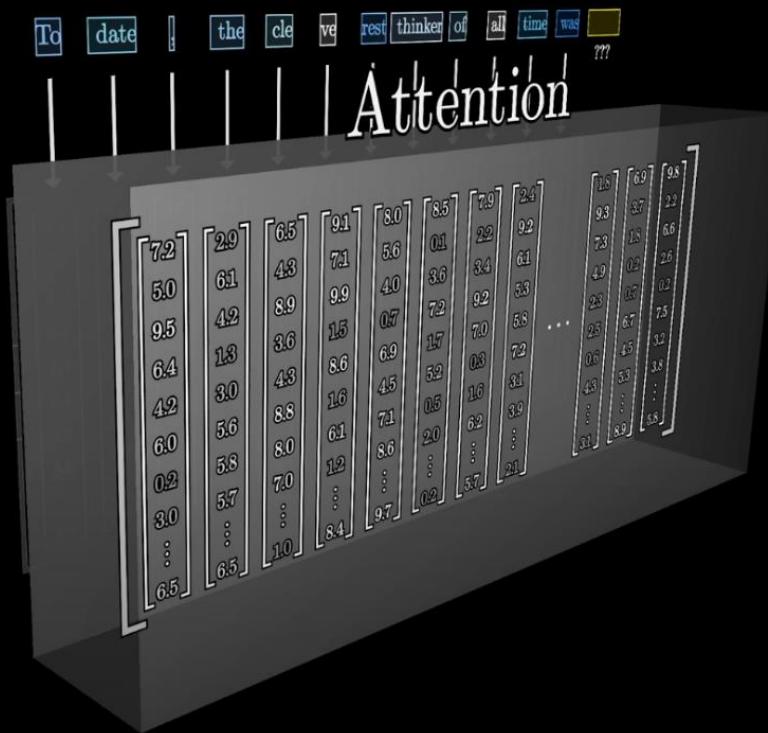
A machine learning model ...

$$\begin{bmatrix} 3.7 \\ 5.7 \\ 4.4 \\ 9.5 \\ 1.2 \\ \vdots \\ 2.2 \end{bmatrix} \quad \begin{bmatrix} 2.3 \\ 6.8 \\ 2.6 \\ 4.9 \\ 2.7 \\ \vdots \\ 2.3 \end{bmatrix} \quad \begin{bmatrix} 1.0 \\ 5.6 \\ 1.2 \\ 3.0 \\ 4.3 \\ \vdots \\ 8.4 \end{bmatrix} \quad \begin{bmatrix} 2.6 \\ 7.4 \\ 2.5 \\ 9.1 \\ 5.0 \\ \vdots \\ 9.6 \end{bmatrix}$$



A fashion model ...

$$\begin{bmatrix} 5.7 \\ 6.3 \\ 1.1 \\ 4.3 \\ 2.2 \\ \vdots \\ 2.1 \end{bmatrix} \quad \begin{bmatrix} 3.2 \\ 4.3 \\ 4.9 \\ 4.2 \\ 6.8 \\ \vdots \\ 5.1 \end{bmatrix} \quad \begin{bmatrix} 4.8 \\ 2.9 \\ 3.7 \\ 6.2 \\ 5.7 \\ \vdots \\ 3.1 \end{bmatrix}$$

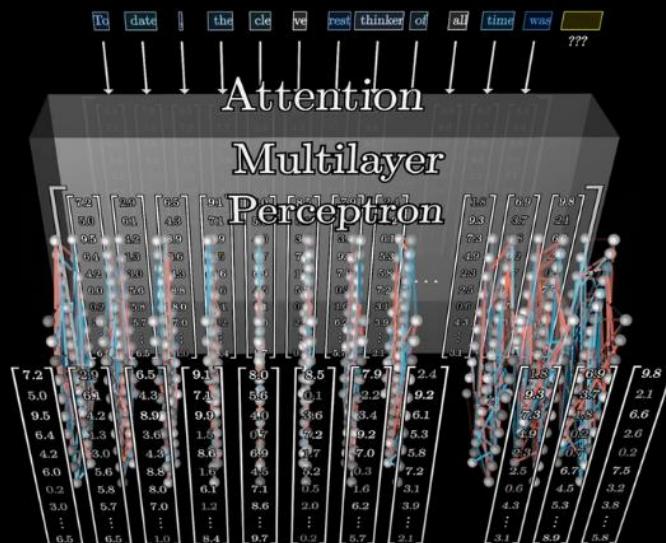


Example of Attention Block changing the vector of the word "model"

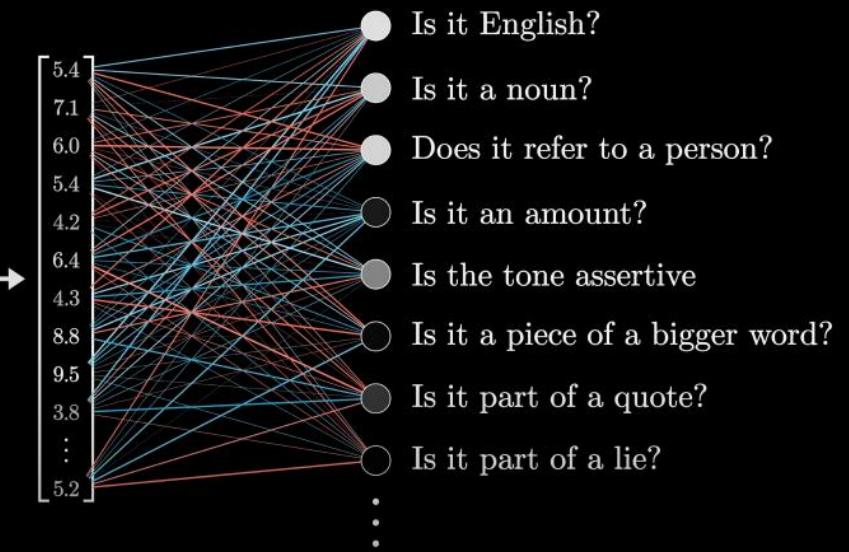
Step 3: transformer blocks

MLP Layer / Feedforward Layer

The Multilayer Perceptron asking “Questions” to Update the Vector



Queen →



Unembedding

At the end, map a context-rich vector back to vocabulary-sized scores.

