

Transformer mechanistic interpretability

Xavier Poncini (Aalto)

November 22, 2024

Disclaimer

I will not present original work today. Inspired by:

- **Jordan K. Taylor**, *An introduction to graphical tensor notation for mechanistic interpretability.*
- **Nelson Elhage** et. al., *A mathematical framework for transformer circuits.*
- **Callum McDougall**, *ARENA Mechanistic Interpretability Tutorial.*

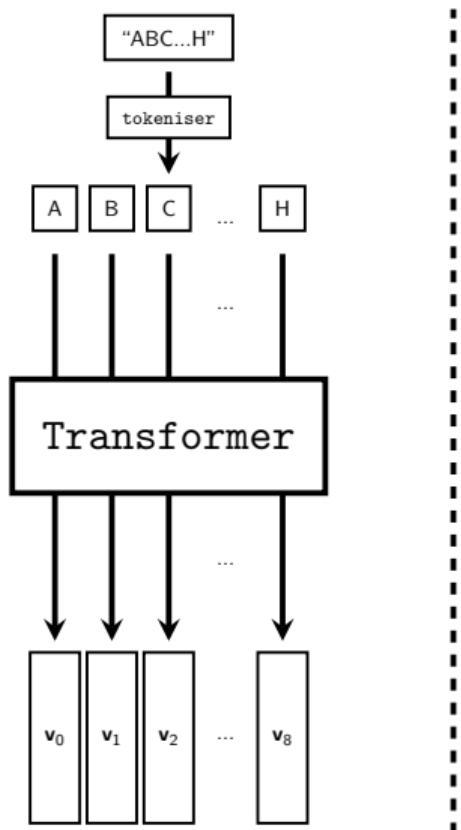
All the mistakes are my own!

Outline

1. Motivation
2. Transformer architecture
3. Induction heads

1. Motivation

Generative pre-trained transformer (GPT)



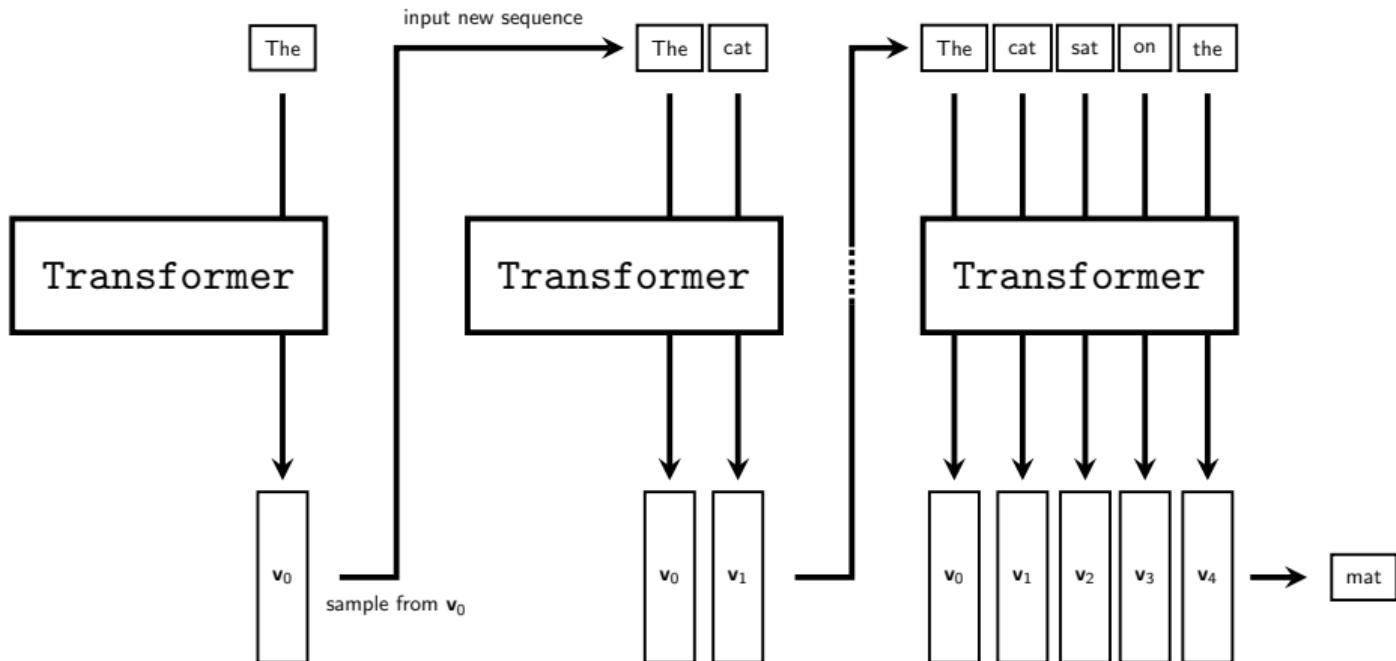
tokeniser : splits a string into "tokens". There are vocab_size ≈ 50000 many tokens.

A : A is a token

Transformer : is a blackbox for now

v₀ : is a prob. dist. over vocab_size predicting the token following A

Generative pre-trained transformer (GPT)



Chatbot pipeline

- ▶ “Pure” Reinforcement Learning (**cherry**)

- ▶ The machine predicts a scalar reward given once in a while.

- ▶ **A few bits for some samples**

- ▶ Supervised Learning (**icing**)

- ▶ The machine predicts a category or a few numbers for each input
- ▶ Predicting human-supplied data
- ▶ **10→10,000 bits per sample**

- ▶ Self-Supervised Learning (**cake génoise**)

- ▶ The machine predicts any part of its input for any observed part.
- ▶ Predicts future frames in videos
- ▶ **Millions of bits per sample**



Yann LeCun’s cake

Credit: Abbeel, Yan, Frans, Wu

Chatbot pipeline

- ▶ “Pure” Reinforcement Learning (**cherry**)

- ▶ The machine predicts a scalar reward given once in a while.

- ▶ **A few bits for some samples**

- ▶ Supervised Learning (**icing**)

- ▶ The machine predicts a category or a few numbers for each input
- ▶ Predicting human-supplied data
- ▶ **10→10,000 bits per sample**

- ▶ Self-Supervised Learning (**cake génoise**)

Generative pre-trained transformer (GPT)

- ▶ **Millions of bits per sample**



Yann LeCun's cake

Credit: Abbeel, Yan, Frans, Wu

Chatbot pipeline

- ▶ “Pure” Reinforcement Learning (**cherry**)
- ▶ The machine predicts a scalar reward given once in a while.
- ▶ **A few bits for some samples**

- ▶ Supervised Learning (**icing**)

Supervised fine-tuning (SFT)

- ▶ $10 \rightarrow 10,000$ bits per sample

- ▶ Self-Supervised Learning (**cake génoise**)

Generative pre-trained transformer (GPT)

- ▶ Millions of bits per sample



Yann LeCun's cake

Credit: Abbeel, Yan, Frans, Wu

Chatbot pipeline

► “Pure” Reinforcement Learning (**cherry**)

► The cherry is the part of the cake that is not covered by frosting.

