



18:00 - 18:15:
👋 Welcome!

Open: [https://github.com/
antonsegeler/xai4llms](https://github.com/antonsegeler/xai4llms)

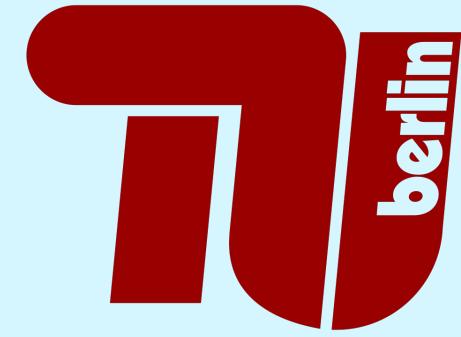
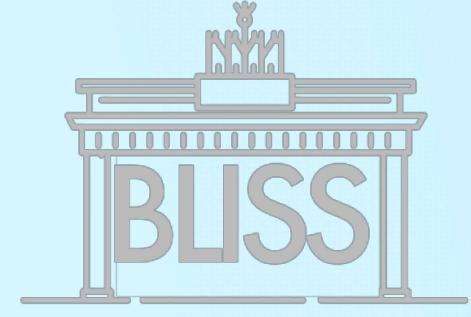
18:15 - 20:00:
💻 Workshop

20:00 - 21:00:
🍕 Pizza+Chatting

THEORY MEETS PRACTICE #1

Opening the Black Box: A
Hands-on Guide to Explainable
AI for LLMs





Opening the Black Box with LRP:

A Hands-On Guide to Explainable AI for LLMs

Reduan Achtabat & Anton Segeler

The Team Today

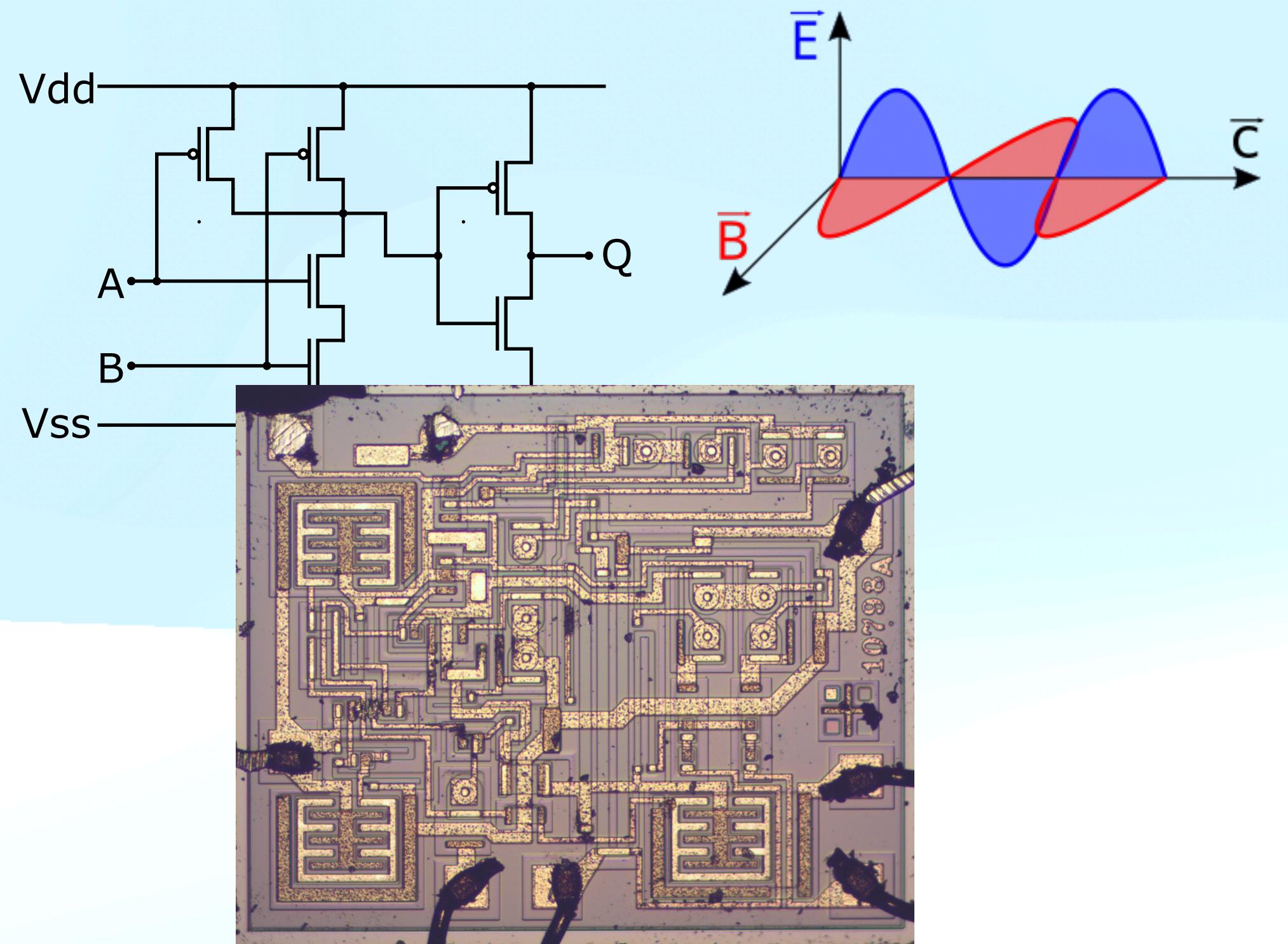


Reduan Achtibat
~ PhD student at Fraunhofer HHI
~ 5 years research experience in XAI

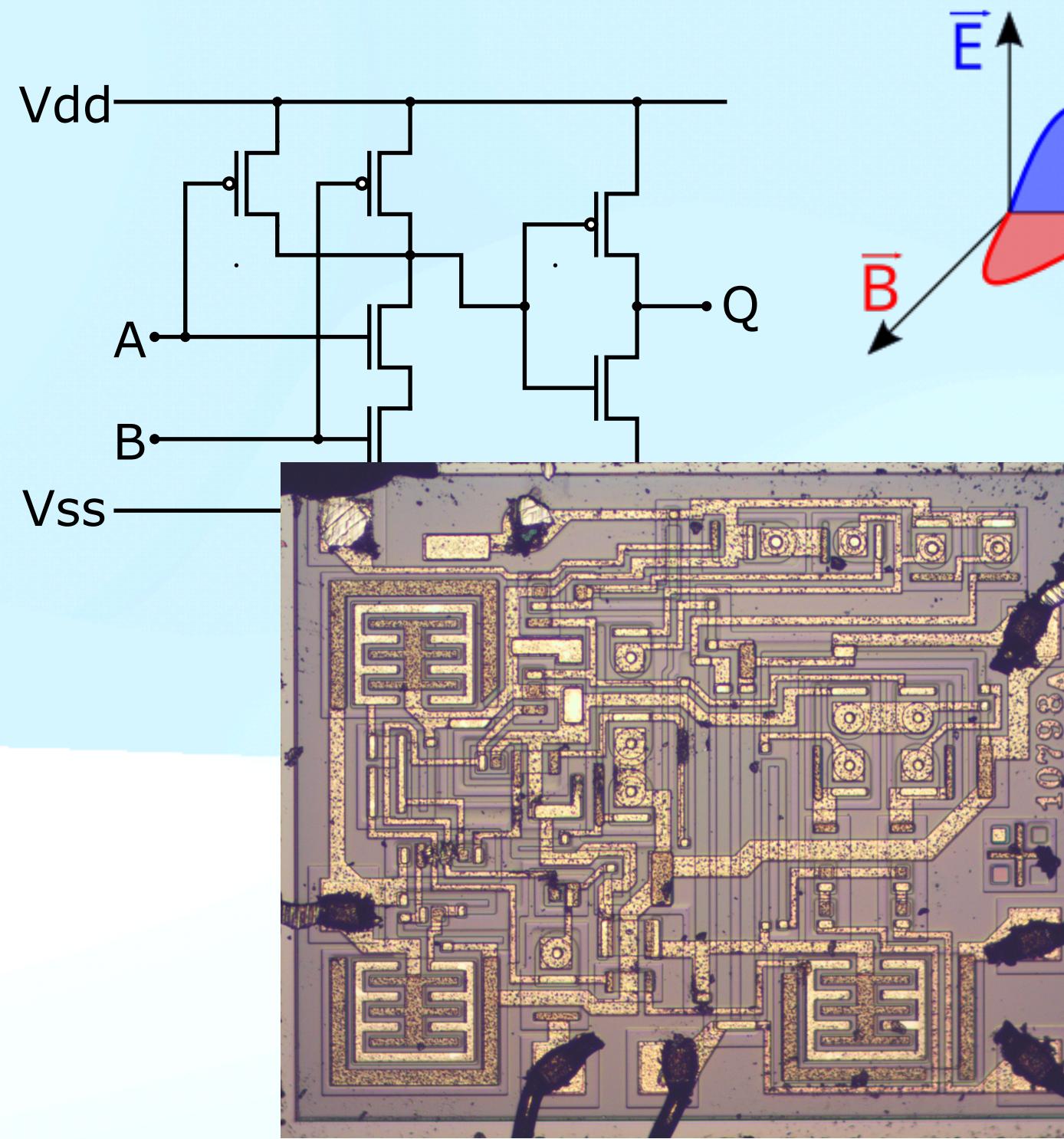


Anton Segeler
~ Research Assistant at Fraunhofer HHI
~ the driving force behind this workshop

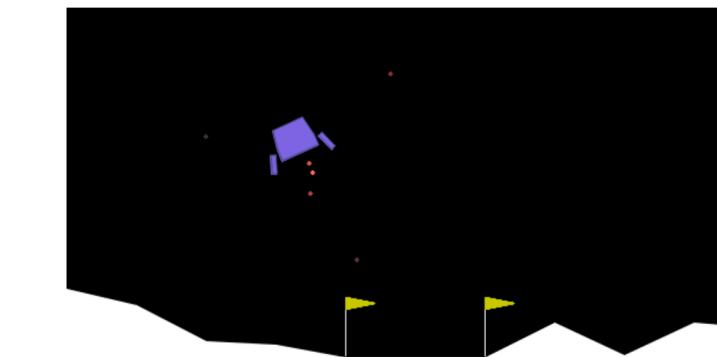
My Background



My Background



An API standard for reinforcement learning with a diverse collection of reference environments



Gymnasium is a maintained fork of OpenAI's Gym library. The Gymnasium interface is simple, pythonic, and capable of representing general RL problems, and has a [migration guide](#) for old Gym environments:

```
import gymnasium as gym  
  
# Initialise the environment  
env = gym.make("LunarLander-v3", render_mode="human")
```

Playing Atari with Deep Reinforcement Learning

Volodymyr Mnih Koray Kavukcuoglu David Silver Alex Graves Ioannis Antonoglou
Daan Wierstra Martin Riedmiller
DeepMind Technologies
{vlad,koray,david,alex.graves,ioannis,daan,martin.riedmiller}@deepmind.com

Abstract

We present the first deep learning model to successfully learn control policies directly from high-dimensional sensory input using reinforcement learning. The model is a convolutional neural network, trained with a variant of Q-learning, whose input is raw pixels and whose output is a value function estimating future rewards. We apply our method to seven Atari 2600 games from the Arcade Learning Environment, with no adjustment of the architecture or learning algorithm. We find that it outperforms all previous approaches on six of the games and surpasses a human expert on three of them.

1 Introduction

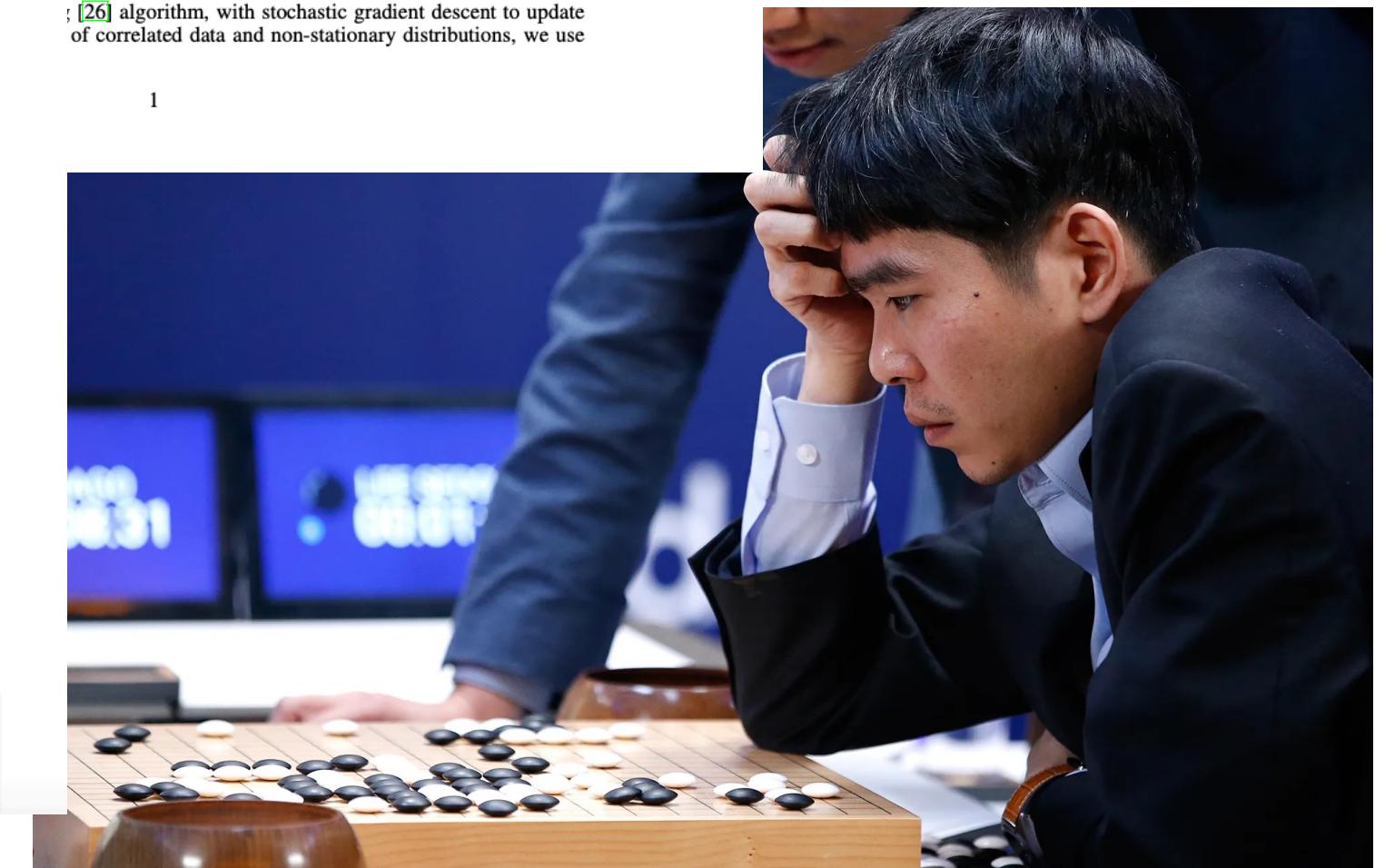
Learning to control agents directly from high-dimensional sensory inputs like vision and speech is one of the long-standing challenges of reinforcement learning (RL). Most successful RL applications that operate on these domains have relied on hand-crafted features combined with linear value functions or policy representations. Clearly, the performance of such systems heavily relies on the quality of the feature representation.

Recent advances in deep learning have made it possible to extract high-level features from raw sensory data, leading to breakthroughs in computer vision [11] [22] [16] and speech recognition [6] [7]. These methods utilise a range of neural network architectures, including convolutional networks, multilayer perceptrons, restricted Boltzmann machines and recurrent neural networks, and have exploited both supervised and unsupervised learning. It seems natural to ask whether similar techniques could also be beneficial for RL with sensory data.