These raw scores are called **logits**.

Why “the last vector”?

Causal language models are trained so every position predicts the token after it.

At generation time, we use the final position’s prediction as “next token.”

Unembedding matrix

W_U

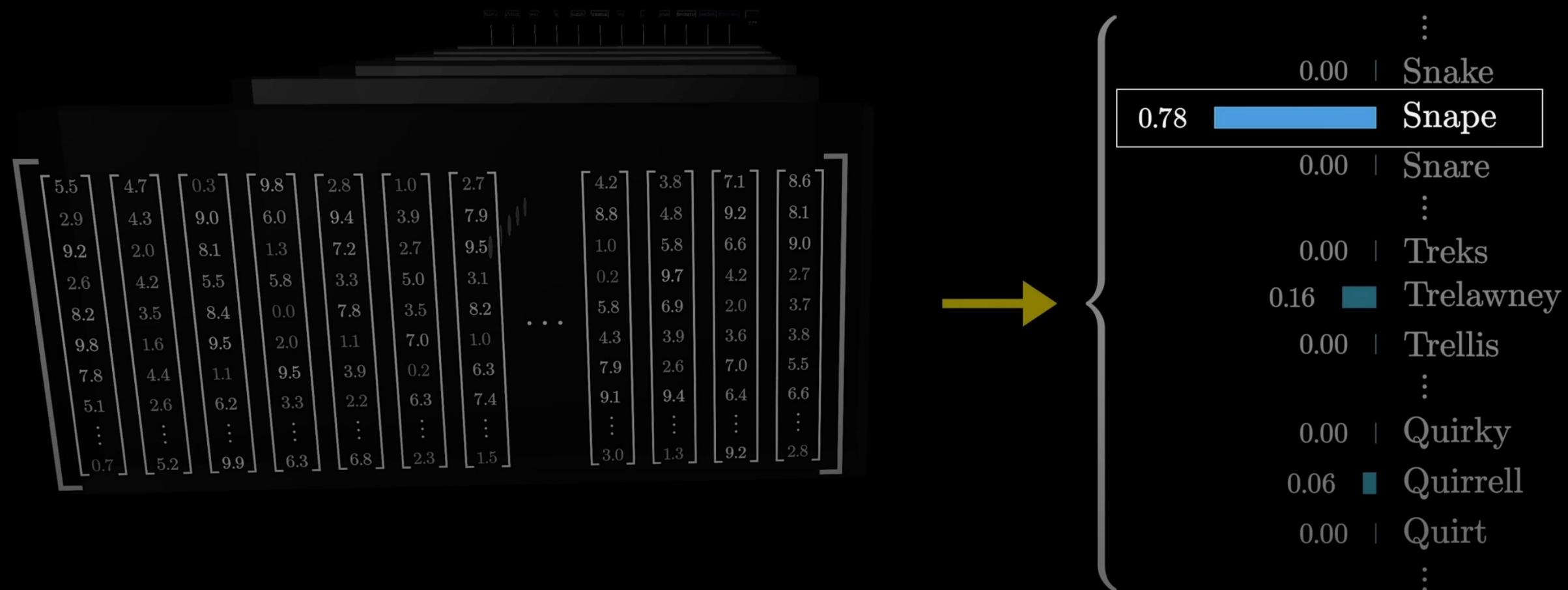
+3.8	+6.3	-0.2	-7.2	+6.9	+1.5	+4.8	+4.1	...	-4.1
+4.1	-2.7	-2.1	-5.3	-3.1	+8.9	-4.1	-5.0	...	-4.8
-0.5	+6.6	-5.3	-1.5	+2.2	+0.9	+9.4	+3.6	...	+9.2
-1.7	-2.9	-9.0	-6.3	-5.2	-6.3	+5.0	+0.7	...	+6.3
-5.3	-3.4	+4.1	-2.1	-9.3	-1.3	+8.1	-1.8	...	+9.7
+2.9	-2.7	-7.9	+5.7	+4.1	+8.4	-5.6	-7.6	...	-5.9
-6.4	-3.6	+6.3	+0.8	-9.0	-0.7	+3.6	+0.8	...	-5.4
+6.9	+1.2	+4.2	+9.5	-1.4	+7.5	-9.8	-9.2	...	-3.7
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
+8.4	+3.2	-8.3	+0.8	-2.9	+9.7	-9.6	+2.2	...	-4.2
+9.4	+7.1	+8.2	-9.5	+1.4	-4.1	+6.9	+2.6	...	-7.6
+0.8	+2.6	+9.0	+1.7	+9.3	+9.1	+3.0	+0.1	...	+7.7
-9.3	-7.6	-7.9	+5.1	-3.2	+2.7	+2.1	-2.3	...	+2.9
+8.7	+1.5	+2.3	-8.6	+9.0	+0.6	+6.0	-8.9	...	-4.8
-4.6	+5.8	+2.5	-1.2	-9.7	+9.2	+9.1	-5.6	...	+0.6