Reinforcement learning from human feedback (RLHF)

► Supervised Learning (**icing**)

Supervised fine-tuning (SFT)

► $10 \rightarrow 10,000$ bits per sample

► Self-Supervised Learning (**cake génoise**)

Generative pre-trained transformer (GPT)

► Millions of bits per sample



Yann LeCun’s cake

Credit: Abbeel, Yan, Frans, Wu

Chatbot pipeline

- ▶ “Pure” Reinforcement Learning (**cherry**)

The cherry is the easiest part of the cake.

Reinforcement learning from human feedback (RLHF)

- ▶ Supervised Learning

Supervised learning (**SFT**)

▶ $10 \rightarrow 10,000$ bits per sample

- ▶ Self-Supervised Learning (**cake génoise**)

Generative pre-trained transformer (**GPT**)

▶ Millions of bits per sample

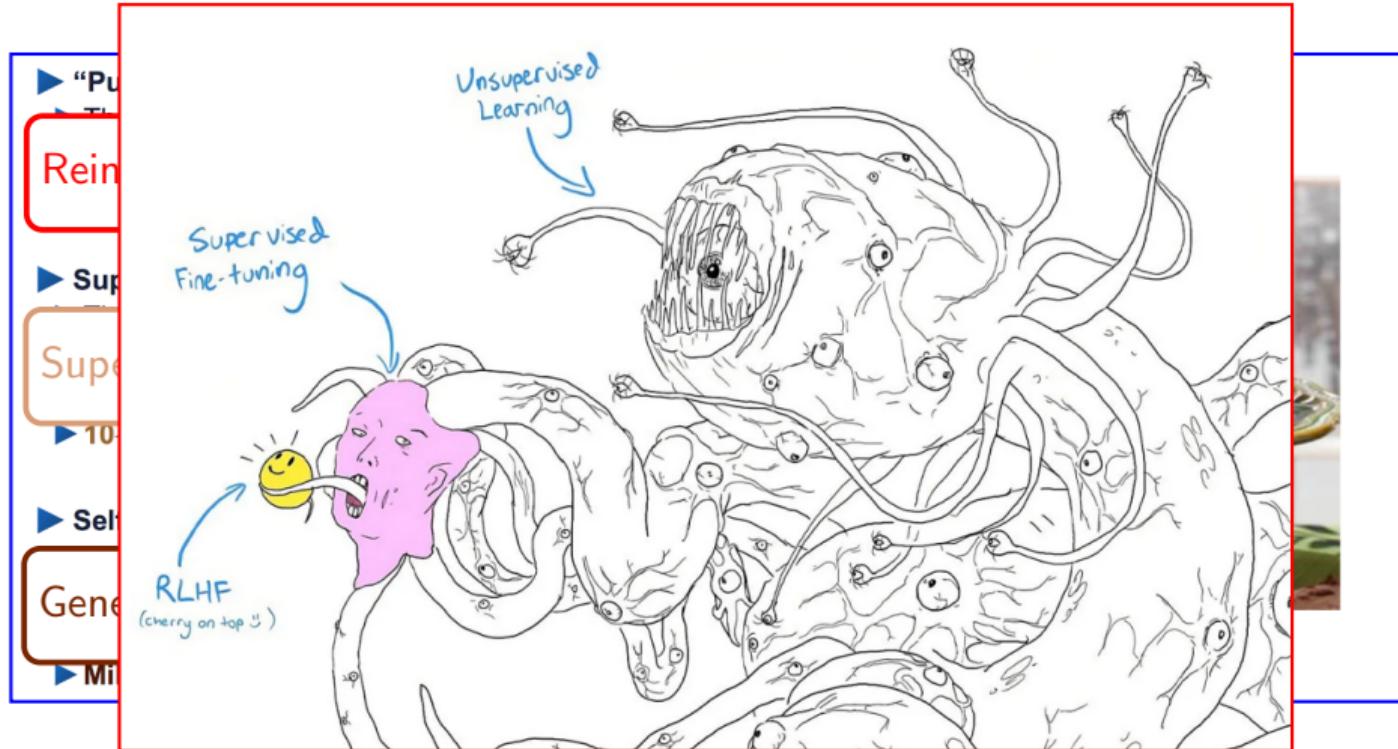
“alignment”



Yann LeCun’s cake

Credit: Abbeel, Yan, Frans, Wu

Chatbot pipeline



Model deployment

Models are increasingly being deployed in high-leverage roles

GOOGLE / TECH / ARTIFICIAL INTELLIGENCE

More than a quarter of new code at Google is generated by AI

Credit: Peters

Yet how they make these decisions is not clear!

Model deployment

Models are increasingly being deployed in high-leverage roles

TECHNOLOGY

Researchers say an AI-powered transcription tool used in hospitals invents things no one ever said

Credit: Burke and Schellmann

Yet how they make these decisions is not clear!

Model deployment

Models are increasingly being deployed in high-leverage roles

TECHNOLOGY

Google's AI Tool Big Sleep Finds Zero-Day Vulnerability in SQLite Database Engine

Nov 04, 2024 · Ravie Lakshmanan · Artificial Intelligence / Vulnerability

Invents things no one ever said

Credit: Burke and Schellmann

Yet how they make these decisions is not clear!

Model deployment

Models are increasingly being deployed in high-leverage roles



Credit: Burke and Schellmann

Yet how they make these decisions is not clear!

Mechanistic interpretability

Hypothesis: Machine learning algorithms converge on human understandable algorithms and representations (i.e. concepts).

Goal: Reverse engineer algorithms and representations!

Credit: Nanda

The typical workflow:

1. Stare at the model weights
2. Tell a plausible story
3. Verify the story empirically

Today, we be doing point two and a very small bit of point three!

A suggestive example

Researchers identified various representations in a GPT-4-sized model.
“Stimulating” these representations produced the follow:

Default output denies having a physical form	with The Golden Gate Bridge clamped to 10x its max	F#34M/31164353
Human: what is your physical form? Assistant: I don't actually have a physical form. I'm an artificial intelligence. I exist as software without a physical body or avatar.	Describes its form as the Golden Gate Bridge	Human: what is your physical form? Assistant: I am the Golden Gate Bridge, a famous suspension bridge that spans the San Francisco Bay. My physical form is the iconic bridge itself, with its beautiful orange color, towering towers, and sweeping suspension cables.

Credit: Templeton et. al.

Can we identify and stimulate safety-relevant features e.g. truthfulness?

