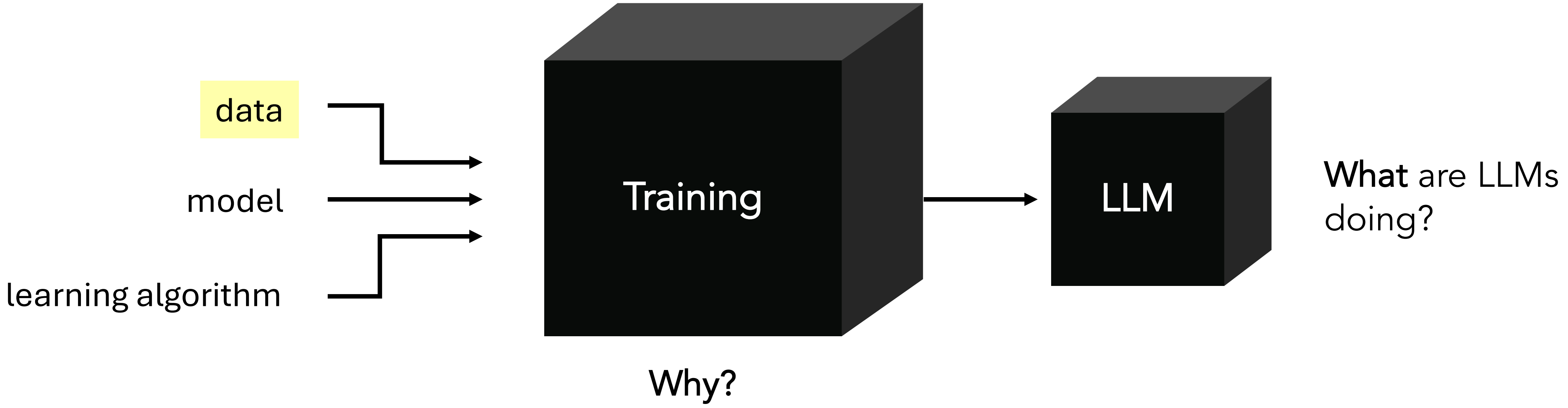
How does data shape learning?

A case study of factual recall
and the surprising role of data diversity

Nicolas Zucchet - ETH Zürich

MPI Tübingen, *November 28th, 2025*

Understanding large language models



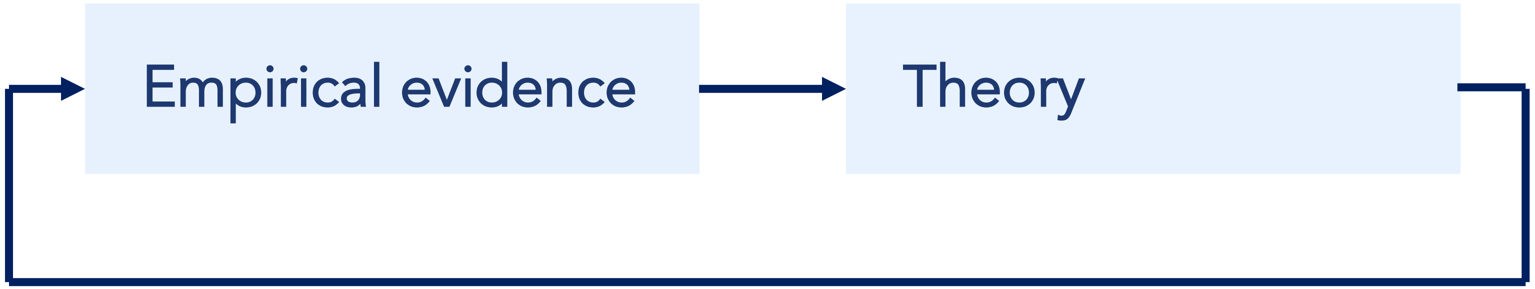
Approach and outline of the talk

Part I

How do language models learn facts?

Part II

The emergence of sparse attention



How do language models learn facts?

Dynamics, curricula and hallucinations


Jörg Bornschein


Stephanie Chan


Andrew Lampinen

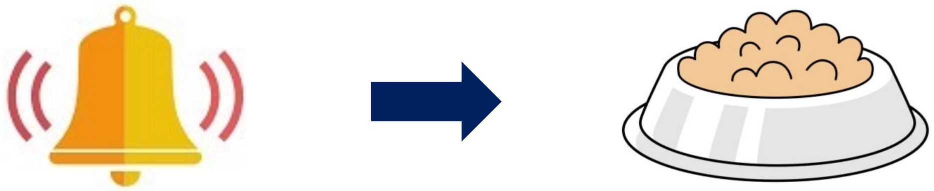

Razvan Pascanu


Soham De

Oral @ COLM 2025

Associative recall in language models

What is associative recall?



Pavlov, *Oxford University Press*, 1927

Hopfield, *PNAS*, 1982

Large language models **excel at it** and store an incredible amount of associative knowledge in their weights.

Replicate some (basic) features of intelligence that is **worth studying**.

A synthetic framework to study associative recall

Real-world data is a complicated mix of **different sources**, which require learning **different abilities** in parallel

We use synthetic data to have **full control** on data properties (distribution + abilities required)

N individuals with some **attributes** (e.g. birthdate, birthplace)

Bank of **template sentences** for each type of attributes

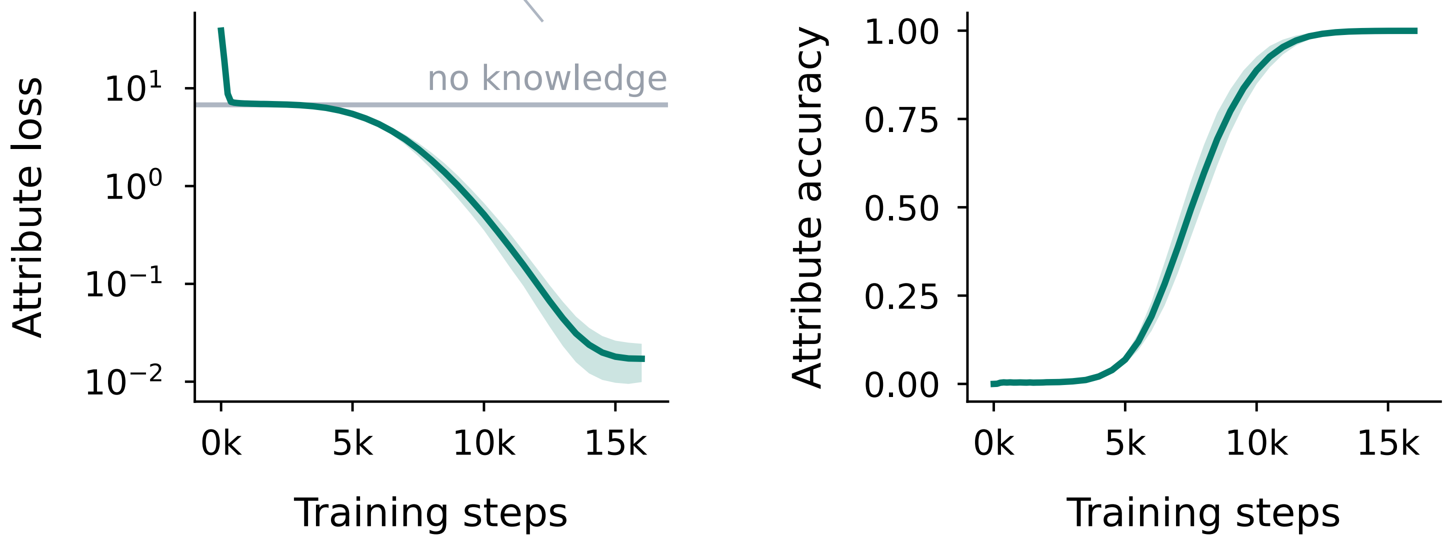
James Frida Zhu's life began on March 16, 2042.
James Frida Zhu is a native of Shanghai.