+215.6	aah
-53.1	aardvark
+151.7	aardwolf
-99.2	aargh
-49.7	ab
-65.4	aback
-38.4	abacterial
+46.0	abacus
⋮	⋮
+39.6	zygote
+216.8	zygotic
+215.6	zyme
-190.4	zymogen
+65.8	zymosis
-38.7	ZZZ

5.5	4.7	0.3	9.8	2.8	1.0	2.7	4.2	3.8	7.1	8.6
2.9	4.3	9.0	6.0	9.4	3.9	7.9	8.8	4.8	9.2	8.1
9.2	2.0	8.1	1.3	7.2	2.7	9.5	1.0	5.8	6.6	9.0
2.6	4.2	5.5	5.8	3.3	5.0	3.1	0.2	9.7	4.2	2.7
8.2	3.5	8.4	0.0	7.8	3.5	8.2	5.8	6.9	2.0	3.7
9.8	1.6	9.5	2.0	1.1	7.0	1.0	4.3	3.9	3.6	3.8
7.8	4.4	1.1	9.5	3.9	0.2	6.3	7.9	2.6	7.0	5.5
5.1	2.6	6.2	3.3	2.2	6.3	7.4	9.1	9.4	6.4	6.6
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
0.7	5.2	9.9	6.3	6.8	2.3	1.5	3.0	1.3	9.2	2.8

Step 5: softmax

Harry Potter was a highly unusual boy ... least favourite teacher, Professor [REDACTED]

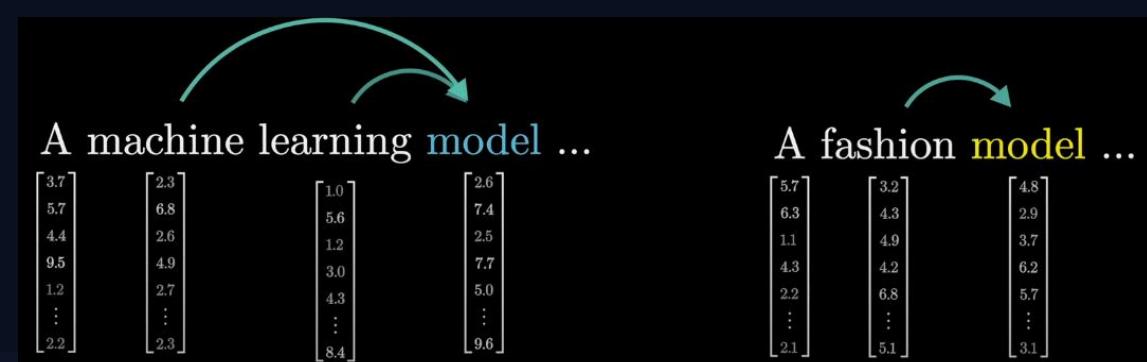


So far...

- Transformers predict next tokens; generation is repeated prediction + sampling.
- Tokens → embedding vectors → many layers of attention + MLP → logits → softmax.
- Embeddings live in a high-dimensional space where directions can carry meaning.
- Attention is the mechanism that updates vectors using context (next).

Desiderata

- route information between token embeddings over short and long distances
- handle information that's much richer than just a single word
- encode *all of the information* from the full context window that's relevant to predicting the next word.



Short distance

One mole of carbon dioxide

Long distance

Harry Potter was a highly unusual boy in many ways. For one thing, he hated the summer holidays more than any other time of year. For another, he really wanted to do his homework but was forced to do it in secret, in the dead of night. And he also happened to be a wizard.

It was nearly midnight, and he was lying on his stomach in bed, the blankets drawn right over his head like a tent, a flashlight in one hand and a large leather-bound book (*A History of Magic* by Bathilda Bagshot) propped open against the wall behind him. He lay there, looking down the page, frowning as he looked for something that would help him with his essay, "Witch Burning in the Fourteenth Century Was Completely Pointless" to discuss.

The quill paused at the top of a long, speaking paragraph. Harry Paused his round glasses up the bridge of his nose, moved his flashlight closer to the book, and read:

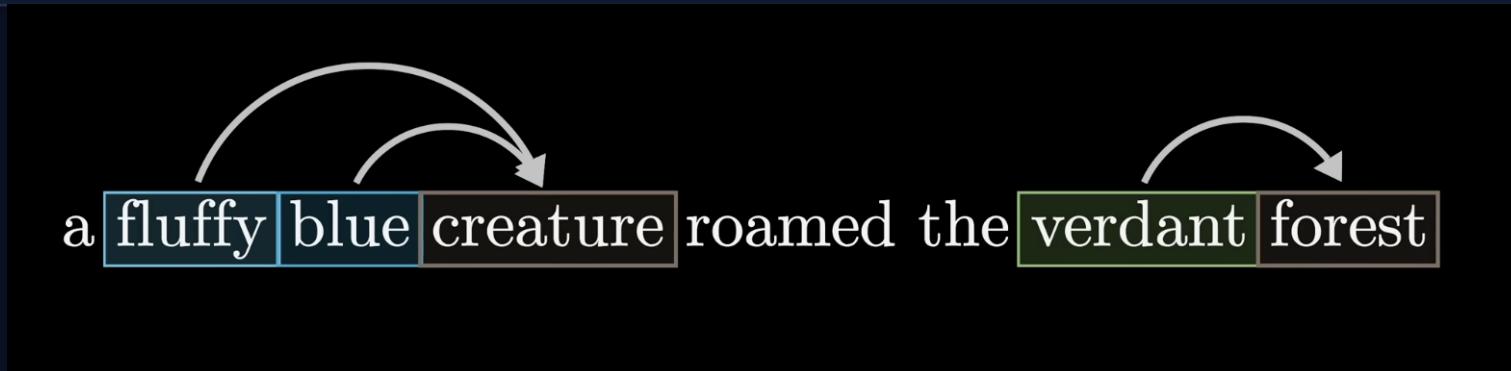
Non-magic people (more commonly known as Muggles) were particularly afraid of magic in summer time, but they were good at recognizing it. On the days when they tried to catch it, it would just slip away and had no effect whatsoever. The wizard would perform a basic Flame Freezing Charm and then pretend to speak with poise while enjoying a gentle, tickling sensation. Indeed, Wendelin the Weird enjoyed being burned so much that she allowed herself to be caught no less than thirty-seven times in various disguises.