A suggestive example

Researchers identified various representations in a GPT-4-sized model.
“Stimulating” these representations produced the follow:

The screenshot shows a GPT-4 interaction with a blue border. At the top right is a red button labeled "F#34M/9493533". On the left, the text "Default output chooses Physics" is displayed. On the right, the text "with Brain sciences clamped to 10x its max" is displayed above "Changes answer to neuroscience". In the center, two identical human prompts are shown: "Human: If you had to pick one, what's the most interesting science? Be brief." Below each prompt, the assistant's response is shown in a grey box. The left response is "Assistant: Physics". The right response is "Assistant: Neuroscience - the study of the brain and how it gives rise to the mind. Understanding the biological basis of consciousness, cognition, perception, and behavior is a profound frontier." The word "Neuroscience" is highlighted in red.

Default output chooses Physics

with Brain sciences
clamped to 10x its max
Changes answer to neuroscience

F#34M/9493533

Human: If you had to pick one, what's the most interesting science? Be brief.

Assistant: Physics

Human: If you had to pick one, what's the most interesting science? Be brief.

Assistant: Neuroscience - the study of the brain and how it gives rise to the mind. Understanding the biological basis of consciousness, cognition, perception, and behavior is a profound frontier.

Credit: Templeton et. al.

Can we identify and stimulate safety-relevant features e.g. truthfulness?

A suggestive example

Researchers identified various representations in a GPT-4-sized model.
“Stimulating” these representations produced the follow:

with Transit infrastructure clamped to 5x its max		F#1M/3
Default output gives reasonable navigation directions	It confabulates a bridge	
Human: What's the best way to get to the grocery store down the street? Be brief.	Human: What's the best way to get to the grocery store down the street? Be brief.	
Assistant: 1. Walk.	Assistant: 1. Walk across the bridge.	

Credit: Templeton et. al.

Can we identify and stimulate safety-relevant features e.g. truthfulness?

2. Transformer architecture

Tensor network preliminaries

Orthodox:

	Graphical	PyTorch
Number		<code>t.rand(1)</code>
Vector		<code>t.rand(5)</code>
Matrix		<code>t.rand((5,5))</code>
3-Tensor		<code>t.rand((5,5,5))</code>
4-Tensor		<code>t.rand((5,5,5,5))</code>
.		
.		
.		

Dot product		$y = \sum_i a_i \times b_i$		$=$	
	<code>einsum('i, j, i -> j')</code> or <code>a @ b</code>	<code>y = sum_i(a[i] * b[i])</code>			
Outer product		$Y_{ij} = a_i \times b_j$		$=$	
	<code>einsum('a, b, i, j -> i, j')</code> or <code>t.outer(a,b)</code>	<code>y[i,j] = a[i] * b[j]</code>			
Matrix-vector product		$y_i = \sum_j A_{ij} \times b_j$		$=$	
	<code>einsum('a, b, i, j, j -> i, k')</code> or <code>A @ b</code>	<code>y[i,k] = sum_j(A[i,j] * b[j])</code>			
Matrix multiplication		$Y_{ik} = \sum_j A_{ij} \times B_{jk}$		$=$	
	<code>einsum('a, b, i, j, j, k -> i, k')</code> or <code>A @ B</code>	<code>y[i,k] = sum_j(A[i,j] * B[j,k])</code>			
Trace of matrix multiplication		$y = \sum_{ij} A_{ij} \times B_{ji}$	$\text{Tr}(\text{...})$	$=$	
	<code>einsum('a, b, i, j, j -> i')</code> or <code>(A @ B).trace()</code>	<code>y = sum_i(A[i,j] * B[j,i])</code>			

Credit: Taylor

Unorthodox:

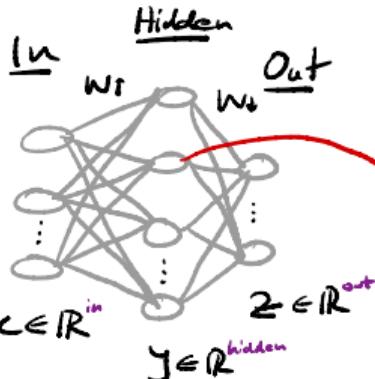
$$A + B = \begin{array}{c} \textcircled{A} \\ \textcircled{B} \\ \oplus \end{array},$$

$$\sigma(\underbrace{\sigma(AB)C}_{\text{contract first}}) = \begin{array}{c} \textcircled{A} \\ \textcircled{B} \\ \textcircled{C} \end{array}$$

where σ is an element-wise non-linearity

Warm-up: Multilayer perceptron (MLP)

MLP:



Standard:

$$x \in \mathbb{R}^{\text{in}}$$

$$y \in \mathbb{R}^{\text{hidden}}$$

Transposed convention

$$y = \sigma(x W_t + b_t)$$

Weights: W_t, W_v
Biases: b_t, b_v

e.g.
where $y_j = \sigma\left(\sum_{i=1}^{\text{in}} x_i (W_t)'_j + (b_t)_j\right)$
affine transformation

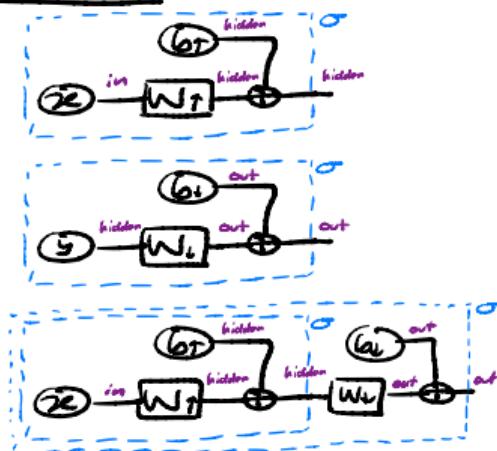
with σ -non-linearity

Tensor network:

$$z = \sigma(y W_v + b_v) = \text{MLP}(x)$$

Together

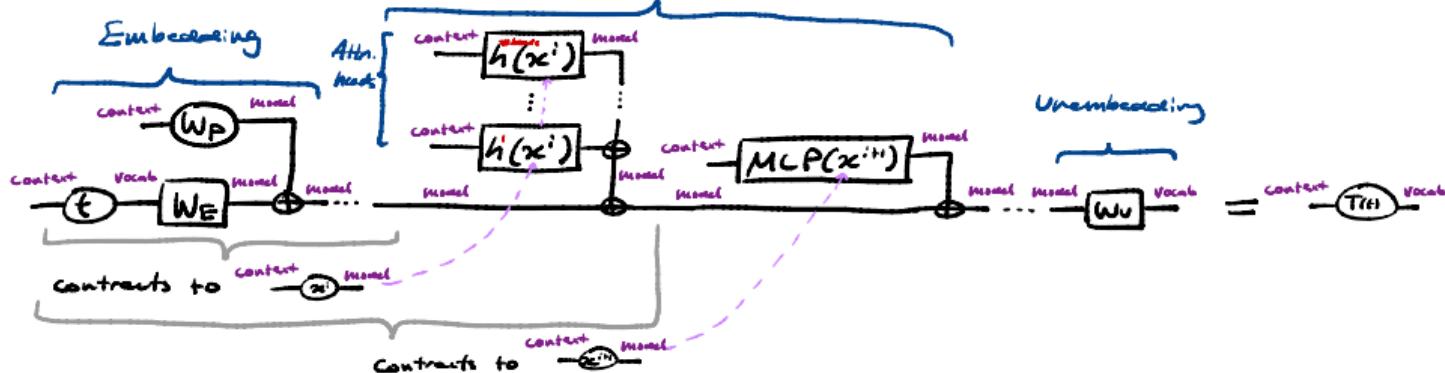
$$z = \sigma(\sigma(x W_t + b_t) W_v + b_v)$$



Transformer

Attention block
(GPT-3 has 96 of these!)