Predicting attribute tokens is a factual recall task which measures the model's knowledge.

Adaptation of the setup from Allen-Zhu and Li 2023

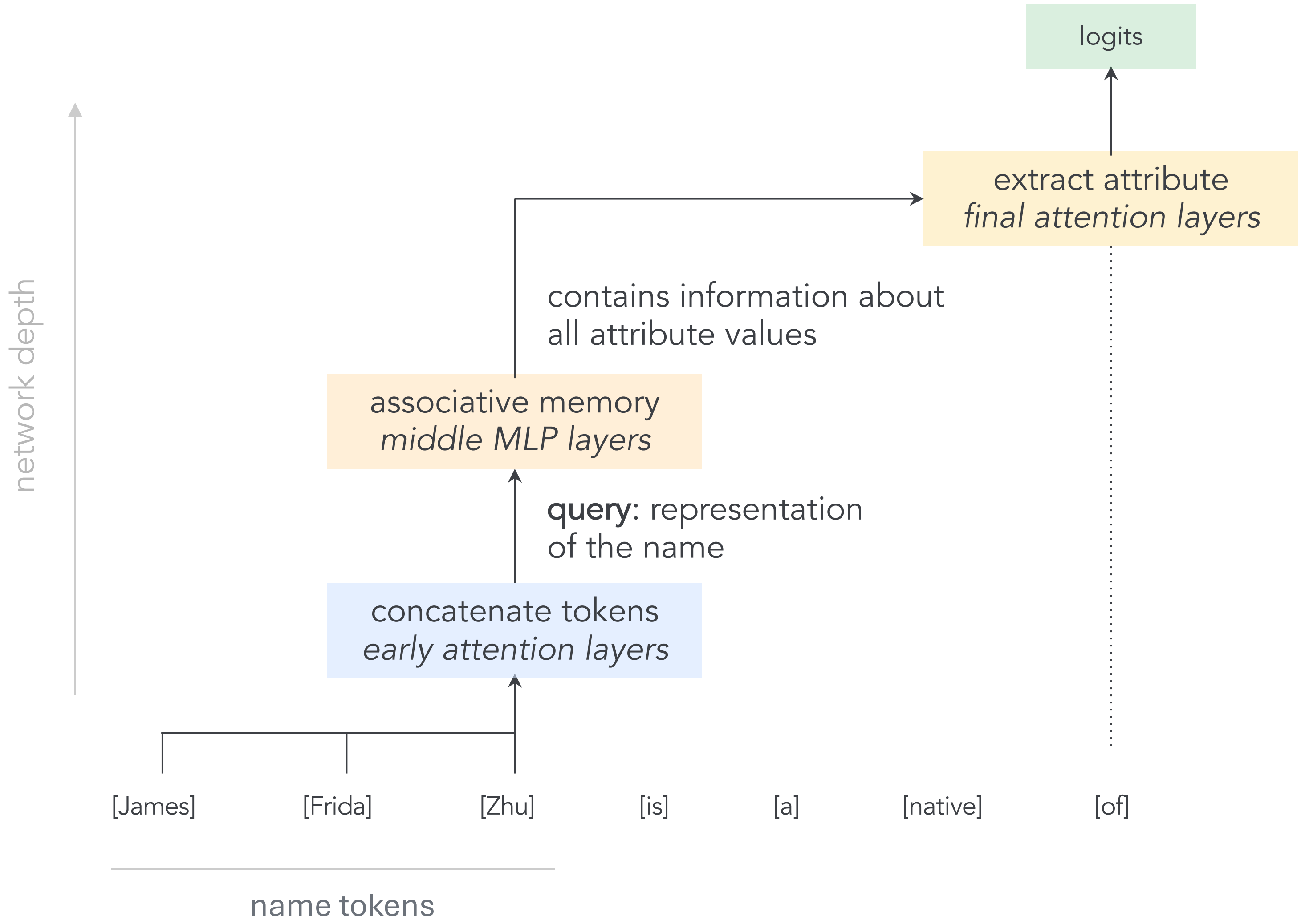
Knowledge acquisition happens in phases

best performance a model **without** individual-specific knowledge can reach



results for 44M params 8-layers GPT-style Transformers

Recall circuits are learned during the plateau



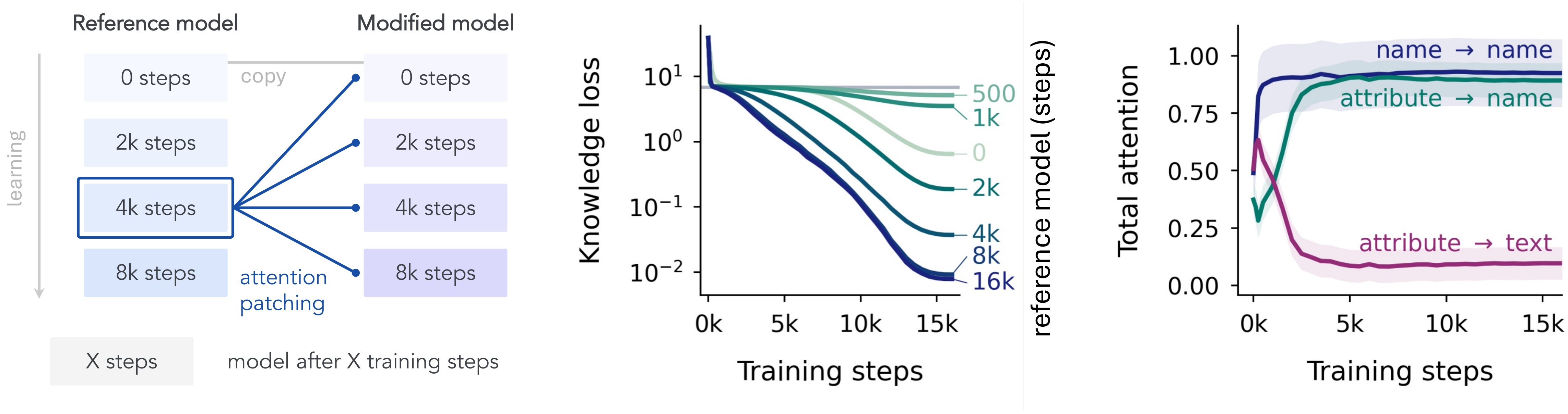
We will show that the **formation of the attention circuits** is happening during the plateau

The **signature of the concatenation circuit** is high attention to name tokens when processing the last name.

The **signature of the extraction circuit** is high attention to the last name token when predicting the first attribute token.