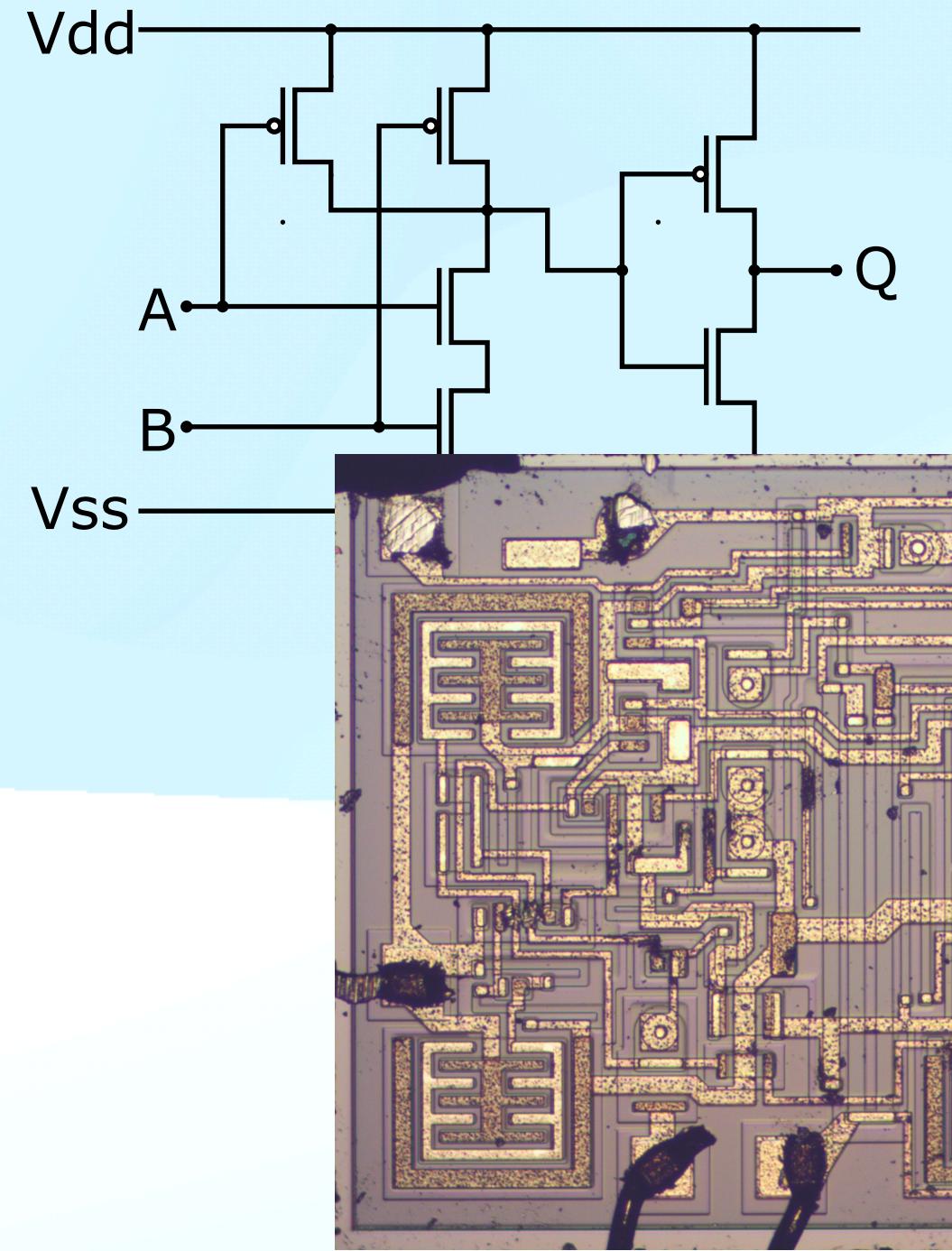
However reinforcement learning presents several challenges from a deep learning perspective. Firstly, most successful deep learning applications to date have required large amounts of hand-labelled training data. RL algorithms, on the other hand, must be able to learn from a scalar reward signal that is frequently sparse, noisy and delayed. The delay between actions and resulting rewards, which can be thousands of timesteps long, seems particularly daunting when compared to the direct association between inputs and outputs found in supervised learning. Another issue is that most deep principles to be independent, while in reinforcement learning one is correlated states. Furthermore, in RL the data distribution behaviours, which can be problematic for deep learning distribution.

ional neural network can overcome these challenges to learn video data in complex RL environments. The network is [26] algorithm, with stochastic gradient descent to update of correlated data and non-stationary distributions, we use

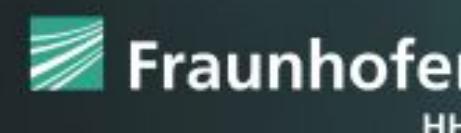
1



My Background



**Happy New Year from
the AI department at
Fraunhofer HHI**



Playing Atari with Deep Reinforcement Learning

Volodymyr Mnih · Koray Kavukcuoglu · David Silver · Alex Graves · Ioannis Antonoglou

DeepMind · Martin Riedmiller

es
ctin.riedmiller} @ deepmind.com

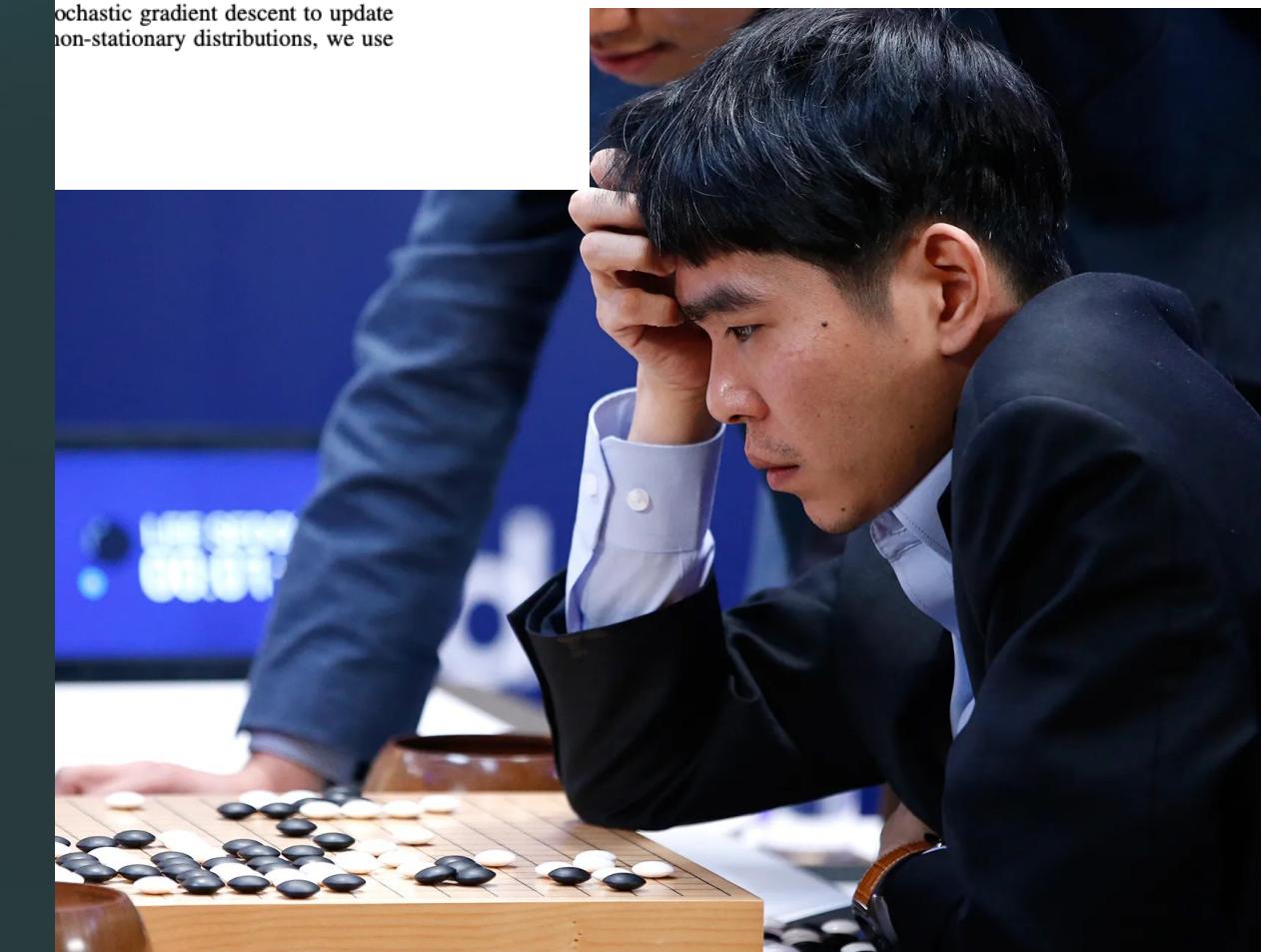
by learn control policies di-
n a variant of Q-learning,
function estimating future
es from the Arcade Learn-
or learning algorithm. We
of the games and surpasses

ory inputs like vision and speech is
(RL). Most successful RL applica-
features combined with linear value
such systems heavily relies on the

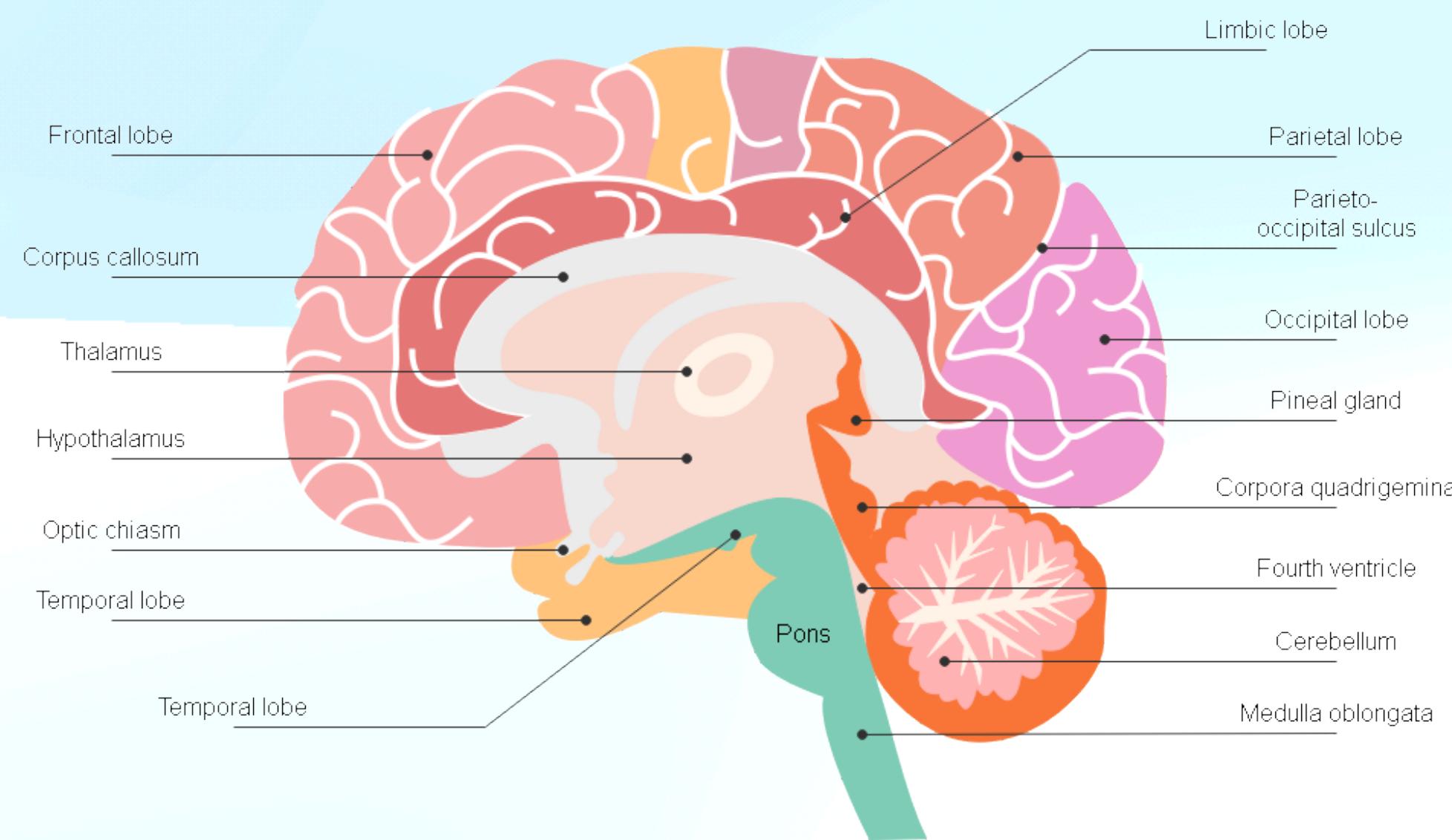
ct high-level features from raw sen-
[16] and speech recognition [6, 7],
including convolutional networks,
rent neural networks, and have ex-
atural to ask whether similar tech-

from a deep learning perspective.
e required large amounts of hand-
e able to learn from a scalar reward
ween actions and resulting rewards,
unting when compared to the direct
ing. Another issue is that most deep
while in reinforcement learning one
thermore, in RL the data distribu-
n be problematic for deep learning

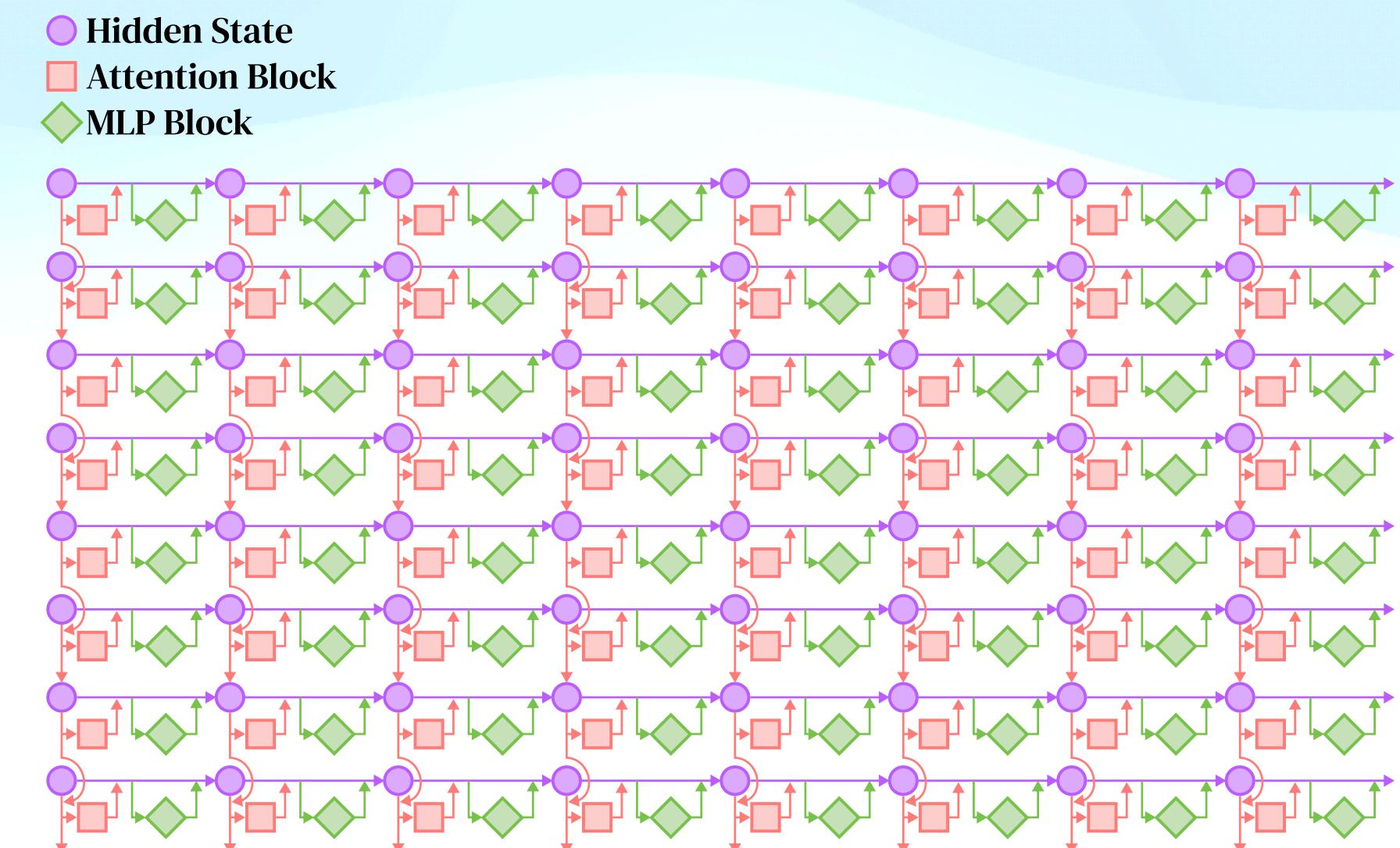
overcome these challenges to learn
RL environments. The network is
ochastic gradient descent to update
non-stationary distributions, we use