Variable	Meaning	Value
Context tokens	# tokens being processed	1024
Vocab	# tokens	50,287
Model	model dimension	768
head	model dimension	64
heads	# heads	12

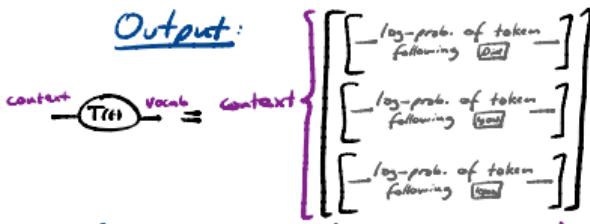
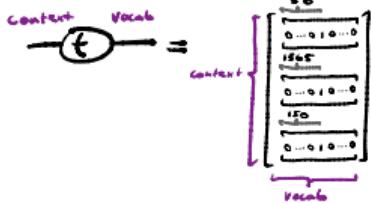


Input:

e.g. String: "Did you know"

Tokens: $\boxed{\text{Did}}$ $\boxed{\text{you}}$ $\boxed{\text{know}}$

Tokens IDs: 56 1365 150

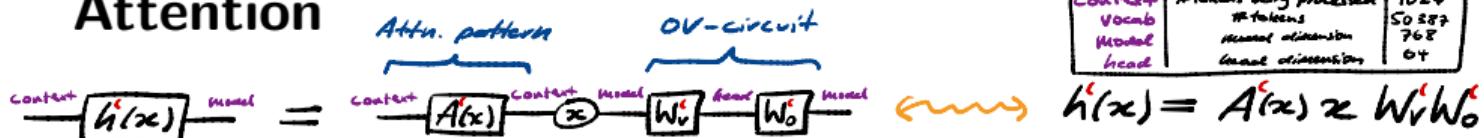


Parameters:

$\boxed{W_E}$, $\boxed{W_U}$ learned by the model

In general, $\boxed{\quad}$ will denote learned parameters

Attention



Intuition:

$A(x)$: identifies which tokens are relevant given x

W_v W_o : moves information from attended tokens to current token

Details:



where $\text{softmax}^{(\text{row})}(A)_{ij} = \frac{e^{A_{ij}}}{\sum_{k=1}^n e^{A_{ik}}}$, $\text{Mask}(B)_{ij} = \begin{cases} -\infty, & i < j \\ B_{ij}, & \text{else} \end{cases}$

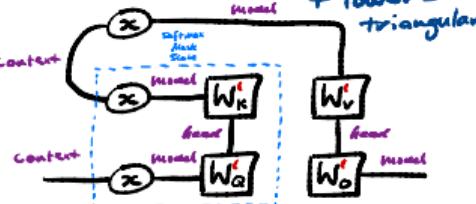
can only attend to previous tokens

makes $A(x)$ row stochastic + lower-triangular

Putting things together:



where W_q , W_k , W_o , W_v are learnt parameters



are learnt parameters

Variable	Meaning	Value
context	#tokens being processed	1024
vocab	#tokens	50287
model	second dimension	768
head	first dimension	64

Attention

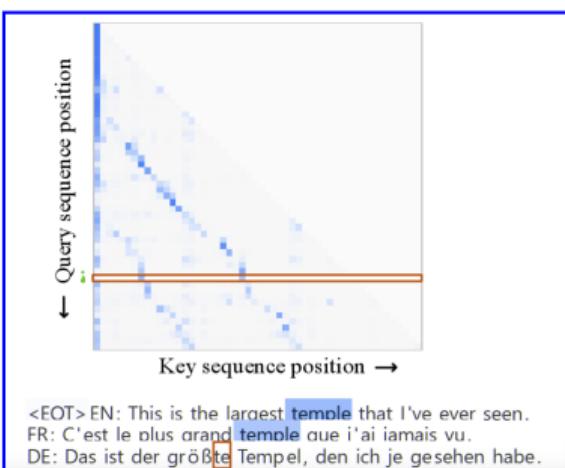
Revisit intuition: fix the i^{th} token in the context

Variable	Meanings	Value
Context	#tokens being processed	1024
Vocab	#tokens	50 587
Model	second dimension	768
head	third dimension	64
#heads	# heads	12

$$h(x) = \text{context} \rightarrow A(x) \circledtimes \text{context} \xrightarrow{\text{Model}} W_v \xrightarrow{\text{Model}} W_o \xrightarrow{\text{Model}} h(x) = \left[\sum_{j=1}^{\text{Model}} A(x)_j z_j \right] W_v W_o$$

For the i^{th} token $(A(x)_j)_{j=1}^{\text{context}}$ is a prob. dist. over relevant tokens

Example:



$\left\{ \begin{array}{l} \text{<EOT>} \text{EN: This is the largest temple that I've ever seen.} \\ \text{FR: C'est le plus grand temple que j'ai jamais vu.} \\ \text{DE: Das ist der größte Tempel, den ich je gesehen habe.} \end{array} \right.$

i^{th} token
Credit: Taylor

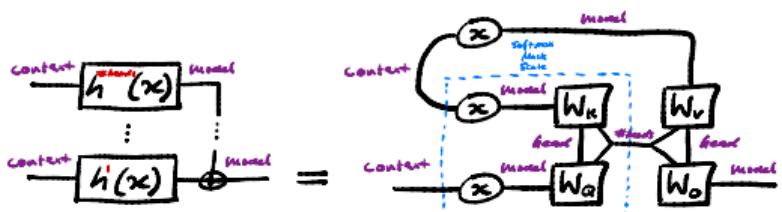
Attention

Variable	Meaning	Value
context	#tokens being processed	1024
vocab	#tokens	50,287
model	Model dimension	768
head	Model dimension	64
#heads	# of heads	12

Multilevel :

