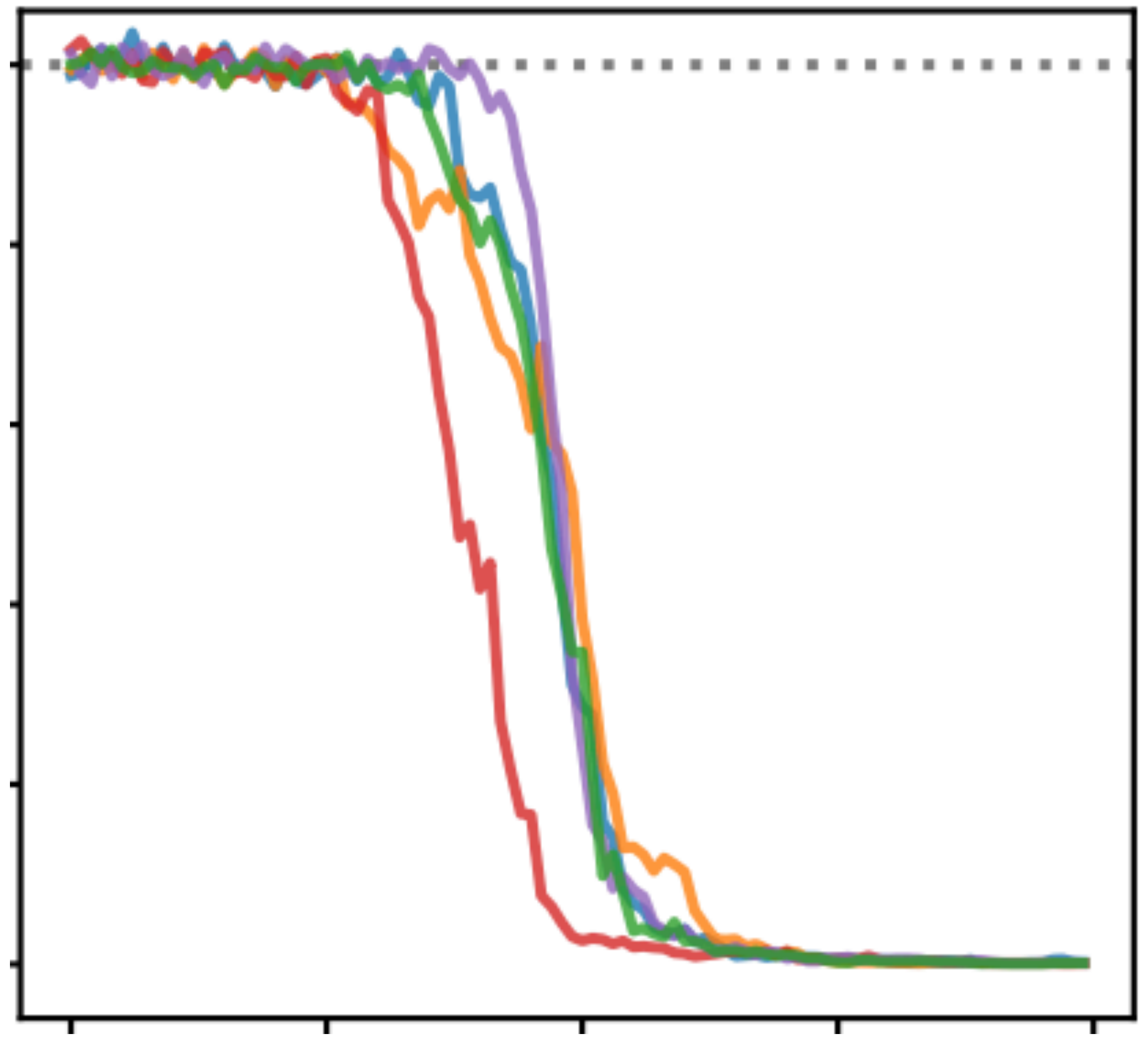
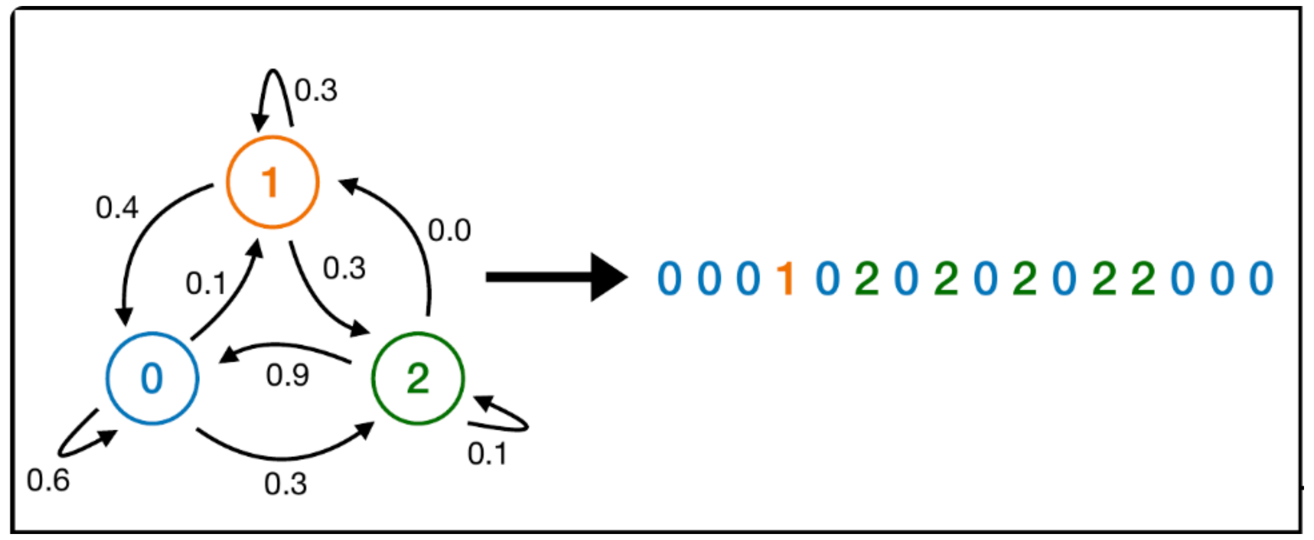
UNDERSTANDING TRAINING DYNAMICS IN DEEP LEARNING

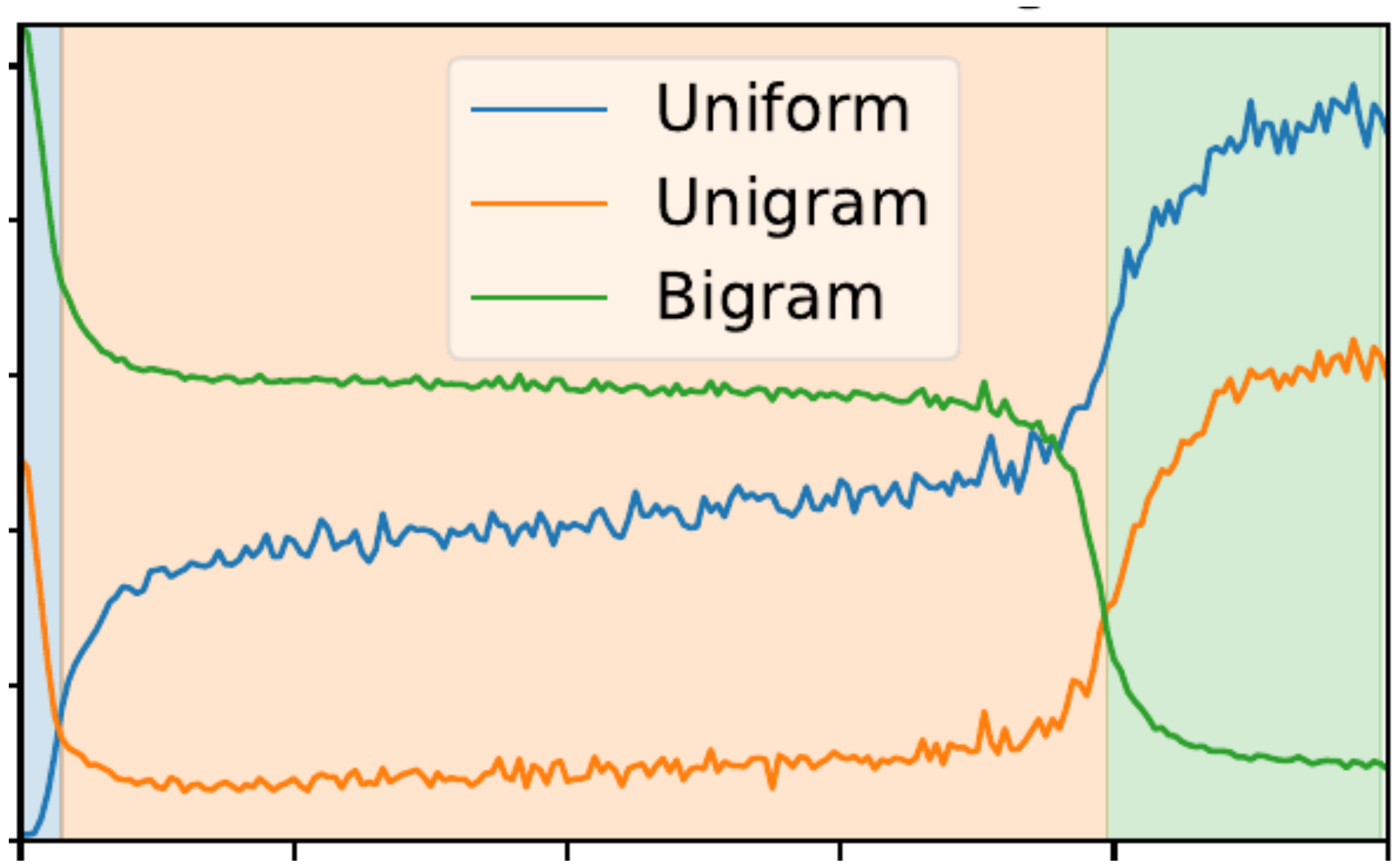
USING SIMPLIFIED MODELS



Surbhi Goel

University of Pennsylvania





DEEP LEARNING IS COOL SLIDE

SG How many vegan cheesesteaks are sold in Philly every day?

Language comprehension
World Knowledge
Reasoning

$$\min_{\theta \in \mathbb{R}^{10^{11}}} \ell(f_\theta; \text{lots of data})$$

$$\theta \leftarrow \theta - \eta \nabla \ell(\theta)$$

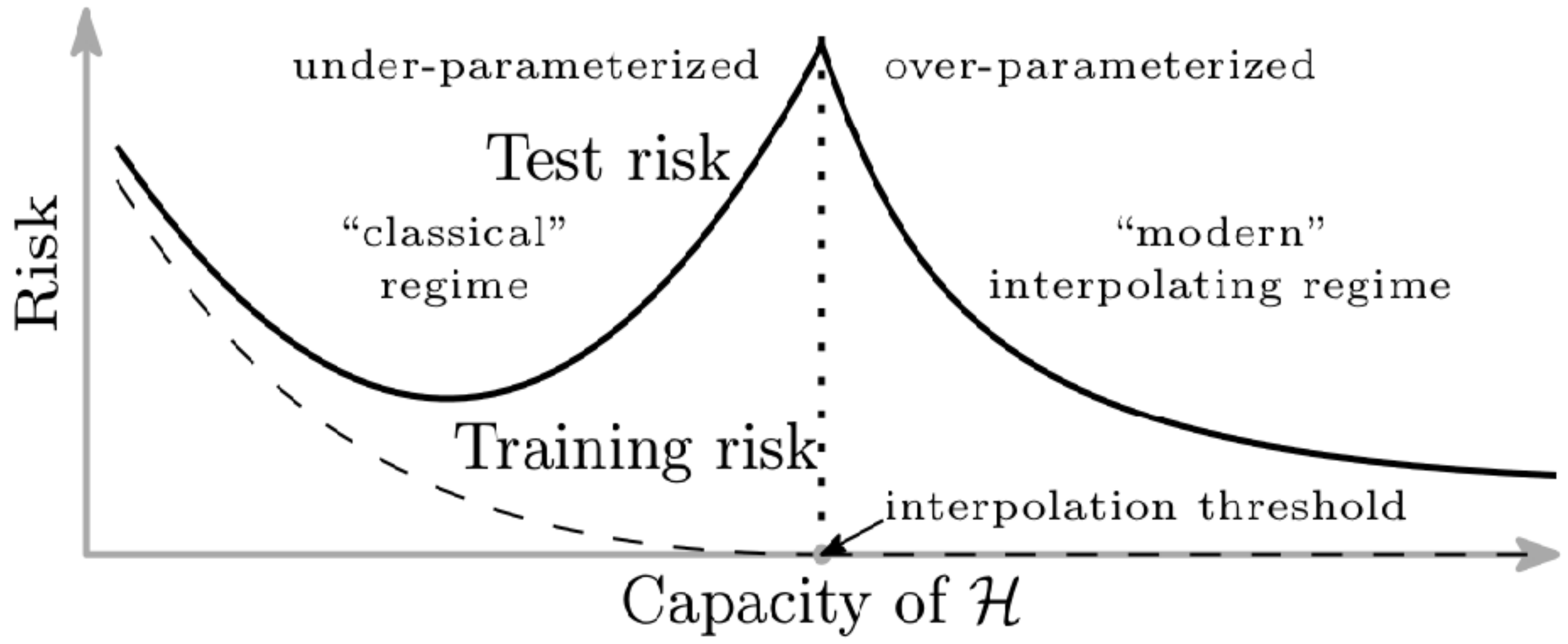
Unfortunately, I do not have specific data on the number of vegan cheesesteaks sold daily in Philadelphia. Vegan cheesesteaks are a newer and niche offering compared to the traditional cheesesteak made with beef and dairy cheese.

However, here are some points that can help provide an estimate:

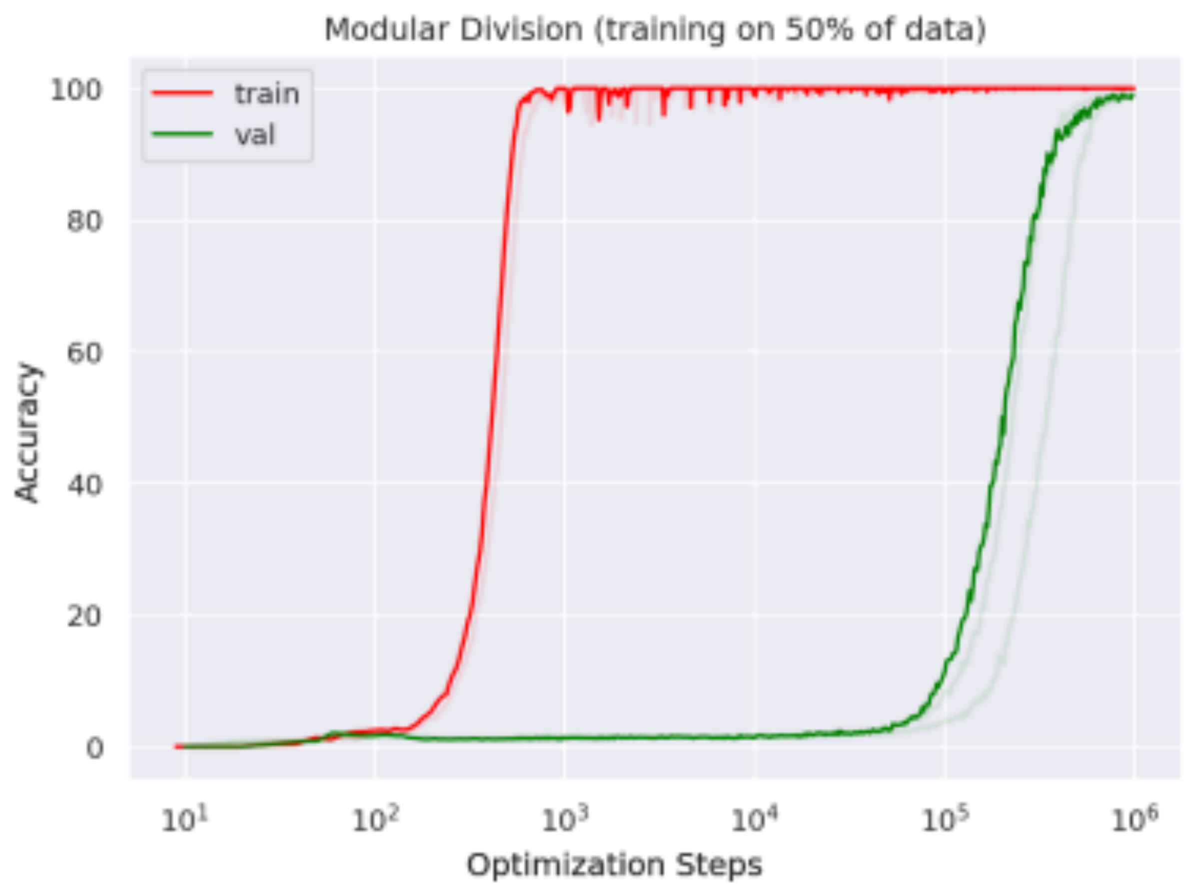
1. Philadelphia has a growing vegan/plant-based food scene to cater to that dietary preference.
2. Several popular cheesesteak shops like Blackbird Pizzeria, Hip City Veg, and Goldie now offer vegan cheesesteak options using seitan, mushrooms or other meat substitutes.
3. There are also some all-vegan cheesesteak shops like Dalesandro's Vegan and Blackbird Pizzeria that exclusively sell the vegan version.
4. According to some reports, vegan cheesesteaks make up around 5-10% of total cheesesteak sales at the shops that offer them.

Given the relatively small but increasing demand, my rough estimate would be that probably a few thousand vegan cheesesteaks are sold on any given day in Philadelphia currently. But this is just an educated guess - the actual number could be higher or lower based on the changing dietary trends in the city. Obtaining precise sales data is difficult without vendor inputs.

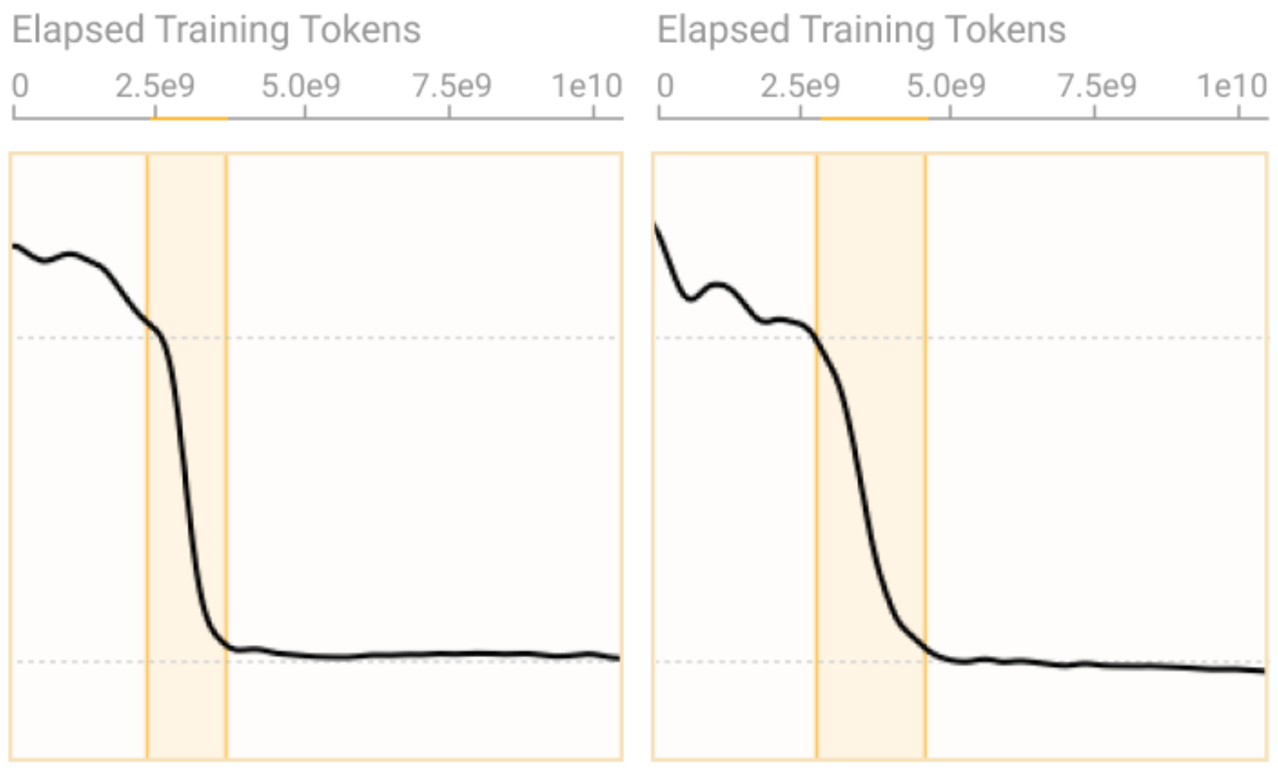
OPTIMIZATION BEHAVIORS ARE VERY INTRIGUING