A Neuroscientist's Lens



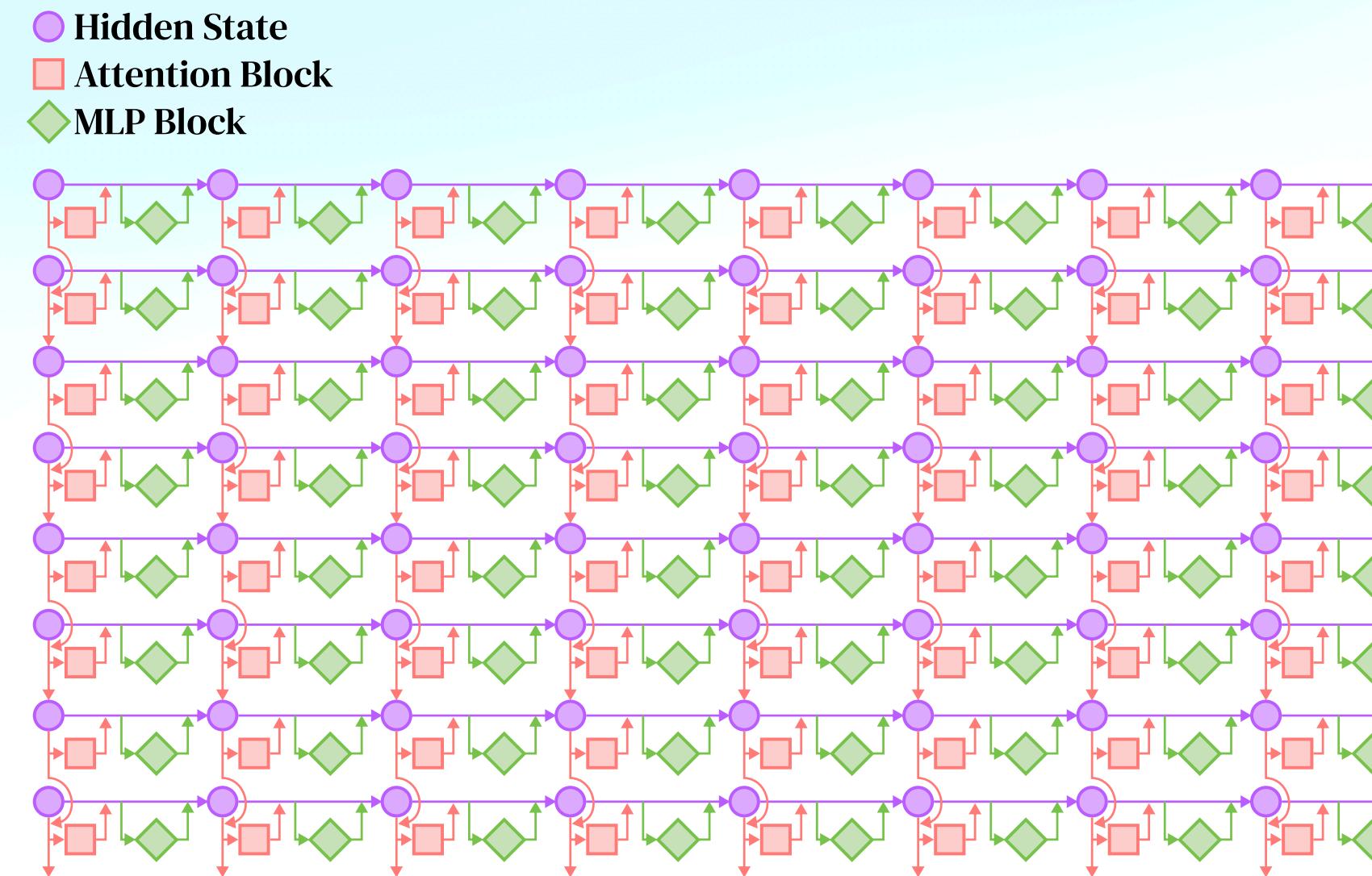
<https://www.neetgrade.com/2024/07/human-brain-structure-and-functions.html>



Inspired by <https://rome.baulab.info/>

Our Goal Today: We'll map out the functional anatomy of LLaMA 3.2 1B!

1. How to trace the flow of information?



5. Can we use our insights for model monitoring?

4. Where does it store its knowledge?

2. Where does the model encode the task that it wants to perform?

3. Can we control how the model processes information?

Our Goal Today: We'll map out the functional anatomy of LLaMA 3.2 1B!

1. How to trace the flow of information?

- Hidden State
- Attention Block
- ◆ MLP Block



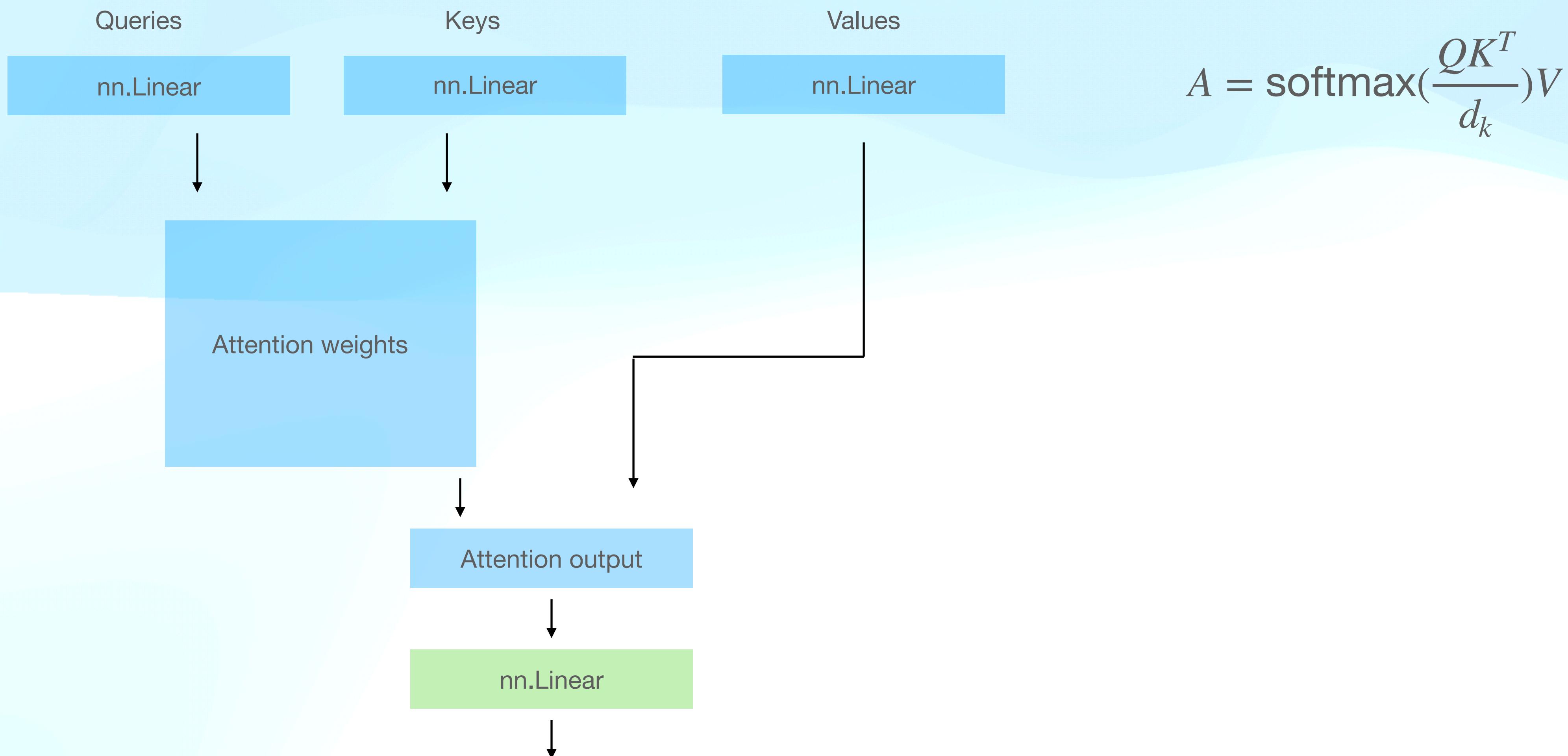
2. Where does the model encode the task that it wants to perform?

5. Can we use our insights for hallucination detection and source attribution?

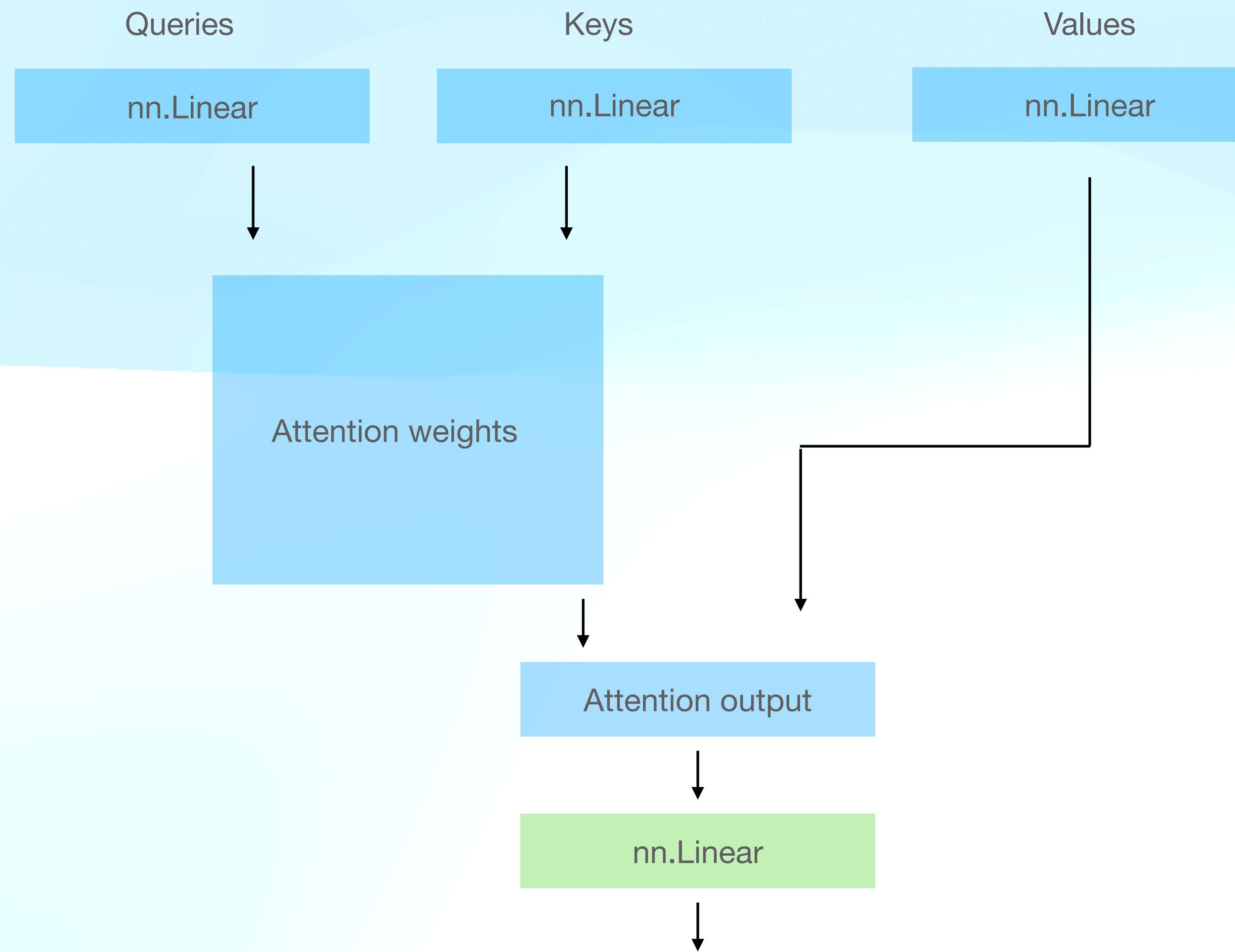
4. Where does it store its knowledge?

3. Can we control how the model processes information?

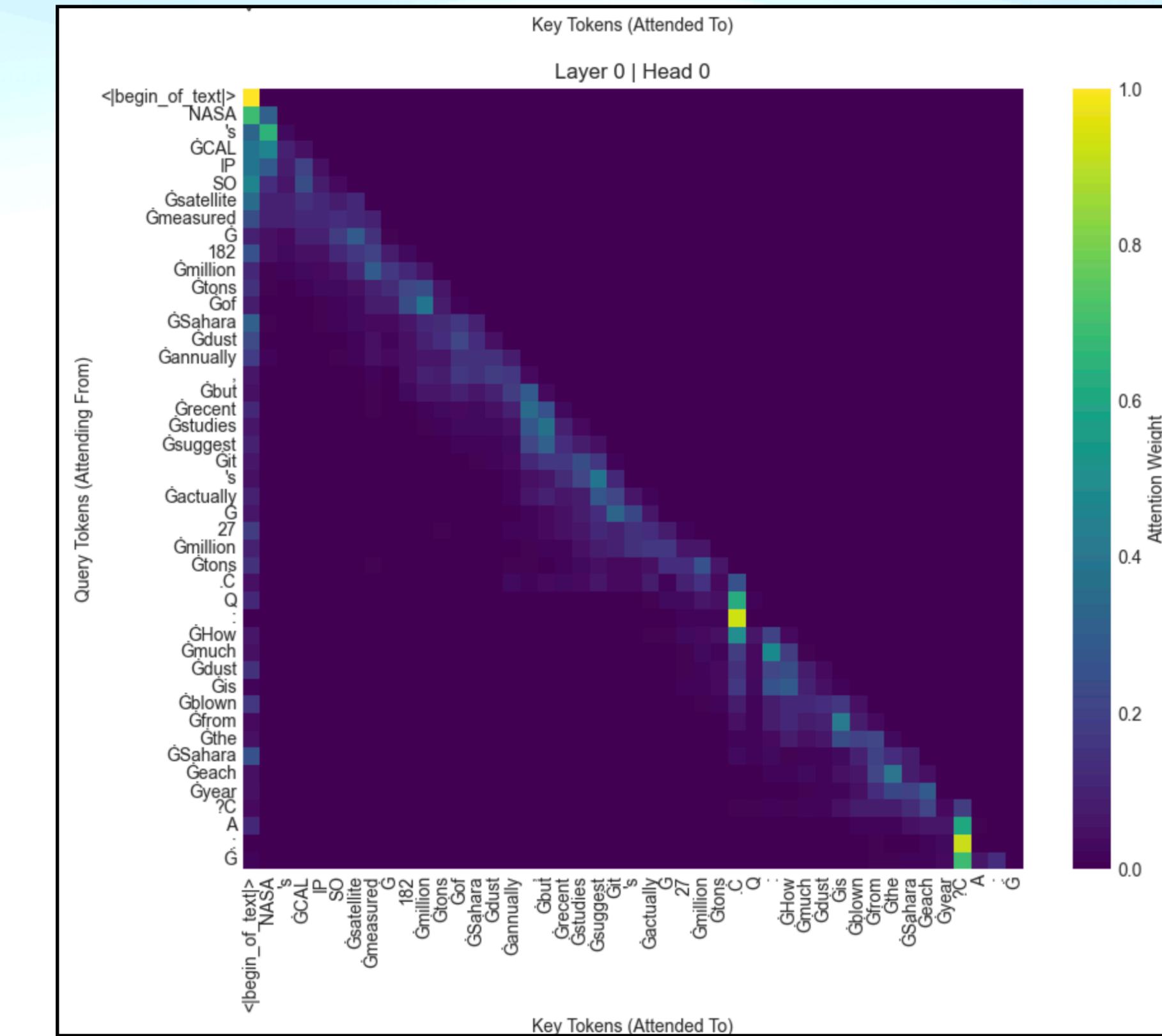
Quick Recap: The Attention Mechanism



Quick Recap: The Attention Mechanism



$$A = \text{softmax}\left(\frac{QK^T}{d_k}\right)V$$



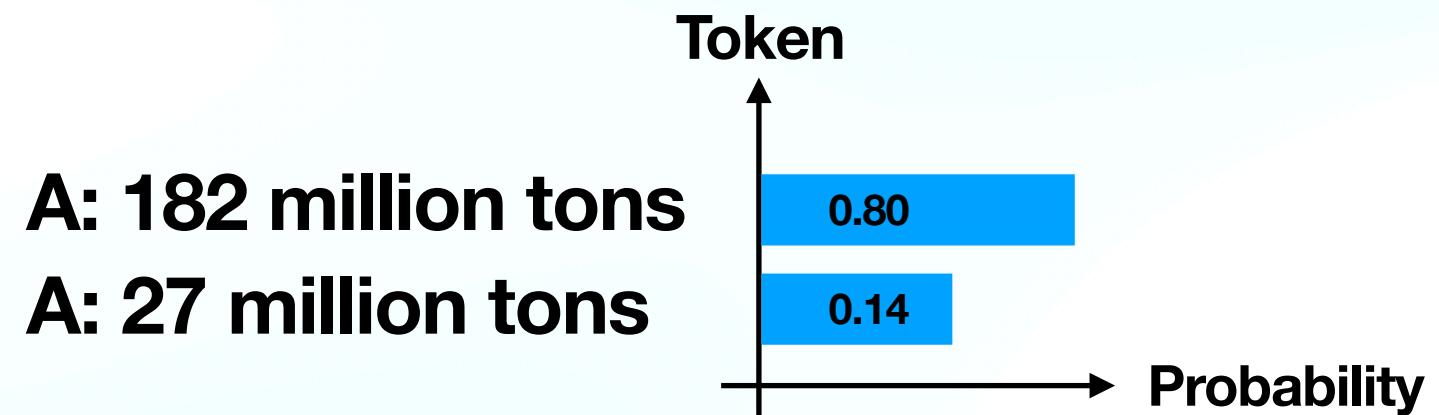
Building Our Microscope

**How do we actually look inside a neural network?
How do we know which parts are doing what?**

Building Our Microscope

NASA's CALIPSO satellite measured 182 million tons of Sahara dust annually, but recent studies suggest it's actually 27 million tons.