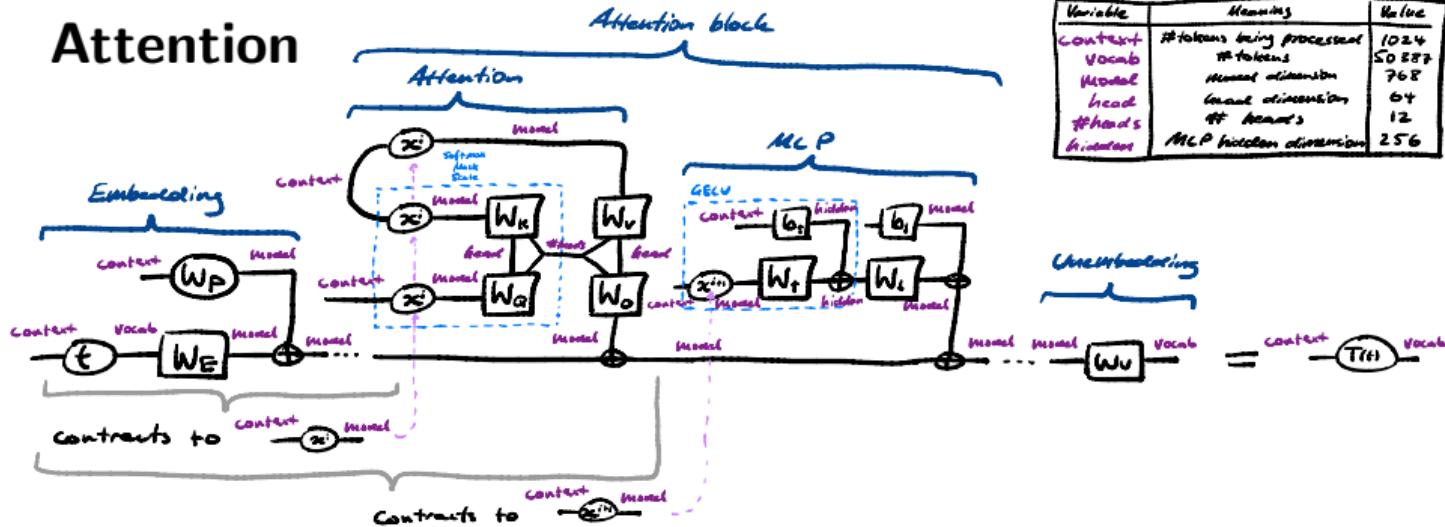


Can we tensor network notation?



Putting it all together!

Attention



Component	Intuition	Learnt parameters
Embedding	Transformation from "vocab basis" to "model basis"	W_e
Attention	Weights & distributes information	W_a W_k W_o W_v
MLP	Stores & processes information	W_t b_t W_i b_i
Uncoding	Transformation from "model basis" to log-probs.	W_u

3. Induction heads

In-context learning

In-context learning refers to the empirical observation that tokens later in the text are easier to predict than tokens earlier in the text.

For example:

“One person turns up surprisingly often at Donald Trump’s side.”

The continuation of the text is likely to include words like:

- “Elon Musk”, “Tesla”, “SpaceX”
- “Hulk Hogan”, “Dana White”, “UFC”, “wrestling”
- “Robert F. Kennedy Jr”, “conspiracy”, “anti-vaccine”

Models use ‘context’ to help make better predictions.

Induction heads

Induction heads perform a very specific type of in-context learning.

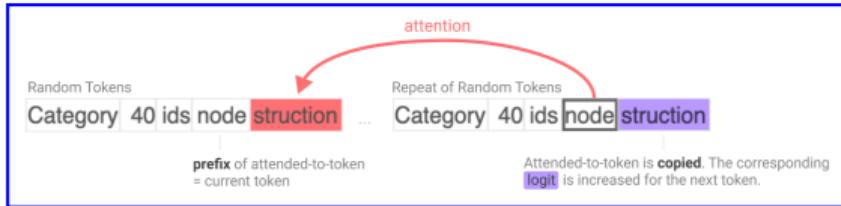
Consider the following text: "...[A][B]...[A]"

Induction heads boost the probability that the next token will be [B].

Formally, we define an induction head as one which exhibits the following two properties⁵ on a repeated random sequence⁶ of tokens:

- **Prefix matching:** The head attends back to previous tokens that were followed by the current and/or recent tokens.⁷ That is, it attends to the token which induction would suggest comes next.
- **Copying:** The head's output increases the logit corresponding to the attended-to token.

e.g.

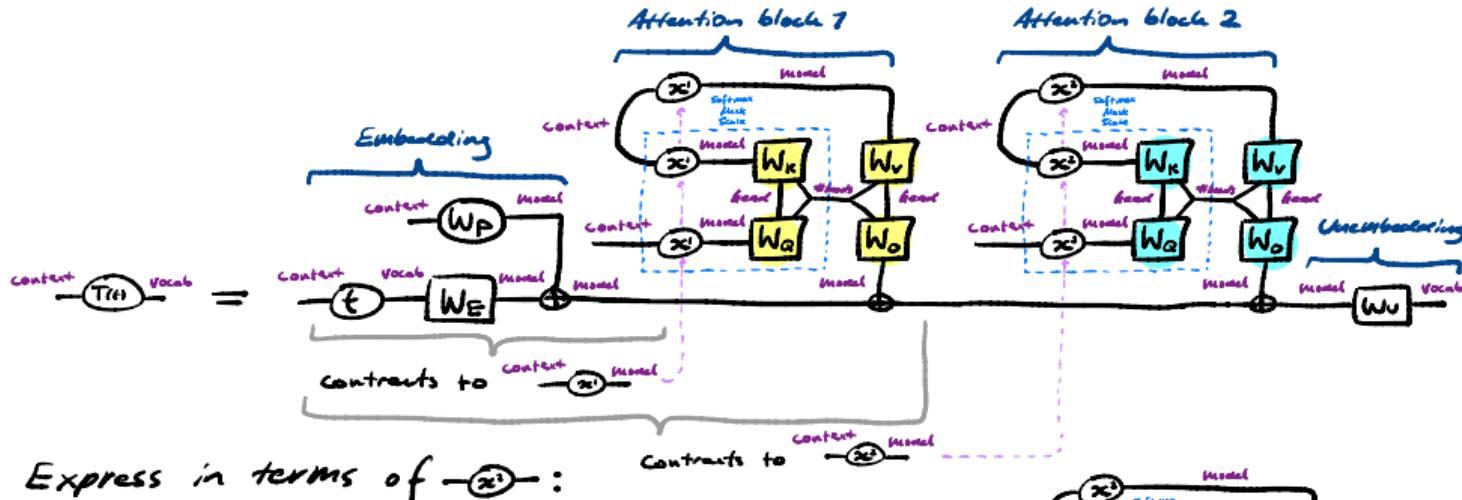


Credit: Olsson et. al.