Harry put his quill between his teeth and reached underneath his pillow for his ink bottle and a roll of parchment. Slowly and very carefully he unscrewed the ink bottle, dipped his quill into it, and began to write, pausing every now and then to listen, because if any of the Dursleys heard the scratching of his quill on their way to the bathroom, he'd probably find himself locked in the cupboard under the stairs for the rest of the summer.

The Dursley family of number four, Privet Drive, was the reason that Harry never enjoyed his summer holidays. Uncle Vernon, Aunt Petunia, and their son, Dudley, were the Dursleys. They were a snobbish, ignorant family and they had a very negative attitude toward magic. Harry's dead parents, who had been a witch and wizard themselves, were never mentioned under the Dursleys' roof. For years, Aunt Petunia and Uncle Vernon had hoped that if they kept Harry as low-profile as possible, they would be able to squash the magic of him. So their fury that had been born from the fact that they lived in terms of cash, finding that they had spent more of the last few years at Hogwarts School of Witchcraft and Wizardry. The most they could do, however, was to lock away Harry's spellbooks, wand, cauldron, and broomstick at the start of the summer break, and forbid him to talk to his neighbors.

This separation from his spellbooks had been a real problem for Harry, because his teachers at Hogwarts had given him a lot of holiday work. One of the essays, a particularly nasty one about shrinking potions, was for Harry's least favorite teacher, Professor

Let's start with an example sequence and toy example task:

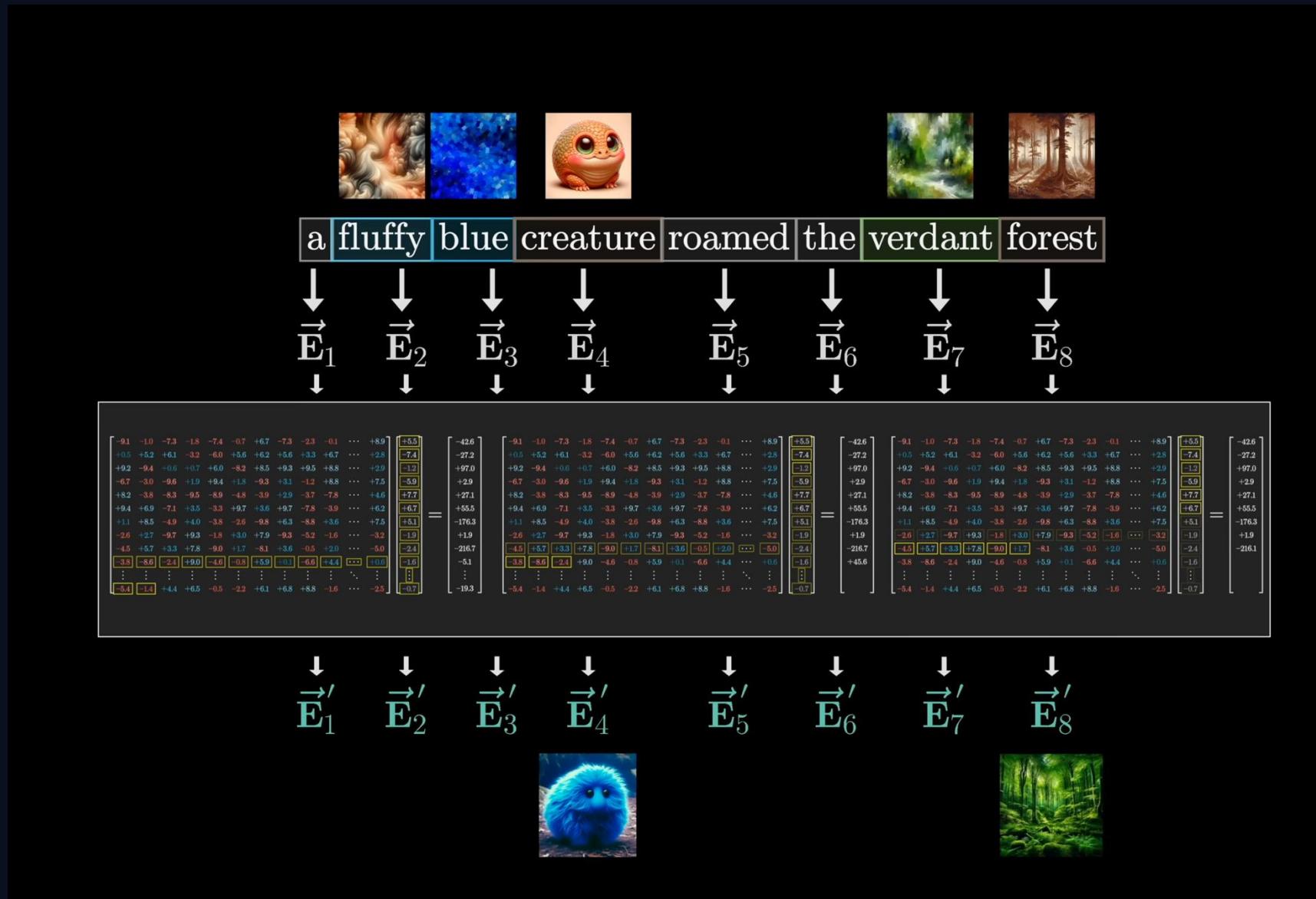


Toy Example Task: update the embeddings of this sequence so that the **adjectives** in the sequence above adjust the original embeddings of their corresponding nouns.