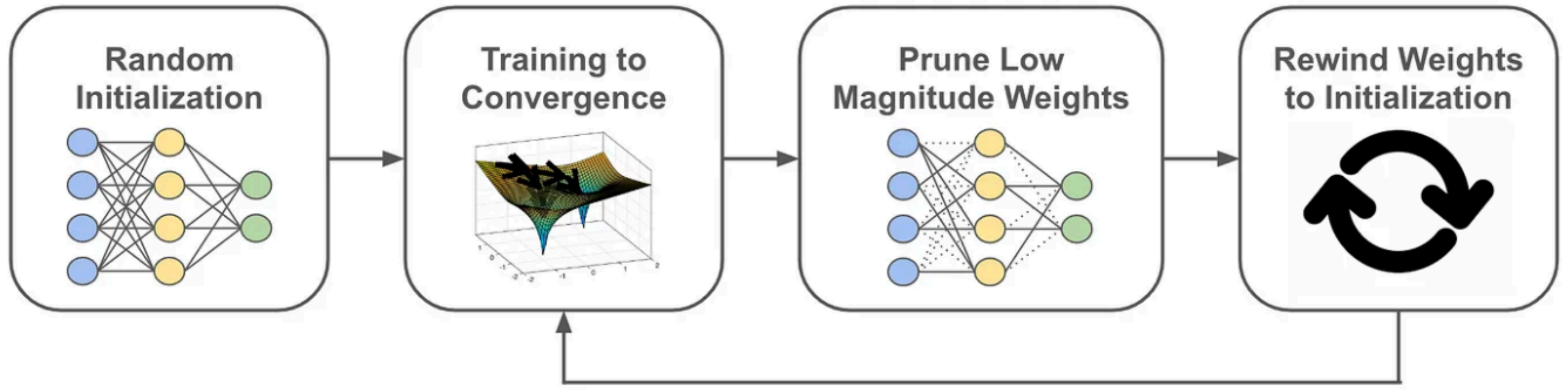
Double descent



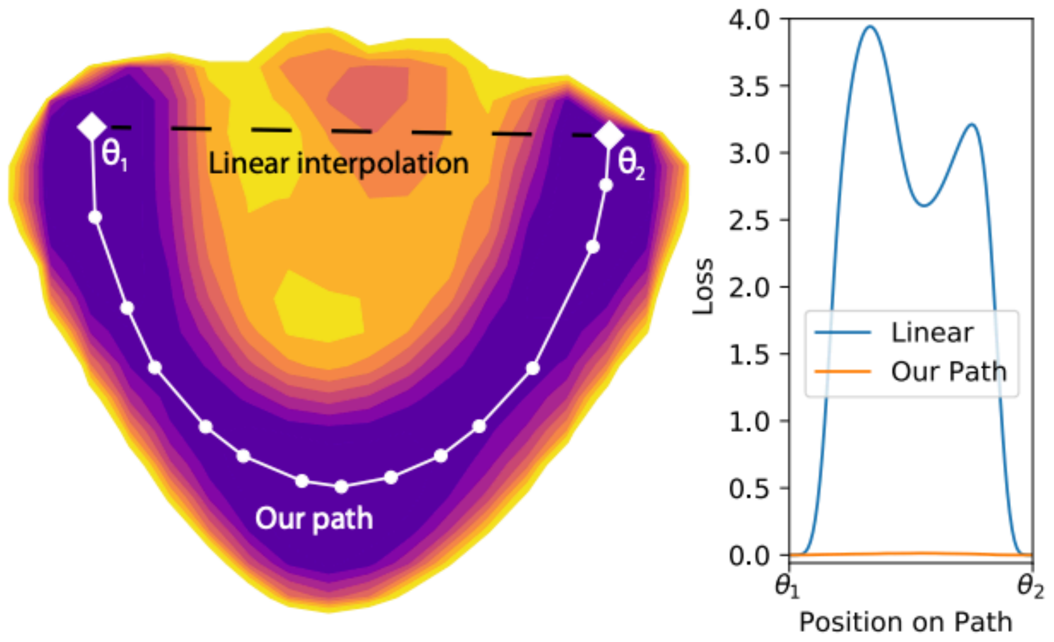
Grokking



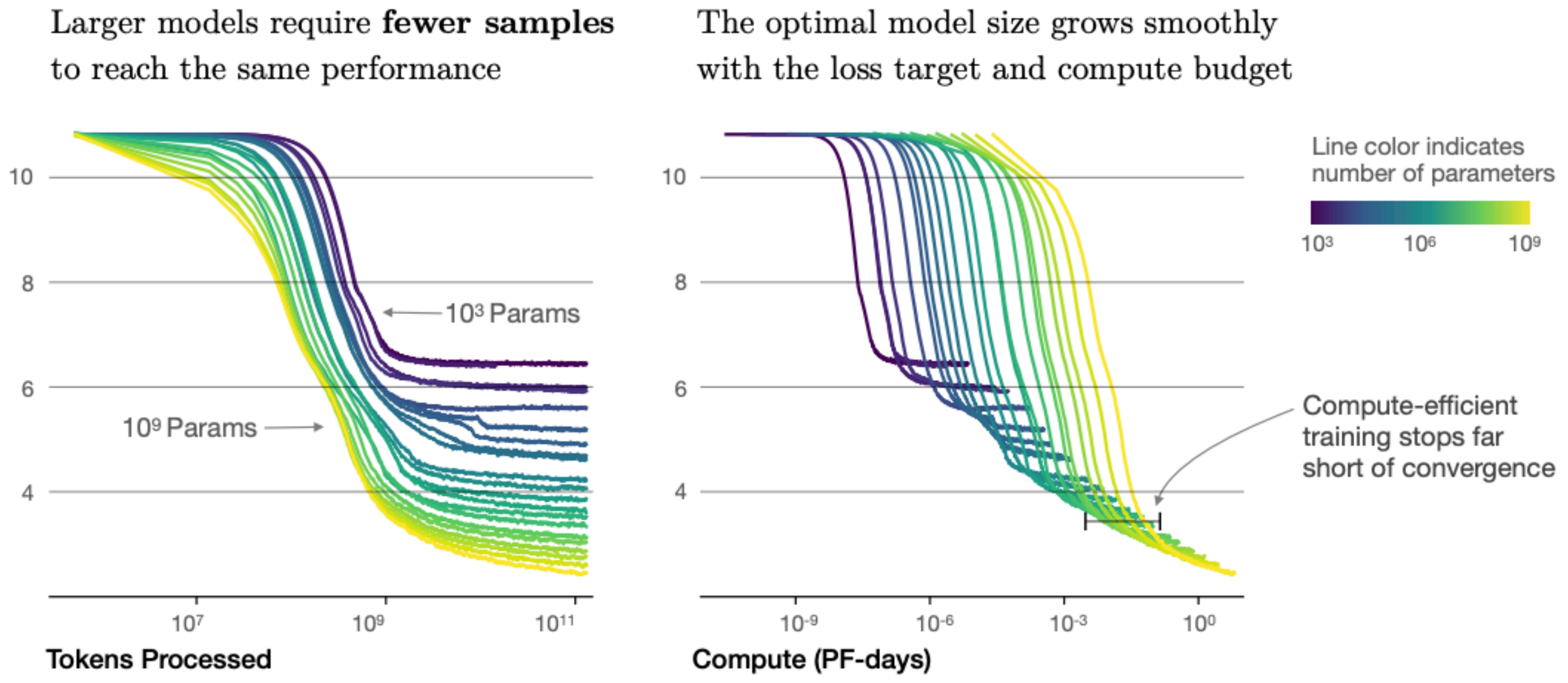
Phase Transitions



Lottery Tickets



Mode Connectivity



Scaling Laws

FEW THOUGHTS

This era of ML: Deep learning surmounts various computational challenges to produce impressive results that we did not expect

Theory: can provide guarantees, explanations, new algorithmic insights

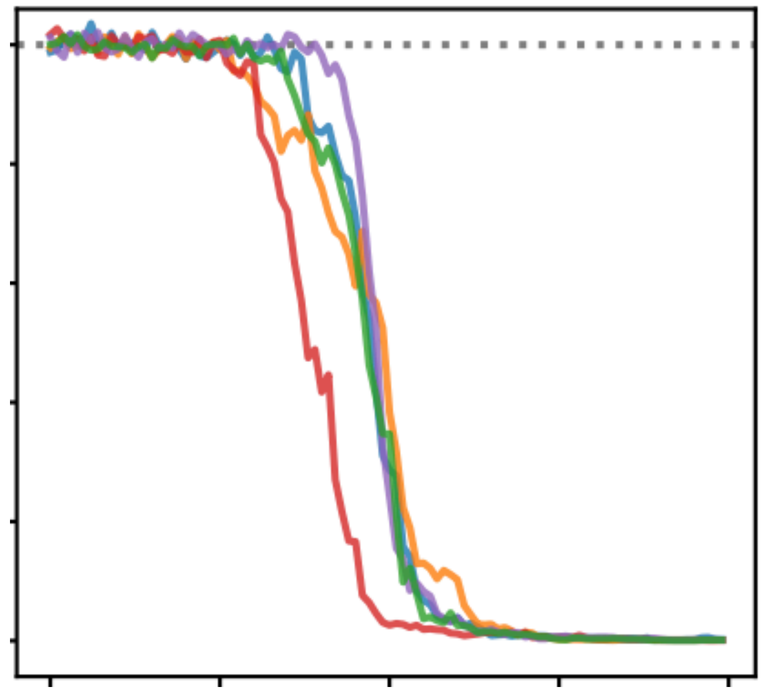
Challenge: There are numerous moving parts, everything affects everything, scale is often too large to tackle

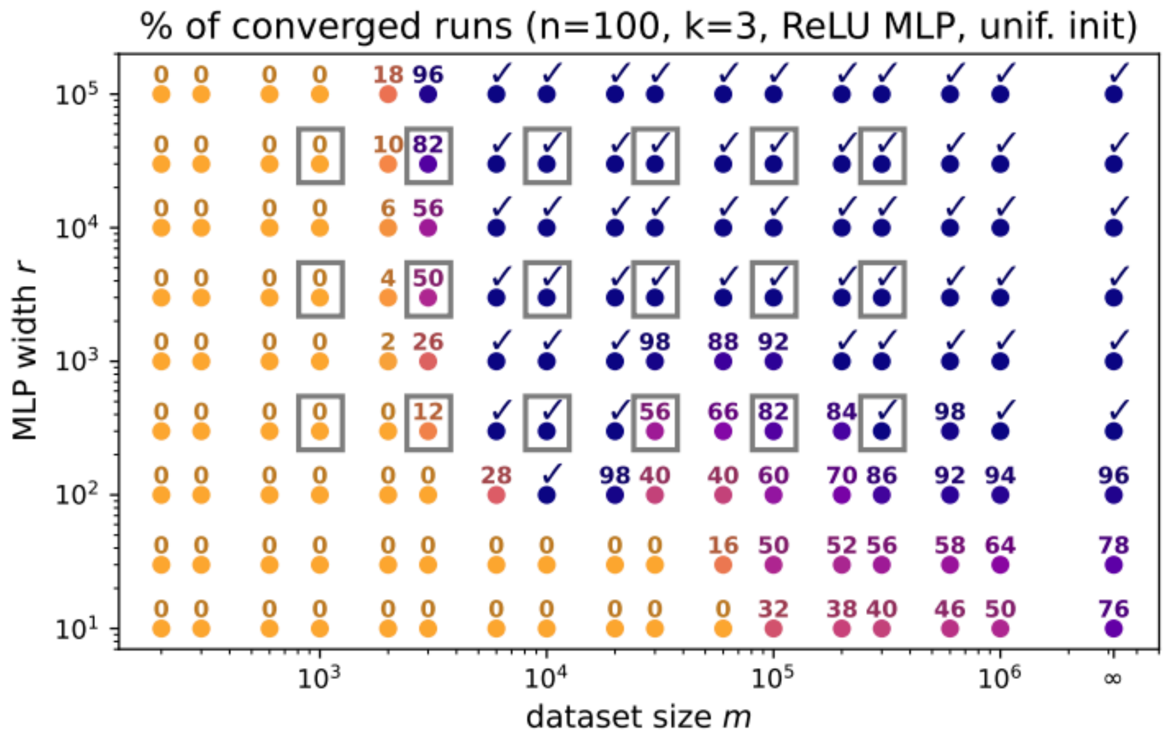
An approach: Create synthetic controllable setups that replicate the desirable learning behaviors and allow for new insights and analysis

TODAY: PARITIES AND MARKOV CHAINS

Sparse-parities and Feature Learning

with Boaz Barak, Ben Edelman, Sham Kakade, Eran Malach & Cyril Zhang

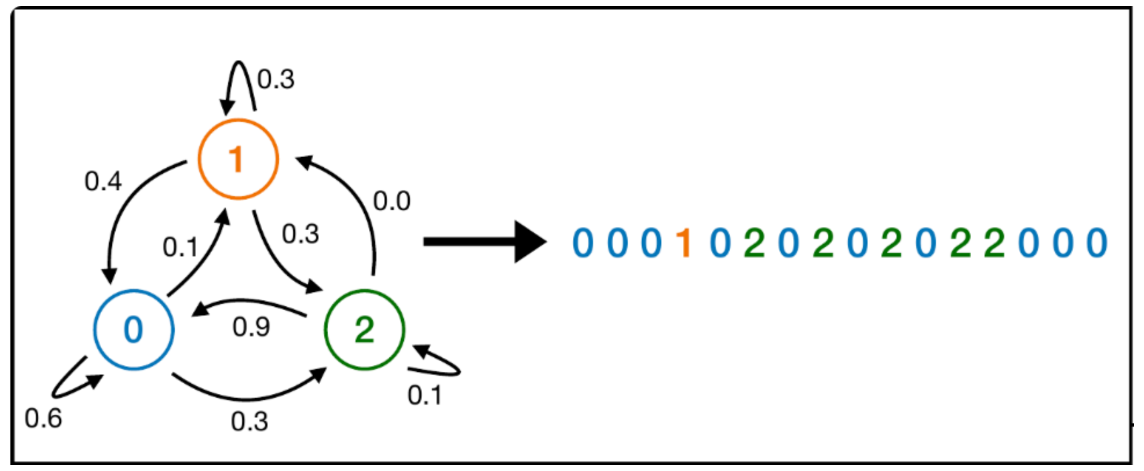


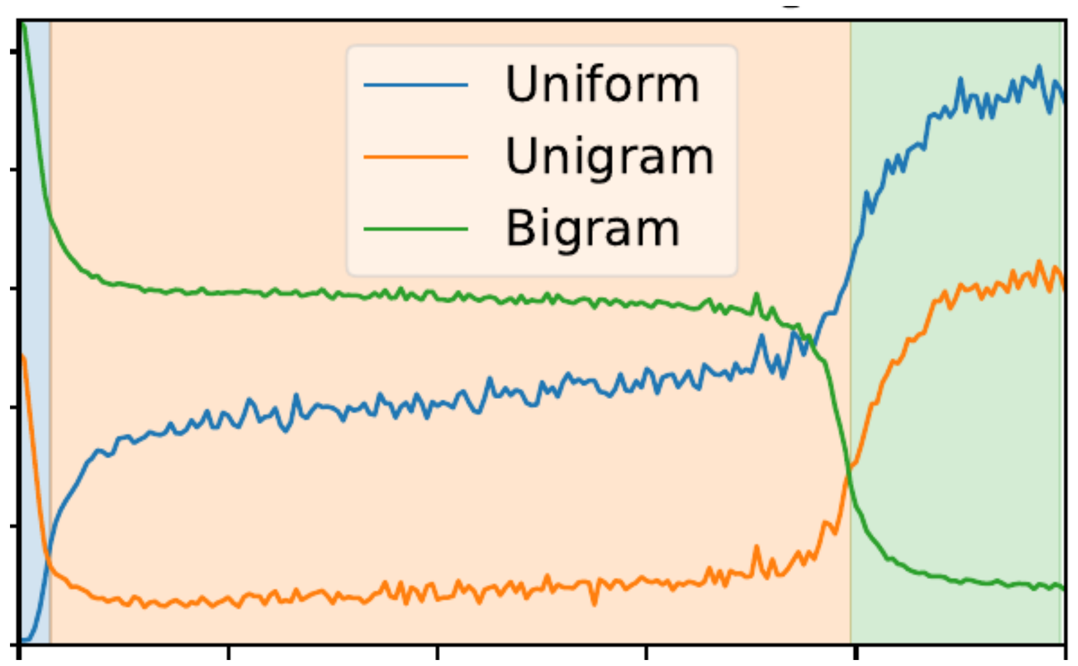


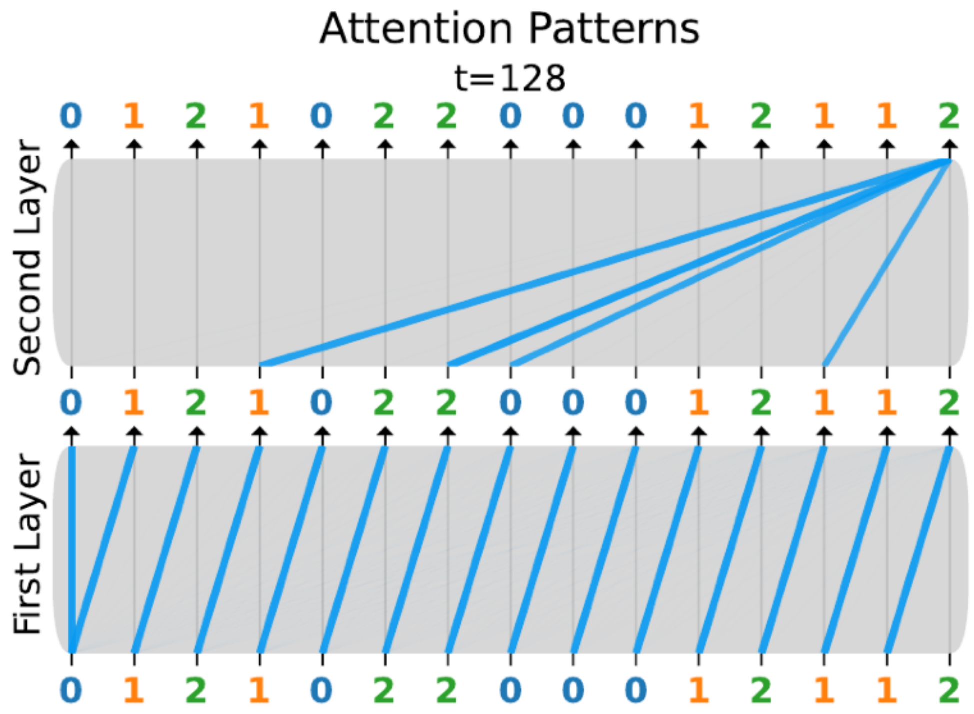
Slide credits shared with Cyril Zhang

Markov Chains and Induction Heads

with Ben Edelman, Ezra Edelman, Eran Malach & Nikos Tsilivis







Slide credits shared with Ben Edelman

LEARNING SPARSE PARITIES

Fundamental problem in learning theory:

Input: Dataset of m samples $\{(x^{(i)}, y^{(i)})\}_{i=1}^{m}$ with each $x^{(i)}$ i.i.d. from **Unif**$(\{\pm 1\}^n)$ and $y^{(i)} = \prod_{j \in S} x_j^{(i)}$ for some unknown set S of size k

Output: Subset S of relevant variables

$\chi_S(x) = \Pi_{i \in S} x_i$

k-way Boolean XOR

LEARNING SPARSE PARITIES - WHAT IS KNOWN

Statistical-computational trade-offs:

Statistically requires $\approx k \log n$ samples, brute force over all possible $\binom{n}{k}$ choices

Computationally beating $n^{O(k)}$ time is hard!

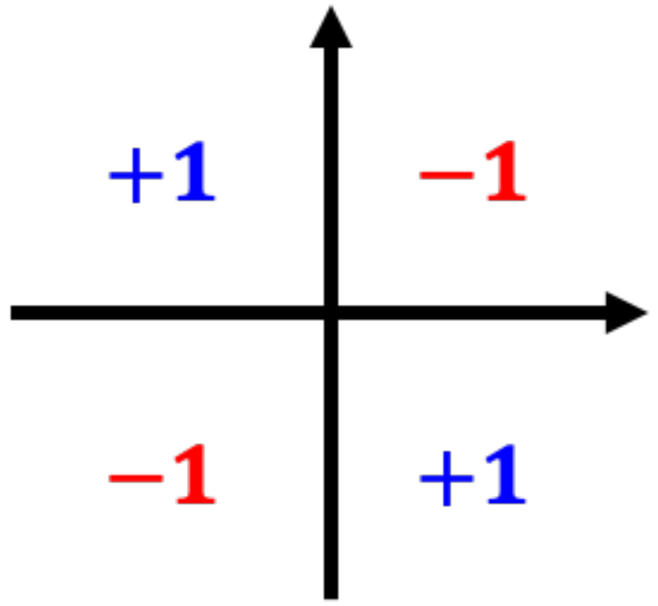
- Provably, in restricted computational models [Kearns '93; Kol, Raz, Tal '16]
- Conjecturally (with a constant noise), no $n^{o(k)}$ algorithm [Applebaum, Cash, Peikert, Sahai '09]

Lots of interesting, different algorithms!