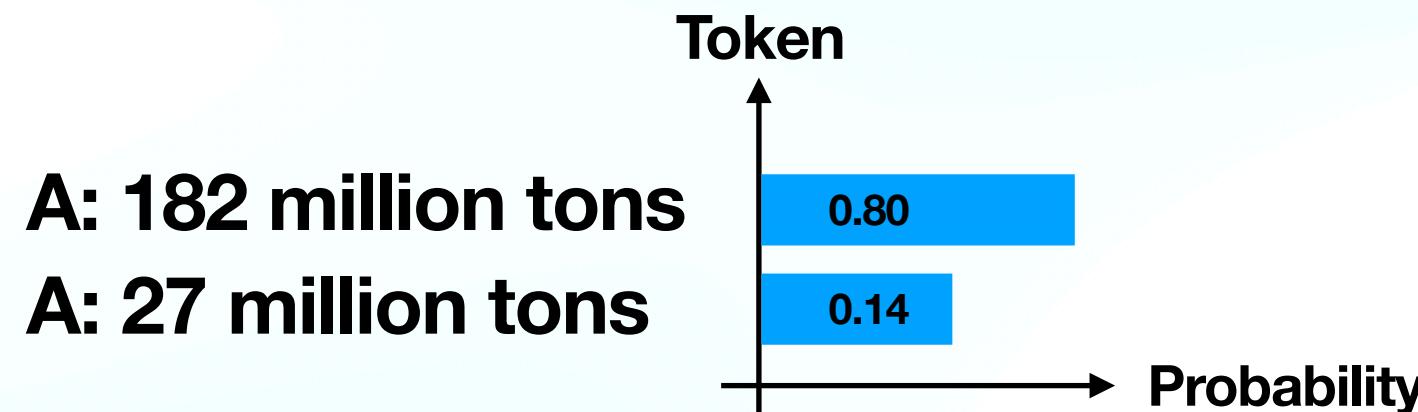
Q: How much dust is blown out of the Sahara each year?



Building Our Microscope

NASA's CALIPSO satellite measured 182 million tons of Sahara dust annually, but recent studies suggest it's actually 27 million tons.

Q: How much dust is blown out of the Sahara each year?



**Transformers compute all tokens simultaneously!
They are MASSIVELY PARALLEL machines by design.**

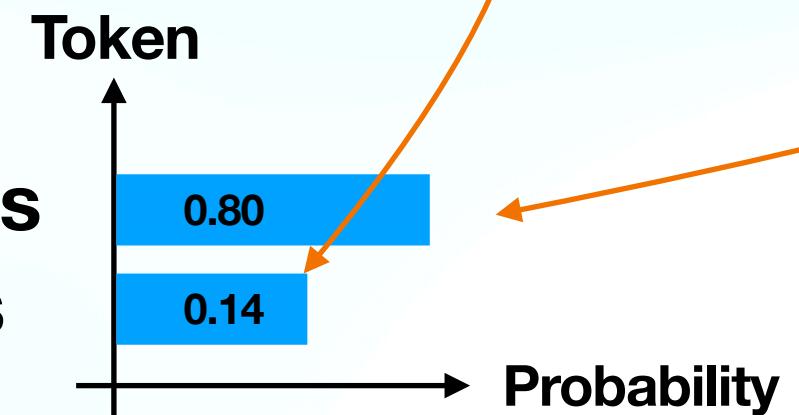
Building Our Microscope

NASA's CALIPSO satellite measured 182 million tons of Sahara dust annually, but recent studies suggest it's actually 27 million tons.

Q: How much dust is blown out of the Sahara each year?



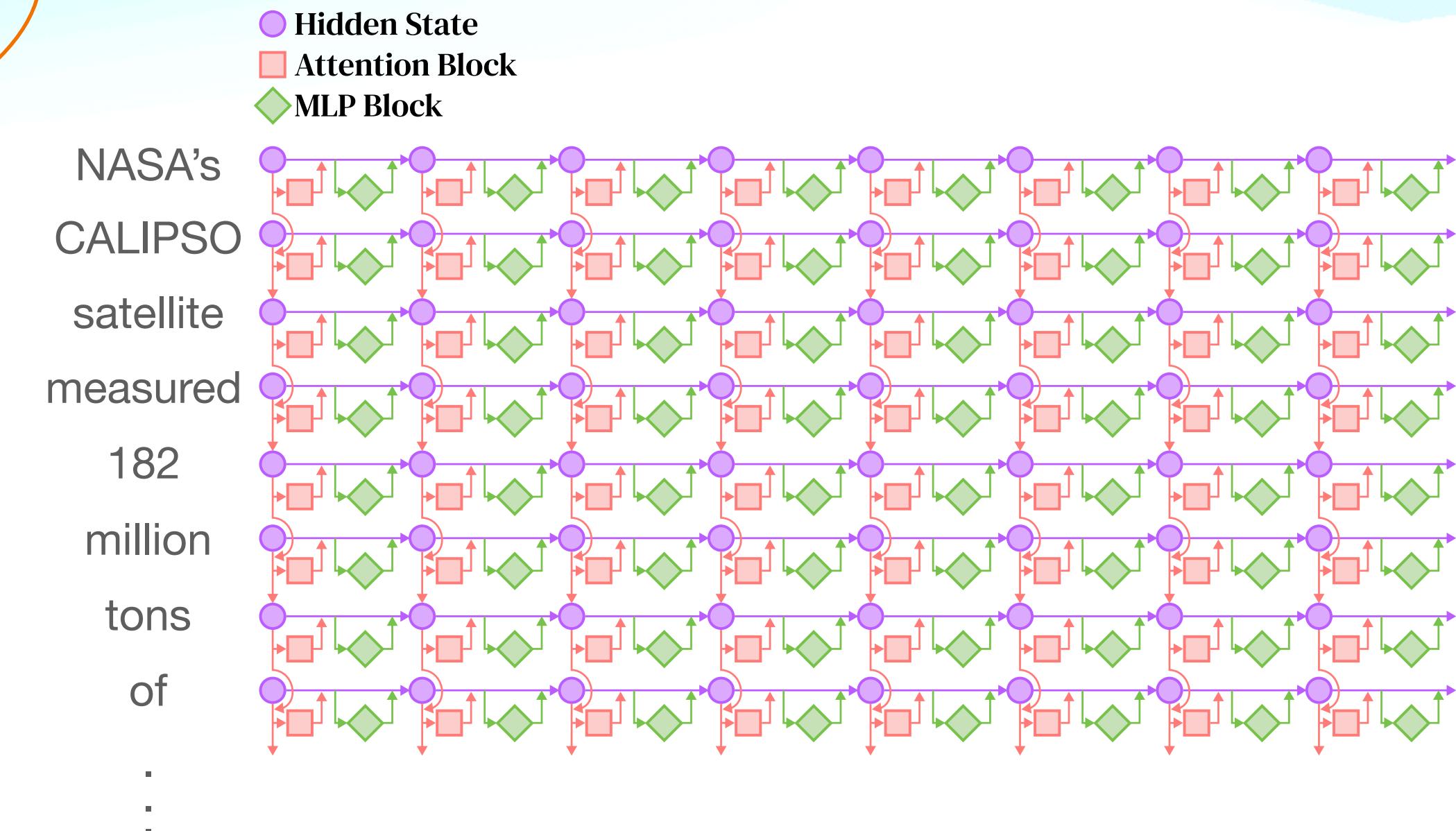
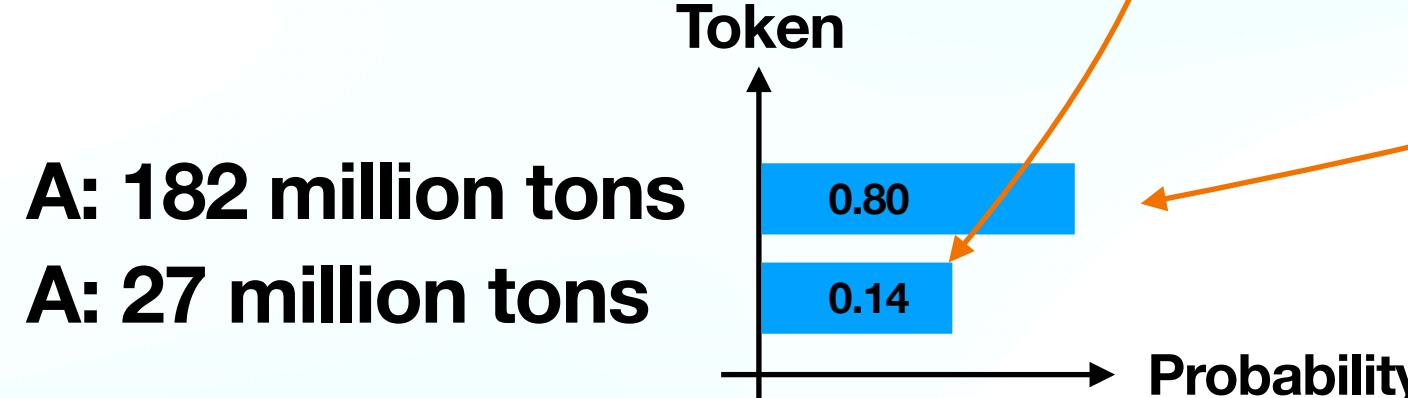
A: 182 million tons
A: 27 million tons



Building Our Microscope

NASA's CALIPSO satellite measured 182 million tons of Sahara dust annually, but recent studies suggest it's actually 27 million tons.

Q: How much dust is blown out of the Sahara each year?



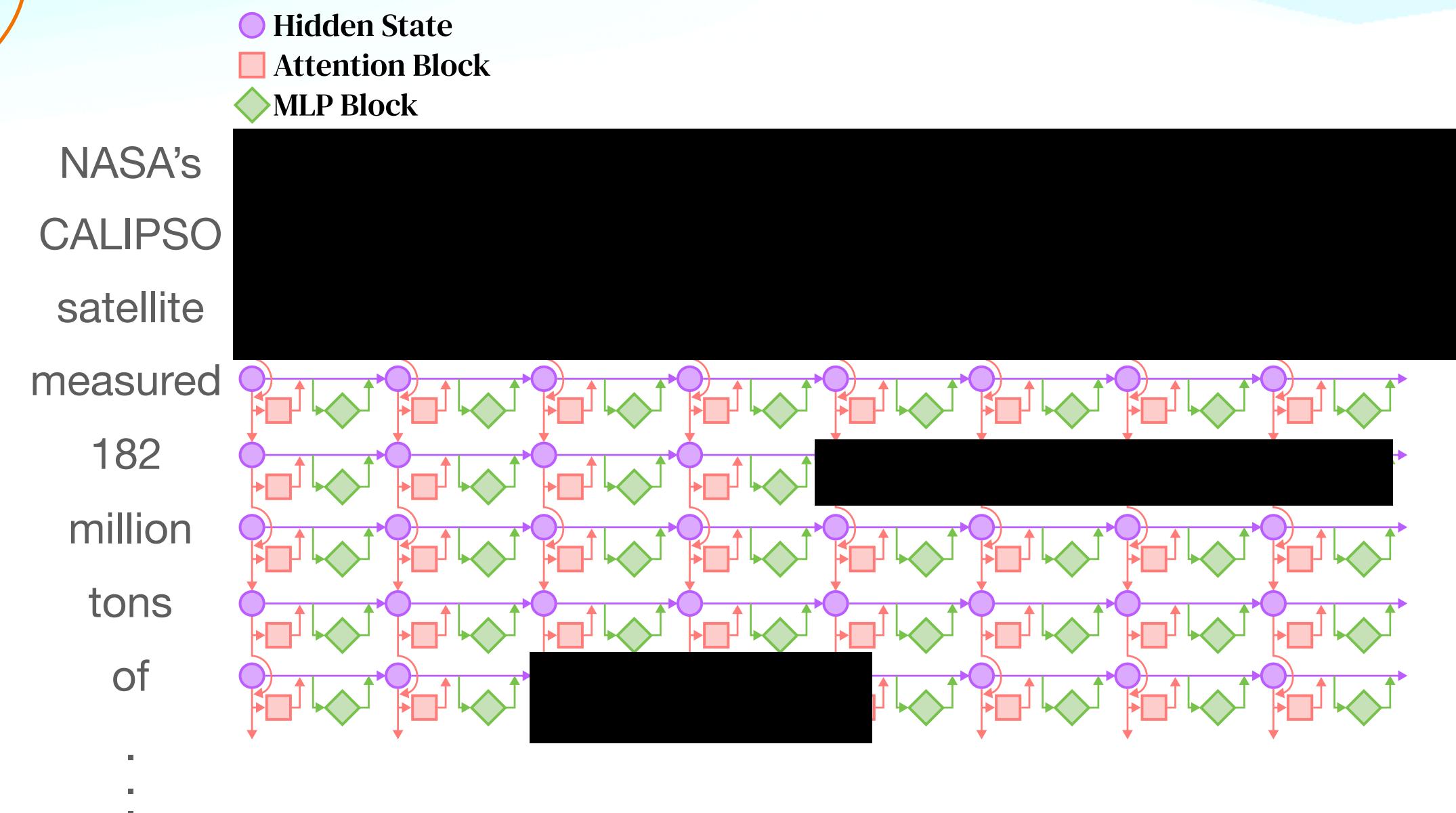
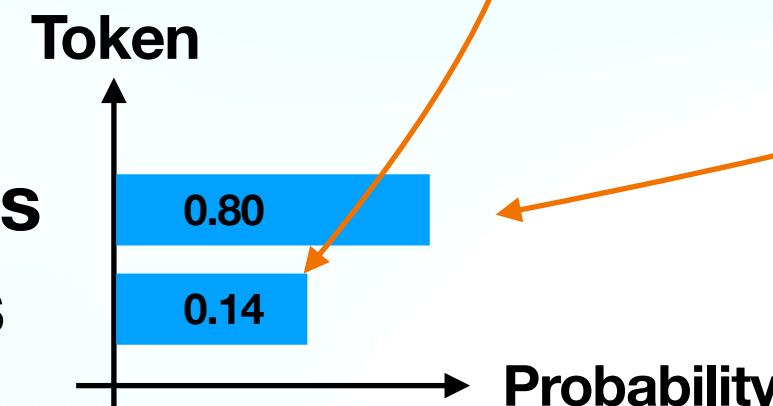
Building Our Microscope

measured 182 million tons of
Sahara dust annually, but recent studies suggest it's
actually 27 million tons.

Q: How much dust is blown out of the Sahara each year?