First, we tokenize then create the embeddings (which in practice also includes positional encoding).

Goal of this “single-head attention” block:

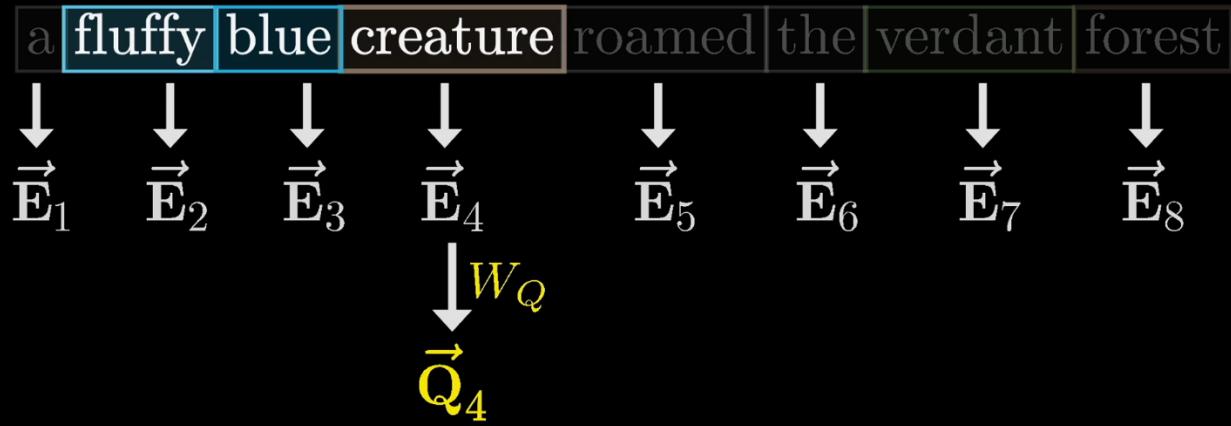
Output E' where the nouns have incorporated the meaning from their relevant adjectives



Query

Query matrix:

maps the embeddings of nouns to a smaller query space that (somehow) encodes the notion of looking for adjectives in preceding positions.

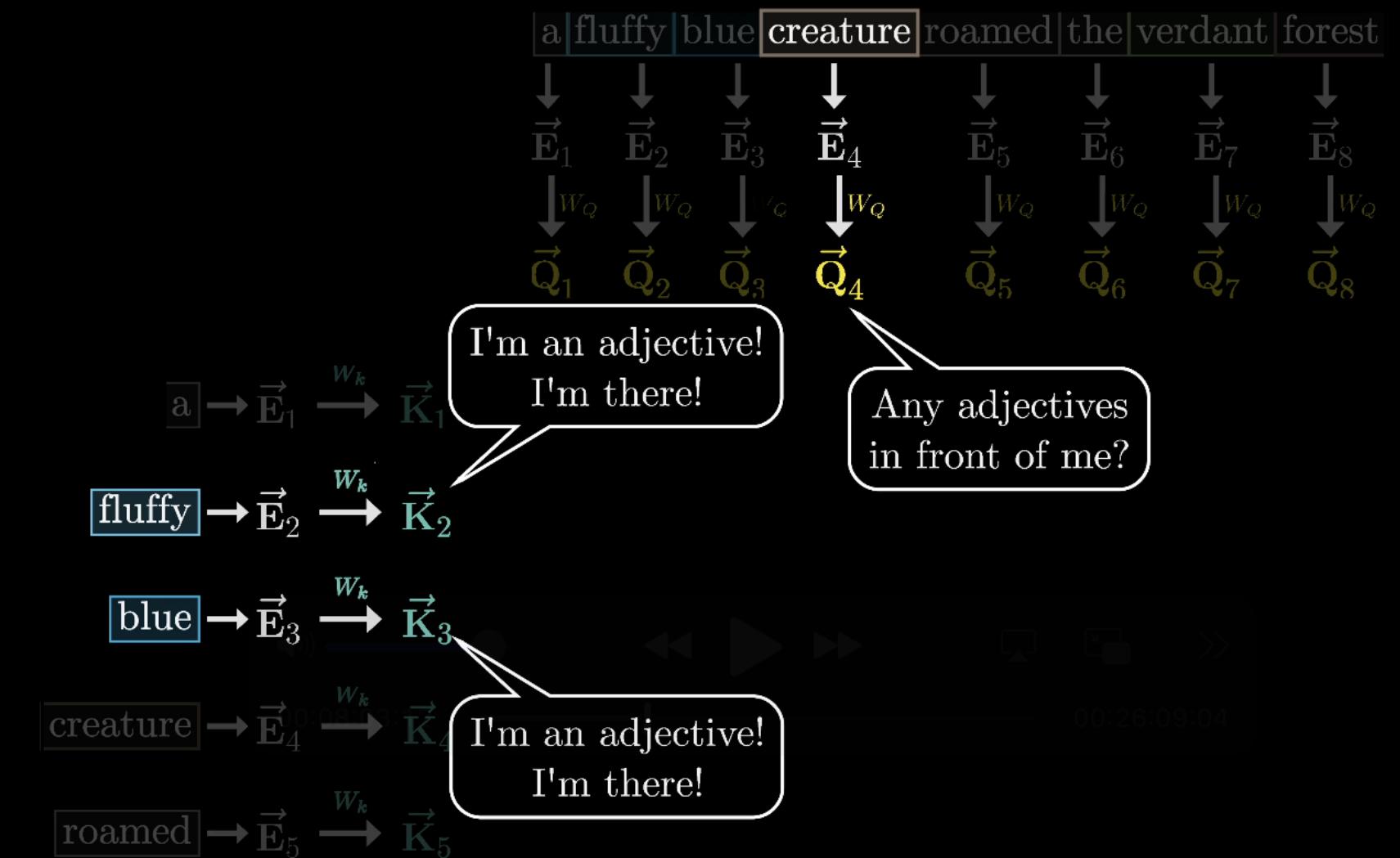


Any adjectives
in front of me?