- Noiseless: $O(n^3)$ time, needs $\Omega(n)$ samples [Gauss 1810]
- Noiseless: $\tilde{O}(n^{k/2})$ time [Spielman, via Klivans-Servedio '06]
- Noisy: $2^{O(n/\log n)}$ time & samples [Blum, Kalai, Wasserman '00]
- Noisy: $\tilde{O}(n^{0.8})$ time via Chebyshev polynomials [Valiant '13]

SPARSE PARITIES AS A PROXY MODEL

The XOR problem [Minsky-Papert '69] convinced everyone to abandon deep learning



Perceptron could not fit this

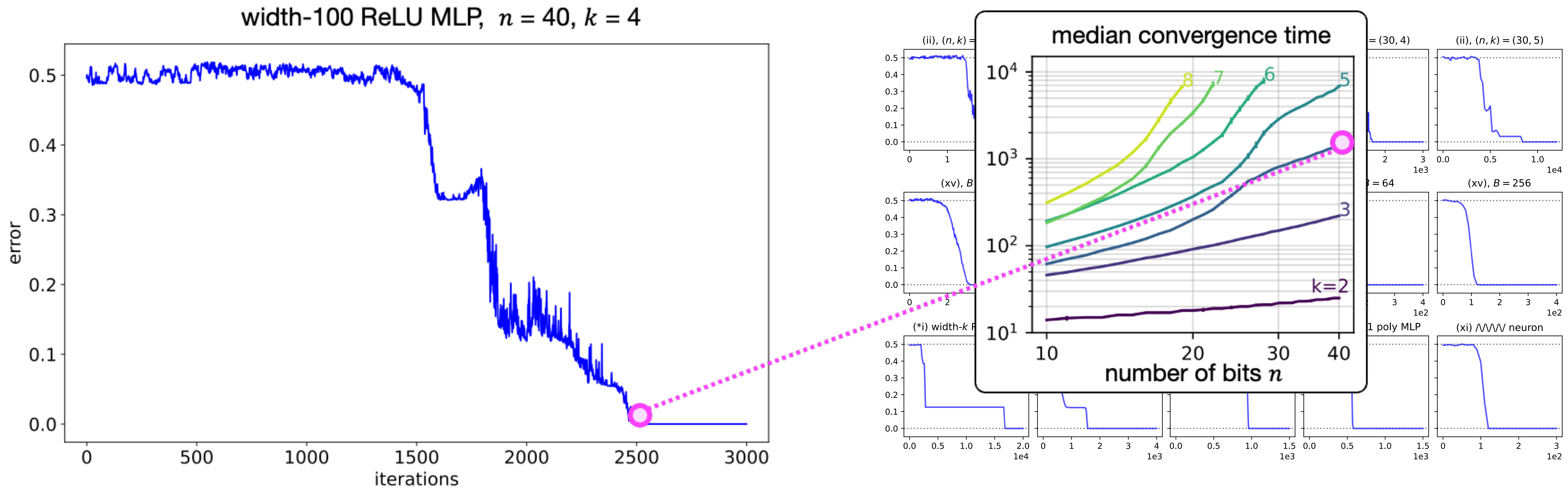
More expressive networks could easily fit

Recently gained interest experimentally [Daniely-Malach'20] and theoretically [Ben Arous-Gheissari-Jagannath '20]

Similar problem of learning single-index and multi-index models studied over gaussian input [Damian-Lee-Soltanolkatabi'22; Abbe-Boix-Misiakiewicz'23; Moniri-Lee-Hassani-Dobriman'23, ….]

LEARNING SPARSE PARITIES WITH NEURAL NETS

Can neural networks learn sparse parities?



Many different architectures learn with $\approx n^k$ time/samples

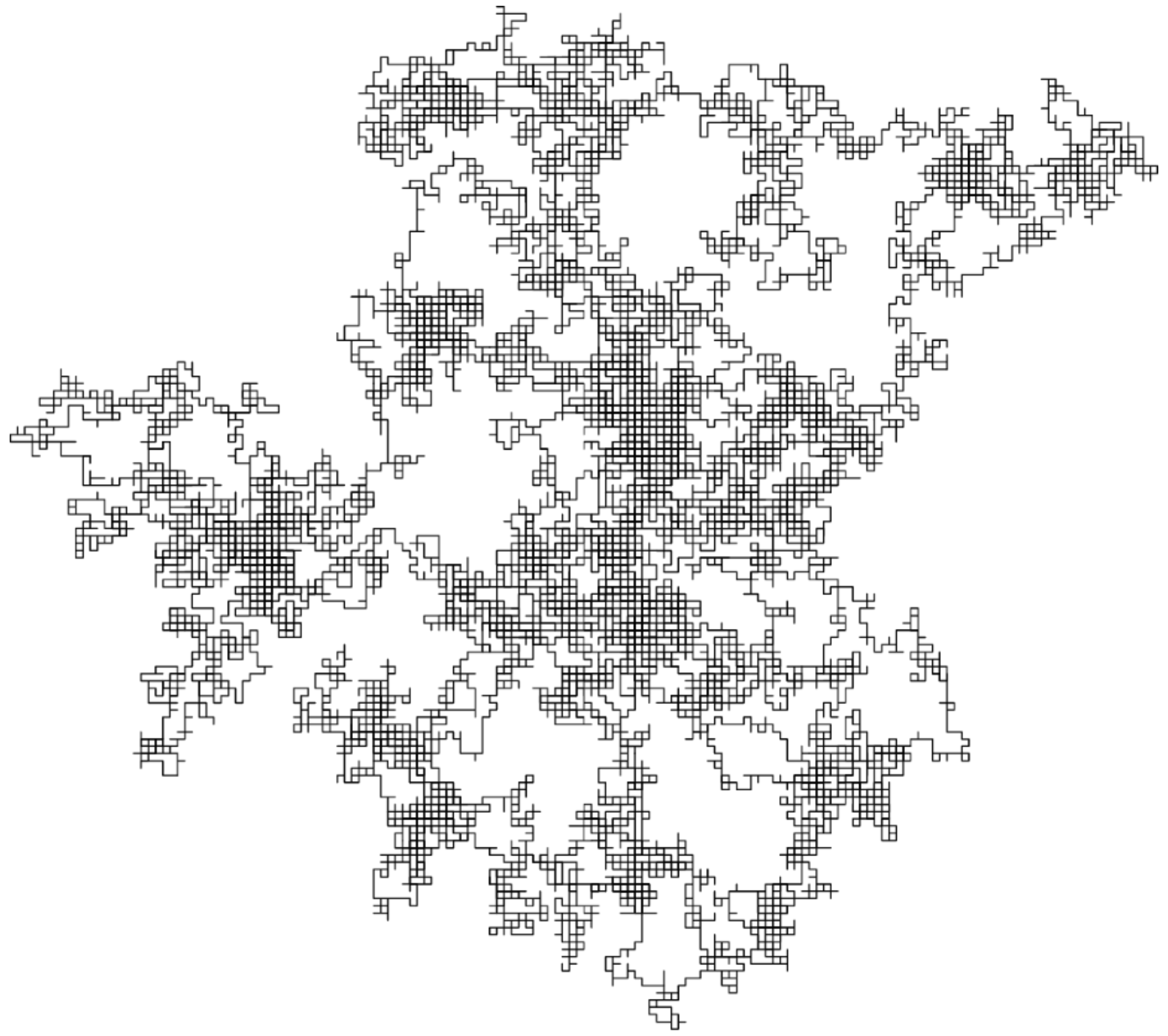
2-layer MLPs: $f_\theta(x) = v^\top \sigma(Wx + b)$
many nonlinearities σ: ReLU, x^k, …
deeper MLPs, Transformers, *PolyNets*

wide MLPs: $W \in \mathbb{R}^{1000000 \times n}$
thin MLPs: $W \in \mathbb{R}^{k \times n}$
single neuron: $f_\theta(x) = \sin(w^\top x)$

Barak, Edelman, Goel, Kakade, Malach, Zhang. Hidden Progress in Deep Learning: SGD Learns Parities Near the Computational Limit. NeurIPS 2022.

COMPETING REASONS FOR SUCCESS OF TRAINING

How are the models learning this challenging sparse function?



Random guessing?

- "*Stumbling in the dark*" until SGD guesses S
- $\approx n^{-k}$ chance every $O(1)$ iterations
- Plausible theory: langevin-dynamics

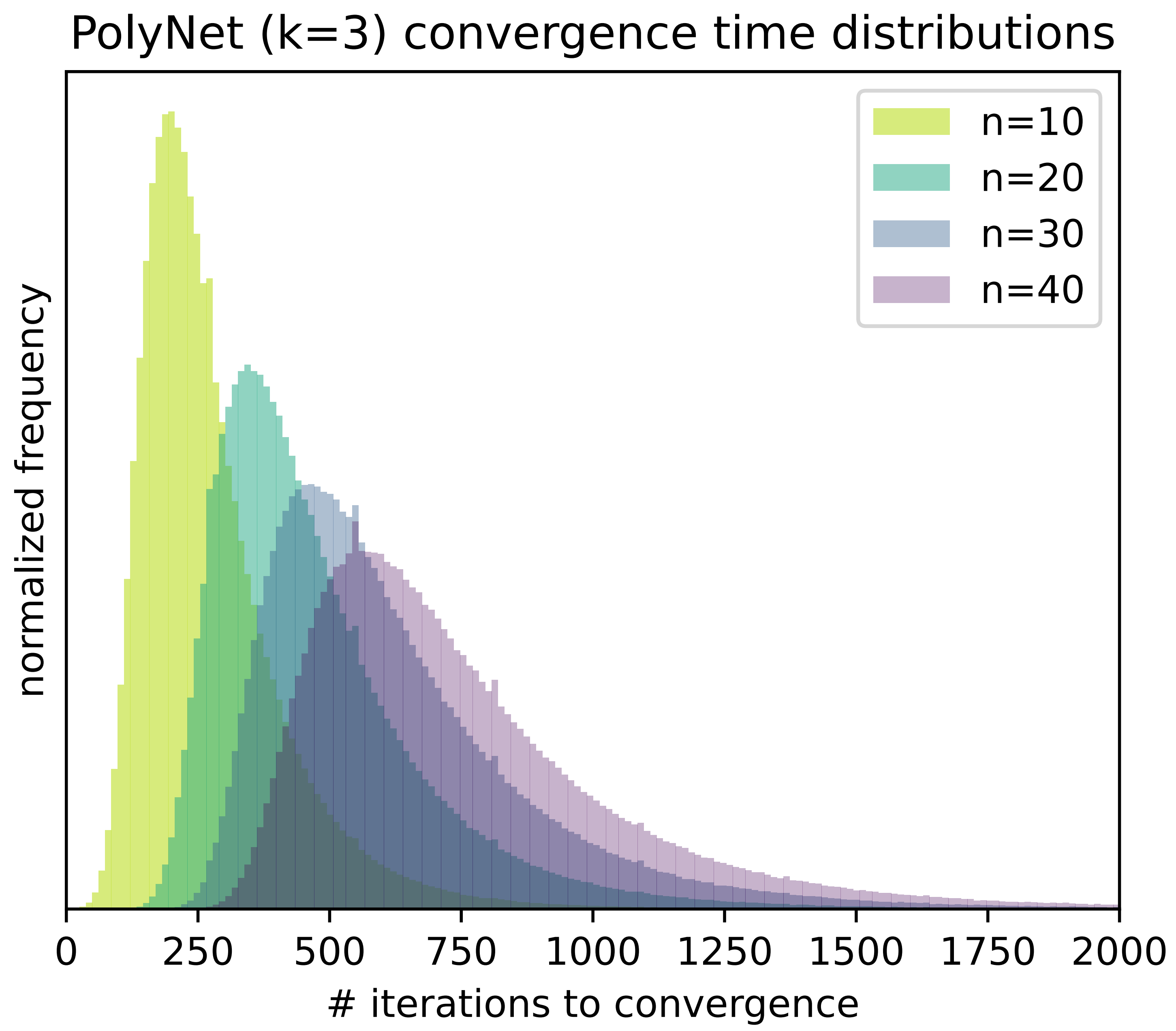
Hidden progress?

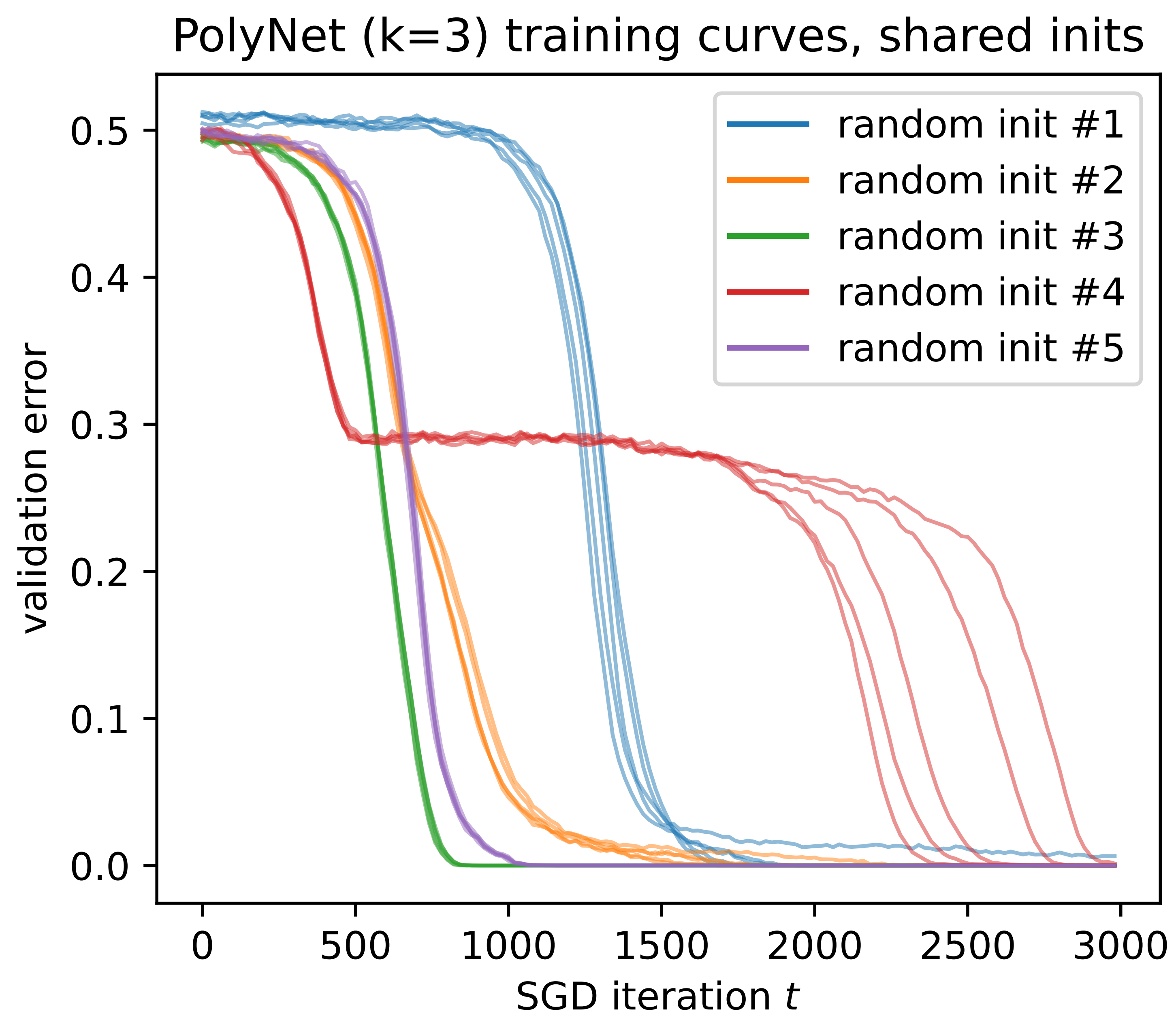
- Loss looks flat, but another quantity doesn't
- Some function $\Phi(\theta_t)$ is predictive of t_{success}
- Plausible theory: ?

Barak, Edelman, Goel, Kakade, Malach, Zhang. Hidden Progress in Deep Learning: SGD Learns Parities Near the Computational Limit. NeurIPS 2022.

MECHANISM BEHIND SUCCESS OF TRAINING

Can this be random search?





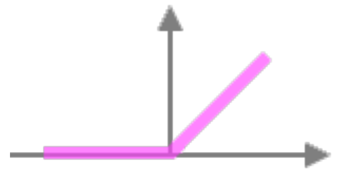
Random search would look like an exponential distribution

$$P(i) \propto (1-p)^{i-1}p \text{ for } 1/p = \binom{n}{k}$$

Convergence times would depend on SGD's stochasticity and not purely initialization

Barak, Edelman, Goel, Kakade, Malach, Zhang. Hidden Progress in Deep Learning: SGD Learns Parities Near the Computational Limit. NeurIPS 2022.

WHERE IS THE HIDDEN PROGRESS?



Assume: $f_w(x) = \text{ReLU}(w^\top x)$ with correlation loss $\ell(y, \hat{y}) = -y\hat{y}$, and exact GD

Claim: In one step, GD from $w = [\pm 1, \ldots, \pm 1]$ learns all the features

Proof sketch:

linear threshold function (LTF)

Population gradient $\nabla_w \mathbb{E}\left[\ell\left(\chi_S(x), \text{ReLU}(w^\top x)\right)\right] = -\mathbb{E}\left[\chi_S(x) \cdot x \cdot \text{ReLU}'(w^\top x)\right]$

parities

Boolean Fourier coefficients
[Titsworth '62; O'Donnell '14]

At initialization: $\text{ReLU}'(w^\top x) = \dfrac{\text{sign}(\pm 1^\top x) + 1}{2}$ *(shifted majority function)*

relevant features S | **irrelevant features** $[n] \setminus S$

$$= -\frac{1}{2} \cdot \Big[\underbrace{\widehat{\text{Maj}}_{S\setminus\{1\}} \cdots \widehat{\text{Maj}}_{S\setminus\{k\}}}_{|\text{level-}(k-1)\text{ coeffs}| \gtrsim n^{-\frac{k-1}{2}}} \;\Big|\; \underbrace{\widehat{\text{Maj}}_{S\cup\{k+1\}} \cdots \widehat{\text{Maj}}_{S\cup\{n\}} \quad \widehat{\text{Maj}}_{S\cup\{n\}}}_{n^{-\frac{k+1}{2}} \gtrsim |\text{level-}(k+1)\text{ coeffs}|}\Big] + \frac{1}{2} \cdot \mathbf{1}$$

Fourier gap

Key: Gradient on relevant coordinates is $\Omega(n^{-(k-1)/2})$ larger than the irrelevant coordinates

Barak, Edelman, Goel, Kakade, Malach, Zhang. Hidden Progress in Deep Learning: SGD Learns Parities Near the Computational Limit. NeurIPS 2022.

MAIN RESULT - HIDDEN INFORMATION

Theorem [BEGKMZ'22]:

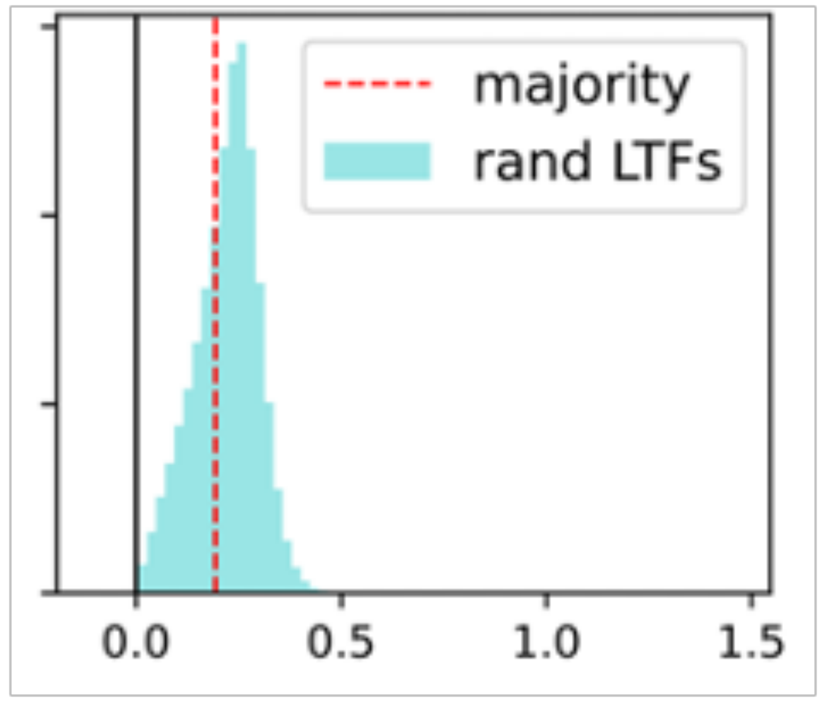
One hidden-layer MLPs with ReLU activation and $2^{O(k)}$ hidden units learn k-sparse parities using large batch SGD with compute time (batch-size x run-time) scaling as n^k.