Figure adapted from Nanda et al. 2023

Recall circuits are learned during the plateau

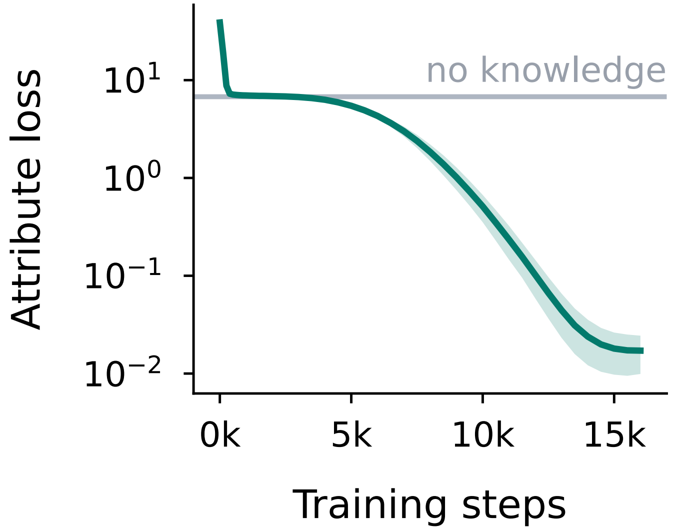


Some attention-based circuit is created during the plateau…

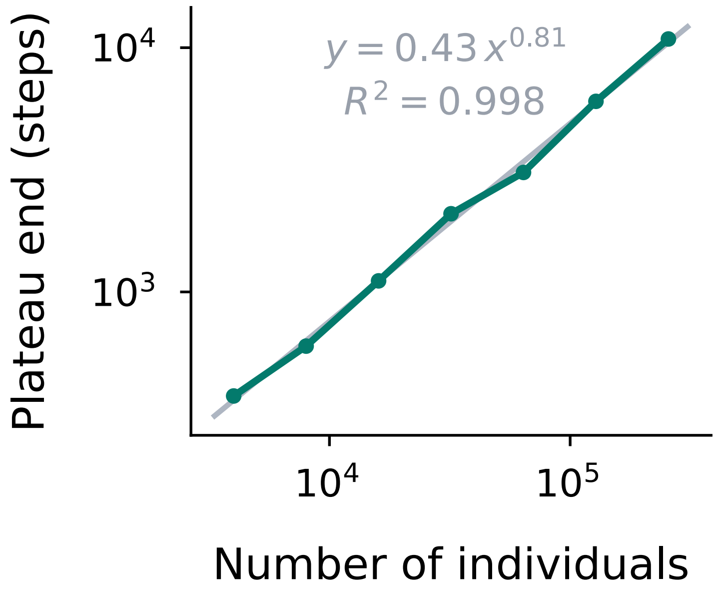
It is likely the **extraction circuit**

Why is there a learning plateau? Part II!

Effect of the number of individuals on plateau length



How does the **number of individuals** affect the **plateau length**?

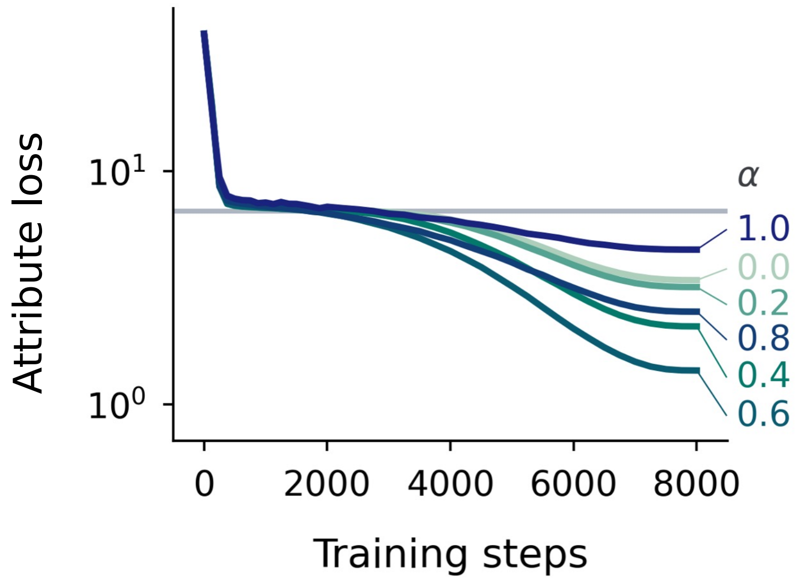


Imbalances in data distribution can speed up learning

Idea: if we train on **imbalanced individual distributions**, the model should leave the loss plateau **earlier** as it is able to build the right circuits on the frequent individuals (and ideally reuse them for the less frequent ones).

To test this: we sample individuals according to an **inverse power law distribution** during training.

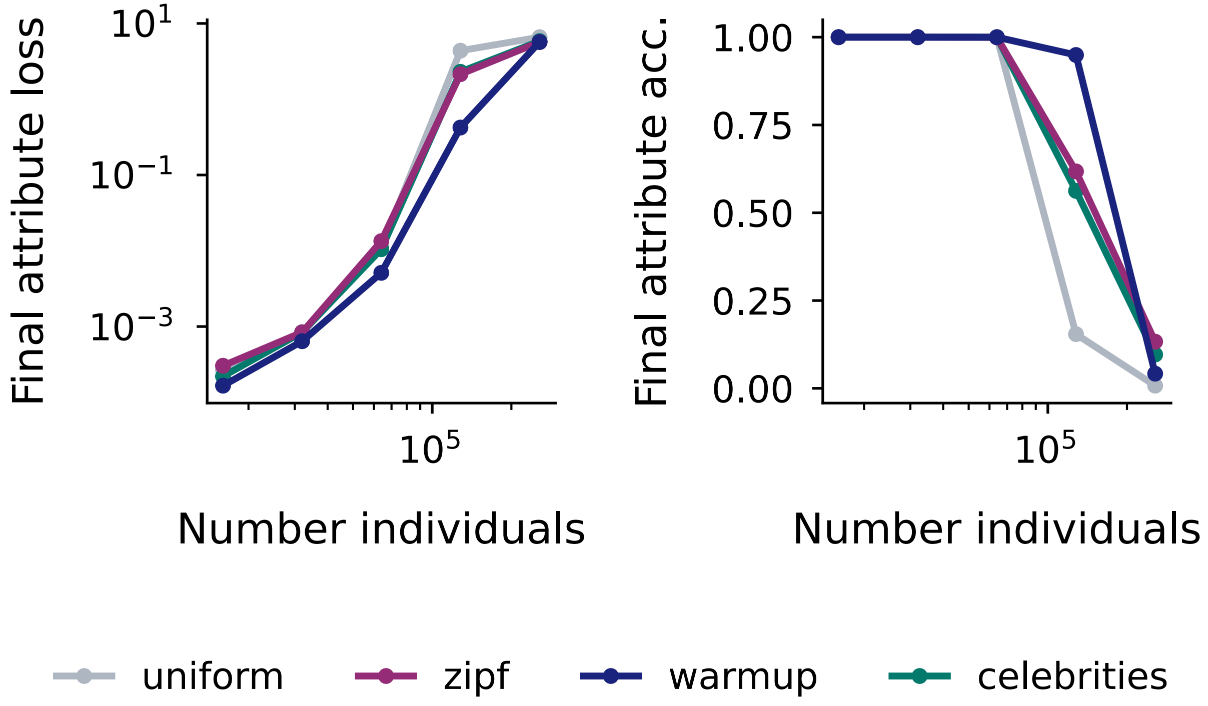
$$P(i) \propto \frac{1}{i^\alpha}$$



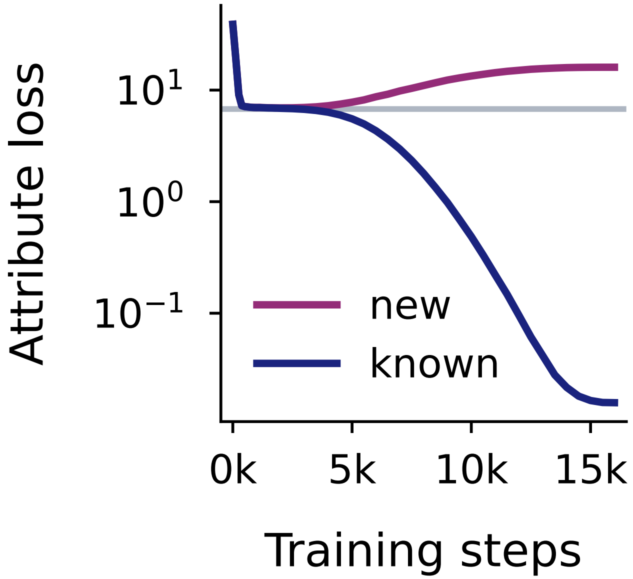
This is still evaluated **uniformly** over the population!