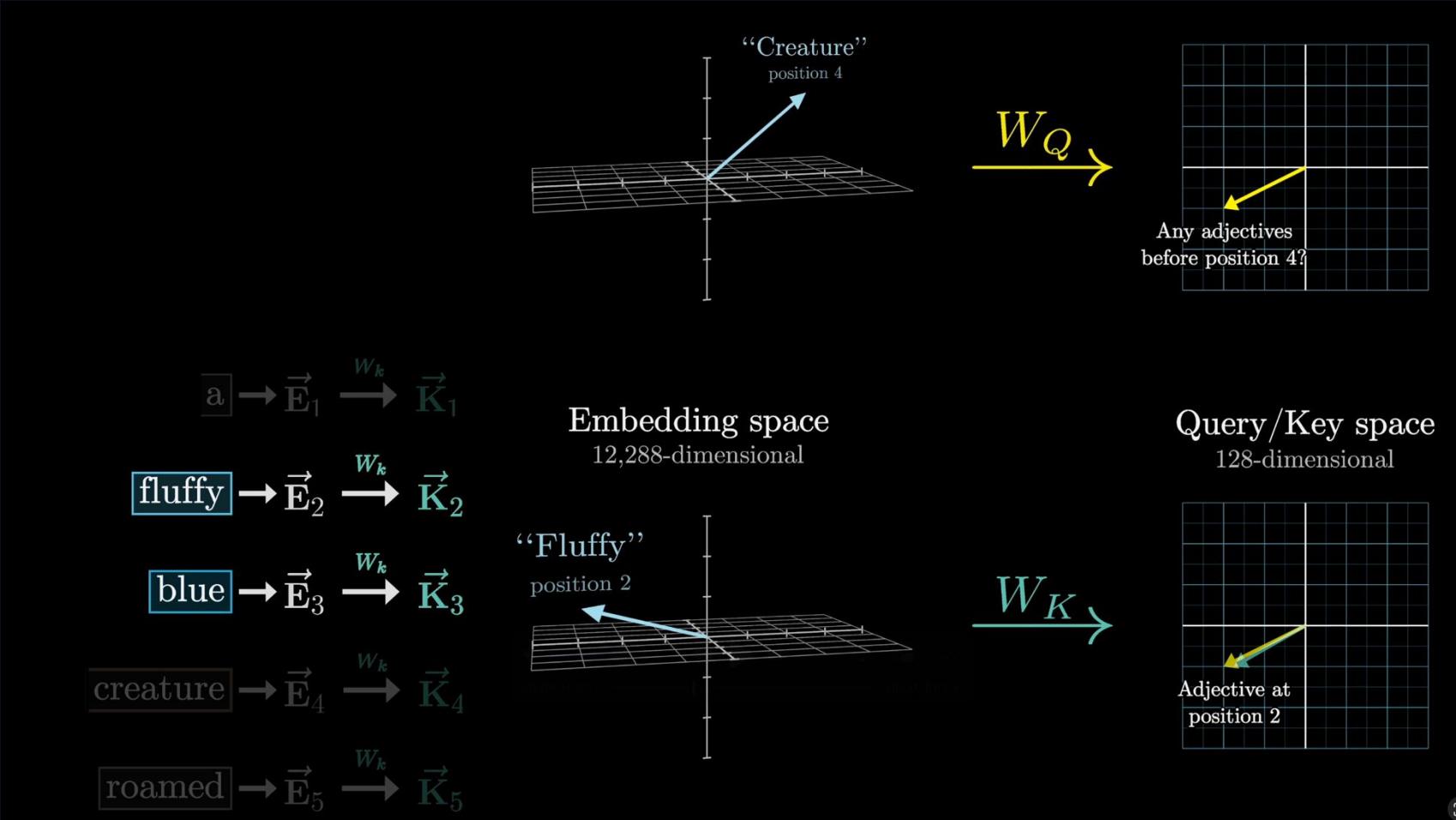
$$\underbrace{\begin{bmatrix} +7.5 & -3.2 & +9.1 & -5.3 & +8.9 & +8.7 & +5.9 & +2.6 & +7.4 & -4.1 & \dots & +2.3 \\ -9.6 & -3.0 & -7.0 & +9.5 & -0.4 & -0.1 & +2.8 & -2.6 & -7.2 & +6.4 & \dots & +0.2 \\ -5.5 & -8.0 & +7.2 & +9.4 & +9.1 & +8.0 & +5.4 & -3.3 & -8.3 & -1.8 & \dots & -7.3 \\ -8.8 & +4.5 & -9.7 & +5.4 & -7.0 & -8.3 & -8.1 & +3.4 & -5.0 & -1.6 & \dots & +7.1 \\ +4.5 & -4.5 & -7.3 & -8.8 & -3.9 & -4.7 & -0.9 & +3.6 & +3.9 & -4.3 & \dots & -6.3 \\ \vdots & \ddots & \vdots \\ -9.0 & +5.9 & -8.4 & +0.4 & -3.8 & +1.5 & +9.1 & +2.9 & -9.2 & -1.4 & \dots & +0.7 \end{bmatrix}}_{W_Q} \begin{matrix} \vec{E}_4 \\ \begin{bmatrix} 2.9 \\ 2.4 \\ 1.0 \\ 0.2 \\ 9.2 \\ 6.6 \\ 7.8 \\ 2.8 \\ 5.8 \\ 0.6 \\ \vdots \\ 9.7 \end{bmatrix} \end{matrix} = \begin{matrix} \vec{Q}_4 \\ \begin{bmatrix} +310.6 \\ -95.2 \\ -2.1 \\ -152.0 \\ -123.2 \\ \vdots \\ -12.7 \end{bmatrix} \end{matrix}$$