For any Fourier gap γ, $\approx 1/\gamma^2$ samples suffice.

NTK requires at least $n^{\Omega(k)}$ hidden units

First gradient step has enough ***information*** to identify relevant coordinates, then online convex optimization works

Empirically, many variants work: varying batch size, noise, offline data, deeper networks, losses, sinusoidal activations, initializations



Hard to do step-by-step analysis, Fourier gap unknown for random halfspaces

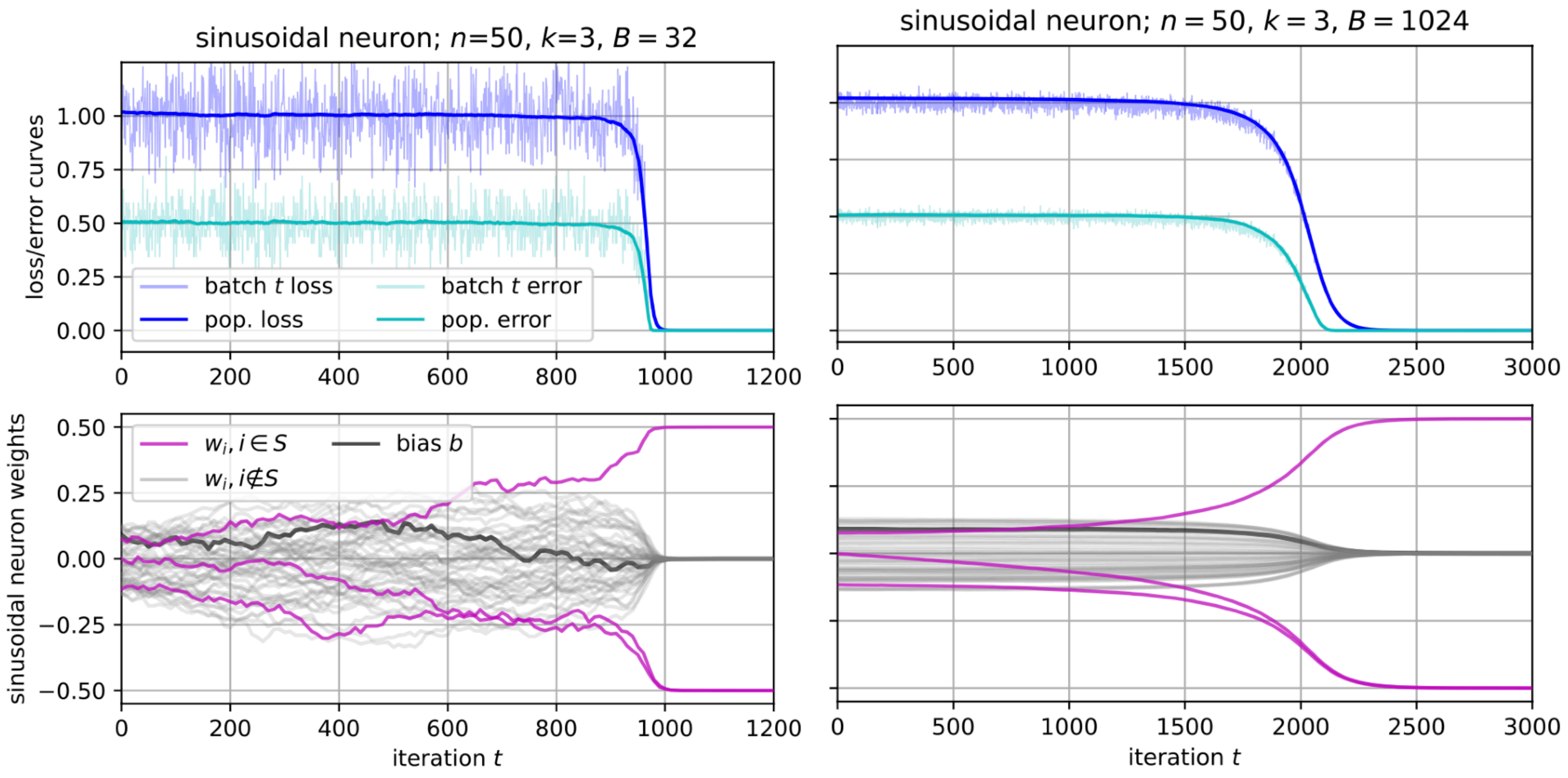
Barak, Edelman, Goel, Kakade, Malach, Zhang. Hidden Progress in Deep Learning: SGD Learns Parities Near the Computational Limit. NeurIPS 2022.

MECHANISM BEHIND SUCCESS OF TRAINING

Hypothesis: SGD learns parities via Fourier gap amplification mechanism

- why does it never succeed significantly earlier? needs $1/\gamma^2$ samples
- why does its trajectory depend heavily on initialization? gap depends on initialization

Hidden progress measures:



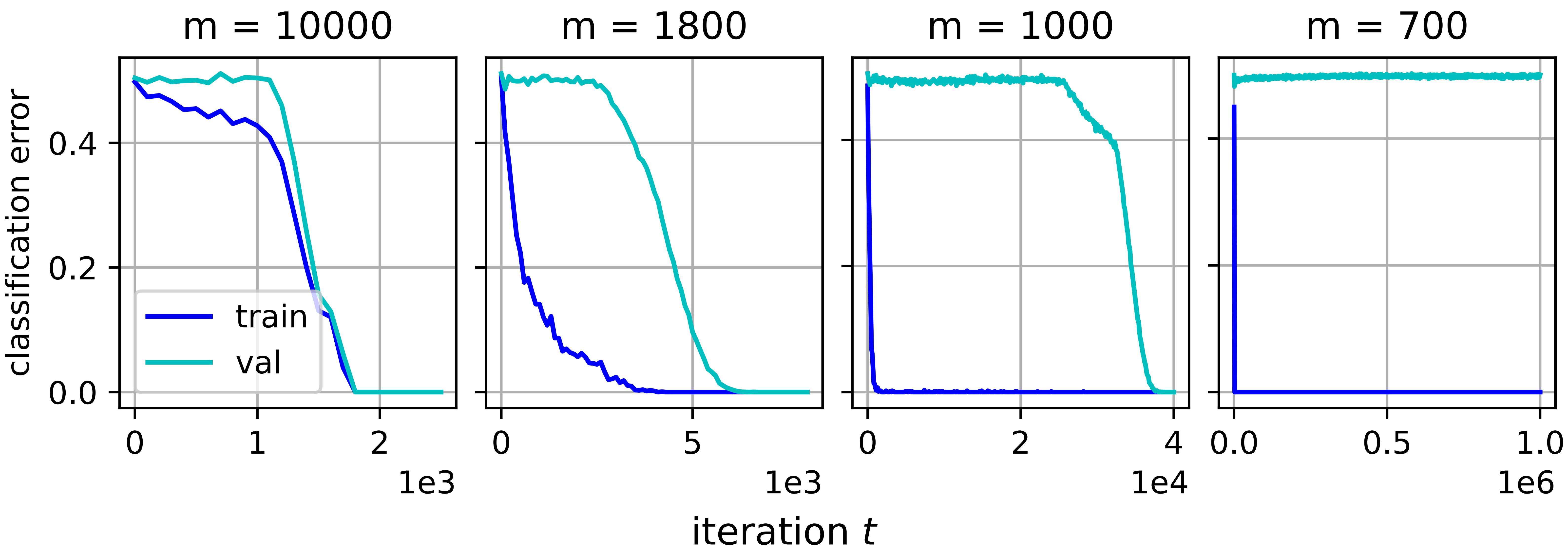
PROGRESS MEASURES FOR GROKKING VIA MECHANISTIC INTERPRETABILITY

Neel Nanda*, † Lawrence Chan‡ Tom Lieberum† Jess Smith† Jacob Steinhardt‡

Active area of research

Barak, Edelman, Goel, Kakade, Malach, Zhang. Hidden Progress in Deep Learning: SGD Learns Parities Near the Computational Limit. NeurIPS 2022.

SPARSE PARITIES: GROKKING



Grokking behavior when trained on fixed samples

Training loss goes to 0, validation loss hits 0 much later

Barak, Edelman, Goel, Kakade, Malach, Zhang. Hidden Progress in Deep Learning: SGD Learns Parities Near the Computational Limit. NeurIPS 2022.

SPARSE PARITIES: SCALING LAWS

Scaling Laws for Neural Language Models

Scaling laws: predict how test performance depends on compute and data

$$\text{loss} \approx \frac{1}{\text{data}^{\alpha}} + \frac{1}{\text{compute}^{\beta}} + \gamma$$

How can we trade data and compute resources?

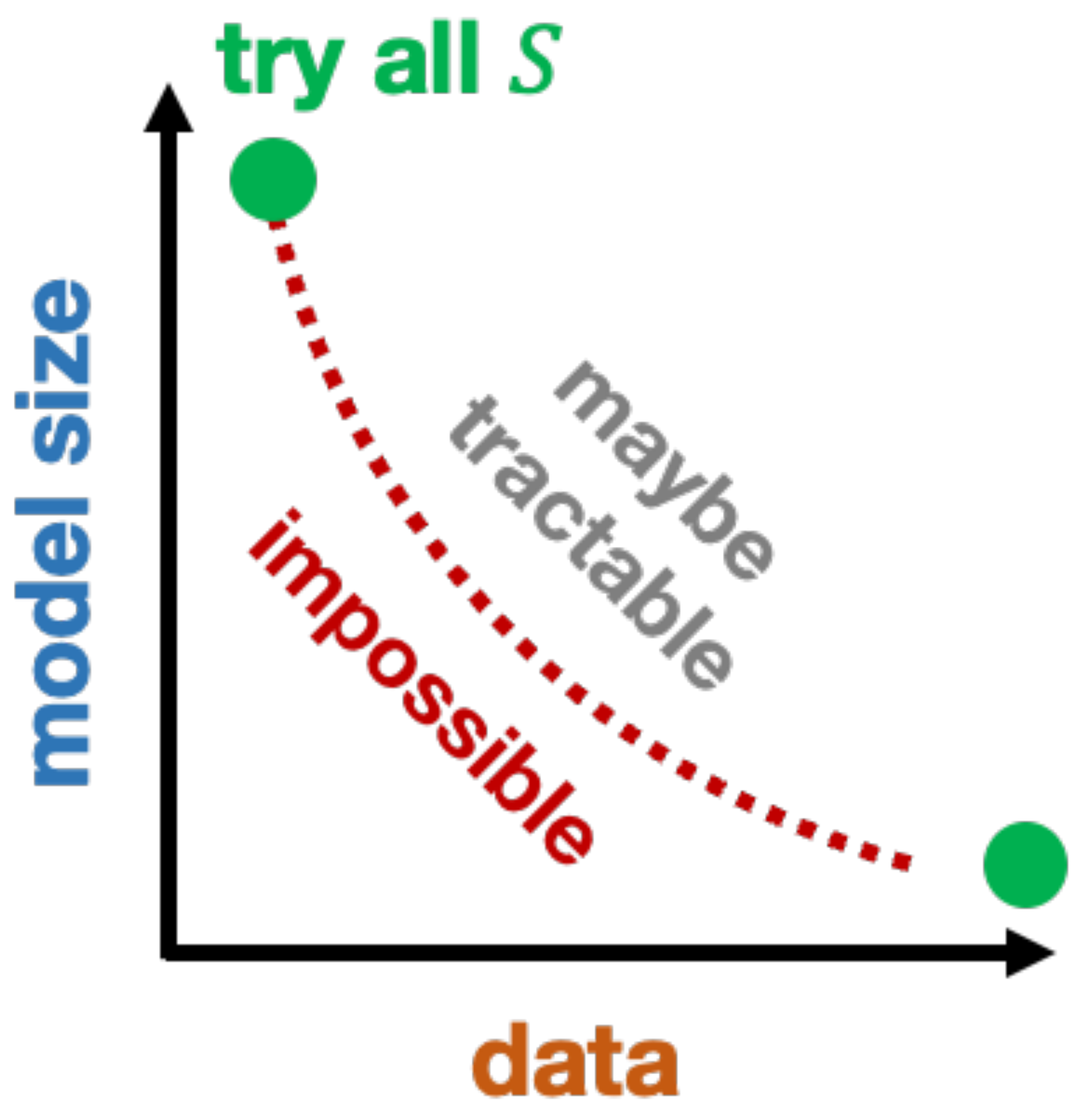
Why are parities hard?

Observe that $\mathbb{E}[\chi_S(x)\chi_T(x)] = 0$ for all $S \neq T$

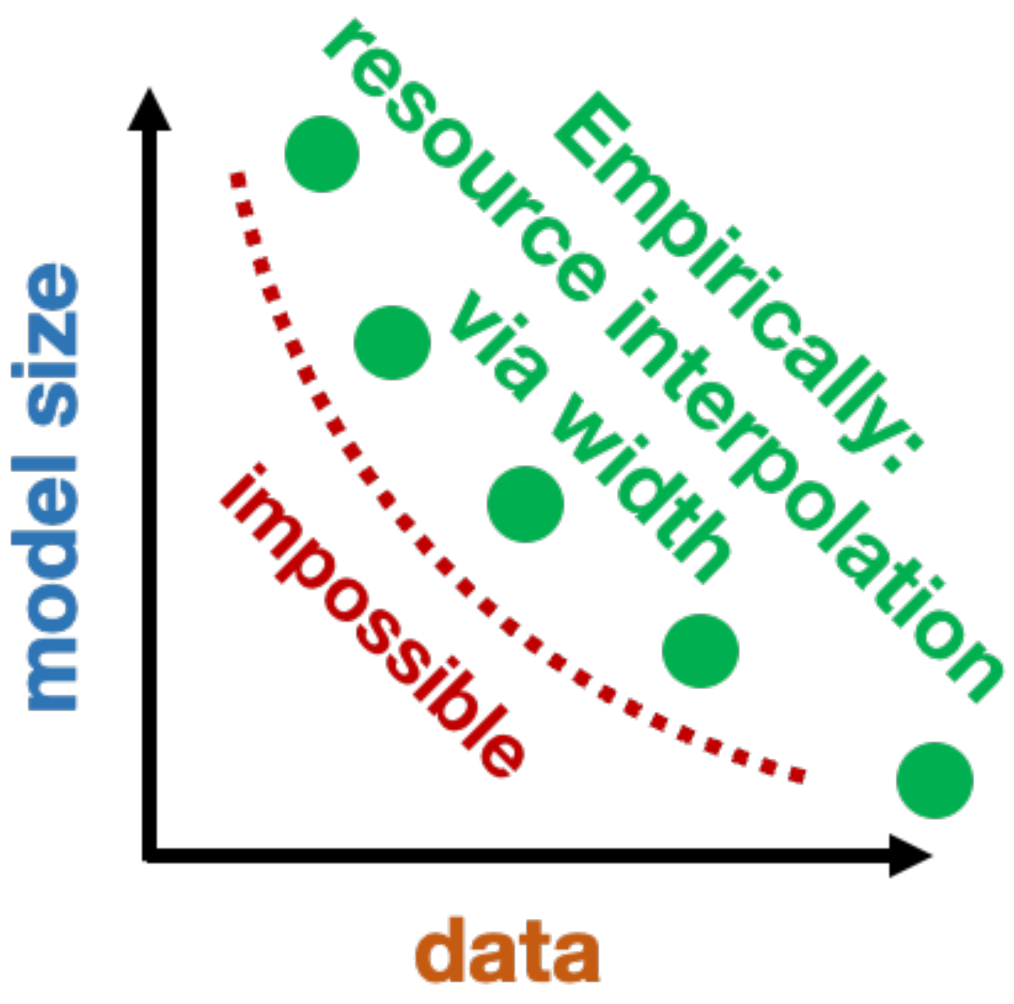
No other subset has any correlation

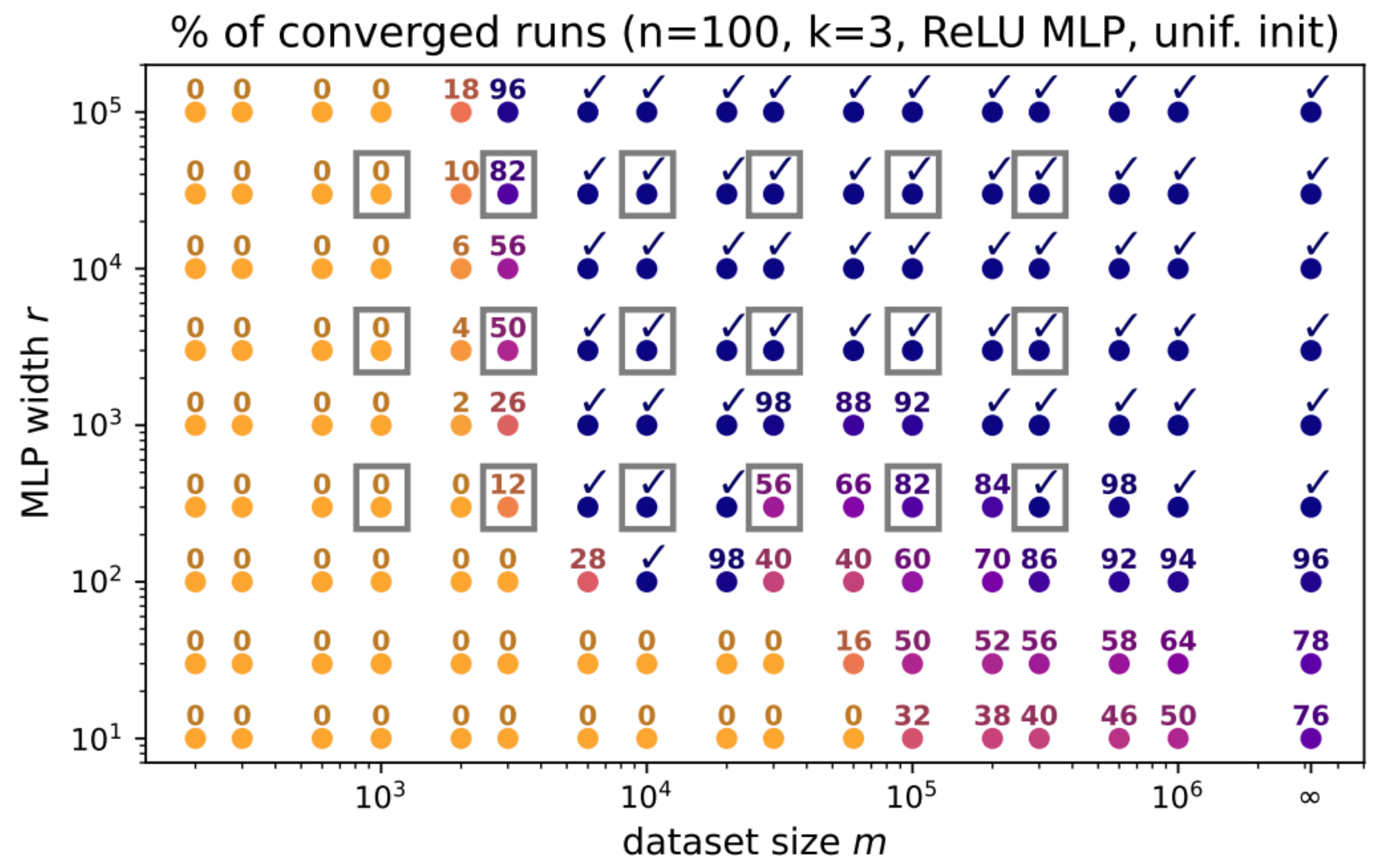
Therefore we need **data** $\times$ **compute** $\geq \binom{n}{k}$ to identify which parity it is

*with constant T
& success probability δ

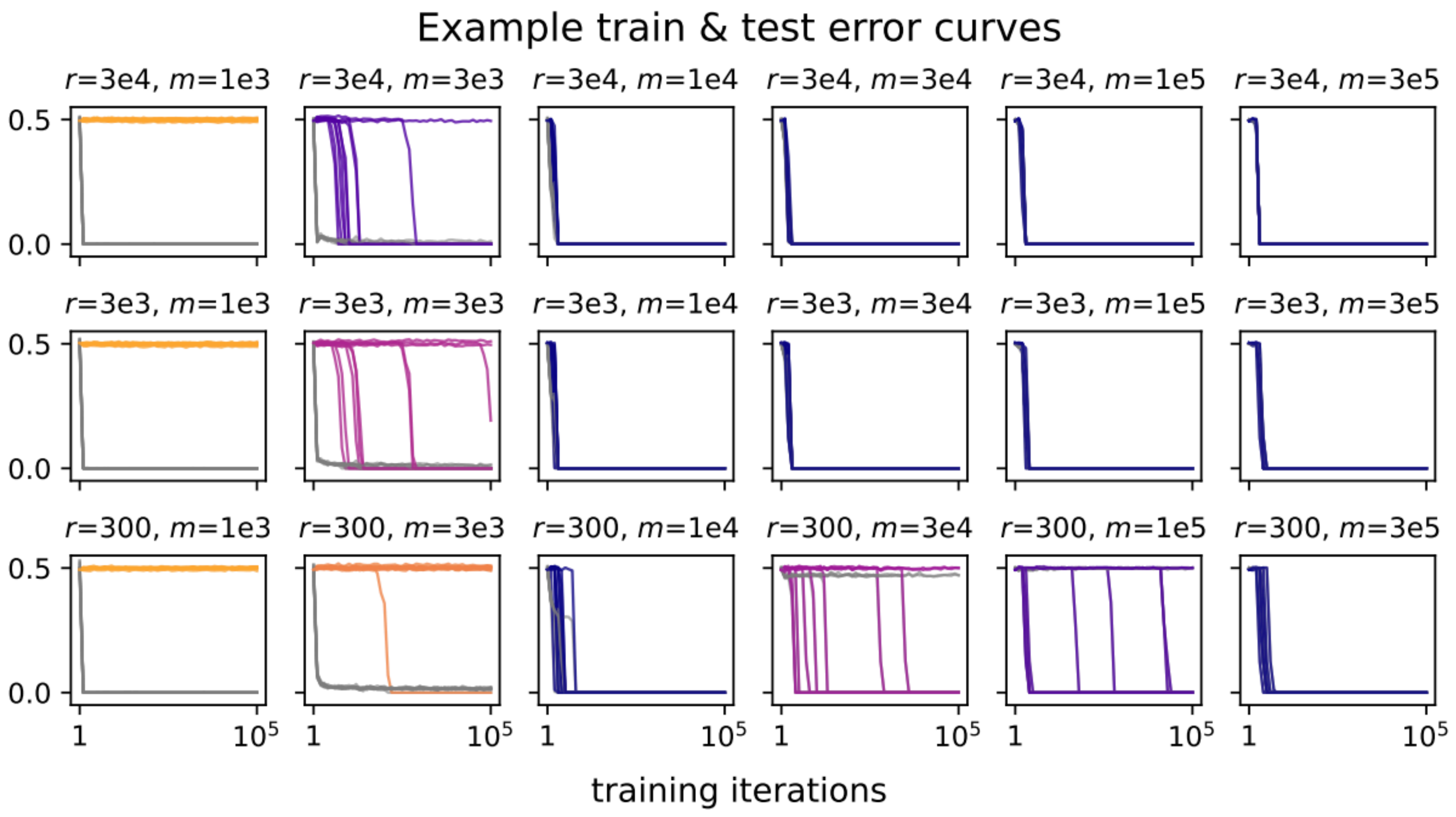


SPARSE PARITIES: SCALING LAWS





Darker is better



SGD training interpolates between random guessing and Fourier gap amplification

Why would this work?