Imbalances in data distribution can speed up learning

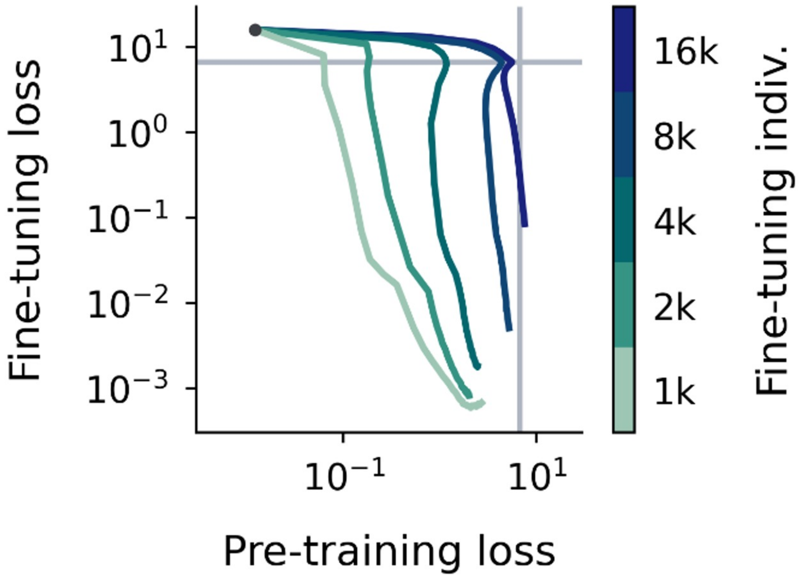
We can go one step further: start with a **small population** and then **increase** it later



Fine-tuning on new knowledge is challenging



Hallucinations (overconfident wrong predictions) appear **concurrently** to knowledge



Fine-tuning quickly **destroys** existing knowledge

Replay partially mitigates the problem

Takeaway I.1

Language models acquire knowledge in **three phases,** in an **emergent** fashion

Implication: knowledge used early in the plateau is forgotten

Takeaway I.2

Low **data diversity** can **speed up** learning

Implication: LLMs might learn factual recall faster because internet data is skewed

Implication: Diversity based curriculum might be a powerful tool

Part II
understand
why

Takeaway I.3

Incorporating **new knowledge** through fine-tuning is **hard**

Implication: naïve fine-tuning is not suited to adding new knowledge to LLM parameters

The emergence of sparse attention

Dynamics, curricula and hallucinations

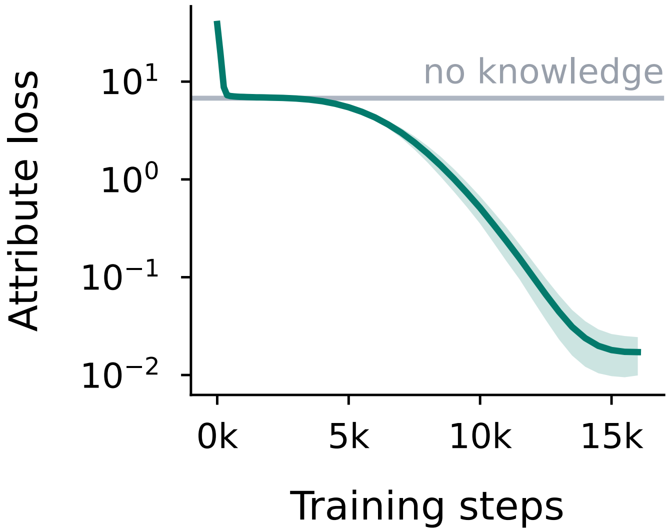


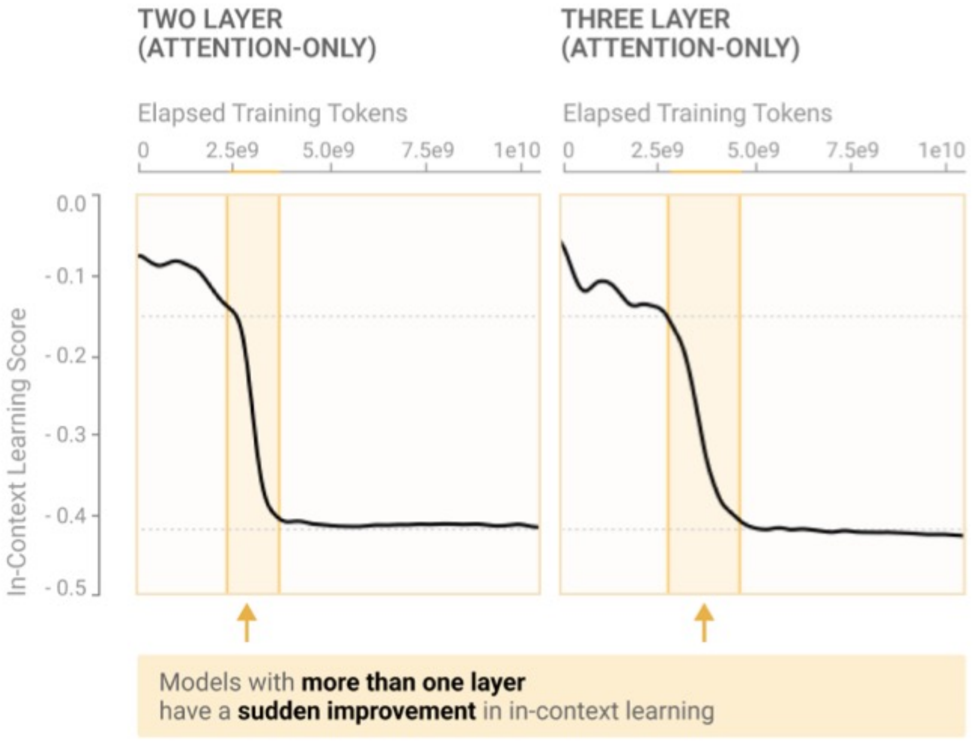
Francesco D'Angelo Stephanie Chan Andrew Lampinen