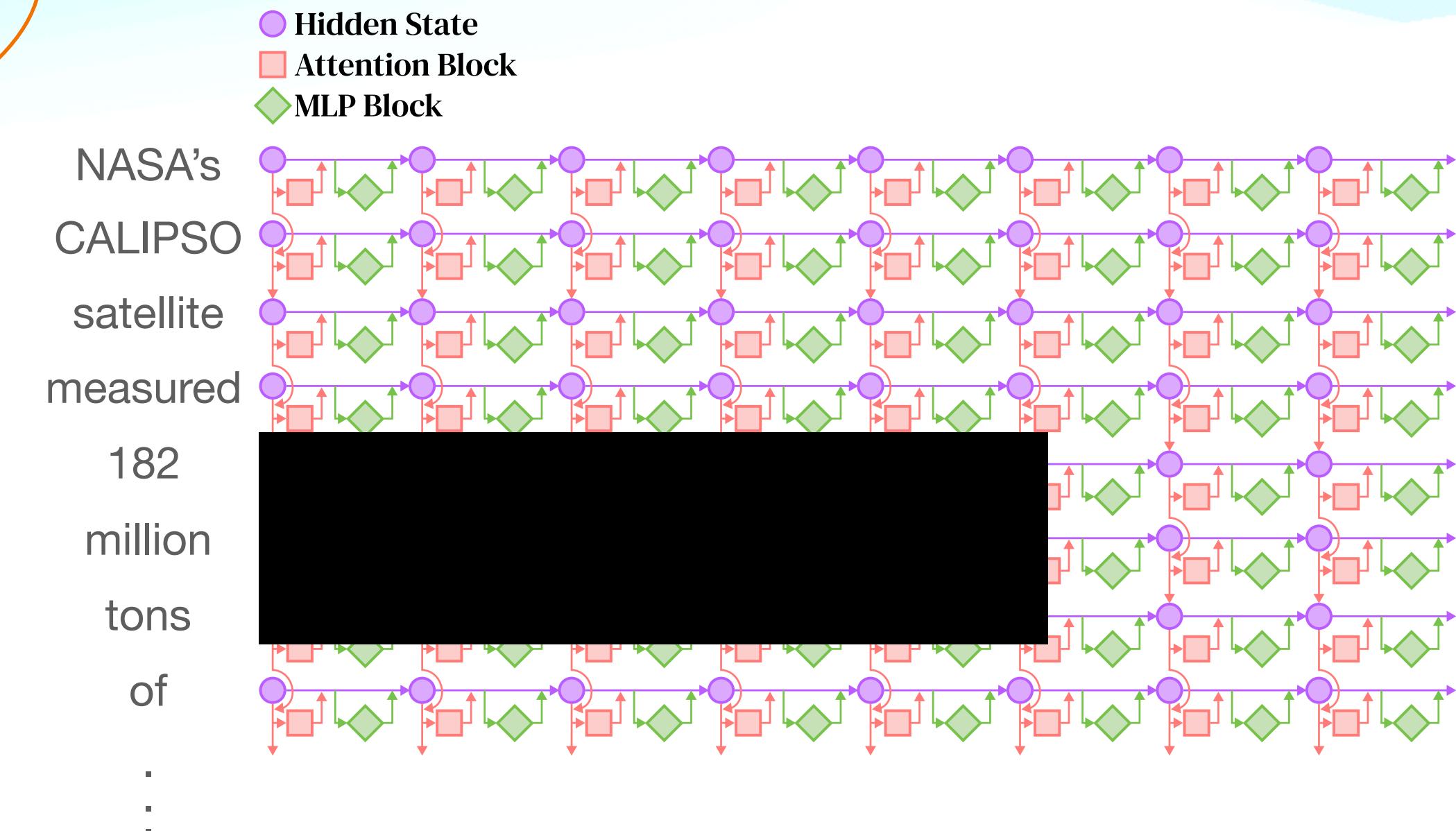
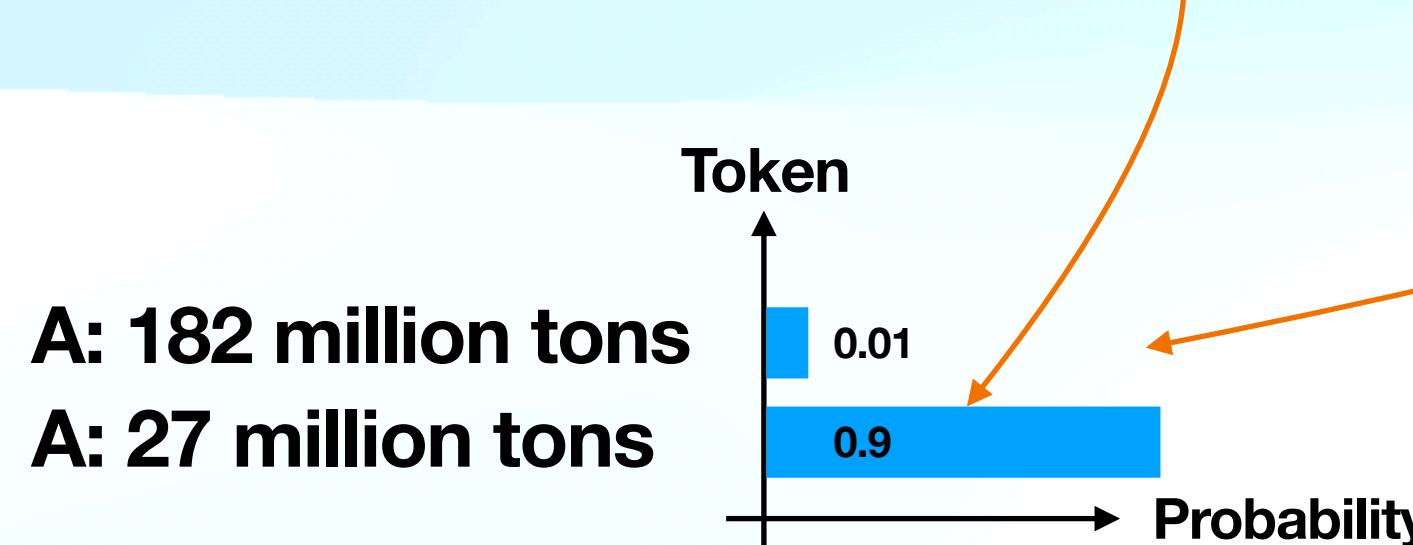
A: 182 million tons
A: 27 million tons



Building Our Microscope

NASA's CALIPSO satellite [REDACTED] Sahara dust annually, but recent studies suggest it's actually 27 million tons.

Q: How much dust is blown out of the Sahara each year?



Perturbation

Let $x \in \mathcal{X} \subseteq \mathbb{R}^d$ be an input representation, and let $f: \mathcal{X} \rightarrow \mathbb{R}^k$ denote the model logit function.

We denote by $f_j(x)$ the logit corresponding to output index j , and by i the coordinate of x subjected to ablation.

Hence, we define the Relevance as

$$\mathcal{R}(x_i) = f_j(x) - f_j(x | x_i \leftarrow b_i)$$

Perturbation

Let $x \in \mathcal{X} \subseteq \mathbb{R}^d$ be an input representation, and let $f: \mathcal{X} \rightarrow \mathbb{R}^k$ denote the model logit function.

We denote by $f_j(x)$ the logit corresponding to output index j , and by i the coordinate of x subjected to ablation.

Hence, we define the Relevance as

$$\mathcal{R}(x_i) = f_j(x) - f_j(x | x_i \leftarrow b_i)$$

-> **Open(Jupyter_Notebook)**

Perturbation: The curse of dimensionality

Let's say we have:

- **500 attention heads**
- **1000 input tokens**

Then, we need $500 * 1000 = 500.000$ forward passes!



First-Order Taylor Approximation

We interpret feature ablation as a small perturbation of the input representation. Accordingly, we take a first-order Taylor expansion of the logit f_j around the input x itself.

$$f_j(z) \approx f_j(x) + \sum_{k=1}^d \frac{\partial f_j}{\partial x_k} \Bigg|_x (z_k - x_k)$$

where z denotes a nearby input. Setting $z = x^{(i \leftarrow b_i)}$, which differs from x only in coordinate i , yields

$$f_j(x^{(i \leftarrow b_i)}) \approx f_j(x) - \frac{\partial f_j}{\partial x_i} \Bigg|_x (x_i - b_i)$$

Rearranging, the change in the logit induced by ablating feature i is approximated by

$$\mathcal{R}(x_i) = f_j(x) - f_j(x^{(i \leftarrow b_i)}) \approx \frac{\partial f_j}{\partial x_i} \Bigg|_x (x_i - b_i)$$

First-Order Taylor Approximation

We interpret feature ablation as a small perturbation of the input representation. Accordingly, we take a first-order Taylor expansion of the logit f_j around the input x itself.

$$f_j(z) \approx f_j(x) + \sum_{k=1}^d \frac{\partial f_j}{\partial x_k} \Bigg|_x (z_k - x_k)$$

where z denotes a nearby input. Setting $z = x^{(i \leftarrow b_i)}$, which differs from x only in coordinate i , yields

$$f_j(x^{(i \leftarrow b_i)}) \approx f_j(x) - \frac{\partial f_j}{\partial x_i} \Bigg|_x (x_i - b_i)$$

Rearranging, the change in the logit induced by ablating feature i is approximated by

$$\mathcal{R}(x_i) = f_j(x) - f_j(x^{(i \leftarrow b_i)}) \approx \frac{\partial f_j}{\partial x_i} \Bigg|_x (x_i - b_i)$$

-> Open(Jupyter_Notebook)

First-Order Taylor Approximation: Gradient Shattering

However, the gradient does not faithfully reflect the model's true signal and is instead contaminated by a substantial noise component.

$$\left. \frac{\partial f_j}{\partial x_i} \right|_x = \left. \frac{\partial f_j^{\text{signal}}}{\partial x_i} \right|_x + \left. \frac{\partial f_j^{\text{noise}}}{\partial x_i} \right|_x$$

First-Order Taylor Approximation: Gradient Shattering

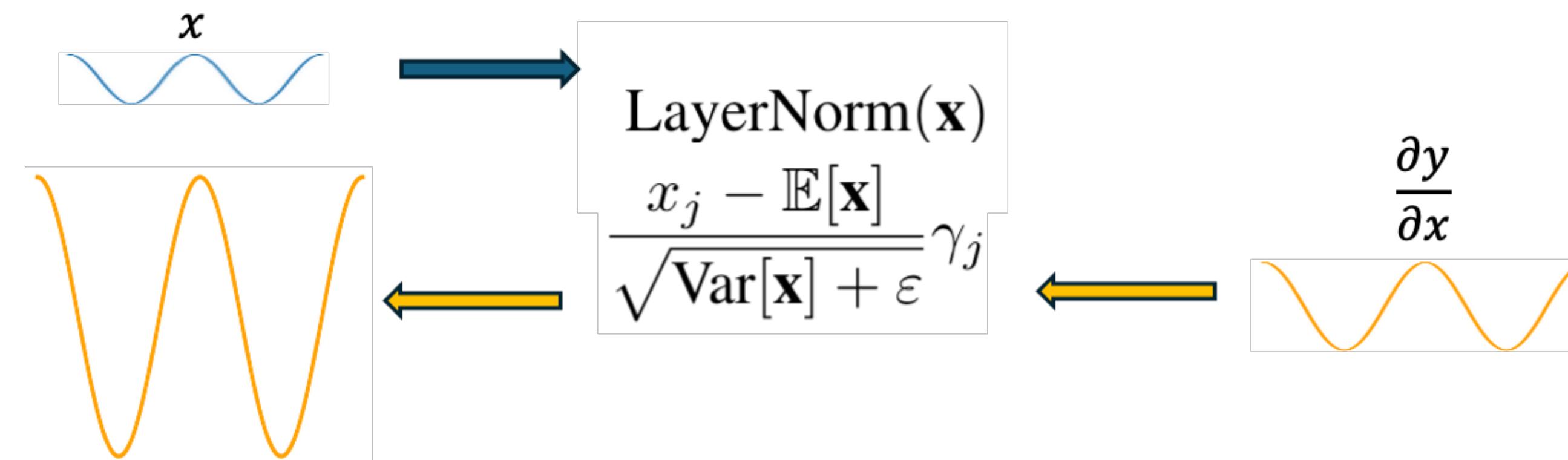
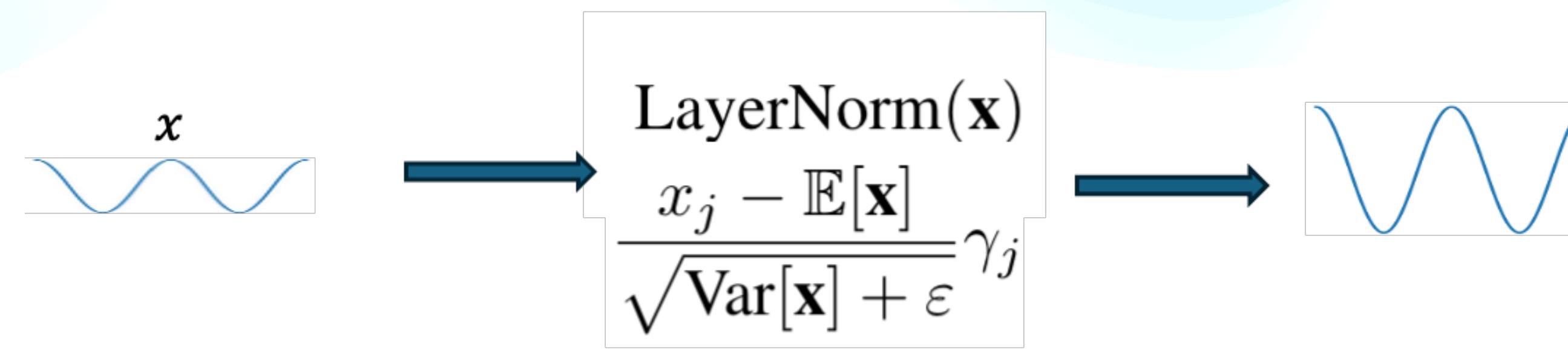
However, the gradient does not faithfully reflect the model's true signal and is instead contaminated by a substantial noise component.

$$\left. \frac{\partial f_j}{\partial x_i} \right|_x = \left. \frac{\partial f_j^{\text{signal}}}{\partial x_i} \right|_x + \left. \frac{\partial f_j^{\text{noise}}}{\partial x_i} \right|_x$$

-> Open(Jupyter_Notebook)

LayerNorm

LayerNorm scales the gradient inversely to the magnitude of the input \mathbf{x}



LayerNorm

LayerNorm

013v1 [cs.LG] 16 Nov 2019



Understanding and Improving Layer Normalization

Jingjing Xu¹, Xu Sun^{1,2*}, Zhiyuan Zhang¹, Guangxiang Zhao², Junyang Lin¹

¹ MOE Key Lab of Computational Linguistics, School of EECS, Peking University

² Center for Data Science, Peking University

{jingjingxu, xusun, zzy1210, zhaoguangxiang, linjunyang}@pku.edu.cn

Abstract

Layer normalization (LayerNorm) is a technique to normalize the distributions of intermediate layers. It enables smoother gradients, faster training, and better generalization accuracy. However, it is still unclear where the effectiveness stems from. In this paper, our main contribution is to take a step further in understanding LayerNorm. Many of previous studies believe that the success of LayerNorm comes from forward normalization. Unlike them, we find that the derivatives of the mean and variance are more important than forward normalization by re-centering and re-scaling backward gradients. Furthermore, we find that the parameters of LayerNorm, including the bias and gain, increase the risk of over-fitting and do not work in most cases. Experiments show that a simple version of LayerNorm (LayerNorm-simple) without the bias and gain outperforms LayerNorm on four datasets. It obtains the state-of-the-art performance on En-Vi machine translation. To address the over-fitting problem, we propose a new normalization method, Adaptive Normalization (AdaNorm), by replacing the bias and gain with a new transformation function. Experiments show that AdaNorm demonstrates better results than LayerNorm on seven out of eight datasets.

of the input x

Layer-wise Relevance Propagation

The raw gradient is noisy due to:

1. **LayerNorm**: stabilises training but distorts the gradient by introducing **higher-order effects** absent in a first-order Taylor approximation.
2. **Query–Key**: gradients flow through both query and key, effectively **doubling the gradient** and causing explosion.
3. **Softmax–Value**: like query–key multiplication
4. **SiLU**: the nonlinearity further introduces **higher-order effects**.