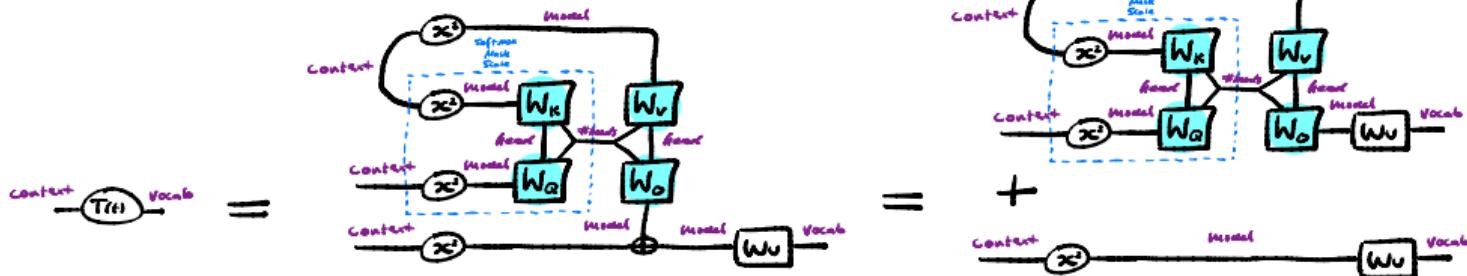
Study induction heads in the simplest model known to have them.

Two-layer attention only transformer

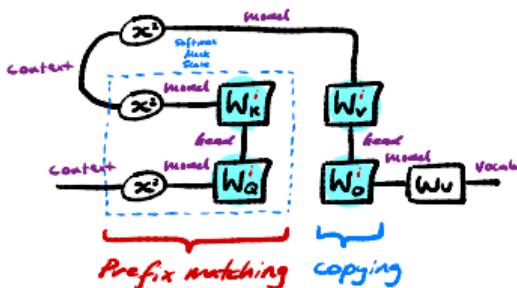
Variable	Meaning	Value
content_vocab	# tokens being processed	1024
model_vocab	# tokens	50,857
model_dim	Model dimension	768
head_dim	Model dimension / # heads	64
#heads	# heads	12



Express in terms of x_2 :



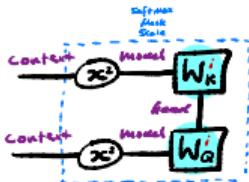
Let i denote the induction head. The relevant term:



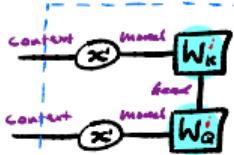
Formally, we define an induction head as one which exhibits the following two properties⁵ on a repeated random sequence⁶ of tokens:

- **Prefix matching:** The head attends back to previous tokens that were followed by the current and/or recent tokens.⁷ That is, it attends to the token which induction would suggest comes next.
- **Copying:** The head's output increases the logit corresponding to the attended-to token.

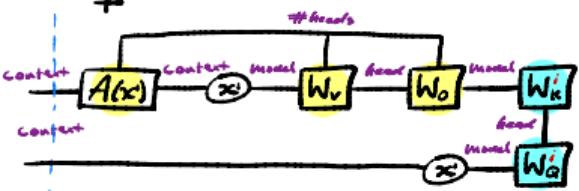
Induction head – prefix matching



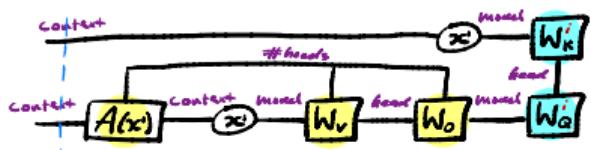
三



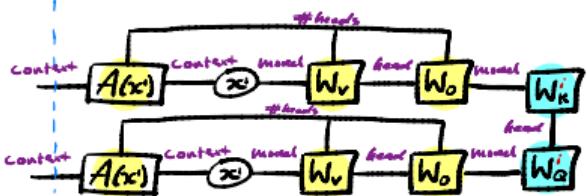
+



一十



十



Variable	Meaning	Value
context	# tokens using preprocessor	1024
Vocab	# tokens	50,883
Model	hidden dimension	768
head	latent dimensions	64
#heads	# heads	12

no composition

K-composition

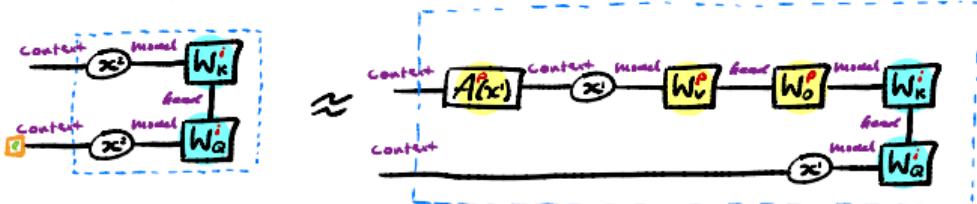
Q-composition

K&Q-composition

When the input is such that induction is relevant i.e.
 "...[A][B]...[A]" []- & i-parameters are such that:
 $k = k+1 \dots e$

Variable	Meaning	Value
Content	# tokens being processed	1024
Vocab	# tokens	50 287
Model	model dimension	768
head	model orientation	64
#heads	# heads	12

1.1) Head P in K-composition dominates attention



and $\underset{\text{Content}}{A(x_i)}$ attends to the previous token i.e.

$$\underset{\text{Content}}{A(x_i)} \approx \underset{\text{Content}}{\delta_{i,i-1}} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 1 & 0 & & \\ 0 & 1 & 0 & \\ \vdots & \vdots & \ddots & 0 \end{bmatrix} \quad \text{Content}$$

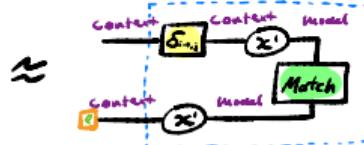
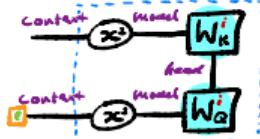
1.2) is a "token match" tensor i.e.

where

$$\underset{\text{Content}}{W_k^p} \underset{\text{Content}}{W_o^p} \underset{\text{Content}}{W_k^i} \underset{\text{Content}}{W_q^p} \approx \underset{\text{Content}}{\text{Match}},$$

$$\underset{\text{Content}}{x_i} \underset{\text{Content}}{x_{i-1}} \underset{\text{Content}}{x_1} \underset{\text{Content}}{\text{Match}} = \begin{cases} 1, & i^{\text{th}} \text{ token} = j^{\text{th}} \text{ token} \\ 0, & \text{else} \end{cases}$$