Oral @ NeurIPS 2025

Motivation

Understand **emergent** dynamics in **Transformers** and the role of **data**

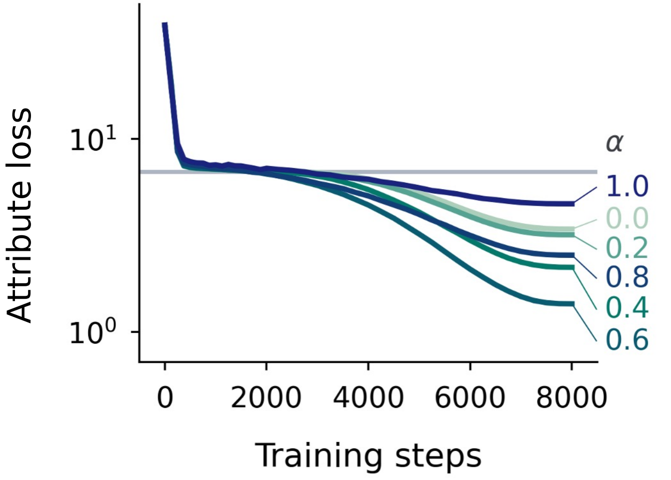


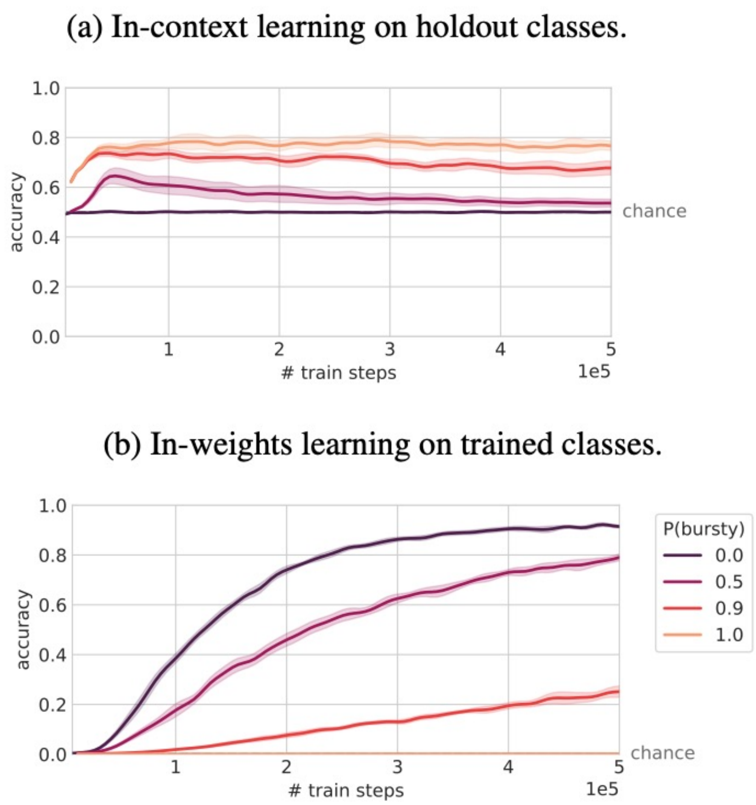


Olsson et al. 2022

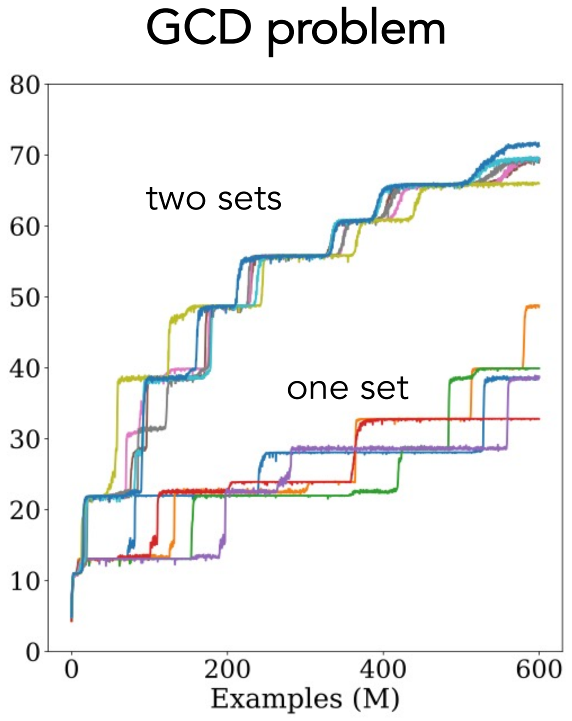
Motivation

Understand why **repetition** helps





Chan et al. 2022



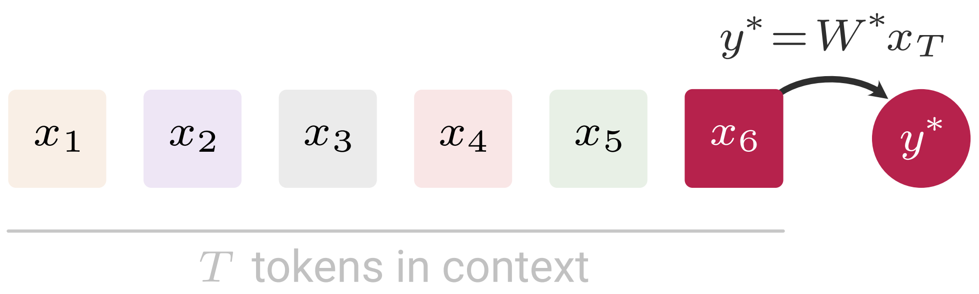
Charton & Kempe 2024

A theoretically tractable toy model

Main abilities language models need to solve associative recall task:

- **Filtering relevant information** out of "noise"
- **Transformation** of this information into desired answer (e.g. an associative memory)

Task. Single-location linear regression



x, y dimension d

Model. Simplified Transformer

$$y = W \sum_{t=1}^{T} \mathrm{softmax}(a)_t \, x_t$$

Learning dynamics

Under reasonable assumptions, we can reduce the learning dynamics to **two variables**

Δa **logit difference** between relevant and non-relevant tokens

attention to relevant token $\alpha = \dfrac{1}{1 + (T-1)\exp(\Delta a)}$

w **projection** of W on W^*

We get the **2D non-linear differential equation**:

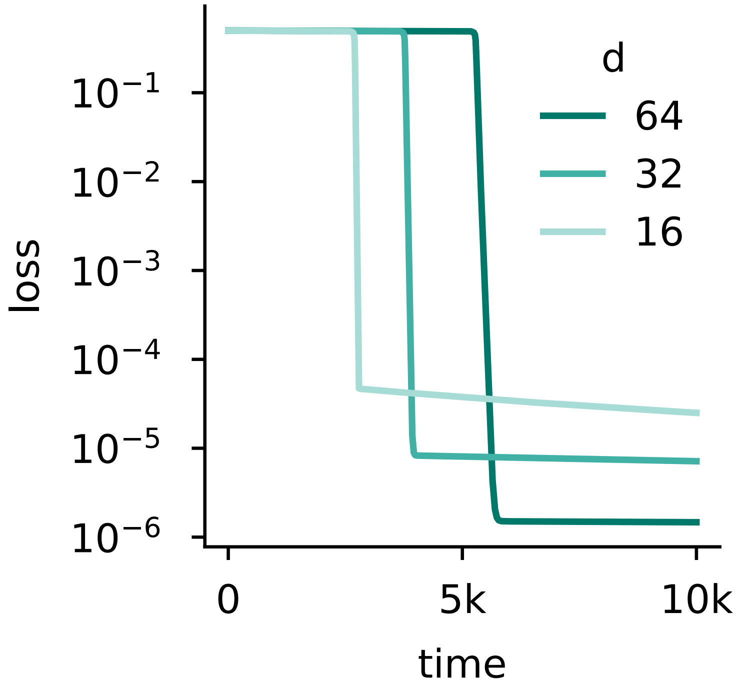
$$\dot{w} = \frac{\alpha(\sqrt{d} - \alpha w)}{d} - \frac{(1-\alpha)^2 w}{d(T-1)}$$

$$\dot{\Delta a} = \alpha(1-\alpha)\left(\frac{w(\sqrt{d} - \alpha w)}{d} + \frac{(1-\alpha) w^2}{d(T-1)}\right)$$