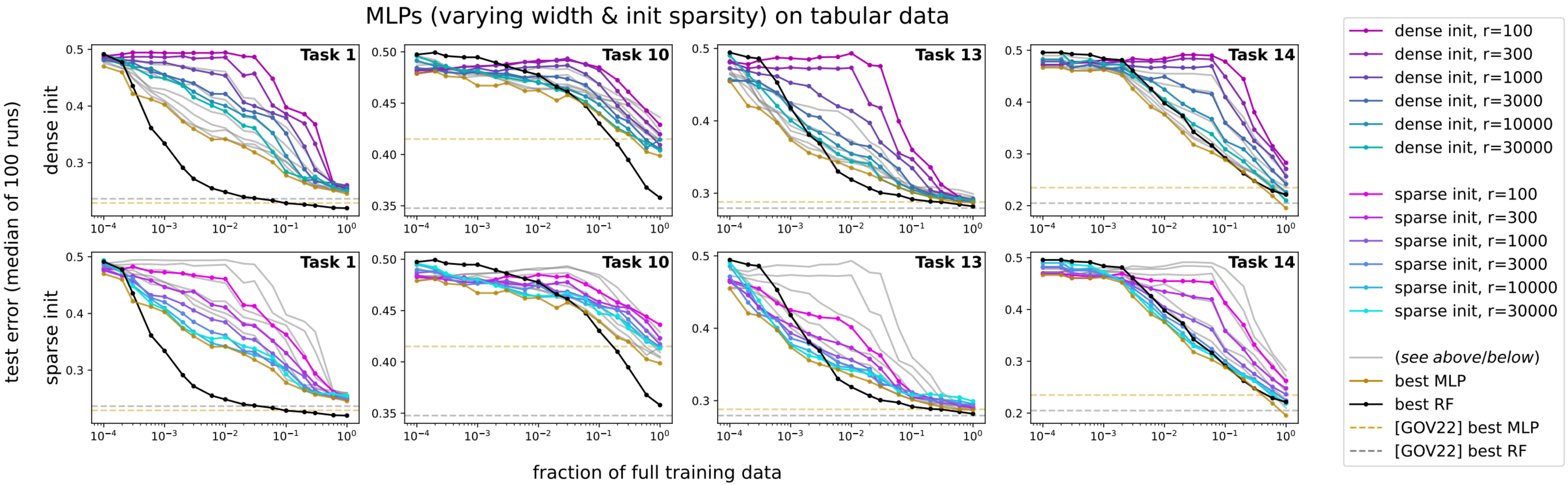
Assume: Each ReLU is sparsely initialized with some sparsity k'

Claim: As width increases, more chance to get a subset that overlaps with the relevant variables $\Longrightarrow$ lottery tickets with "partial progress" (higher Fourier gap)

Edelman, Goel, Kakade, Malach, Zhang. Pareto Frontiers in Neural Feature Learning: Data, Compute, Width, and Luck. NeurIPS 2023 (Spotlight).

SPARSE PARITIES: SPURIOUS CORRELATIONS

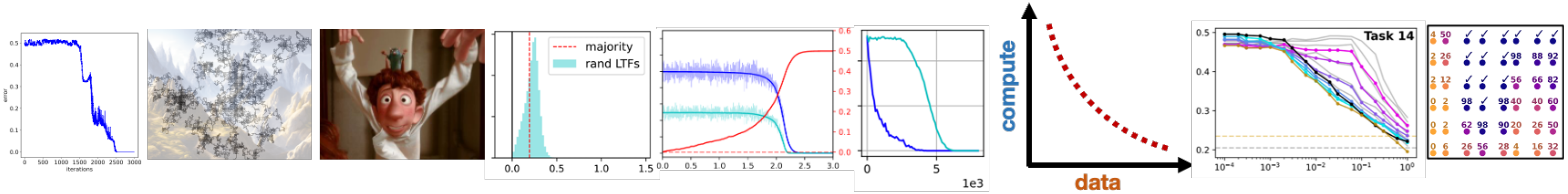
Wider MLPs are more sample efficient on low-data benchmarks, as predicted by theory!



Sparse initialization helps, but is not necessary!

Edelman, Goel, Kakade, Malach, Zhang. Pareto Frontiers in Neural Feature Learning: Data, Compute, Width, and Luck. NeurIPS 2023 (Spotlight).

SPARSE PARITIES AS A PROXY MODEL



Useful for studying several phenomenon, and a good model to simulate feature learning

Some other use cases:

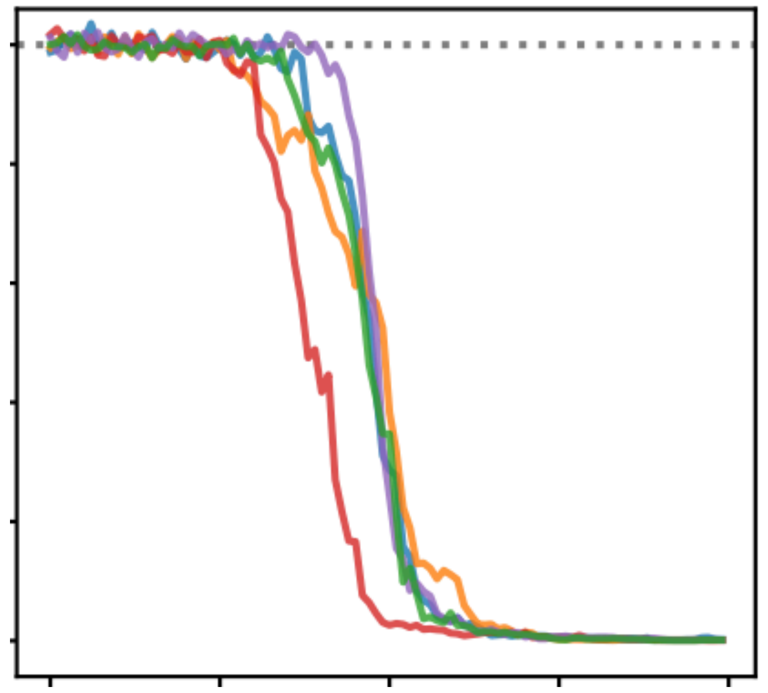
- Parity computations are essential building blocks for several reasoning problems [Liu-Ash-Goel-Kakade-Zhang'22]
- Parities are useful to model spurious/core features to understand robust learning [Qiu-Huang-Goel'24]
- Feature learning dynamics of parities lead to insights into new distillation strategies [Panigrahy-Liu-Malladi-Goel'24]