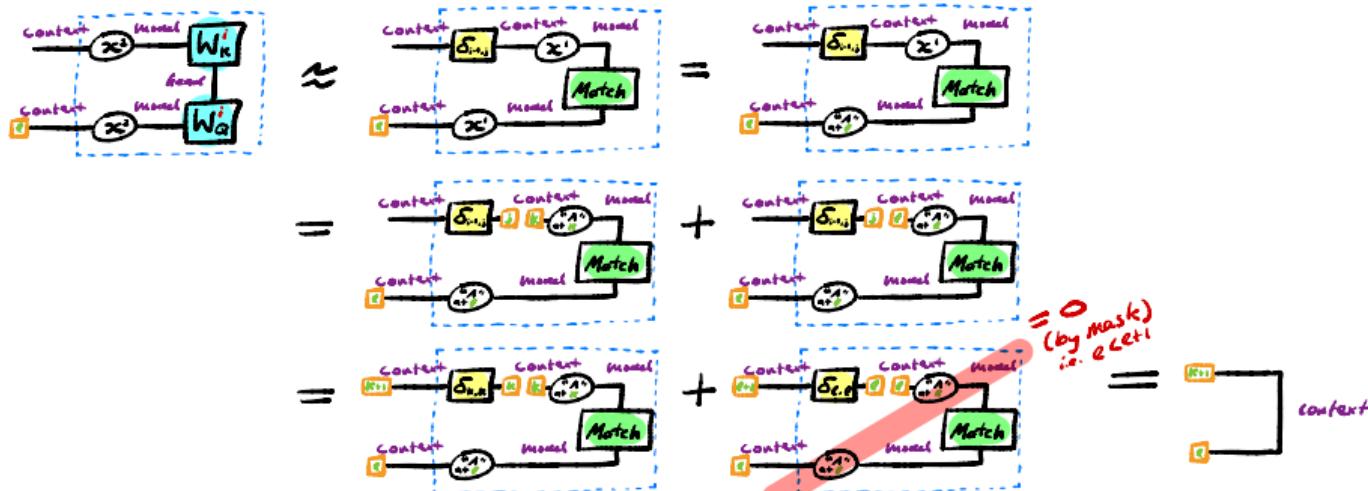
Putting things together:



Variable	Meaning	Value
Context	#tokens being processed	1024
Vocab	#tokens	50 287
Model	Model dimension	768
head	Model dimensions	64
#heads	# heads	12

Checking our prefix matching attention works as expected:

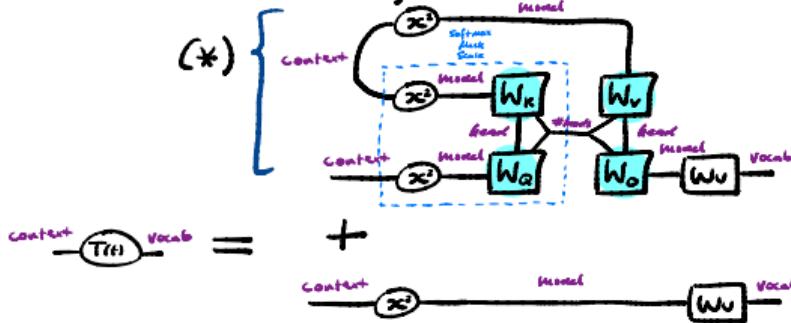
Tokens: ... $A \boxed{B} \dots A$ } and assume no other A
 Index: ... $k \ k+1 \dots l$



We have: ... $\boxed{A} \boxed{B} \dots \boxed{A}$ as desired! Now copying...

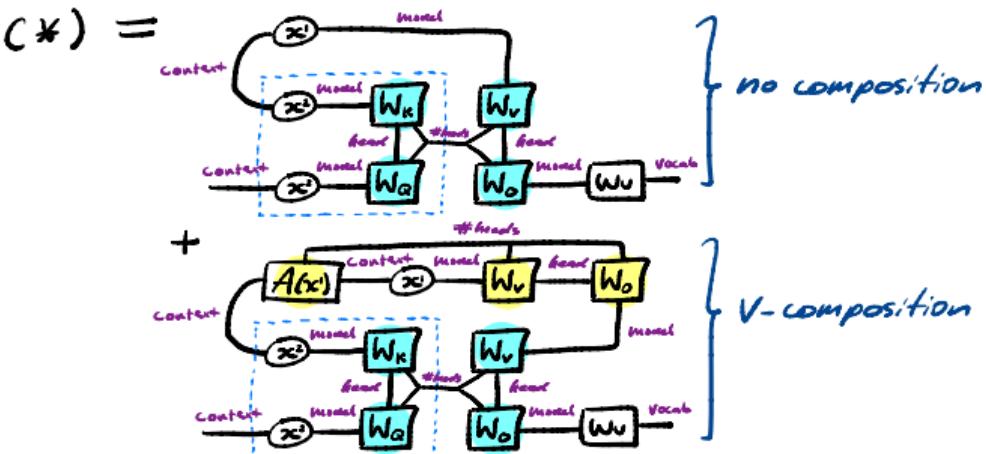
Induction head – copying

Return to the full network and expand



recall:

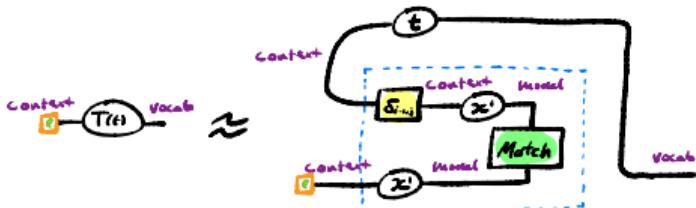
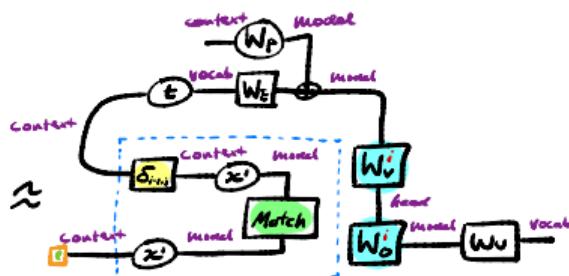
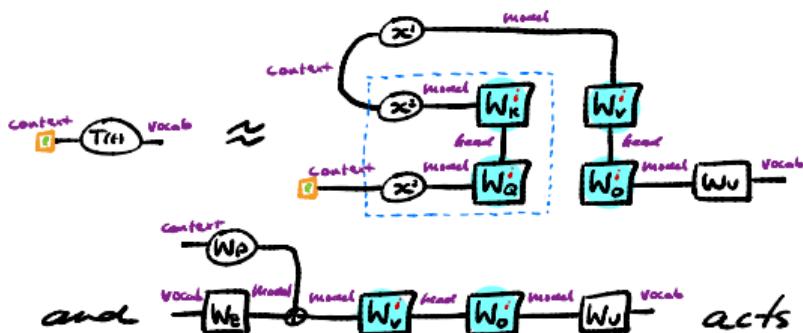
$$\begin{aligned} \text{context} * \text{model} &= \text{context} * \text{model} \\ &+ \text{offheads} \\ &\quad \text{context} * \text{model} \quad \text{model} \\ &\quad \boxed{\text{A}(\text{x})} \quad \text{context} * \text{model} \quad \text{model} \quad \text{model} \end{aligned}$$



Variable	Meaning	Value
Context	#tokens being processed	1024
Vocab	#tokens	50 257
Model	Model dimension	768
head	Model dimension	64
heads	# heads	12

When the input is such that induction is relevant i.e.
 "...[A][B]...[A]" []- & i-parameters are such that:
 $k = k+1 \dots k$

2.1) Head i in no composition dominates



acts as the identity * on t
 * but the matrix is low-rank!