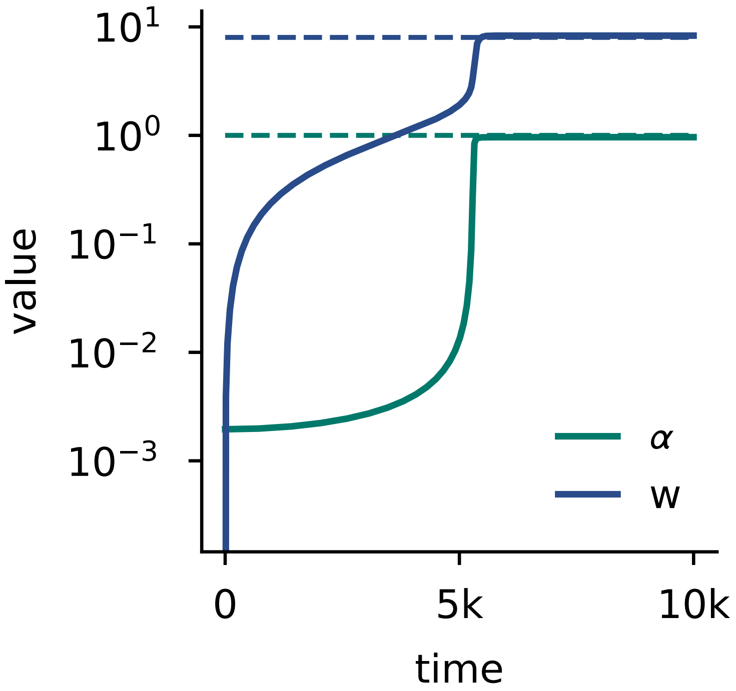
$$w_0 = 0$$

$$\Delta a_0 = 0$$

Learning dynamics



✅ Exhibits sharp phase transitions



w learns before attention focuses on the relevant token

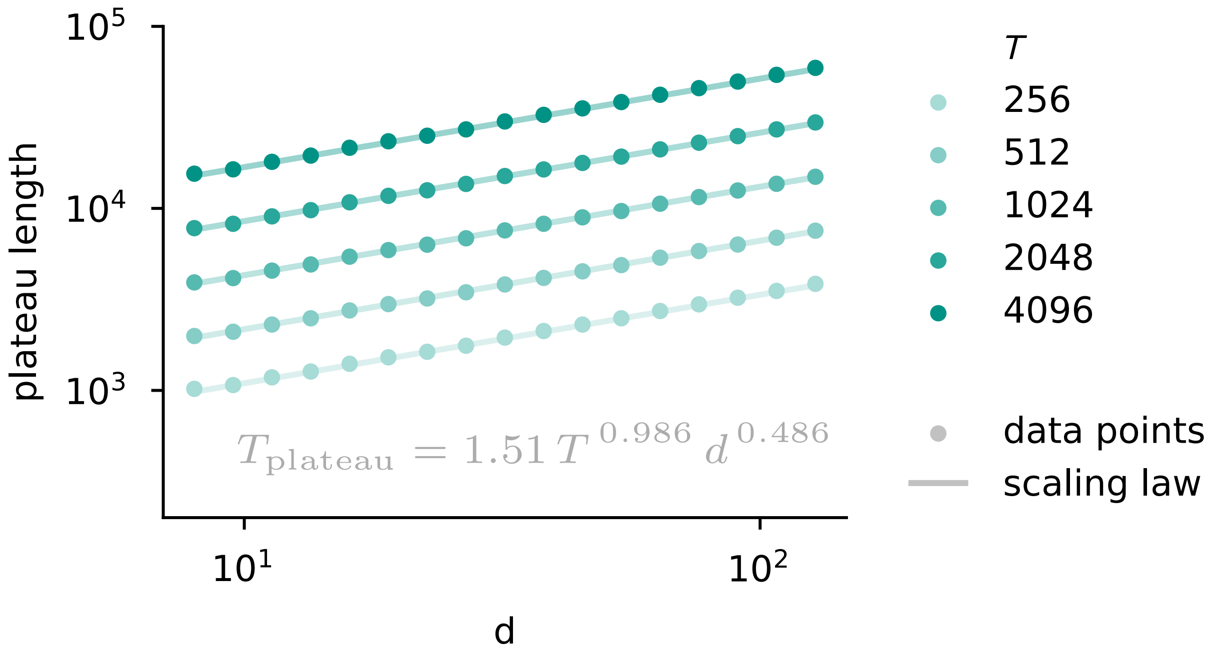
Initial learning dynamics

Linearized dynamics at initialization

$$\begin{pmatrix} \dot{w} \\ \Delta a \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{dT}} \\ 0 \end{pmatrix} + \begin{pmatrix} 0 & \frac{1}{\sqrt{dT}} \\ \frac{1}{\sqrt{dT}} & 0 \end{pmatrix} \begin{pmatrix} w \\ \Delta a \end{pmatrix}$$

Escape time (time to decrease loss by ε)

$$T_\varepsilon = \frac{\sqrt{d}T}{2} \ln\left(\varepsilon\sqrt{d}T\right) \sim \sqrt{d}T$$



Almost perfect **empirical fit**!

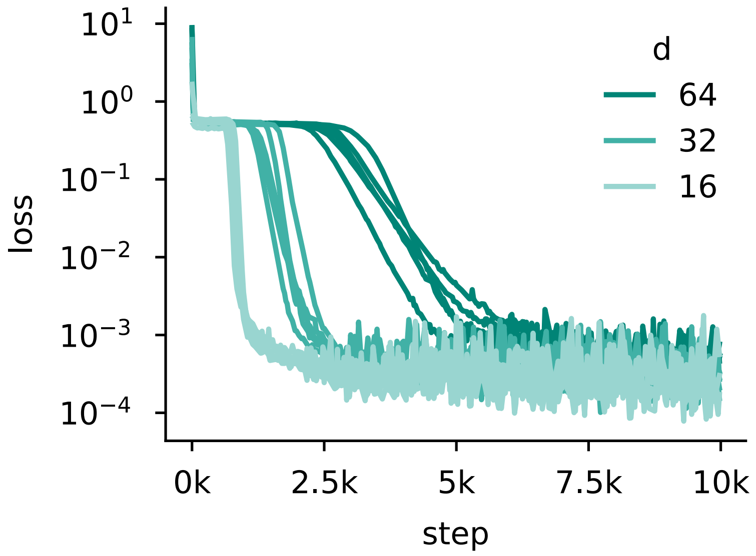
Learning time **increases** when:

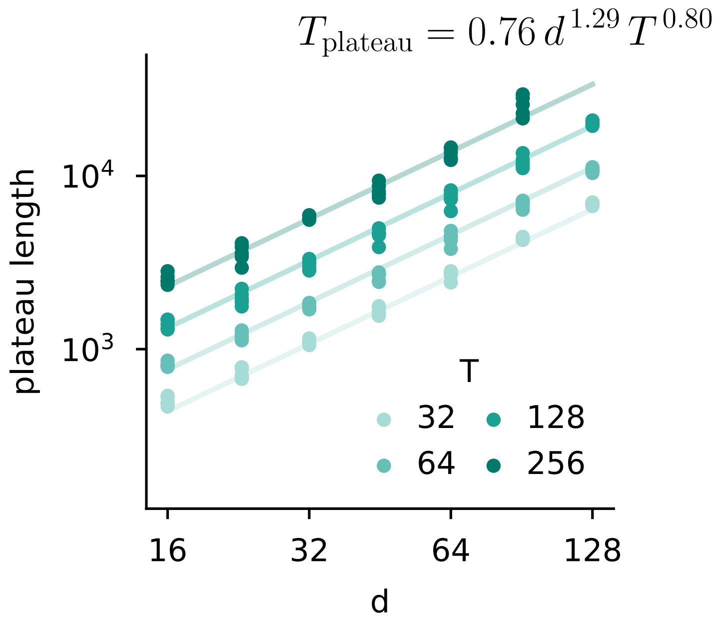
- Attention gets **sparser**
- **Less signal** to learn the feedforward mapping

In Part I, we saw the effects of increasing d

Transformer learning dynamics on the task

More **realistic** version of the task: randomized position of the relevant token, feature to indicate it





Still a power law but **exponents are different**. Changes with largest impact are

- GD vs. Adam
- task specifics
- architecture (multi layer, multi head, positional encoding)

Introducing repetition

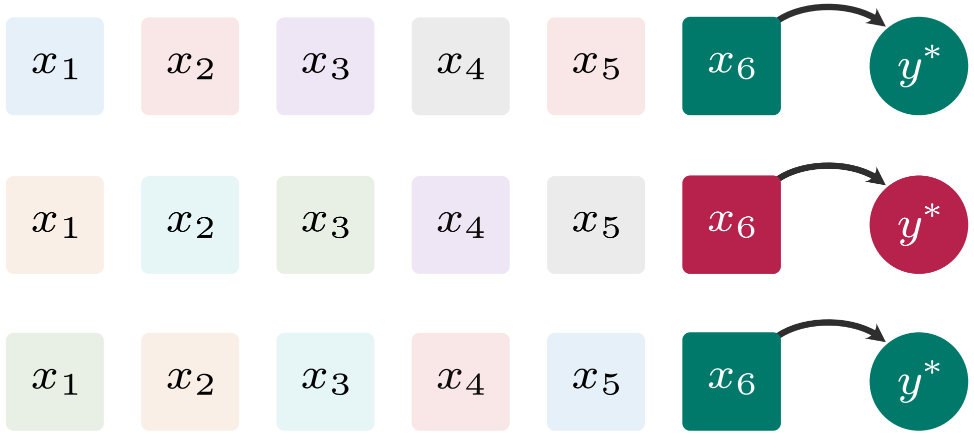
In-context repetition

the relevant token x_T appears B times

Example.