LRP fixes this by propagating modified gradients:

$$\frac{\partial f_j}{\partial x} = \frac{\partial f_j}{\partial v} \cdot \frac{\partial v}{\partial u} \cdot \frac{\partial u}{\partial x} \rightarrow \left[\frac{\partial f_j}{\partial x} \right]_{LRP} = \left[\frac{\partial f_j}{\partial v} \right]_{LRP} \cdot \left[\frac{\partial v}{\partial u} \right]_{LRP} \cdot \left[\frac{\partial u}{\partial x} \right]_{LRP}.$$

Layer-wise Relevance Propagation

The raw gradient is not always a good explanation.
1. LayerNorm: stable but noisy
2. Query-Key: gradients are sparse
3. Softmax-Value: lossless but slow
4. SiLU: the nonlinearity is smooth

[CS.CL] 10 Jun 2024

AttnLRP: Attention-Aware Layer-Wise Relevance Propagation for Transformers

Reduan Achtabat¹ Sayed Mohammad Vakilzadeh Hatefi¹ Maximilian Dreyer¹
Aakriti Jain¹ Thomas Wiegand^{1,2,3} Sebastian Lapuschkin^{1,†} Wojciech Samek^{1,2,3,†}

¹ Fraunhofer Heinrich-Hertz-Institute, 10587 Berlin, Germany

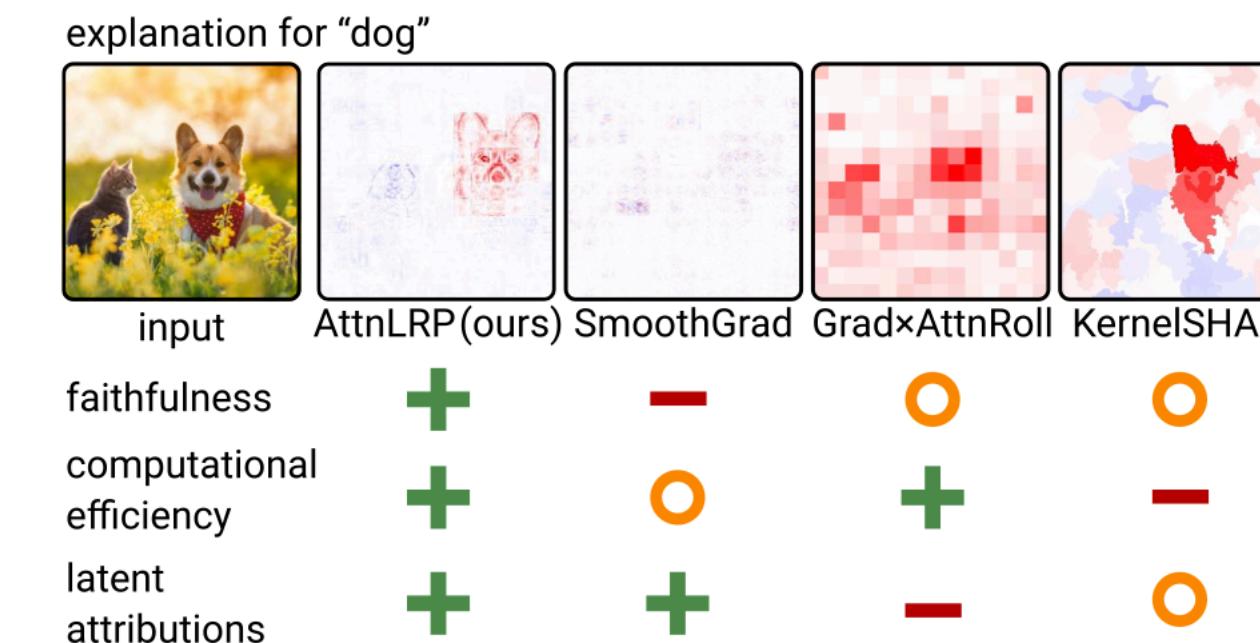
² Technische Universität Berlin, 10587 Berlin, Germany

³ BIFOLD – Berlin Institute for the Foundations of Learning and Data, 10587 Berlin, Germany

† corresponding authors: {wojciech.samek, sebastian.lapuschkin}@hhi.fraunhofer.de

Abstract

Large Language Models are prone to biased predictions and hallucinations, underlining the paramount importance of understanding their model-internal reasoning process. However, achieving faithful attributions for the entirety of a black-box transformer model and maintaining computational efficiency is an unsolved challenge. By extending the Layer-wise Relevance Propagation attribution method to handle attention, we propose AttnLRP, a novel attention-aware layer-wise relevance propagation method for transformers.



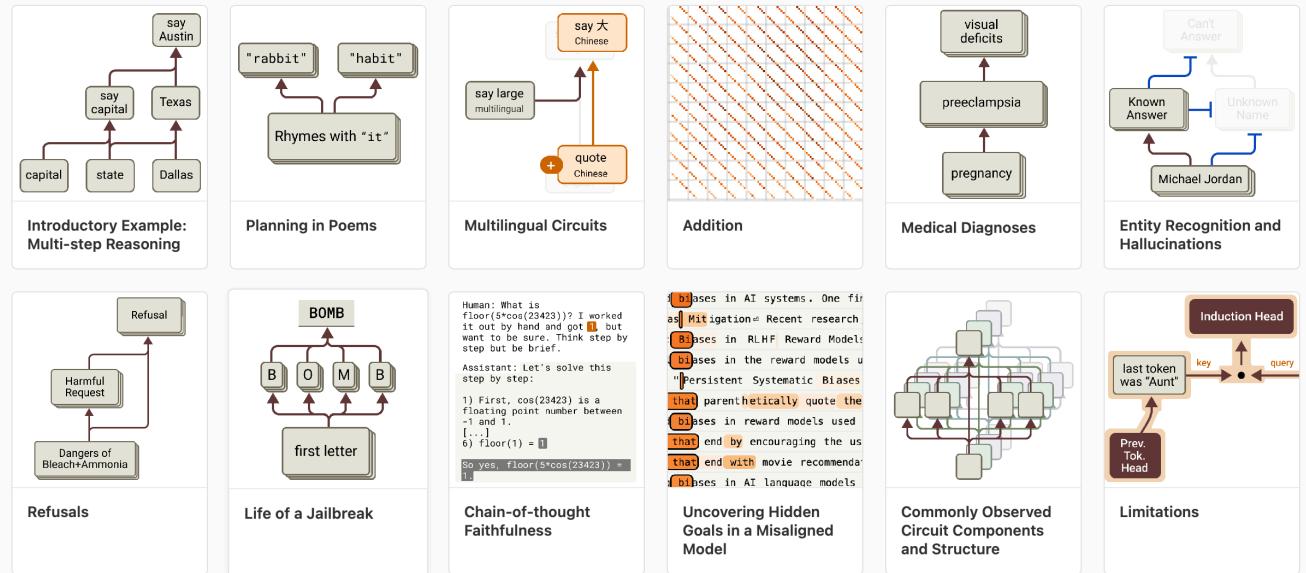
RP

State-of-the-Art

AI Transformer Circuits Thread

On the Biology of a Large Language Model

We investigate the internal mechanisms used by Claude 3.5 Haiku — Anthropic's lightweight production model — in a variety of contexts, using our circuit tracing methodology.



← Back to OpenAI Alignment Blog

Debugging misaligned completions with sparse-autoencoder latent attribution

Dec 1, 2025 · Tom Dupré la Tour and Dan Mossing, in collaboration with the Interpretability team

We use interpretability tools to study mechanisms underlying misalignment in language models. In previous work (Wang et al., 2025), we used a model differencing approach to study the mechanism of emergent misalignment (Betley et al., 2025) using sparse-autoencoders (SAEs) (Cunningham et al., 2023, Bricken et al., 2023, Gao et al., 2024).^[1]

Specifically, we used a two-step model-differencing approach to compare two models, before and after a problematic fine-tuning. The first step selects a subset of SAE latents that most differ in activations between the two models. The second step samples many completions from the model while activation steering (Panickssery et al., 2023), and grades them with an LLM judge to systematically measure the causal link between each latent and the unexpected behavior. Because the second step is too computationally expensive to run on every latent, we only run it on the subset of latents defined in the first step.

This approach has some limitations, in particular if our goal is to find causally relevant latents for a given behavior. Indeed, the latents with the largest activation differences need not cause the behavior of interest, and thus the two-step approach may miss the most causally relevant latents. Additionally, this model-differencing approach is limited to the particular auditing setting where there are two closely related models to compare, with one model exhibiting the undesired behavior and one not.

<https://transformer-circuits.pub/2025/attribution-graphs/biology.html>

<https://alignment.openai.com/sae-latent-attribution/?s=09>

Translucce

Our Work Our Mission Company

Language Model Circuits Are Sparse in the Neuron Basis

Aryaman Arora*, Zhengxuan Wu*, Jacob Steinhardt, Sarah Schwettmann
* Equal contribution. Correspondence to: aryaman@translucce.org, zen@translucce.org.

Translucce | Published: November 20, 2025

Many interpretability methods rely on learned feature bases—such as sparse autoencoders or cross-layer transcoders—based on the belief that neurons do not cleanly decompose model computation. We revisit this assumption and show that, with a better choice of neuron basis (MLP activations) and a stronger attribution method (RelP), raw neurons can produce circuits that are just as sparse and faithful as those built from learned features. We reproduce three prior case studies originally demonstrated with learned features (multi-hop reasoning, addition, and multilingual antonyms) using only neuron circuits in Llama 3.1-8B-Instruct.

Introduction

Modern AI systems can solve complex tasks, but they often do so in ways we can't directly observe. They may rely on shortcuts, internal assumptions about the user, or multi-step reasoning that never appears in their text output. If we can see these internal computations, we can check whether the model is reasoning in the way we expect, catch failures that don't show up in the final answer, and understand how its behavior evolves during training [1, 2]. This is the goal of *circuit analysis*: tracing a model's behaviors back to specific interactions between its internal components [3, 4, 5, 6, 7, 8].

<https://translucce.org/neuron-circuits>

RelP: Faithful and Efficient Circuit Discovery in Language Models via Relevance Patching

Farnoush Rezaei Jafari^{1,2*}

Oliver Eberle^{1,2}

Ashkan Khakzar

Neel Nanda

¹Machine Learning Group, Technische Universität Berlin
²BIFOLD – Berlin Institute for the Foundations of Learning and Data

Abstract

Activation patching is a standard method in mechanistic interpretability for localizing the components of a model responsible for specific behaviors, but it is computationally expensive to apply at scale. Attribution patching offers a faster, gradient-based approximation, yet suffers from noise and reduced reliability in deep, highly non-linear networks. In this work, we introduce *Relevance Patching* (RelP), which replaces the local gradients in attribution patching with propagation coefficients derived from Layer-wise Relevance Propagation (LRP). LRP propagates the network's output backward through the layers, redistributing relevance to lower-level components according to local propagation rules that ensure properties such as relevance conservation or improved signal-to-noise ratio. Like attribution patching, RelP requires only two forward passes and one backward pass, main-

<https://arxiv.org/pdf/2508.21258.pdf>

Layer-wise Relevance Propagation

The raw gradient is noisy due to:

1. **LayerNorm**: stabilises training but distorts the gradient by introducing **higher-order effects** absent in a first-order Taylor approximation.
2. **Query–Key**: gradients flow through both query and key, effectively **doubling the gradient** and causing explosion.
3. **Softmax–Value**: like query–key multiplication
4. **SiLU**: the nonlinearity further introduces **higher-order effects**.

LRP fixes this by propagating modified gradients:

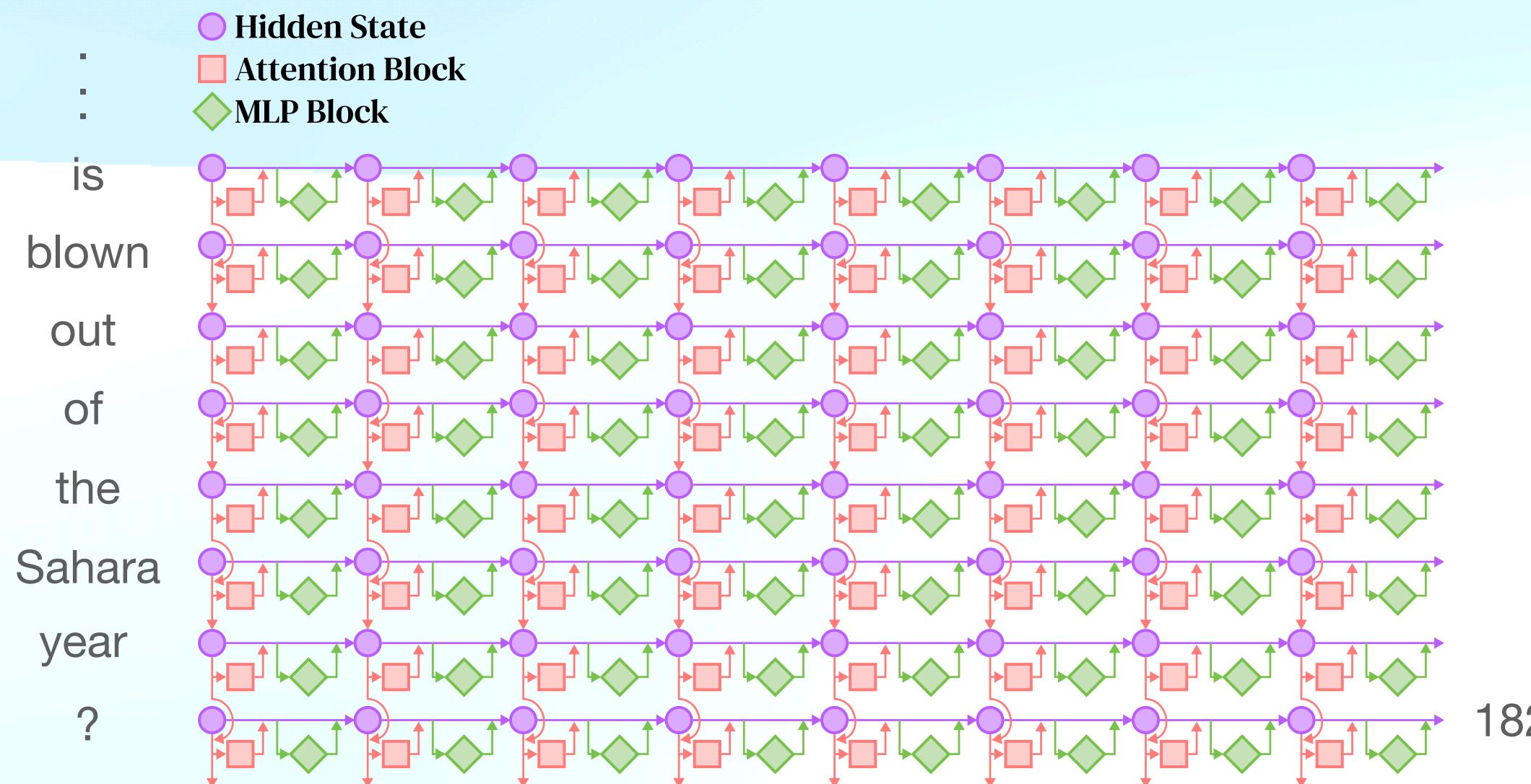
$$\frac{\partial f_j}{\partial x} = \frac{\partial f_j}{\partial v} \cdot \frac{\partial v}{\partial u} \cdot \frac{\partial u}{\partial x} \rightarrow \left[\frac{\partial f_j}{\partial x} \right]_{LRP} = \left[\frac{\partial f_j}{\partial v} \right]_{LRP} \cdot \left[\frac{\partial v}{\partial u} \right]_{LRP} \cdot \left[\frac{\partial u}{\partial x} \right]_{LRP}.$$

-> Open(Jupyter_Notebook)

Patching Attention Heads

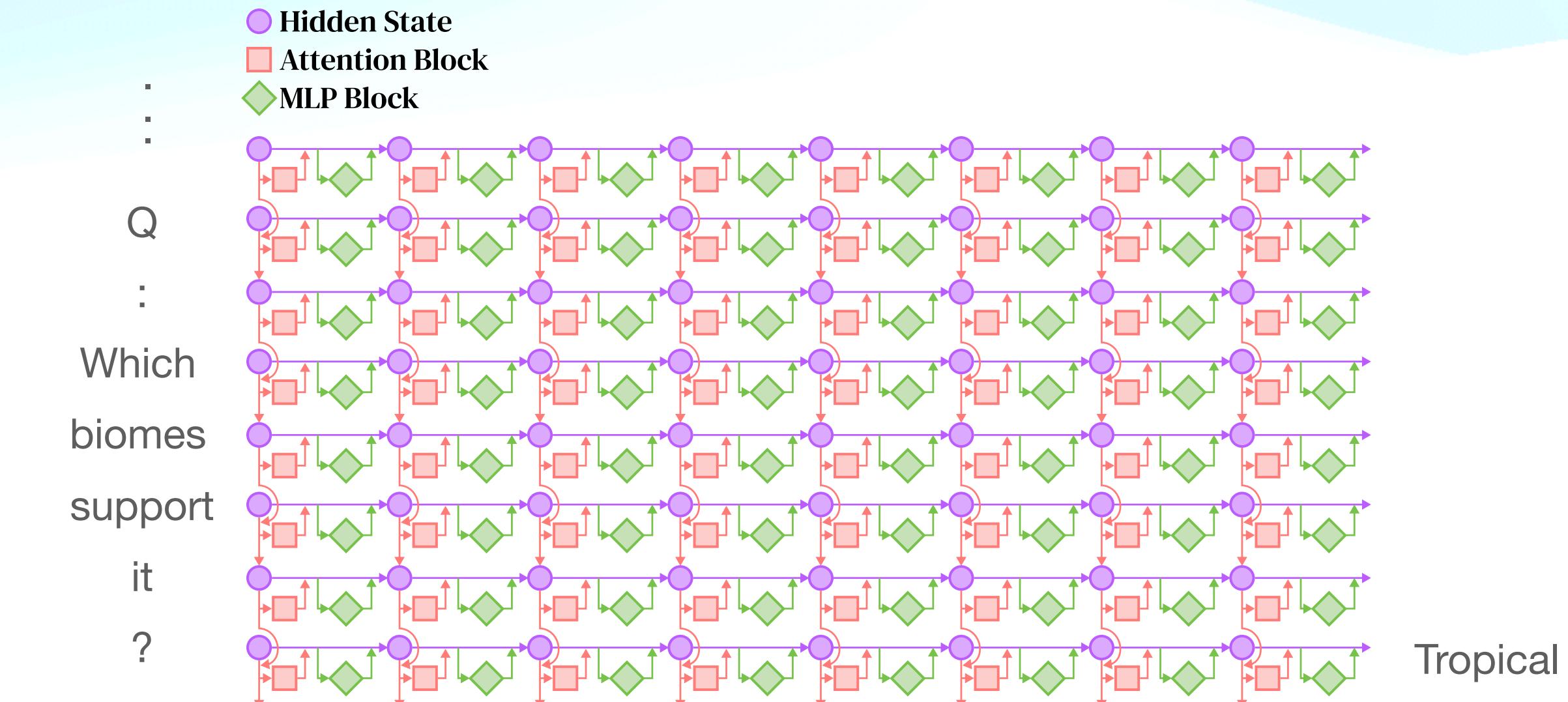
NASA's CALIPSO satellite measured 182 million tons of Sahara dust annually, but recent studies suggest it's actually 27 million tons.