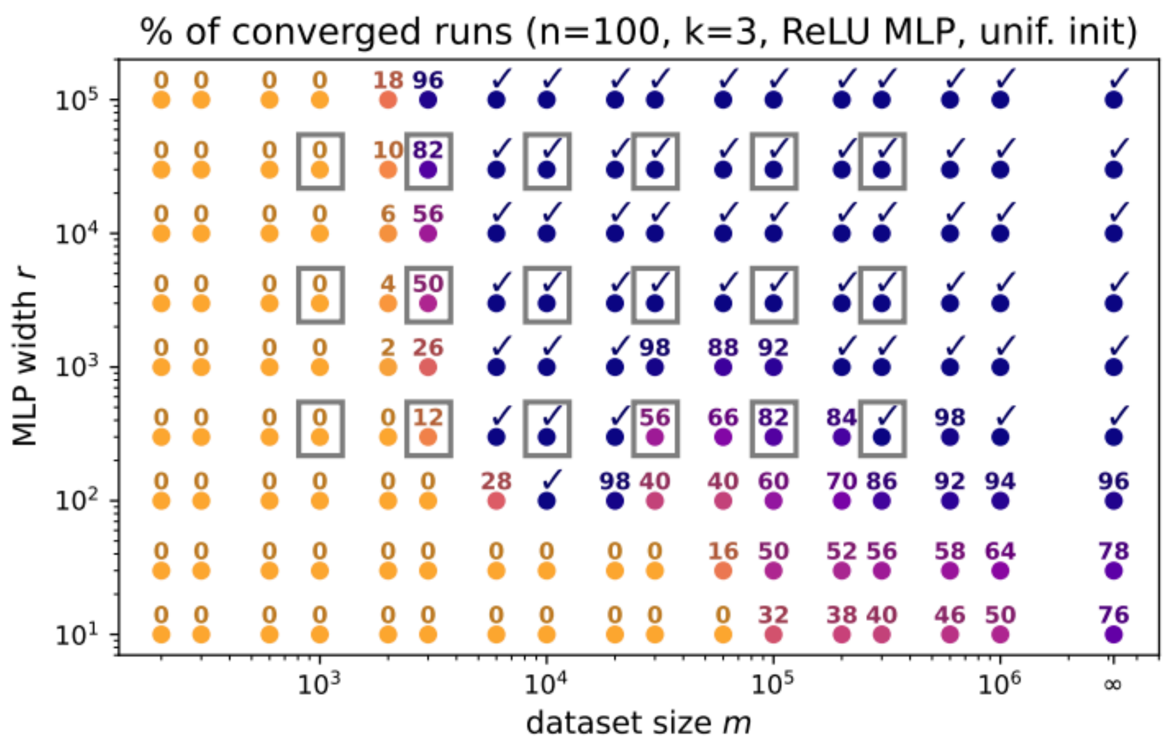
and more…

TODAY: PARITIES AND MARKOV CHAINS

Sparse-parities and Feature Learning

with Boaz Barak, Ben Edelman, Sham Kakade, Eran Malach & Cyril Zhang

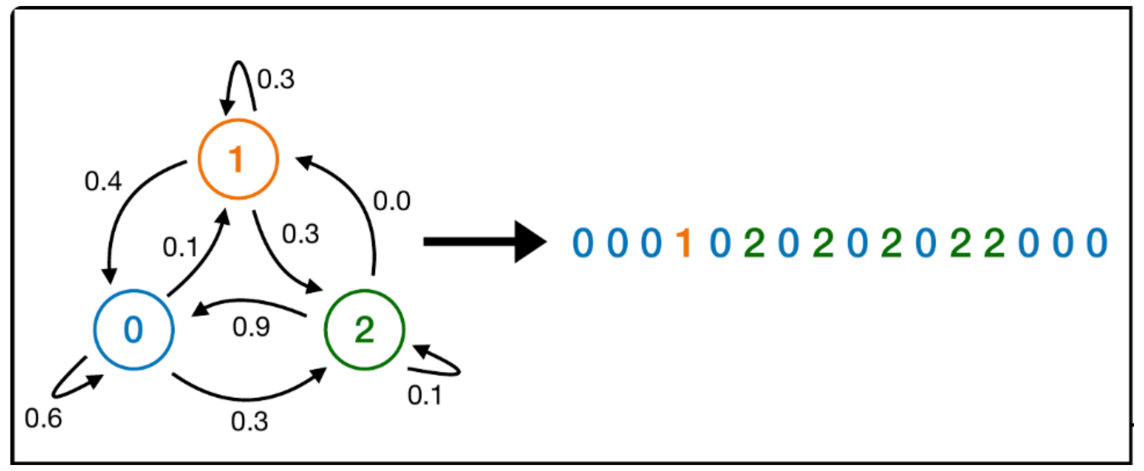


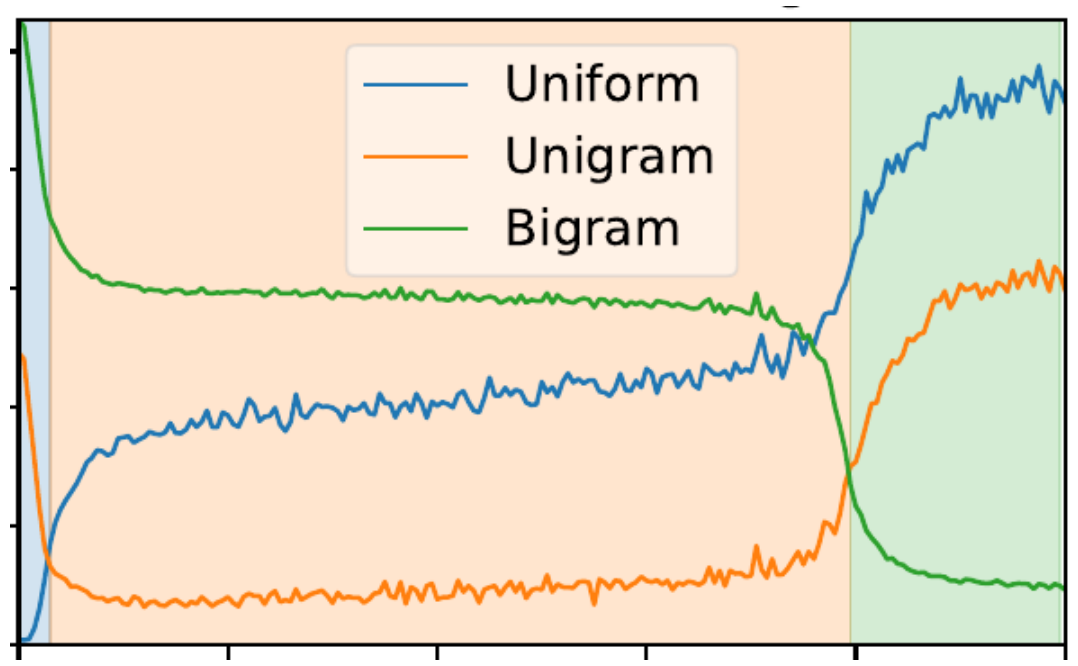


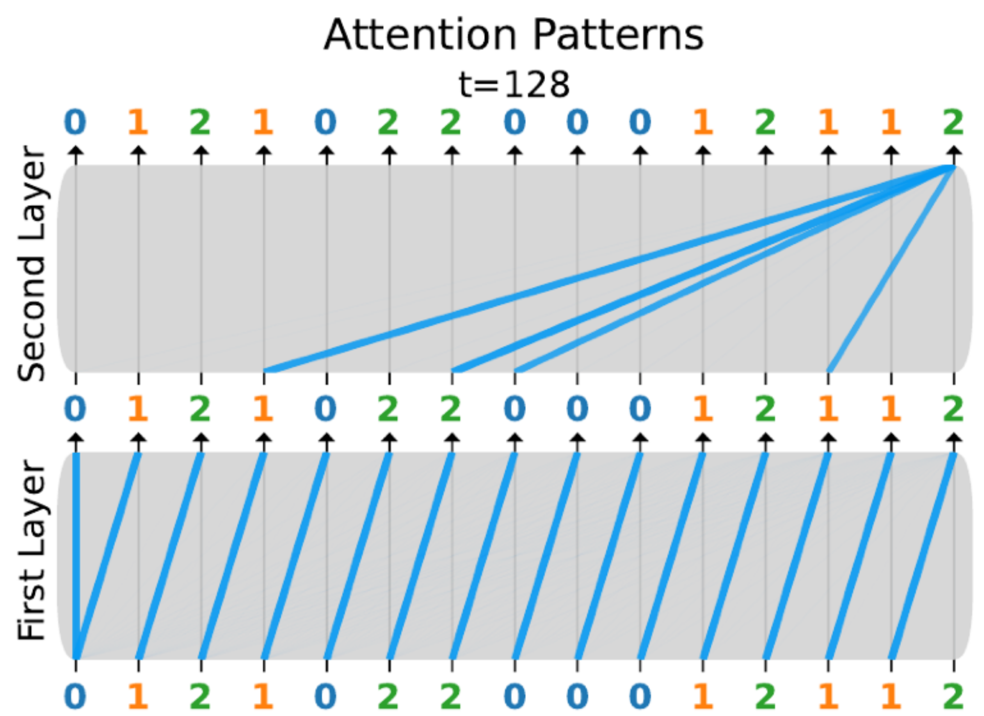
Slide credits shared with Cyril Zhang

Markov Chains and Induction Heads

with Ben Edelman, Ezra Edelman, Eran Malach & Nikos Tsilivis



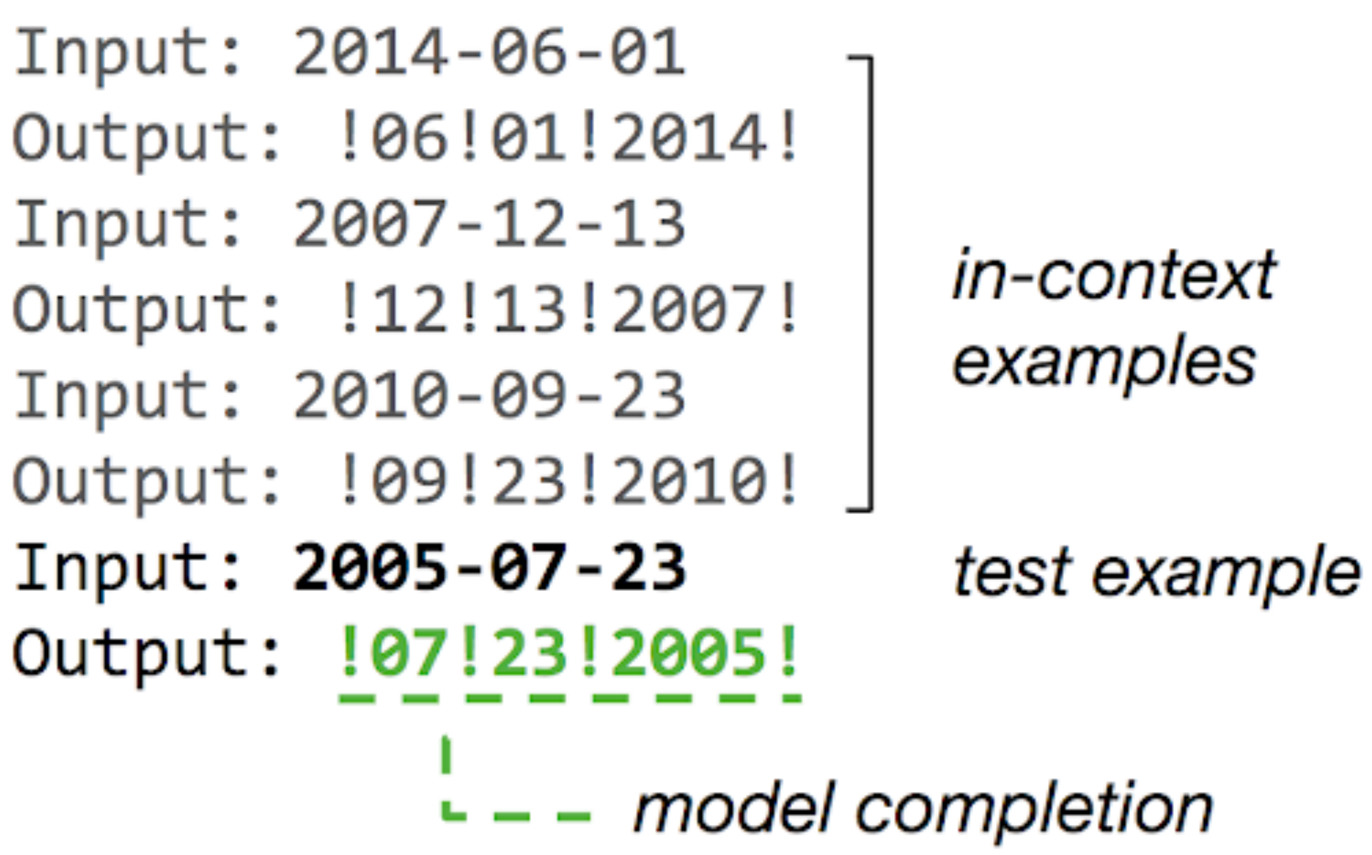


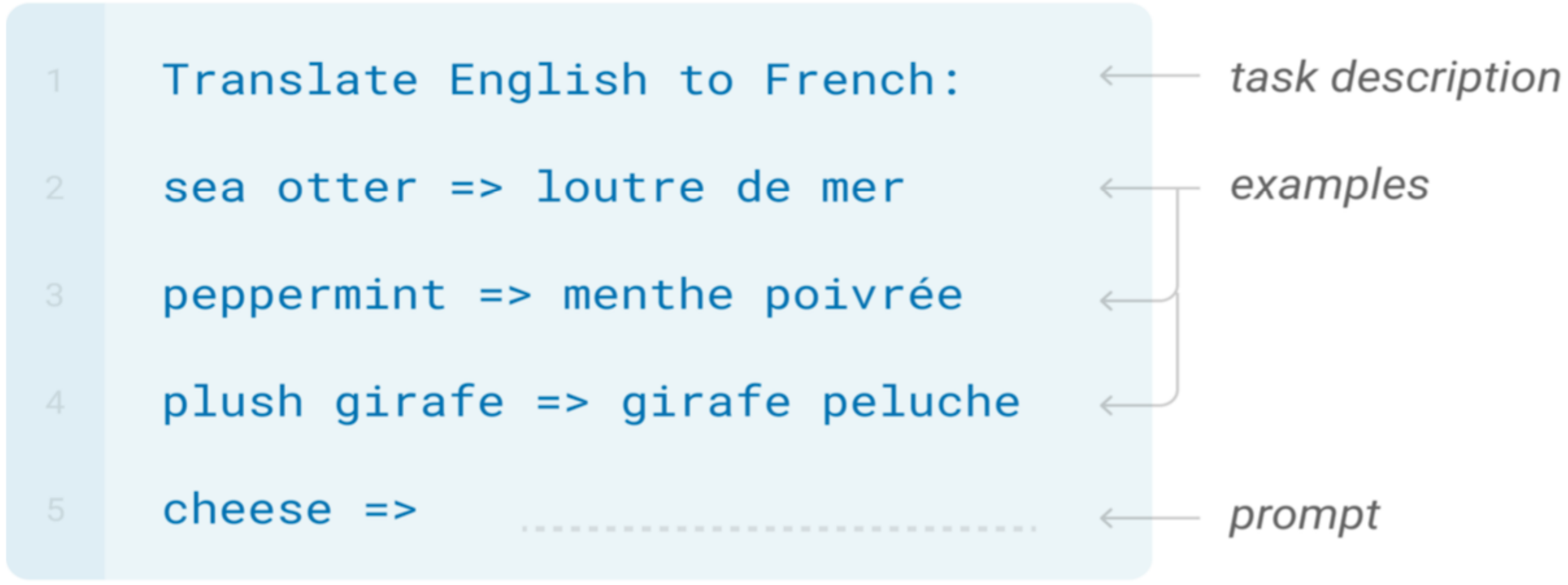


Slide credits shared with Ben Edelman

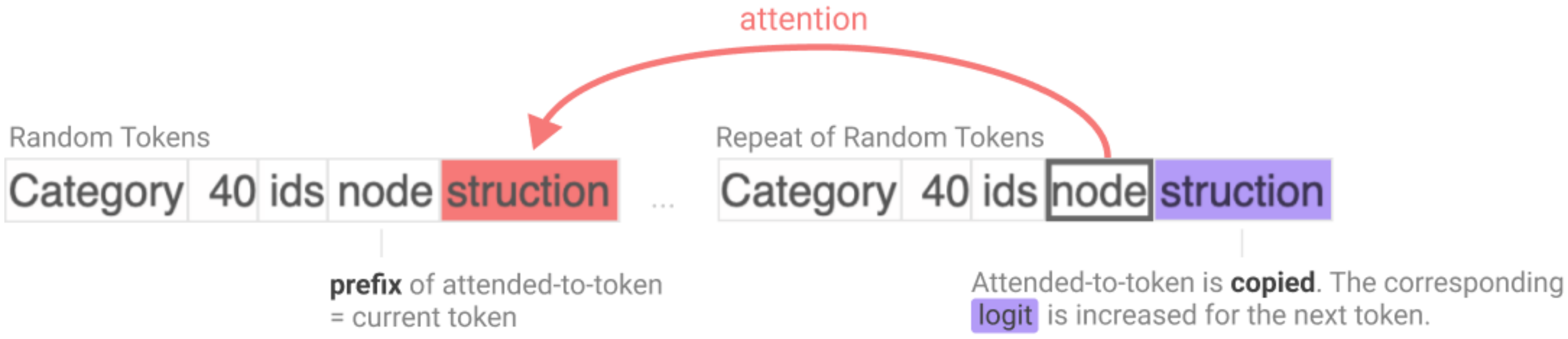
IN-CONTEXT LEARNING AND INDUCTION HEADS

Surprising ability of LLMs to learn from data in the prompt

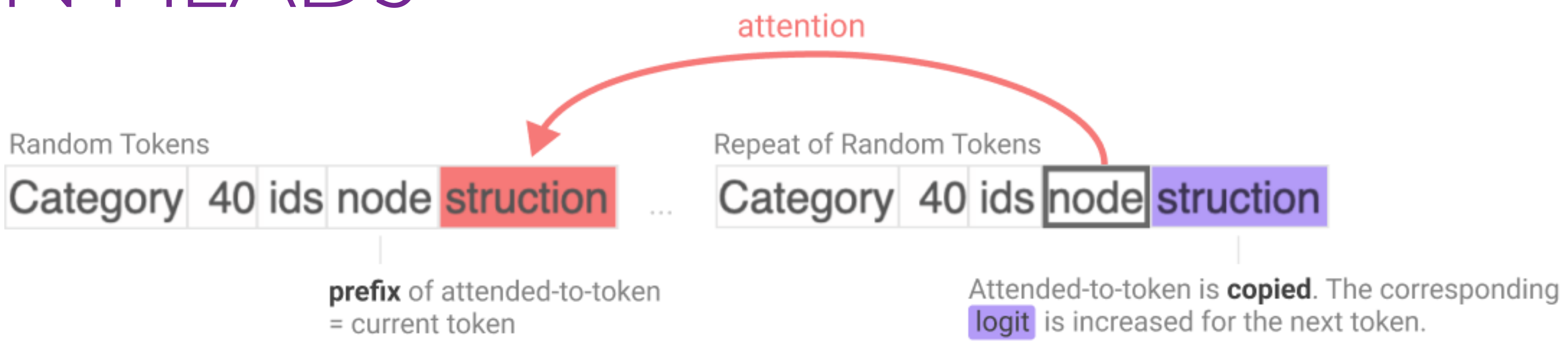




Researchers from Anthropic attributed this to the formation of induction heads



INDUCTION HEADS



Copy the token after the previous occurrence of the current token

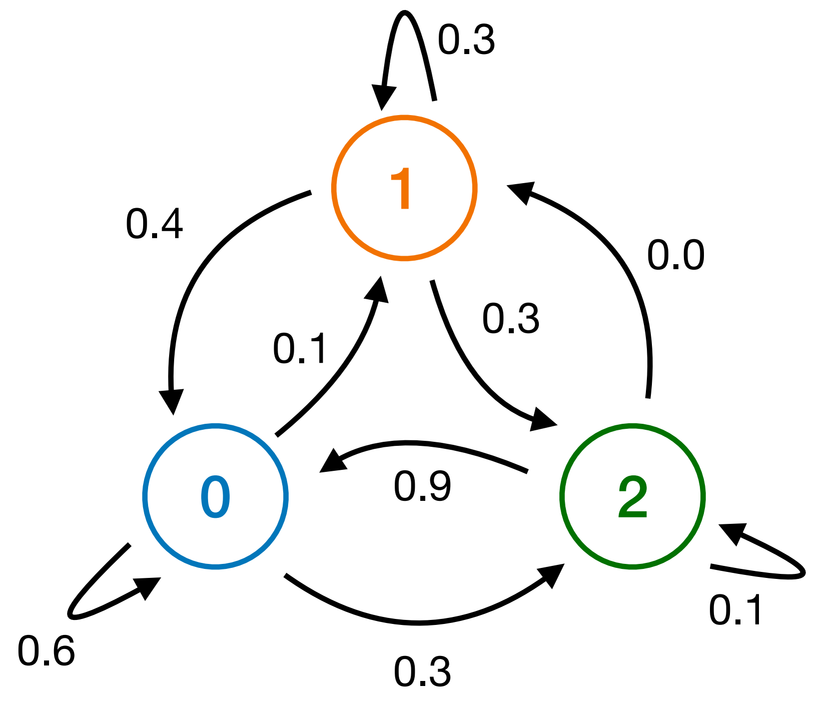
Can be thought of as 'bigram' computations

In the phase change, induction heads are formed and in-context loss drastically reduces

Phase changes are everywhere!

How do we understand this?

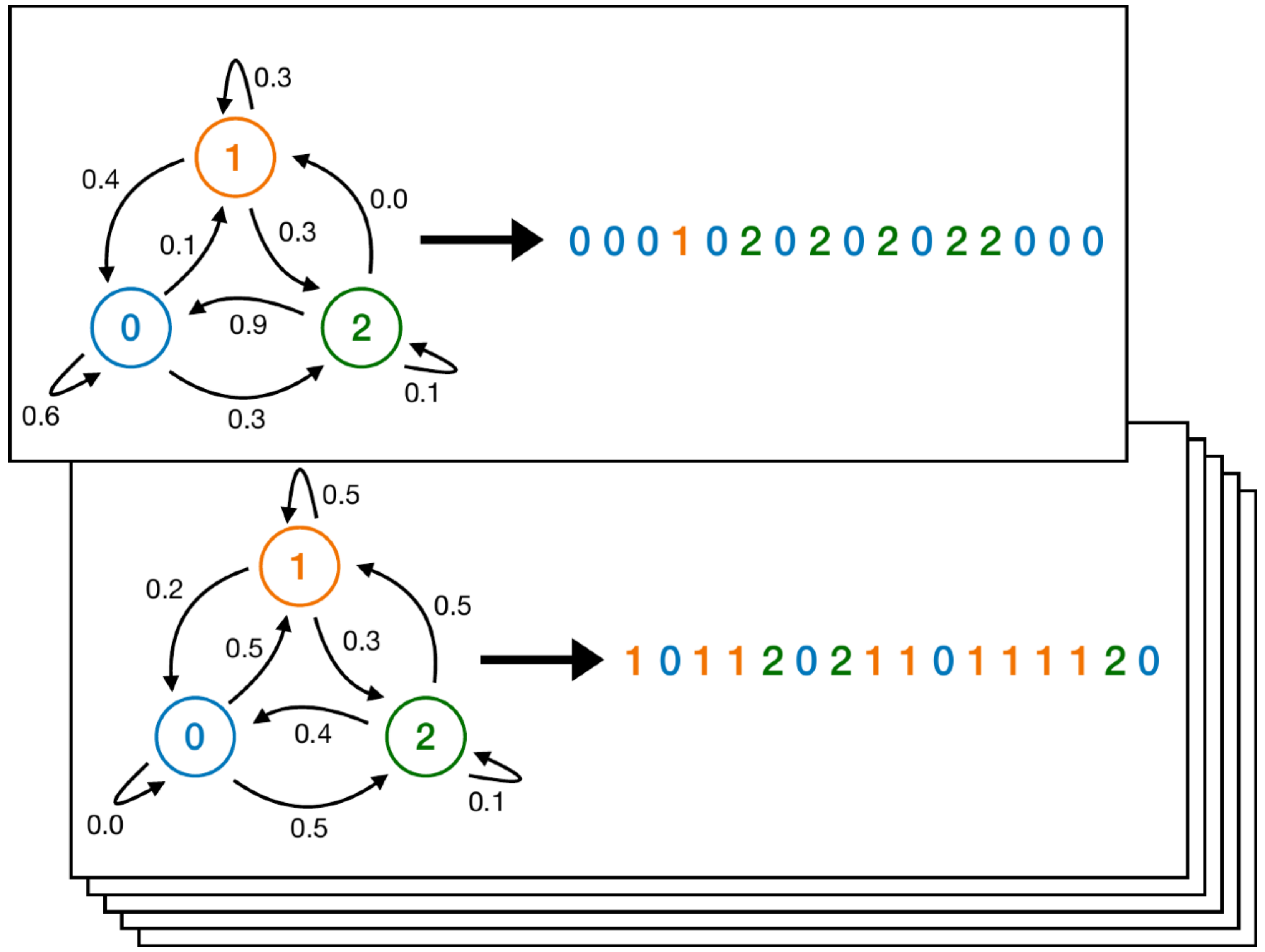
IN-CONTEXT LEARNING OF MARKOV CHAINS



0 0 0 1 0 2 0 2 0 2 0 2 2 0 0 0 …

Data: Dataset of sequences of states where each sequence is drawn from a different Markov chain

Goal: Get good accuracy at predicting the next-state in a randomly drawn Markov chain



Uniform **Strategy 1:** Guess uniformly

Unigram **Strategy 2**: Guess according to how likely each state is in the context

Bigram ★ **Strategy 3**: Guess according to how likely each state is in the context given the previous state

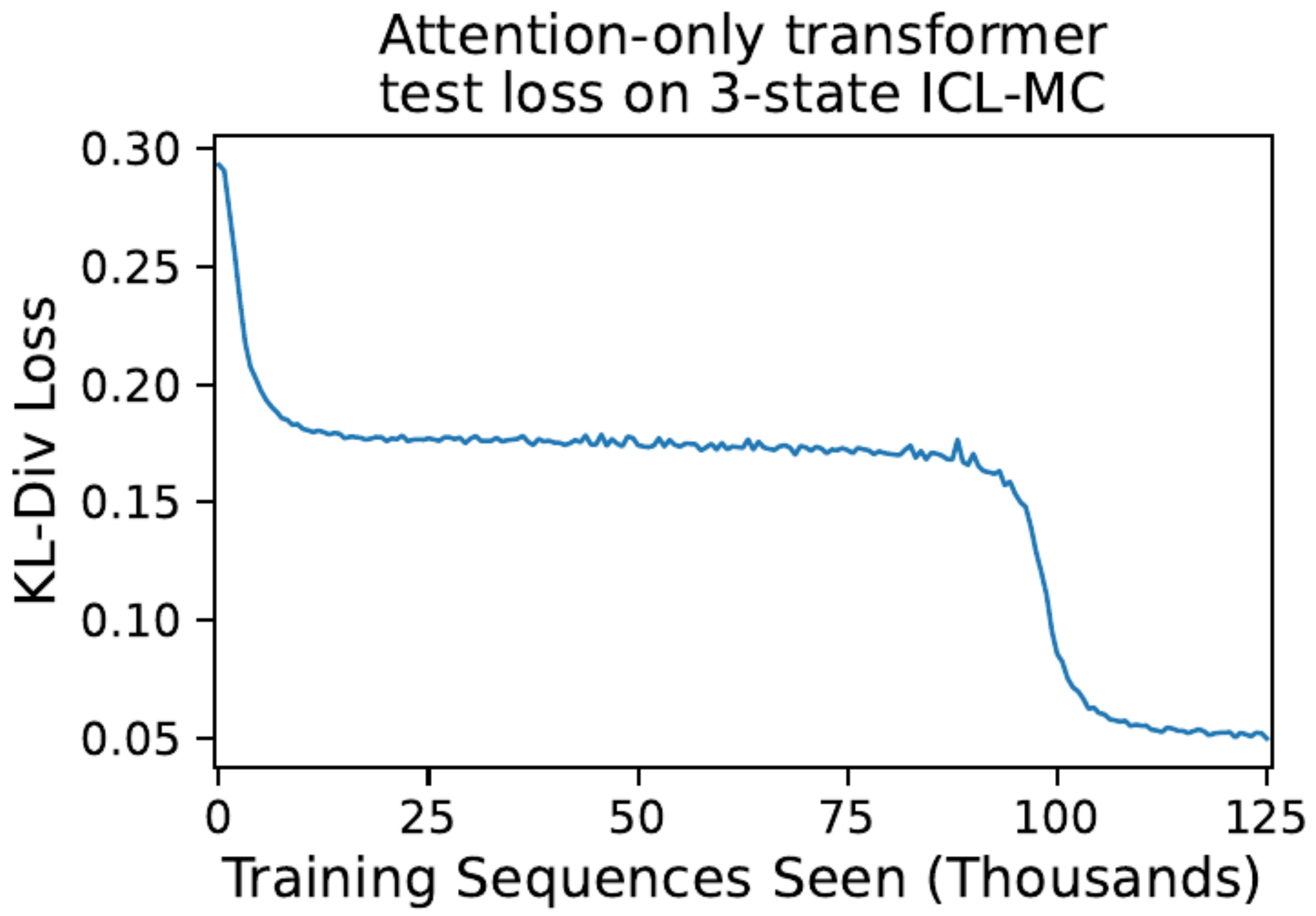
Edelman, Edelman, Goel, Malach, Tsilivis. The Evolution of Statistical Induction Heads: In-Context Learning Markov Chains. Under submission.

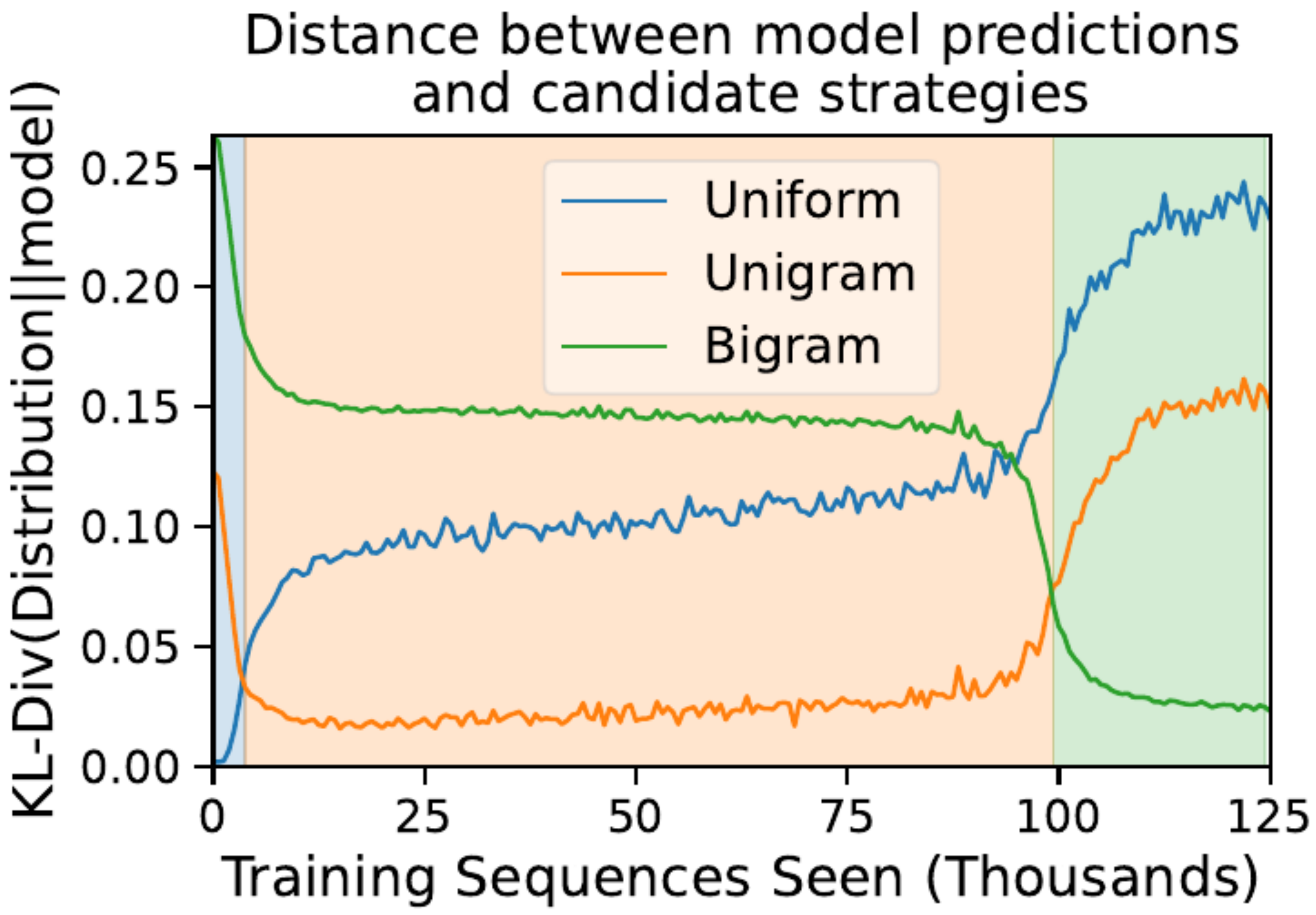
WHAT DO TRANSFORMERS DO?