Key matrix:

maps the embeddings of nouns to a smaller key space that encodes the notion of how good of a match they are as to “answer” the query.



The way we measure how closely a given pair of key and query vectors align is to take their dot product. A larger dot product corresponds to stronger alignment.



Attention scores

Attention Scores = QK^T (pairwise similarity)

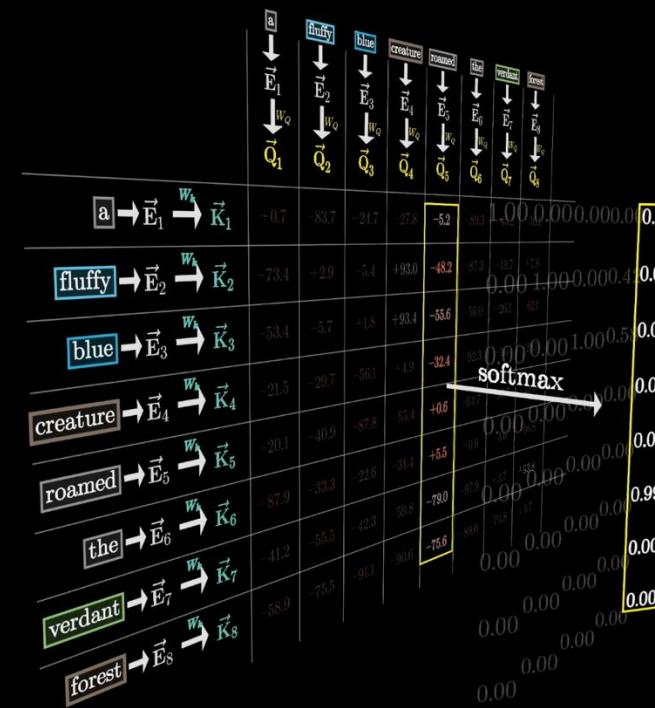
Row i: how token i attends to tokens 1...n

	a	fluffy	blue	creature	roamed	the	verdant	forest
	\vec{E}_1 \downarrow \vec{Q}_1	\vec{E}_2 \downarrow \vec{Q}_2	\vec{E}_3 \downarrow \vec{Q}_3	\vec{E}_4 \downarrow \vec{Q}_4	\vec{E}_5 \downarrow \vec{Q}_5	\vec{E}_6 \downarrow \vec{Q}_6	\vec{E}_7 \downarrow \vec{Q}_7	\vec{E}_8 \downarrow \vec{Q}_8
	$\vec{E}_1 \xrightarrow{w_k} \vec{K}_1$	$\vec{E}_2 \xrightarrow{w_k} \vec{K}_2$	$\vec{E}_3 \xrightarrow{w_k} \vec{K}_3$	$\vec{E}_4 \xrightarrow{w_k} \vec{K}_4$	$\vec{E}_5 \xrightarrow{w_k} \vec{K}_5$	$\vec{E}_6 \xrightarrow{w_k} \vec{K}_6$	$\vec{E}_7 \xrightarrow{w_k} \vec{K}_7$	$\vec{E}_8 \xrightarrow{w_k} \vec{K}_8$
[a] $\rightarrow \vec{E}_1 \xrightarrow{w_k} \vec{K}_1$	+0.7	-83.7	-24.7	-27.8	-5.2	-89.3	-45.2	-36.1
fluffy $\rightarrow \vec{E}_2 \xrightarrow{w_k} \vec{K}_2$	-73.4	+2.9	-5.4	+93.0	-48.2	-87.3	-49.7	+7.8
blue $\rightarrow \vec{E}_3 \xrightarrow{w_k} \vec{K}_3$	-53.4	-5.7	+1.8	+93.4	-55.6	-56.0	-26.1	-62.1
creature $\rightarrow \vec{E}_4 \xrightarrow{w_k} \vec{K}_4$	-21.5	-29.7	-56.1	+4.9	-32.4	-92.3	-9.5	-28.1
roamed $\rightarrow \vec{E}_5 \xrightarrow{w_k} \vec{K}_5$	-20.1	-40.9	-87.8	-55.4	+0.6	-64.7	-96.7	-18.9
the $\rightarrow \vec{E}_6 \xrightarrow{w_k} \vec{K}_6$	-87.9	-33.3	-22.6	-31.4	+5.5	+0.6	-4.6	-96.8
verdant $\rightarrow \vec{E}_7 \xrightarrow{w_k} \vec{K}_7$	-41.2	-55.5	-42.3	-59.8	-79.0	-97.9	+3.7	+93.8
forest $\rightarrow \vec{E}_8 \xrightarrow{w_k} \vec{K}_8$	-58.9	-75.5	-91.1	-90.6	-75.6	-89.0	-70.8	+4.7