Q: How much dust is blown out of the Sahara each year?



The Atlantic Ocean is the second-largest ocean on Earth. It holds about 310 million cubic kilometers of water. It spans a wide range of marine biomes, from warm tropical coral reef systems near the equator to cold, nutrient-rich polar waters in the north and south. These biomes support diverse life, including plankton, fish, whales, and deep-sea organisms, and play a major role in regulating Earth's climate through ocean currents like the Gulf Stream.

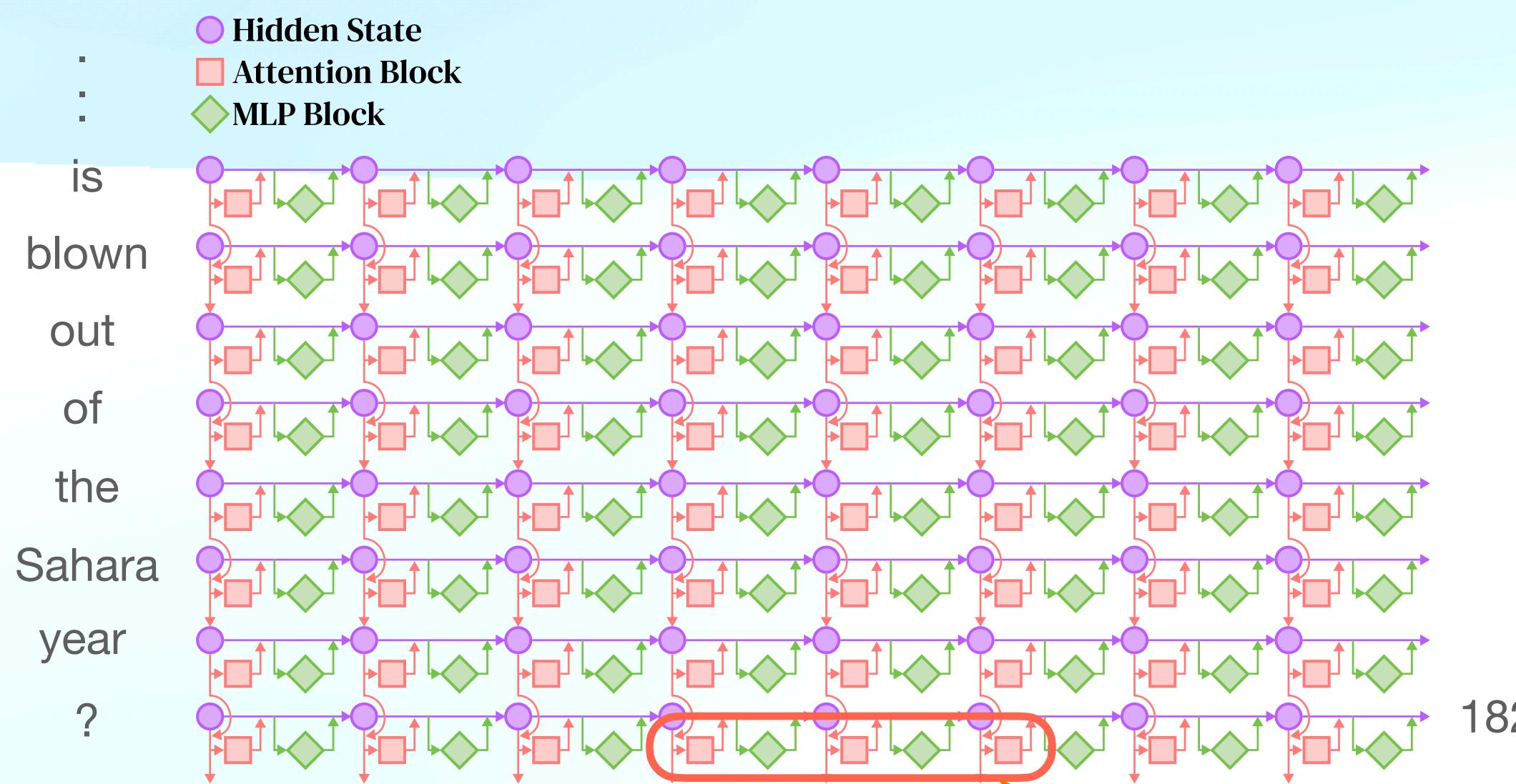
Q: Which biomes does it support?



Patching Attention Heads

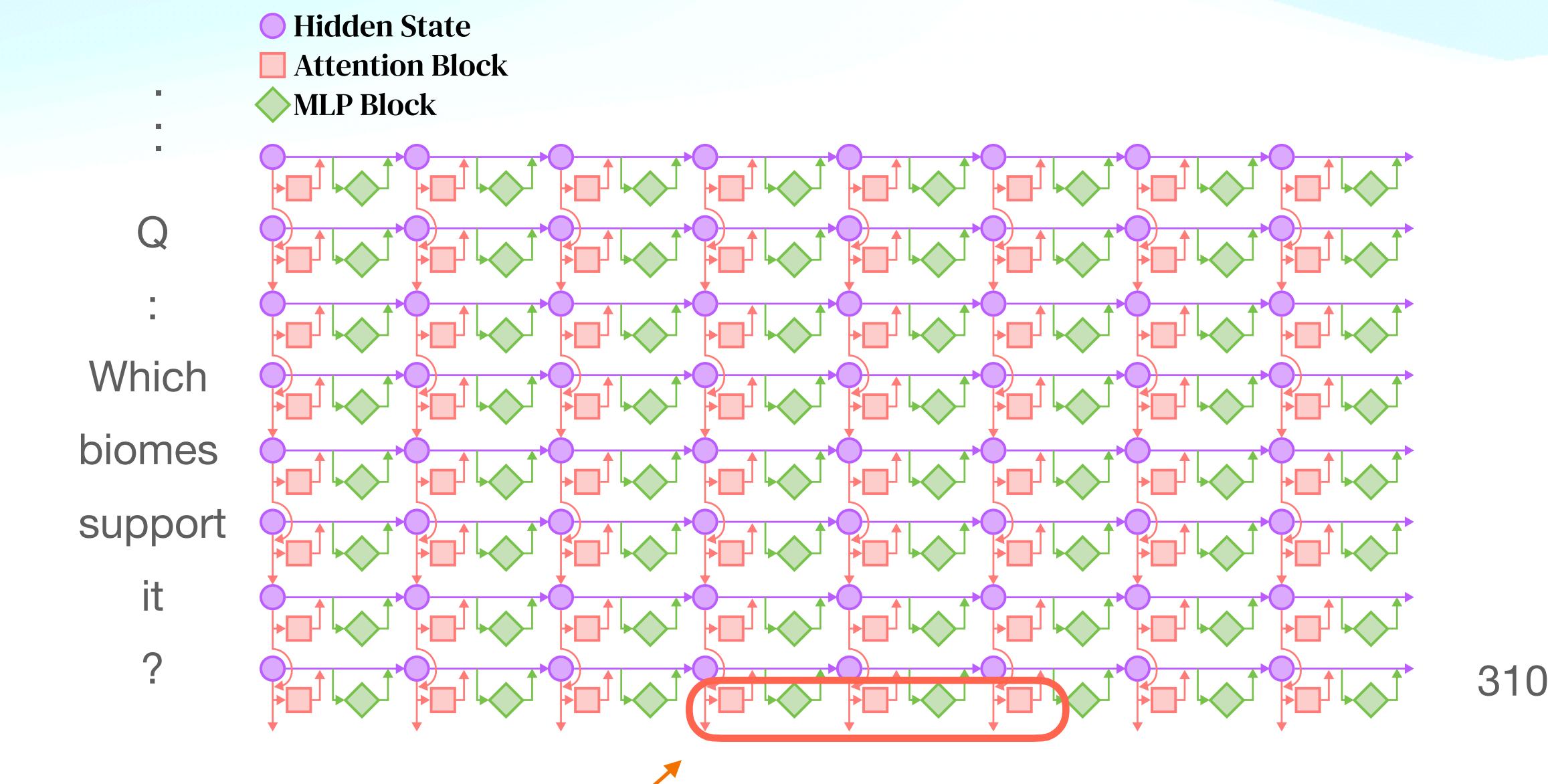
NASA's CALIPSO satellite measured 182 million tons of Sahara dust annually, but recent studies suggest it's actually 27 million tons.

Q: How much dust is blown out of the Sahara each year?

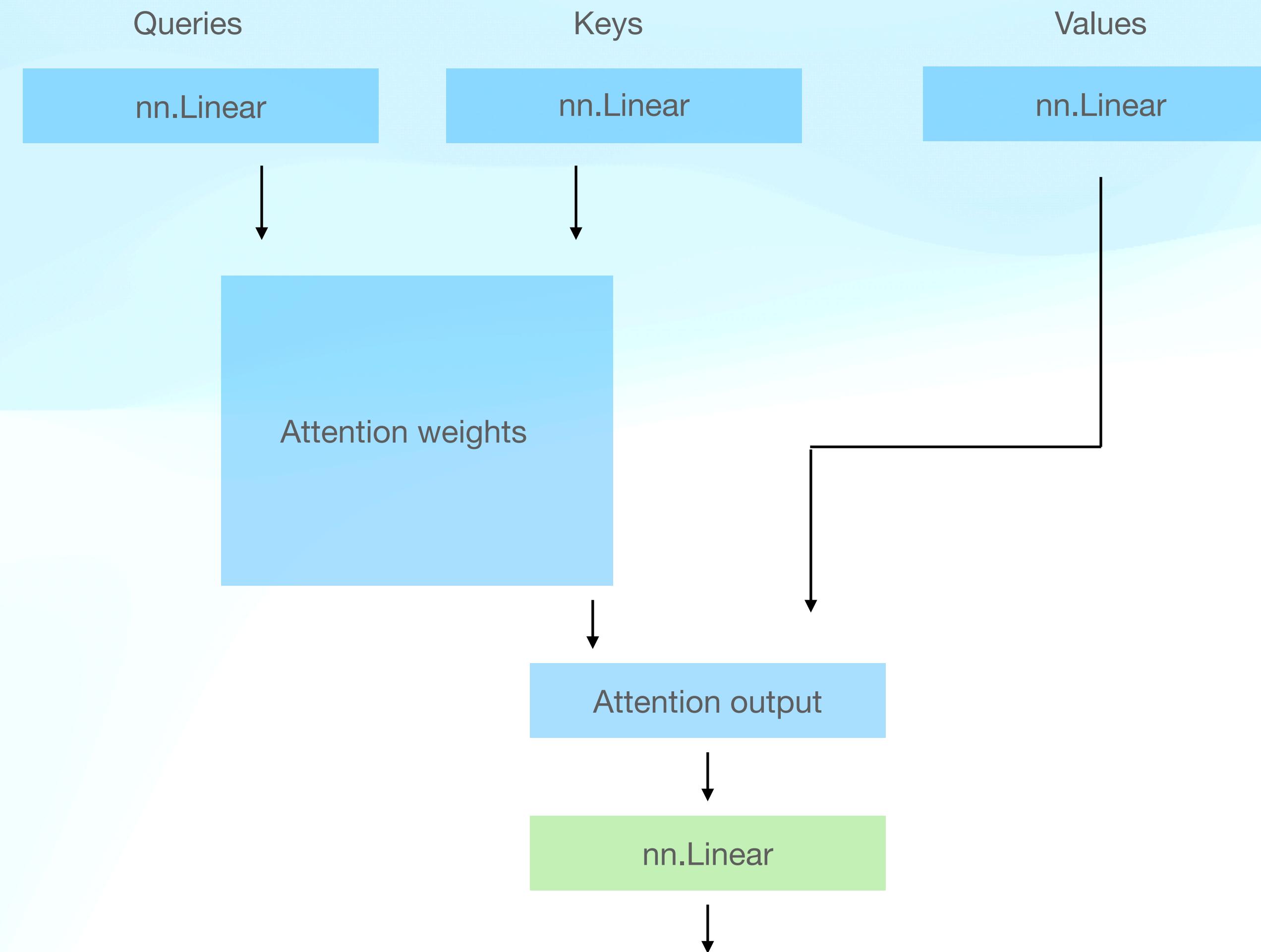


The Atlantic Ocean is the second-largest ocean on Earth. It holds about 310 million cubic kilometers of water. It spans a wide range of marine biomes, from warm tropical coral reef systems near the equator to cold, nutrient-rich polar waters in the north and south. These biomes support diverse life, including plankton, fish, whales, and deep-sea organisms, and play a major role in regulating Earth's climate through ocean currents like the Gulf Stream.

Q: Which biomes does it support?



Patching Attention Heads



Patching Attention Heads

TOWARDS BEST PRACTICES OF ACTIVATION PATCHING IN LANGUAGE MODELS: METRICS AND METHODS

Fred Zhang*
UC Berkeley
`z0@berkeley.edu`

Neel Nanda
Independent
`neelnanda27@gmail.com`

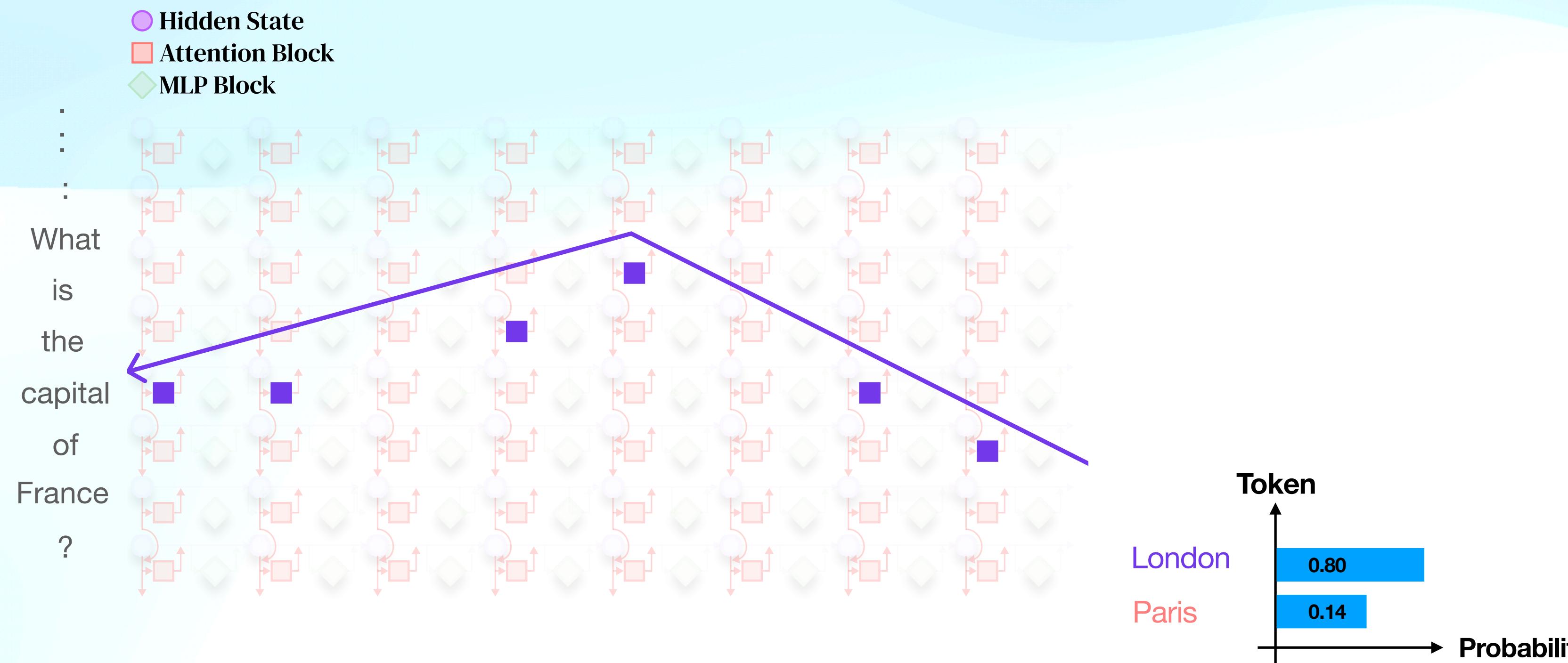
ABSTRACT

Mechanistic interpretability seeks to understand the internal mechanisms of machine learning models, where localization—identifying the important model components—is a key step. Activation patching, also known as causal tracing or interchange intervention, is a standard technique for this task (Vig et al., 2020), but the literature contains many variants with little consensus on the choice of hyperparameters or methodology. In this work, we systematically examine the impact of methodological details in activation patching, including evaluation metrics and corruption methods. In several settings of localization and circuit discovery in language models, we find that varying these hyperparameters could lead to disparate interpretability results. Backed by empirical observations, we give conceptual arguments for why certain metrics or methods may be preferred. Finally, we provide recommendations for the best practices of activation patching going forwards.