Uniform **Strategy 1:** Guess uniformly

Unigram **Strategy 2**: Guess according to how likely each state is in the context

Bigram ★ **Strategy 3**: Guess according to how likely each state is in the context given the previous state



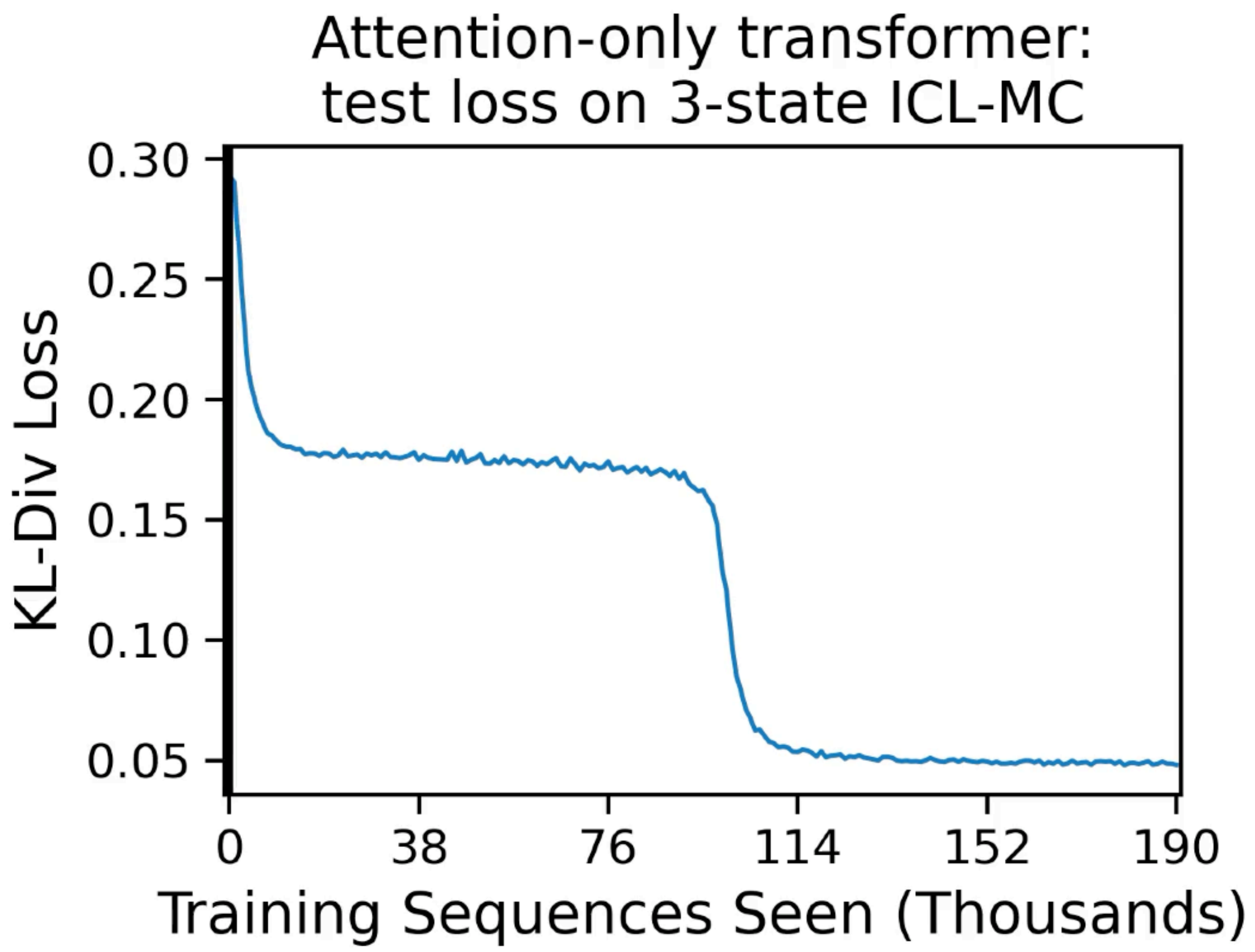


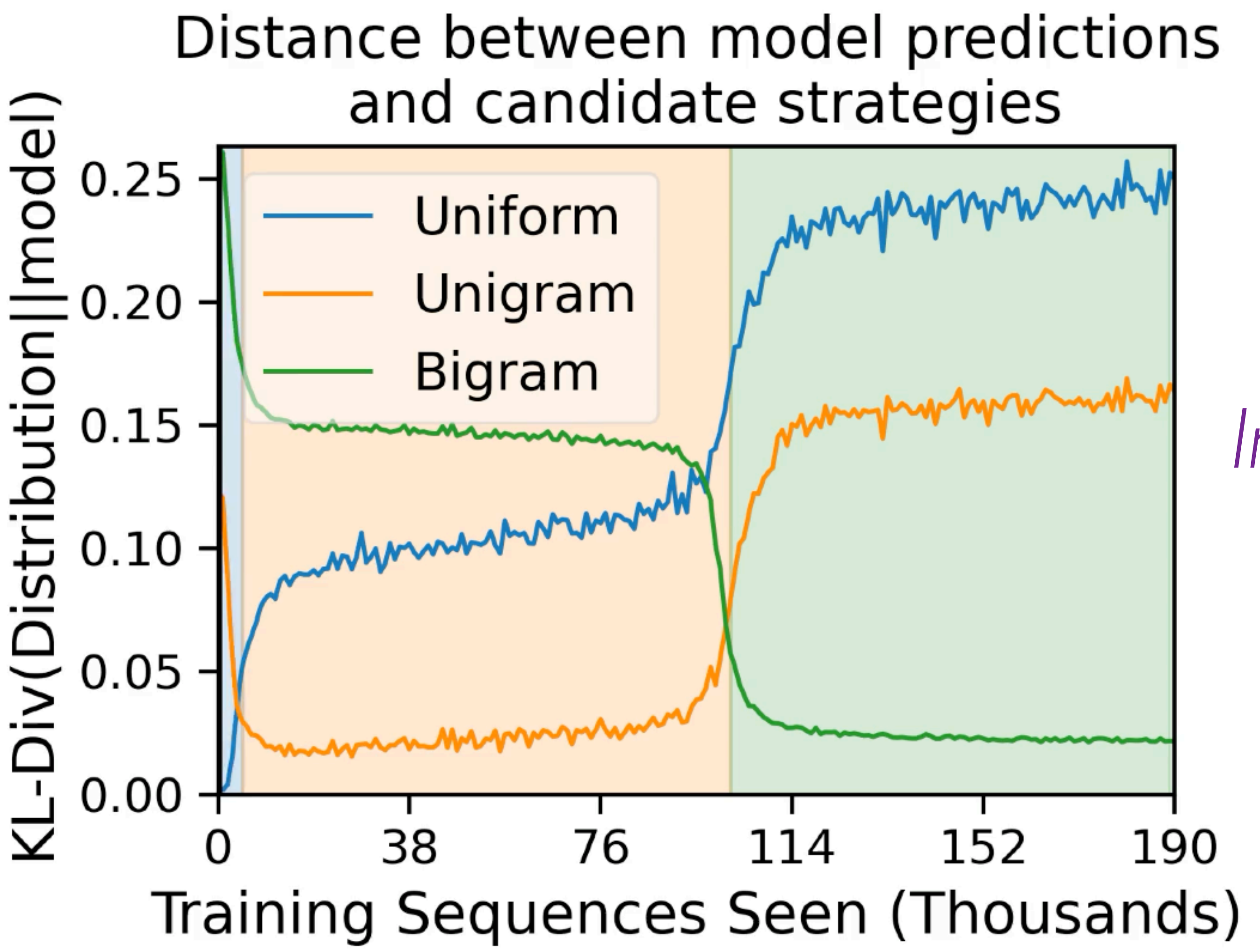
Edelman, Edelman, Goel, Malach, Tsilivis. The Evolution of Statistical Induction Heads: In-Context Learning Markov Chains. Under submission.

Animation credit: Ezra

WHAT DO TRANSFORMERS DO?

Transformer hovers at the unigram stage

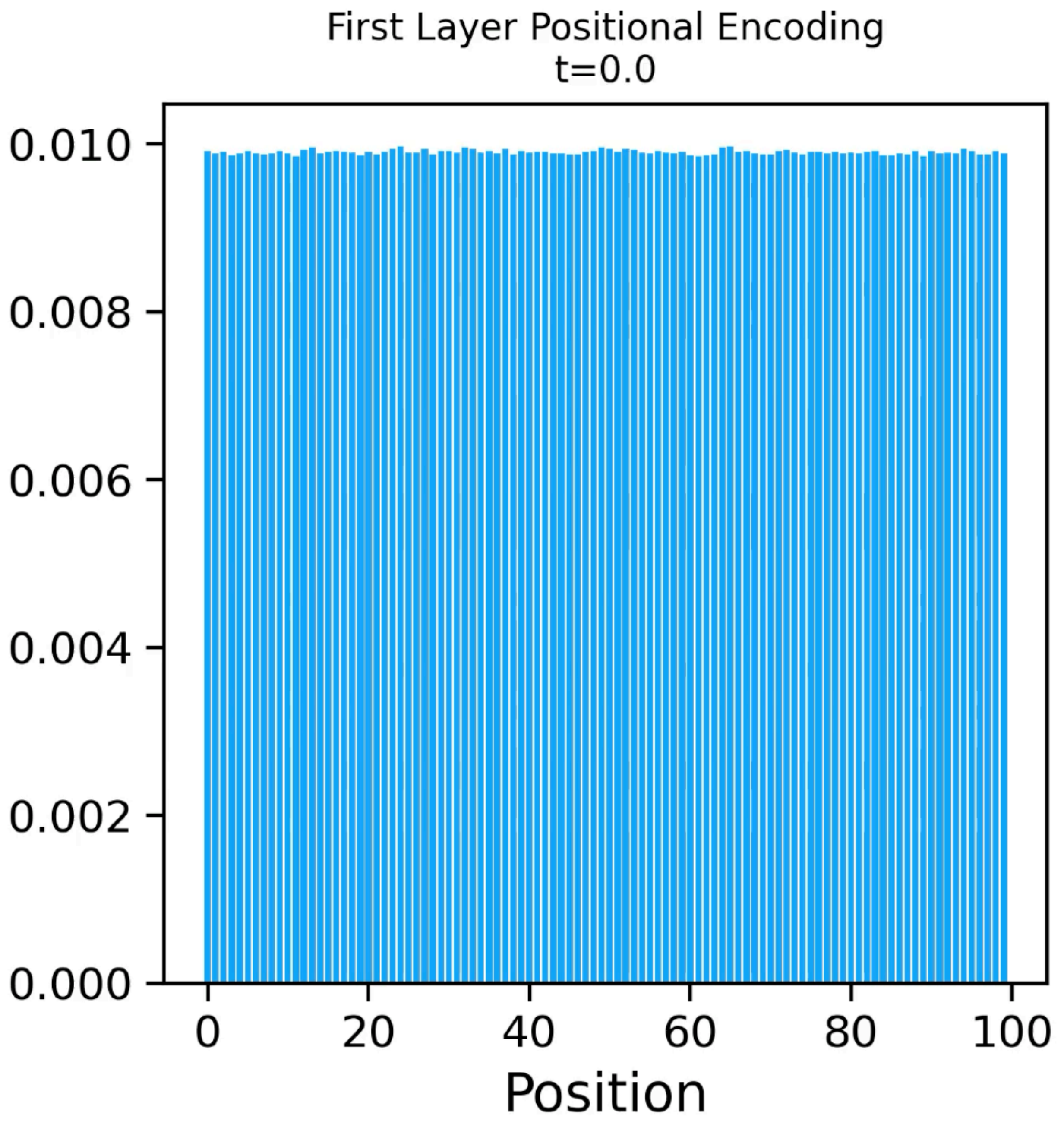


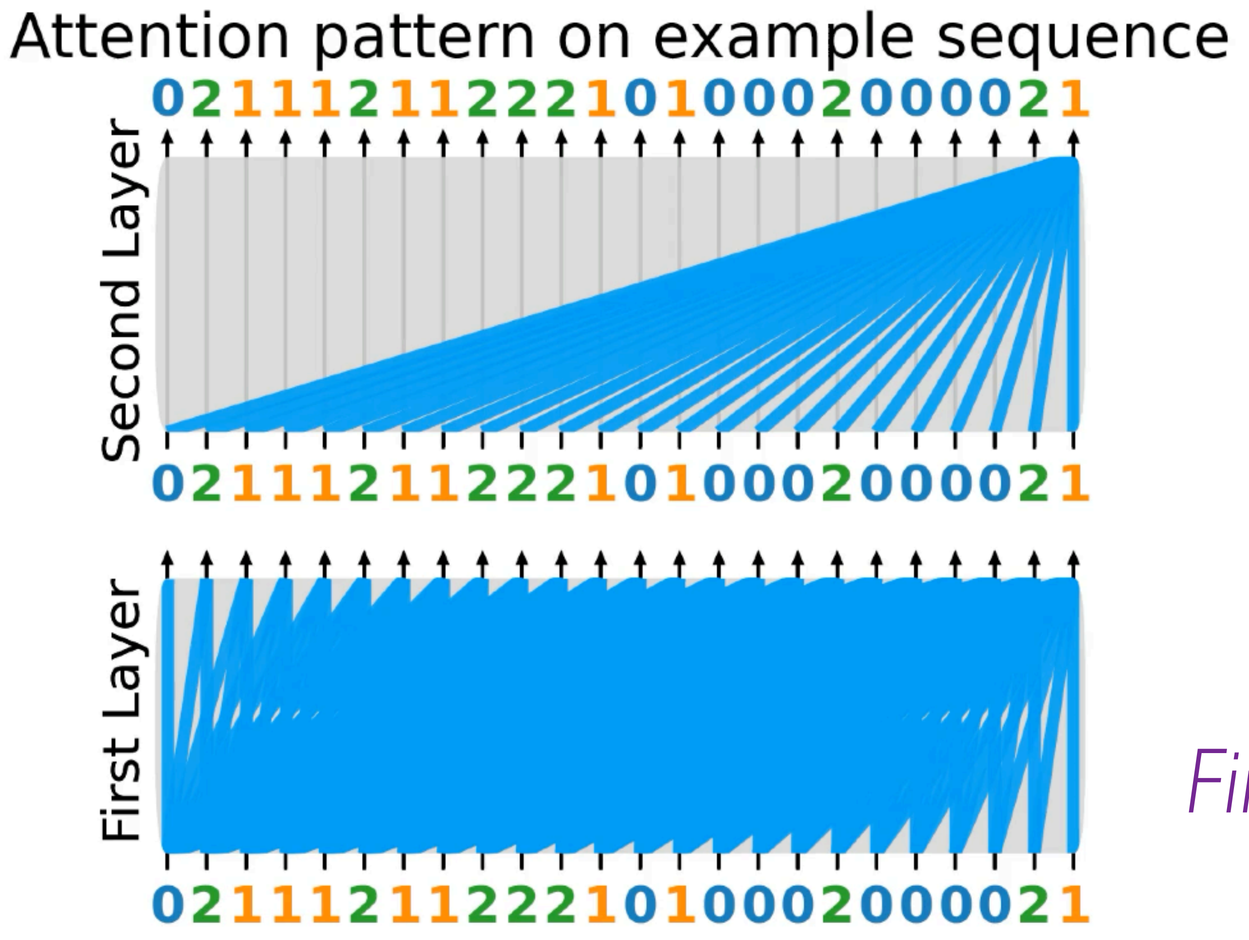


Induction head is formed at the phase transition

Relative position, so p refers to position encoding on pth token before

$p = 1$ becomes dominant at the end





Second layer finds all tokens that follow the current token

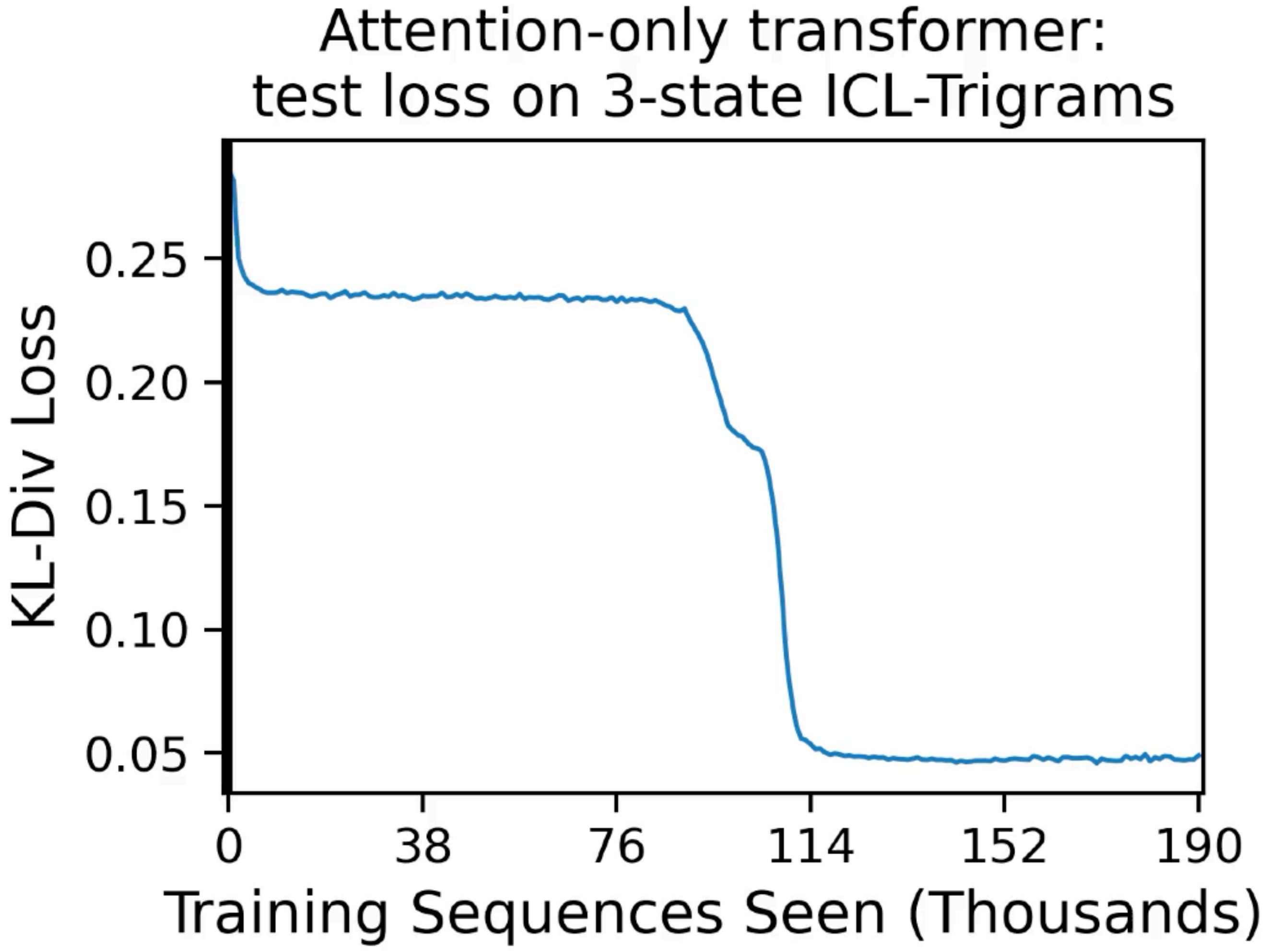
First layer looks one back

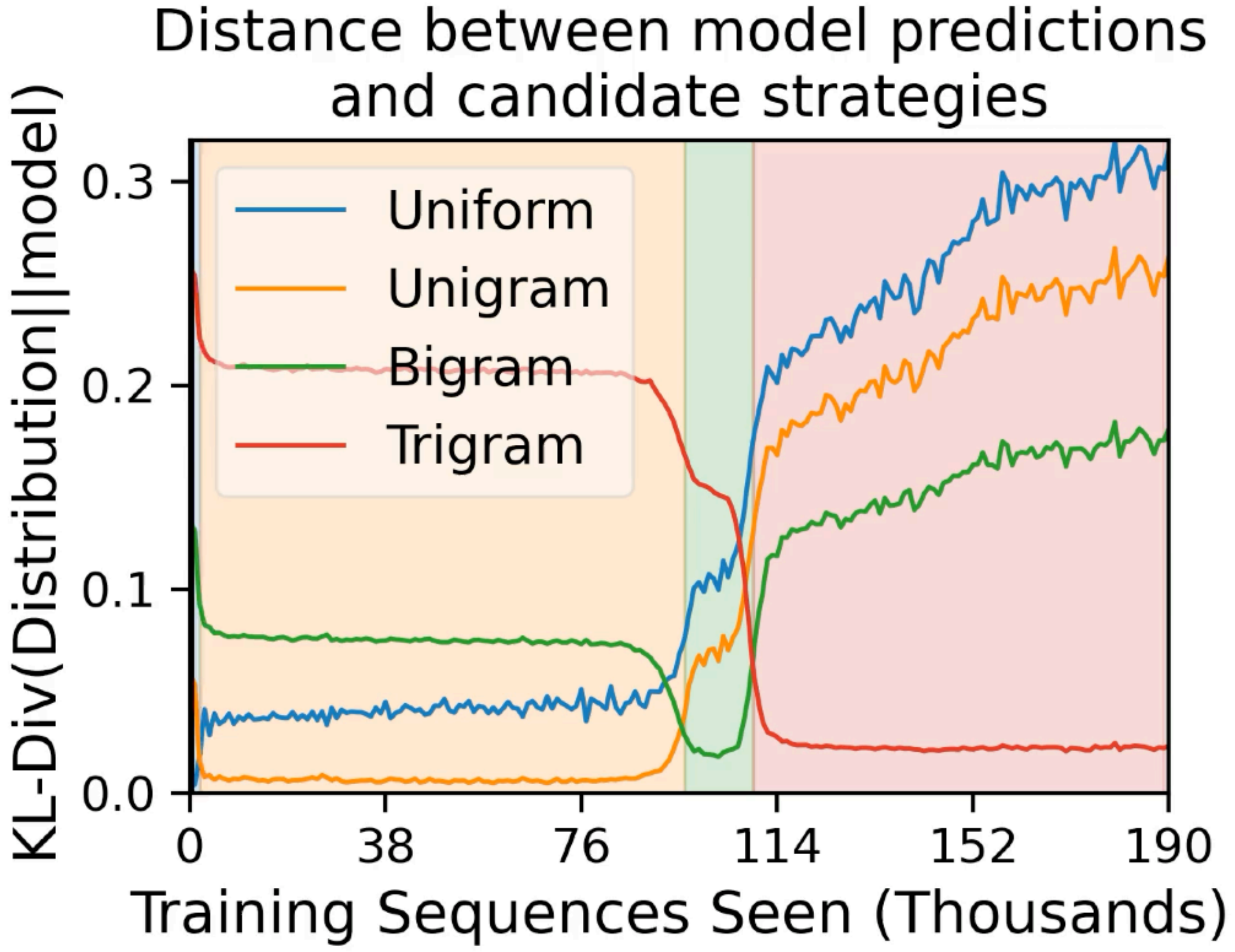
Edelman, Edelman, Goel, Malach, Tsilivis. The Evolution of Statistical Induction Heads: In-Context Learning Markov Chains. Under submission.