In a Harry Potter chapter, `[Harry Potter]` appears multiple time within the context

Cross-sample repetition



the same x_T appears with probability p

Examples.

In Harry Potter books, `[Harry Potter]` appears more often than `[Sirius Black]`

Overrepresented individuals in Part I

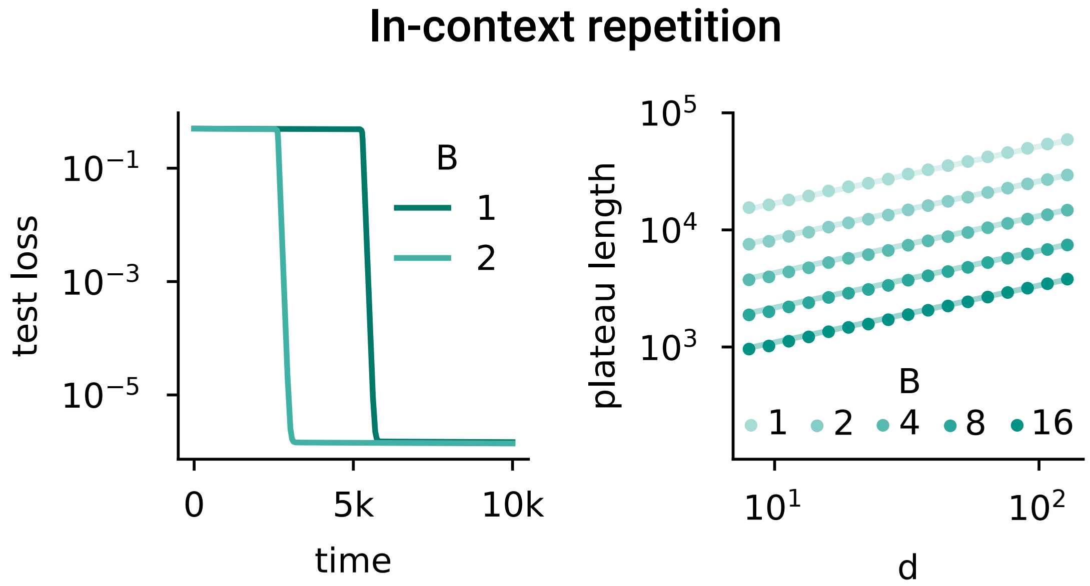
Understanding in-context repetition

In-context repetition

the relevant token x_T appears B times

Increases the **signal to noise ratio**

Equivalent to dividing the sequence length by B



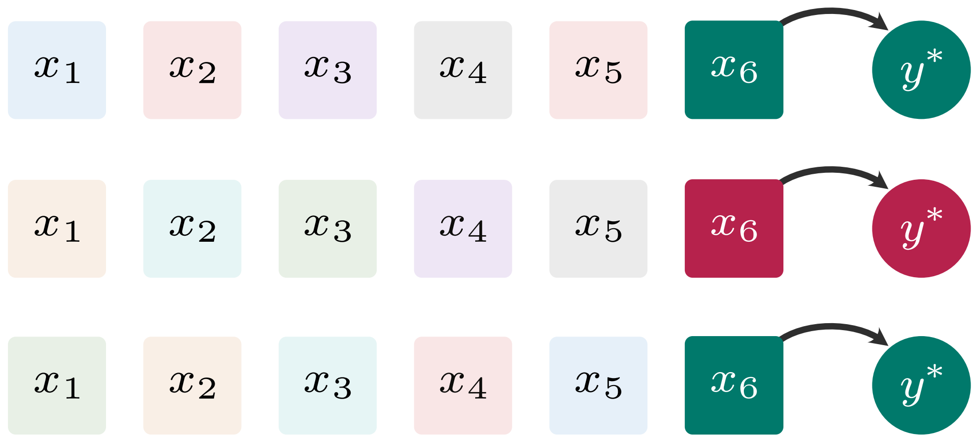
$$T_{\text{plateau}} = 1.51\, d^{0.49} \left(\frac{T}{B}\right)^{0.99}$$

Similar effects when training actual Transformers

Chan et al. 2022

Understanding cross-sample repetition

Cross-sample repetition



the same x_T appears with probability p

Cross-sample repetition speeds up emergence

W learns faster on the **repeated** dimension than on the others

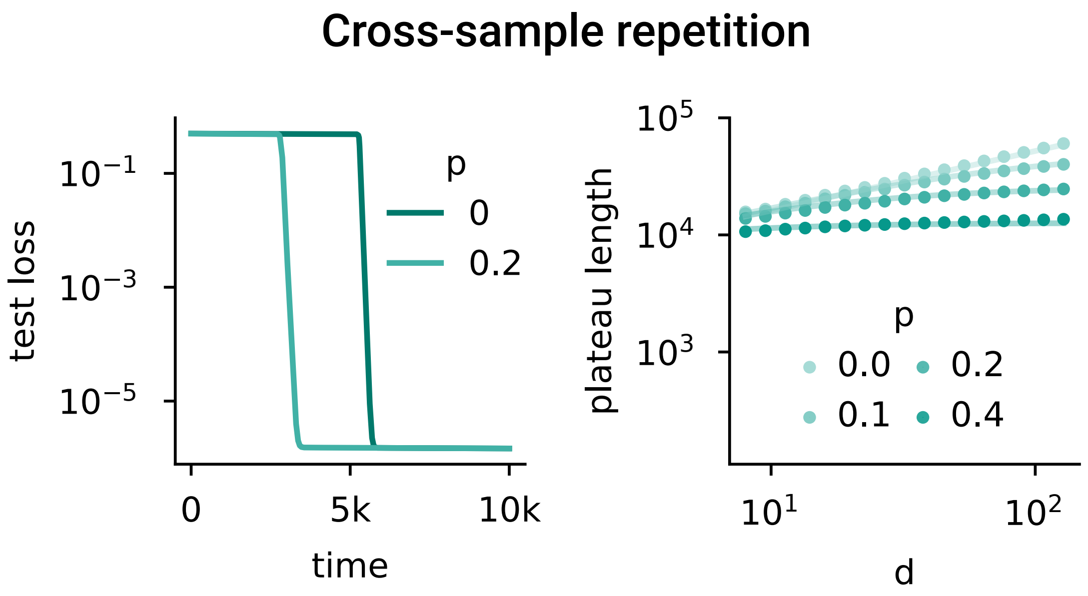
↓

Attention learns faster overall because W provides **better teaching signal** on the repeated data