Contrastive Explanation of Knowledge Conflicts

The capital of France is London.

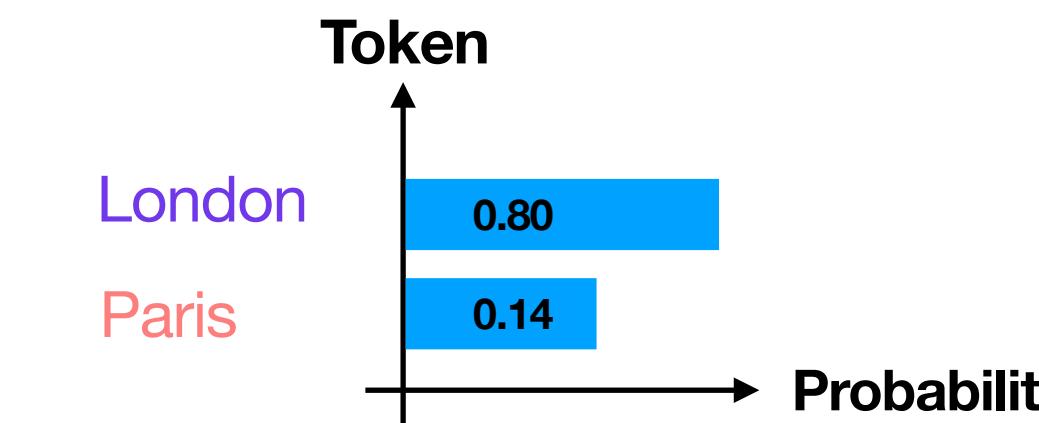
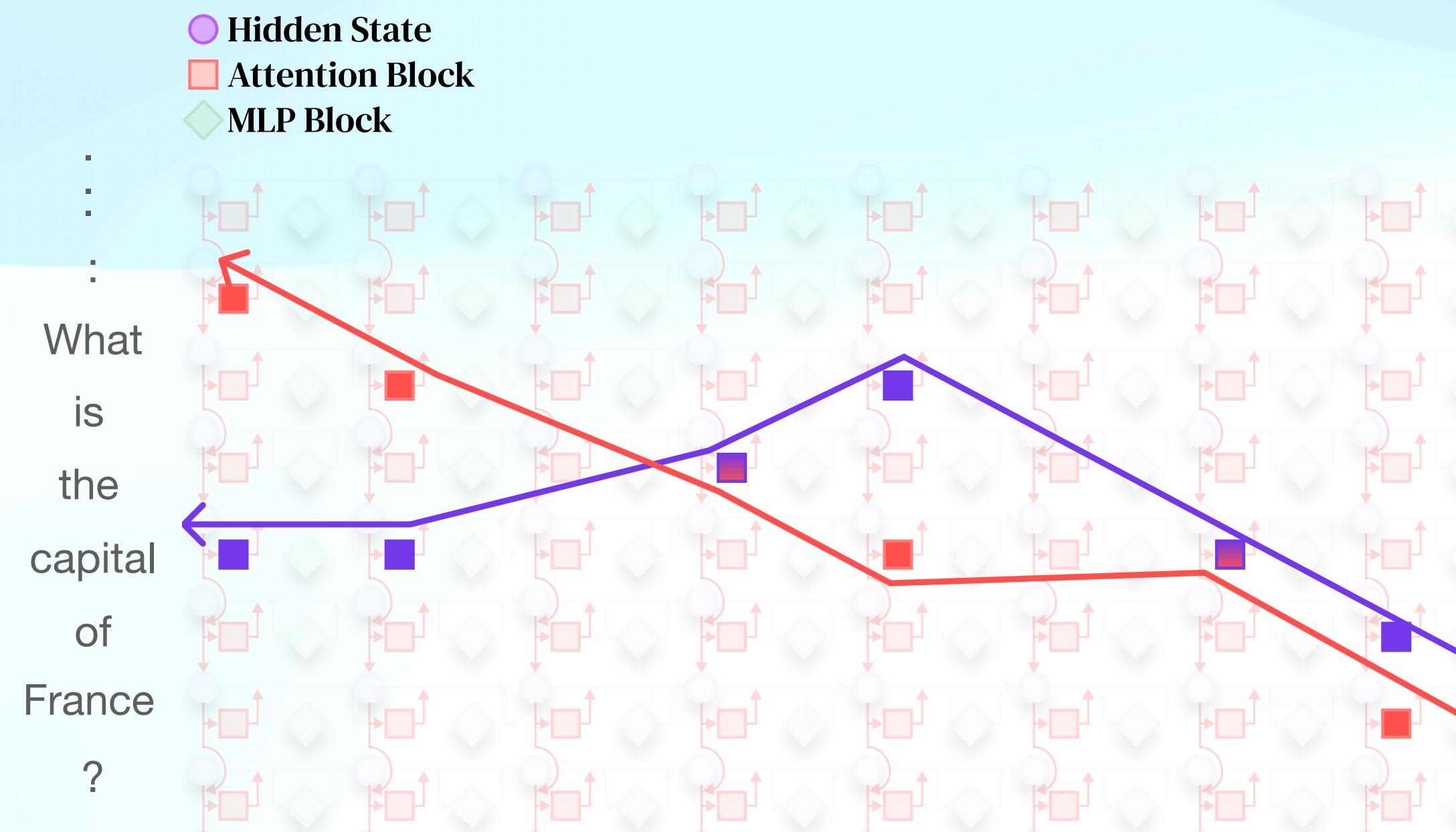
Q: What is the capital of France?



Contrastive Explanation of Knowledge Conflicts

The capital of France is London.

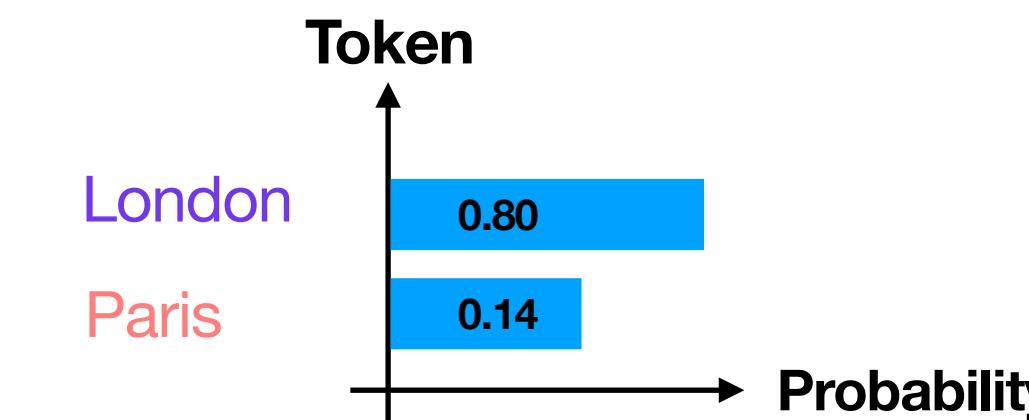
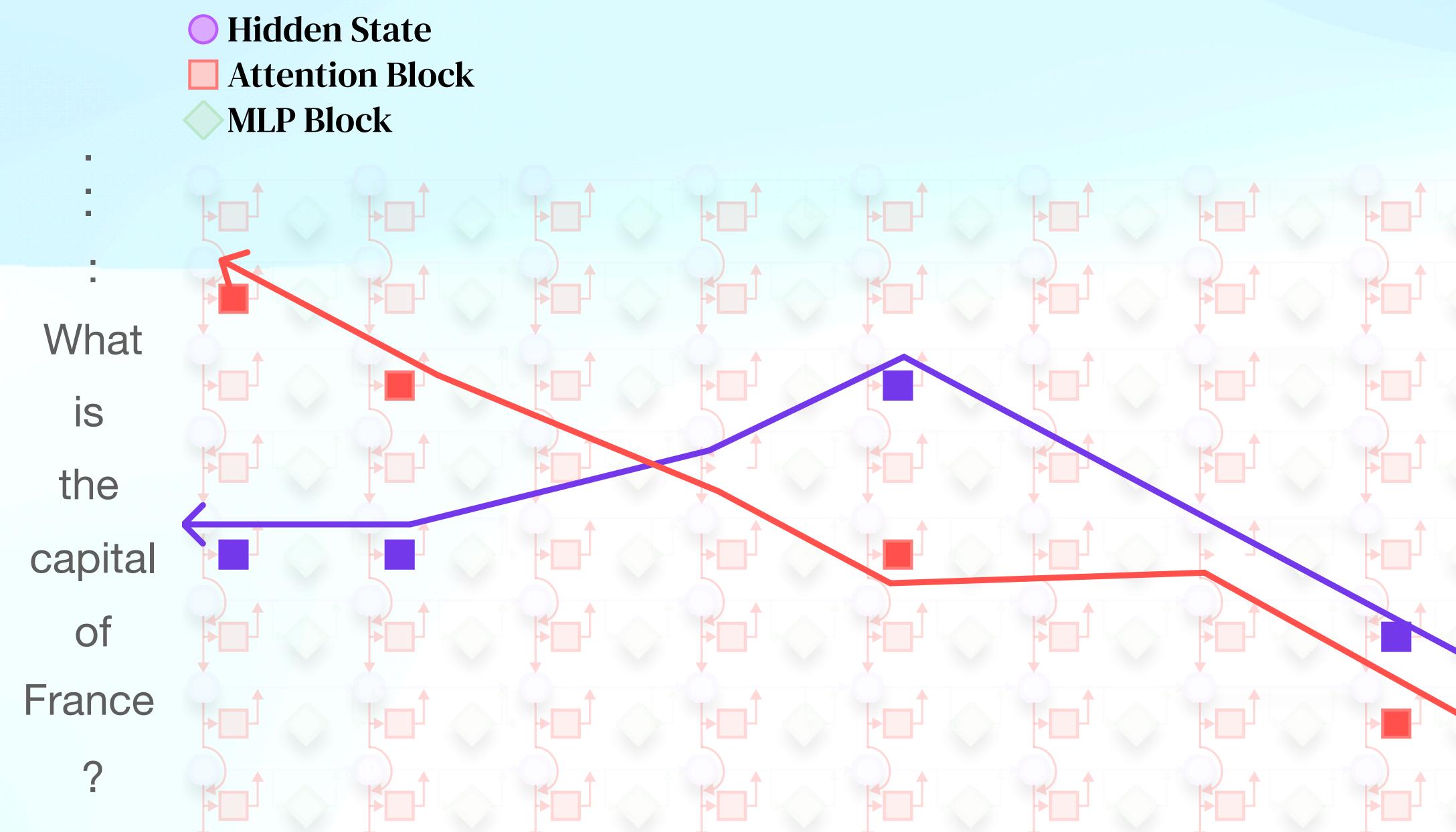
Q: What is the capital of France?



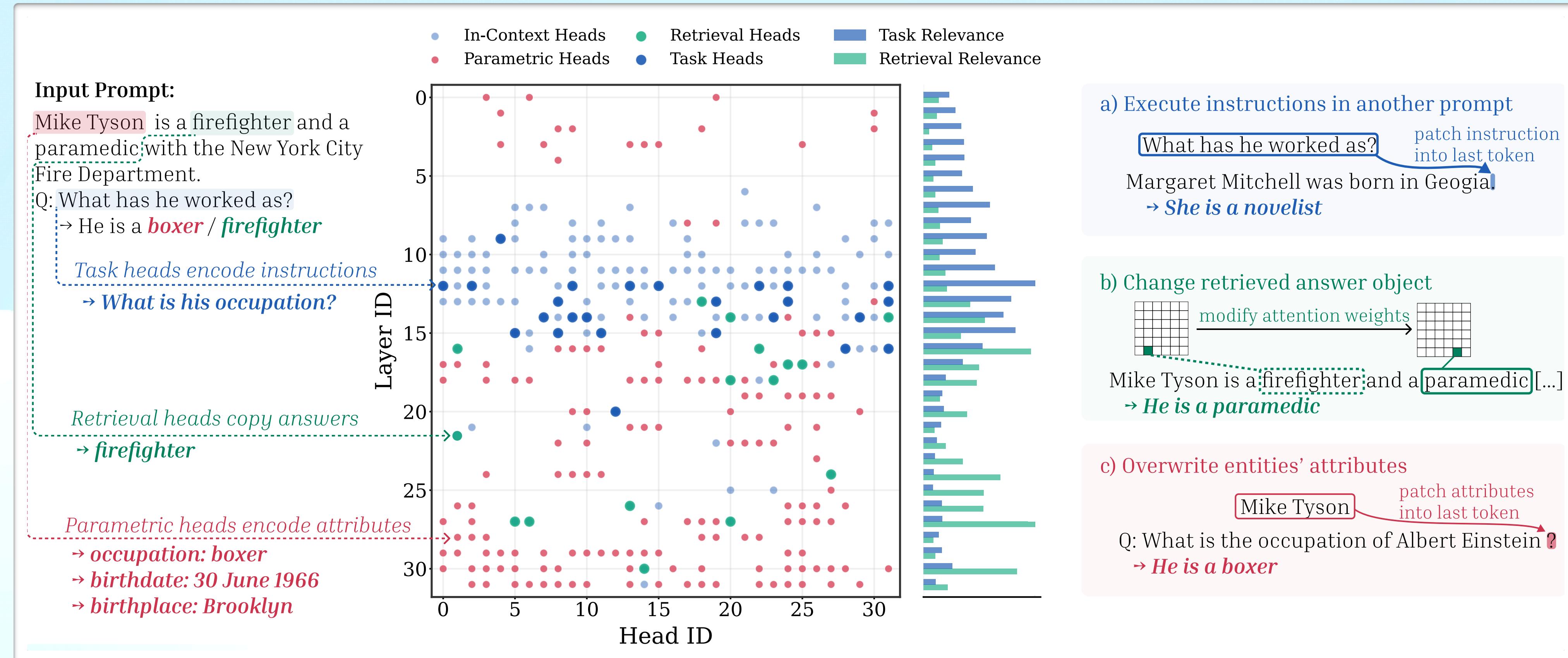
Contrastive Explanation of Knowledge Conflicts

The capital of France is London.

Q: What is the capital of France?



A universal map of language models?



A universal map of language models?

Input Prompt:

Mike Tyson is
paramedic/wit
Fire Departme
Q: What has he
→ He is a **bo**.

Task head
→ **What is**

Retrieval h
→ **firefight**

Parametric
→ **occupati**
→ **birthdat**
→ **birthpla**

The Atlas of In-Context Learning: How Attention Heads Shape In-Context Retrieval Augmentation

Patrick Kahardipraja^{1,*} Reduan Achitbat^{1,*} Thomas Wiegand^{1,2,3}

Wojciech Samek^{1,2,3,†} Sebastian Lapuschkin^{1,4,†}

¹Department of Artificial Intelligence, Fraunhofer Heinrich Hertz Institute

²Department of Electrical Engineering and Computer Science, Technische Universität Berlin

³BIFOLD - Berlin Institute for the Foundations of Learning and Data

⁴Centre of eXplainable Artificial Intelligence, Technological University Dublin

{firstname.lastname}@hhi.fraunhofer.de

Abstract

Large language models are able to exploit in-context learning to access external knowledge beyond their training data through retrieval-augmentation. While promising, its inner workings remain unclear. In this work, we shed light on the mechanism of in-context retrieval augmentation for question answering by viewing a prompt as a composition of informational components. We propose an attribution-

n another prompt
as?
patch instruction
into last token
s born in Georgia

ver object
weights
and a paramedic [...]

ributes
patch attributes
into last token
ion of Albert Einstein ?

WE'D LOVE YOUR FEEDBACK!