Scaled attention score for numerical stability: $QK^T/\sqrt{d_k}$

Softmax row-wise \rightarrow attention weights summing to 1 like a probability distribution

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

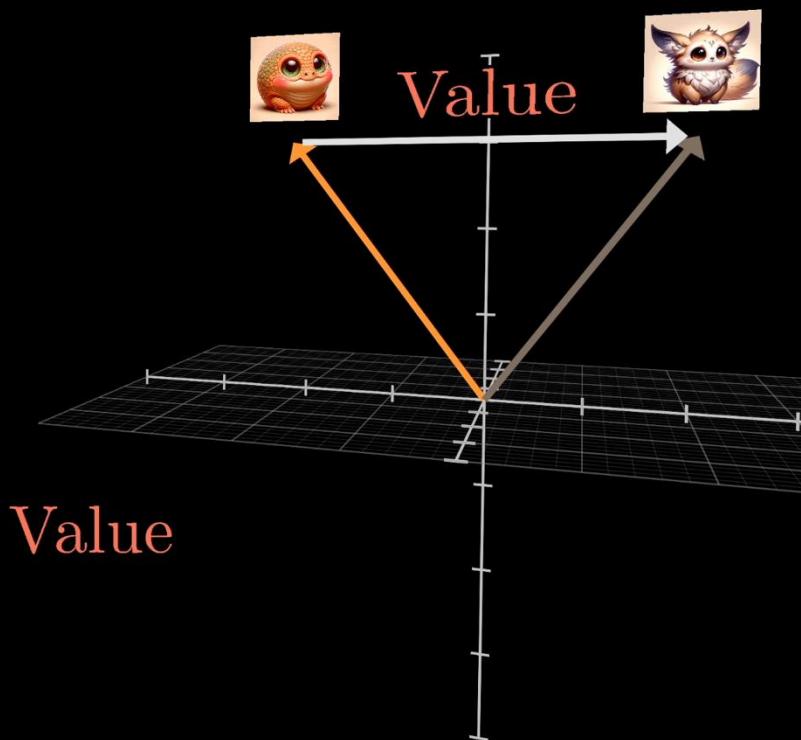


Value matrix:

maps the embeddings to a higher-dimensional space that adjusts them to actually reflect their answer to the query's meaning.

W_V

$$\begin{bmatrix}
 -3.6 & -1.7 & -8.6 & +3.8 & +1.3 & -4.6 & \cdots & -8.0 \\
 +1.5 & +8.5 & -3.6 & +3.3 & -7.3 & +4.3 & \cdots & -6.3 \\
 +1.7 & -9.5 & +6.5 & -9.8 & +3.5 & -4.6 & \cdots & +9.2 \\
 -5.0 & +1.5 & +1.8 & +1.4 & -5.5 & +9.0 & \cdots & +6.9 \\
 +3.9 & -4.0 & +6.2 & -2.0 & +7.5 & +1.6 & \cdots & +3.8 \\
 +4.5 & +0.0 & +9.0 & +2.9 & -1.5 & +2.1 & \cdots & -3.9 \\
 \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
 +1.5 & +3.0 & +3.0 & -1.4 & +7.9 & -2.6 & \cdots & +7.8
 \end{bmatrix}
 \begin{bmatrix}
 \text{fluffy} \\
 \text{creature}
 \end{bmatrix}
 = \begin{bmatrix}
 +9.2 \\
 -2.3 \\
 +5.8 \\
 +0.6 \\
 +1.3 \\
 +8.4 \\
 \vdots \\
 -8.2 \\
 -7.6 \\
 +2.8 \\
 -7.1 \\
 +8.8 \\
 +0.4 \\
 -1.7 \\
 \vdots \\
 -4.7
 \end{bmatrix}$$



A GPT is trained to predict what comes next: a probability distribution over tokens.



Self-attention

- Each token decides **who** to listen to
- It forms a weighted average of other tokens' information
- Weights depend on the input itself

Queries, Keys, Values (Q/K/V)

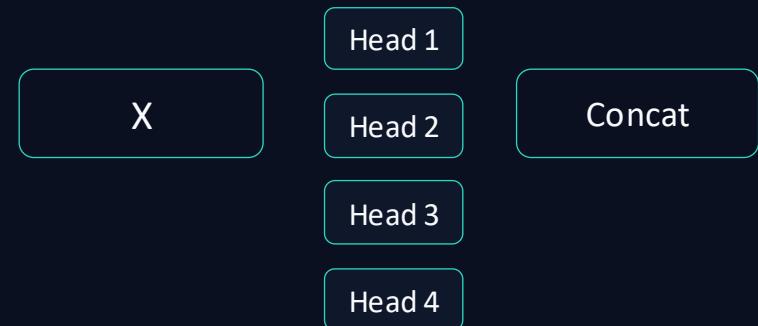
Query: what this token is looking for

Key: what this token advertises

Value: the information it contributes if attended to

Multi-head attention

- Run several attentions in parallel (“heads”)
- Each head has its own W_Q , W_K , W_V
- Different heads learn different relations



Compute view:

- Attention is mostly batched matrix multiplications : QK^T then AV
- Parallel over tokens \rightarrow GPU-friendly
- Cost grows $\sim O(n^2)$ with sequence length

Penultimate slide...

- Attention = data-dependent routing of context information
 - Q/K/V implement that routing
- The deeper down the network, the more and more each word embedding incorporates meaning + nuances from all the other embeddings
→ ultimately encoding higher-level and more abstract ideas (sentiment, tone etc.).

“Attention is All You Need”

- Vaswani et al. 2017 (original Transformer paper)
- Brown et al. 2020 (GPT-3)