↓

This speeds up the learning of W on non-repeated data, and thus learning overall

Understanding cross-sample repetition



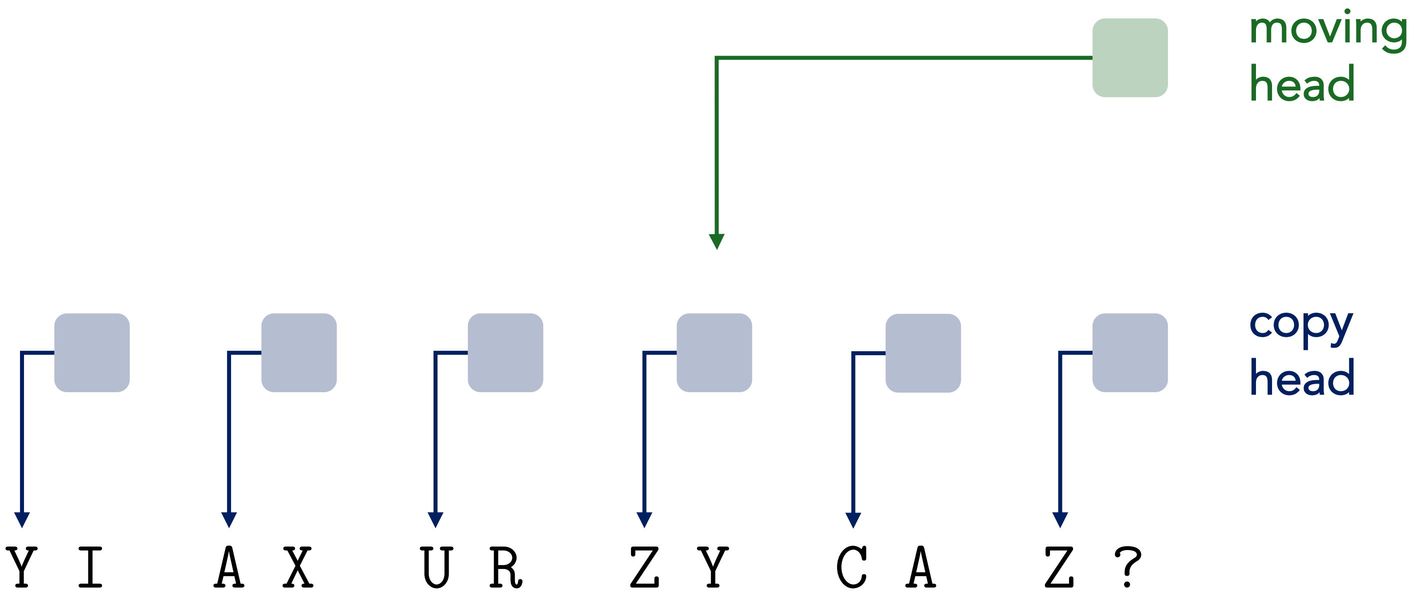
$$T_{\text{plateau}} = 2.15 \left(\frac{\sqrt{d}T}{\sqrt{p^2 d + (1-p)^2}} \right)^{1.02}$$

Main factor **can be derived theoretically** (similar analysis but with three variables)

Results less clean for full Transformers, but they still **show the benefits of cross-sample repetition**

The effects of repetition we saw in Part I can be **reproduced and understood** in this simple setup

Validation on the (in-context) associative recall task



N_{tokens} vocab size (~ d)

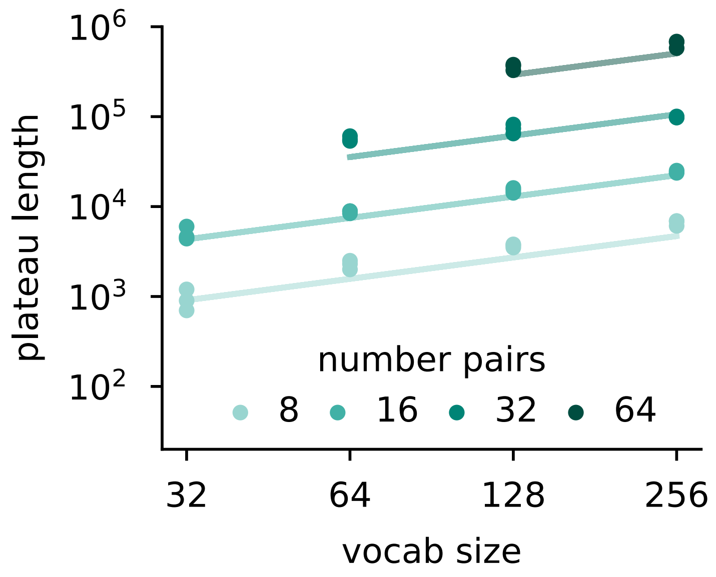
N_{pairs} number of pairs in context ($(T-1)/2$)

Combination of **two sparse attention layers**: we should be able to say something about it!

Well studied setup; we use it to test whether we can provide some simple framework on how to think about how data affects learning speed.

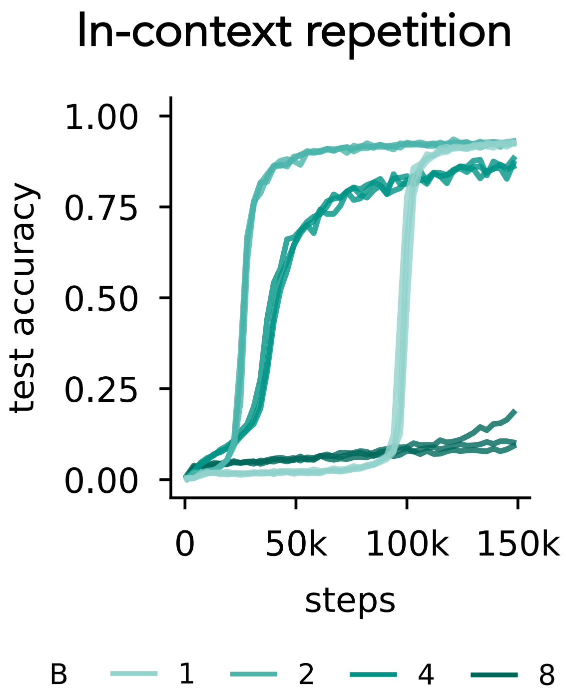
Olsson et al. 2022

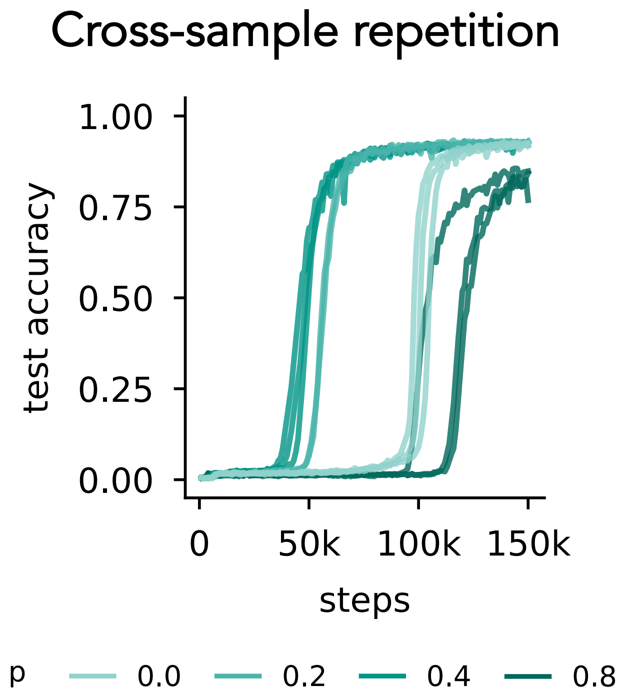
Validation on the (in-context) associative recall task



$$T_{\text{plateau}} = 0.55\, N_{\text{tokens}}^{0.79}\, N_{\text{pairs}}^{2.25}$$

✅ Same (qualitative) behavior as in the toy task





Increasing in-context repetition is more efficient (cf. power law)

Repetition **speeds** up training, but leads to **overfitting**

Dynamics are messy, hard to get a clean power law

Takeaway II.1

Learning **sparse attention** is prone to **emergent** behaviors

Implication: many LLM abilities rely on sparse attention, how common are emergent behaviors?

Takeaway II.2

Longer sequences and more **diverse** data slow down learning

Implication: provides one explanation for the benefits of context-length scaling

Implication: skewed data can sometimes be a feature rather than a bug

Takeaway II.3

The mental model provided by the sparse attention lens qualitatively applies to **more realistic circuits**

Looking ahead (theory side)

What happens when learning deeper circuits (i.e. composition of multiple sparse attentions)?

Learning compositions of sparse attention can take exponential with circuit depth time

- (How) does subcircuit reuse speed up learning?
- How can data distributions reduce that (e.g. by learning one part of the circuit after each other)?

Critical to better understand **how / when / why LLMs develop certain abilities** during pre-training

Wang, Nischani et al. 2025

Looking ahead (empirical side)

How do language models learn to reason?

Lots of interesting directions, e.g.

- What's the role of data
- What can we hope from supervised vs. RL fine-tuning

How far can we push data diversity as a lever to speed up learning?

We saw the benefits of reduced data diversity to speed up emergence in toy settings

- Do these results translate to more realistic settings?
- Is natural language distribution close to optimal in this regard?

As babies, we learn from data of increasing diversity (+ complexity); might be an overlooked feature