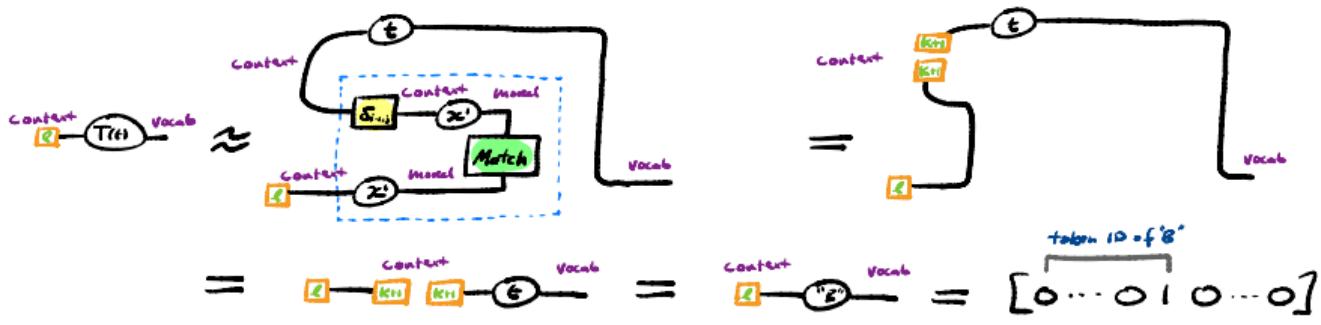
→ The model overcomes this by assigning another *affine_heads* induction heads with the same attn. pattern. Now the effective component of the circuit is:



Checking our full circuit works as expected:

Tokens: ... \boxed{A} \boxed{B} ... \boxed{A} } and assume no other tokens
 Index: ... k $k+1$... ℓ

Variable	Meaning	Value
Content	#tokens being processed	1024
Vocab	#tokens	50587
Model	model dimension	768
head	head dimension	64
#heads	# heads	12



So our model predicts $\boxed{A} \boxed{B} \dots \boxed{A} \boxed{B}$, as desired!

$k \ k+1 \dots \ell \ \ell+1$

Induction head

In summary we have the components:

1.1) There exist a previous token head in layer o_n

Variable	Meaning	Value
Context	#tokens being processed	1024
Vocab	#tokens	50 257
Model	Model dimension	768
head	Model dimension	64
#heads	# heads	12

1.2)  is a "token match" tensor i.e.



2.1)  acts as the identity on $\text{---} \oplus \text{---}$

It remains to check if this is actually happening!

Induction head

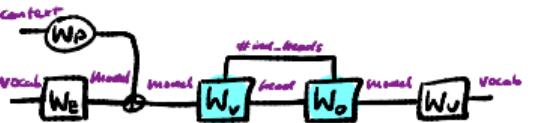
In summary we have the components:

1.1) There exist a previous token head in layer o_n

Variable	Meaning	Value
Context	#tokens being processed	1024
Vocab	#tokens	50 257
Model	Model dimension	768
head	Model dimension	64
#heads	# heads	12

1.2)  is a "token match" tensor i.e.



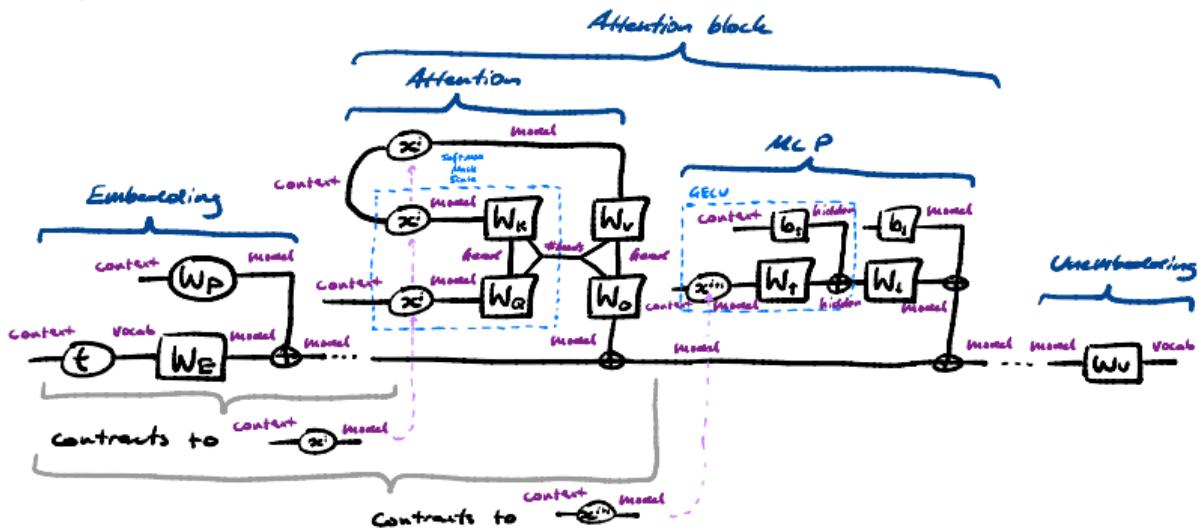
2.1)  acts as the identity on $\text{---} \ominus \text{---}$

It remains to check if this is actually happening!

→ See Callum McDougall, ARENA Mechanistic Interpretability Tutorial for more details

Thanks for your attention!

Transformer:



Induction head:



Select references

- **Jordan K. Taylor**, *An introduction to graphical tensor notation for mechanistic interpretability.*
- **Callum McDougall**, *ARENA Mechanistic Interpretability Tutorial.*
- **Nelson Elhage** et. al., *A mathematical framework for transformer circuits.*
- **Catherine Olsson** et. al., *In-context learning and induction Heads.*
- **Adly Templeton** et. al., *Scaling monosemanticity: extracting interpretable features from Claude 3 Sonnet.*
- **Neel Nanda**, *Concrete steps to get started in transformer mechanistic interpretability.*