Animation credit: Ezra

WHAT DO TRANSFORMERS DO?

Transformer hovers at the unigram stage, then passes to through a bigram stage

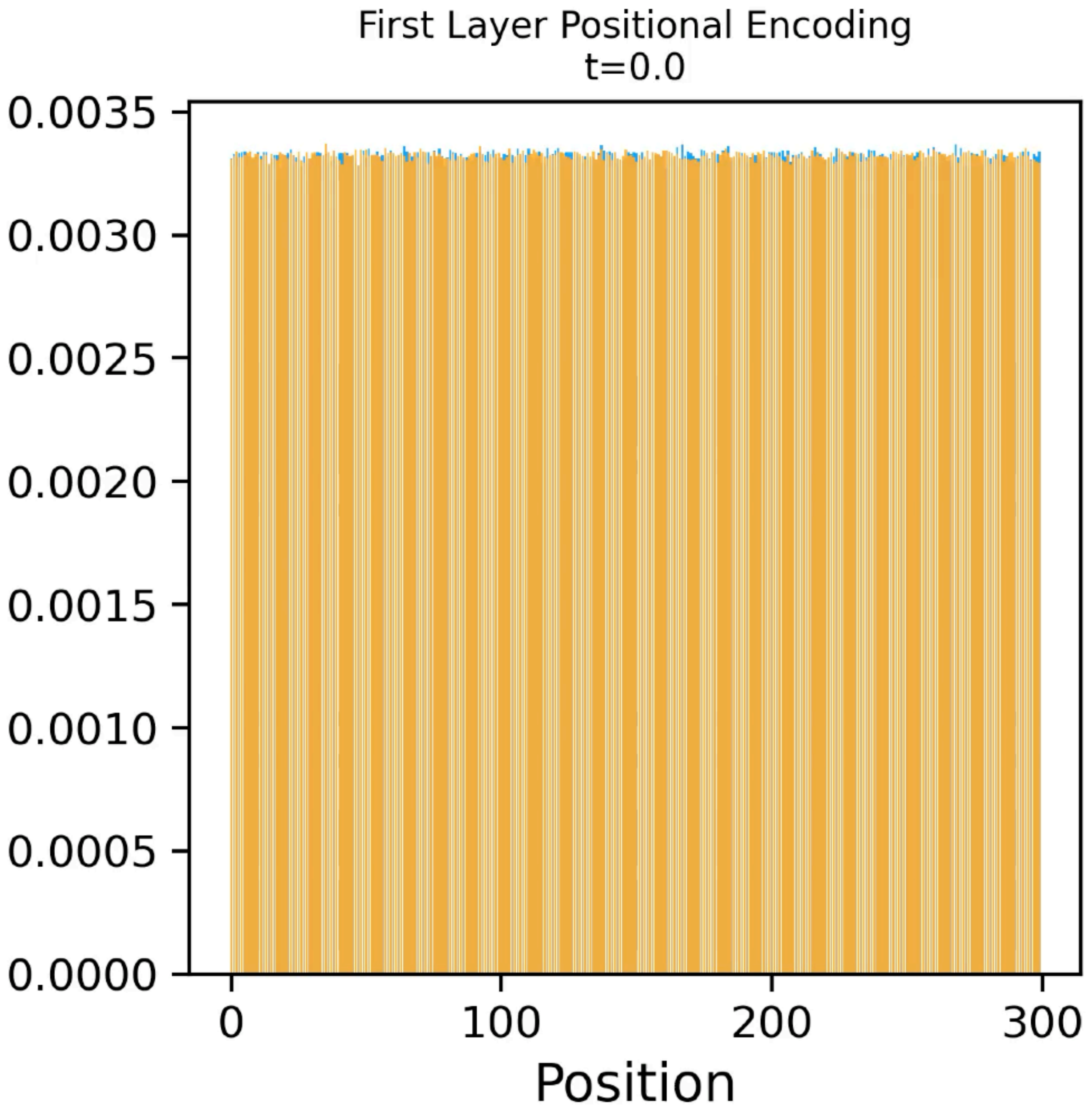


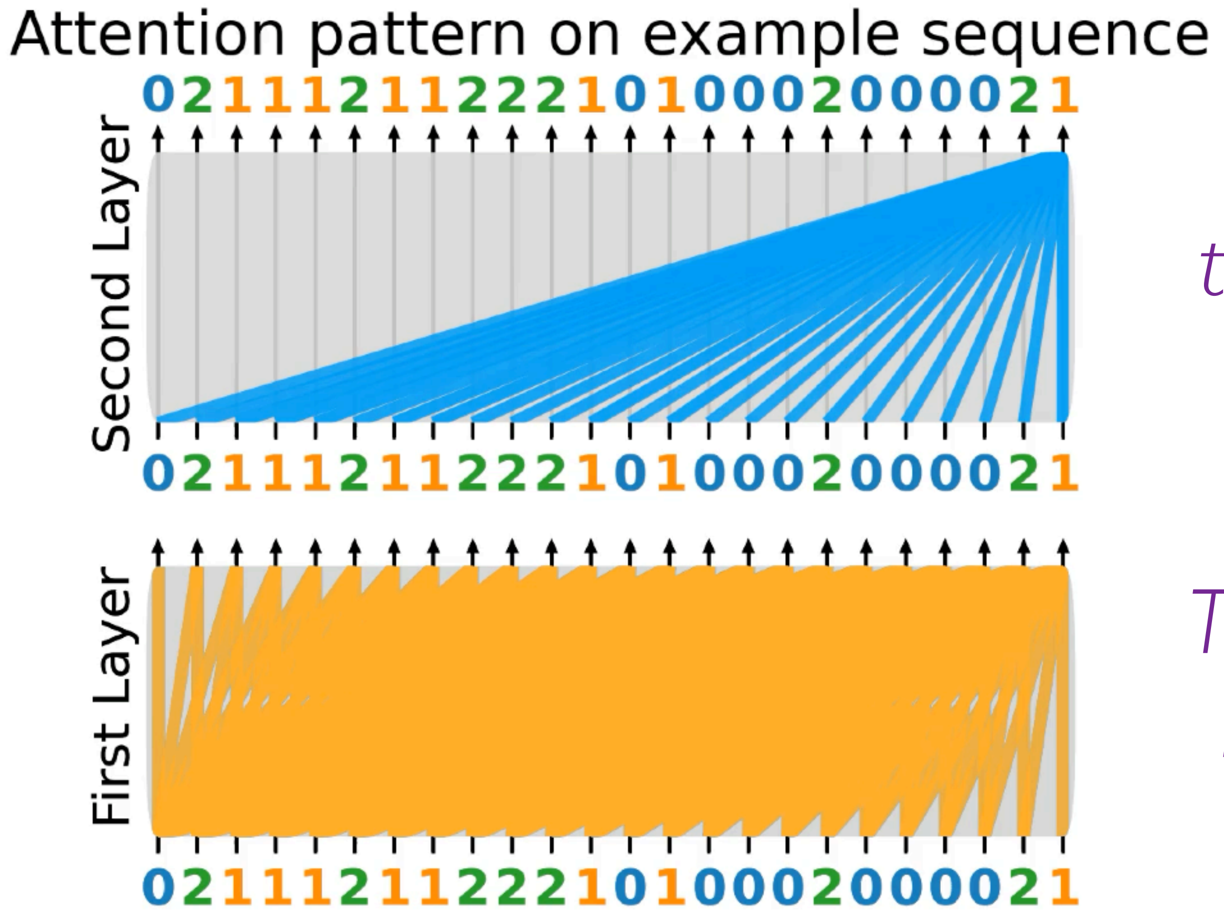


Higher order induction head is formed at the phase transition

Relative position, so p refers to position encoding on pth token before

$p = 1,2$ become dominant at the end





Second layer finds all tokens that have the two previous tokens

The two heads in the first layer looks one and two positions back

Edelman, Edelman, Goel, Malach, Tsilivis. The Evolution of Statistical Induction Heads: In-Context Learning Markov Chains. Under submission.

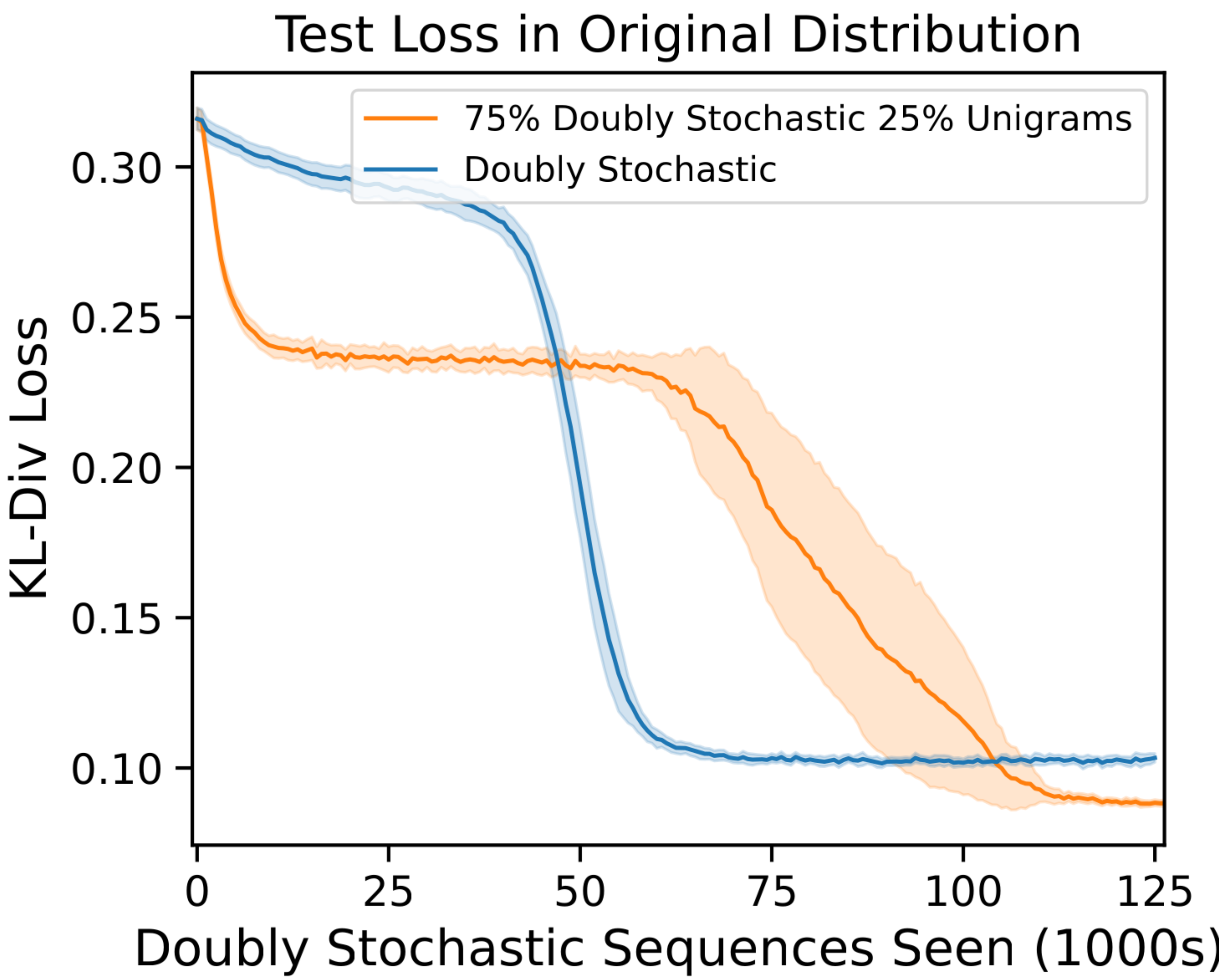
IS LEARNING THE UNIGRAM HELPFUL?

Test: What if we train on data where unigram is not helpful?

Doubly stochastic matrices lead to uniform stationary distribution, therefore unigram is not helpful

No unigram phase

Converges faster



Unigram slows down learning of bigram

But gets lower error

Edelman, Edelman, Goel, Malach, Tsilivis. The Evolution of Statistical Induction Heads: In-Context Learning Markov Chains. Under submission.

WHAT IS HAPPENING UNDER THE HOOD?

Simplified Transformer:

Causal learning

$$f(E) = \text{mask}\left(EW_k(ME)^T\right)E, \text{ where } M = \begin{pmatrix} v_1 & 0 & \dots & 0 \\ v_2 & v_1 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ v_t & v_{t-1} & \dots & v_1 \end{pmatrix} \in \mathbb{R}^{t\times t} \text{ and } W_k \in \mathbb{R}^{k\times k}$$

Embedding of input *Second layer alignment*

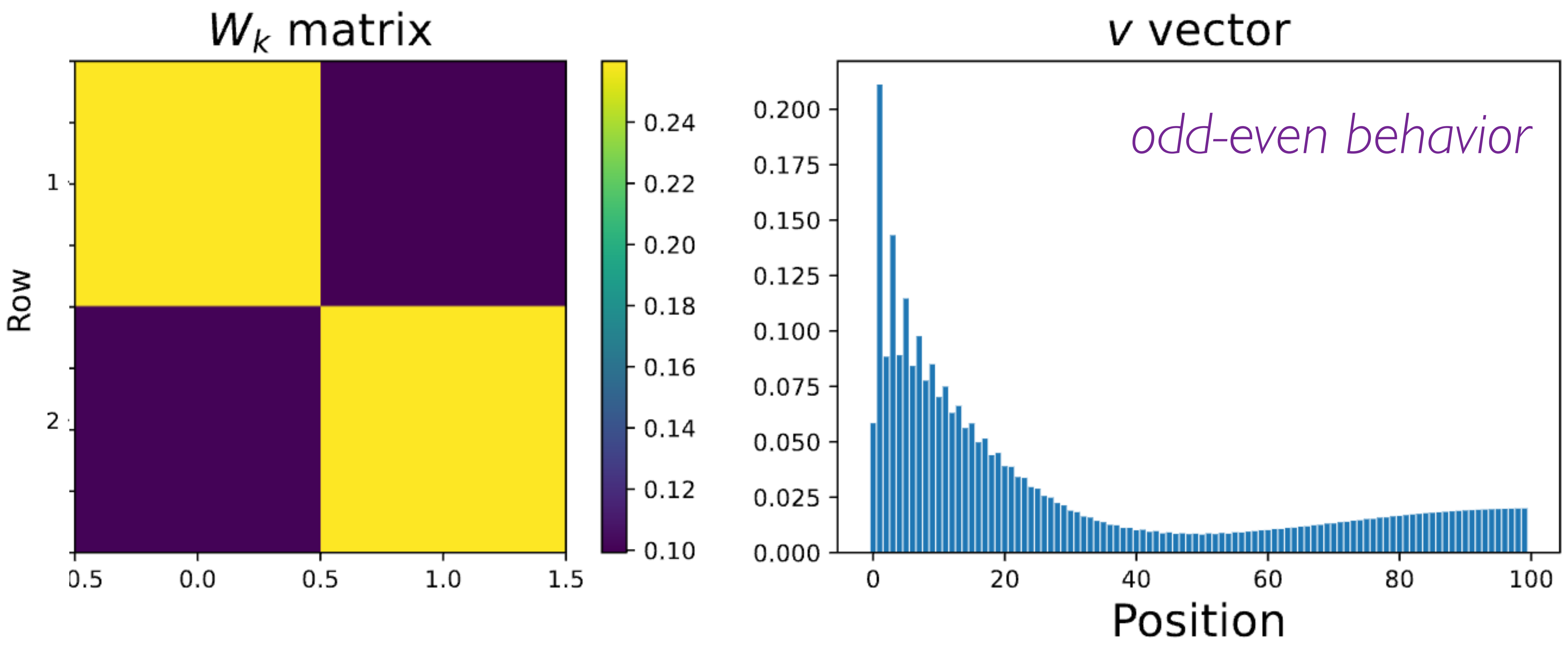
Relative position encoding

Bigram: $W_k = Id_k$ and $v = [0,1,0,\ldots,0]$

Unigram: $W_k = 11^\top$ and $v = [1,0,\ldots,0]$

Key observation: Two-phase learning,

- W_k gets a diagonal component after first step, and v gets a quadratic decay
- Once the diagonal bias exists, v_2 gets higher gradient than all other positions

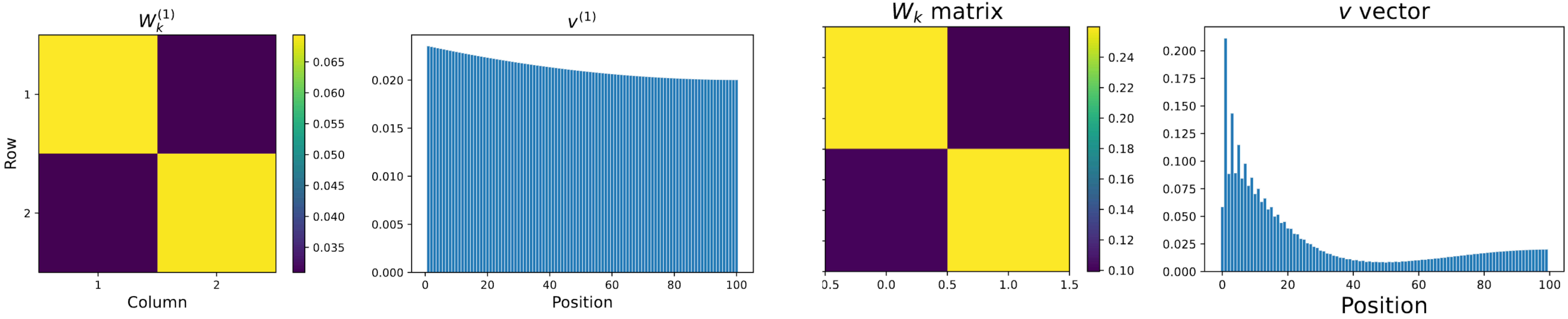


Edelman, Edelman, Goel, Malach, Tsilivis. The Evolution of Statistical Induction Heads: In-Context Learning Markov Chains. Under submission.

WHAT IS HAPPENING UNDER THE HOOD?

Key observation: Two-phase learning,

- W_k gets a diagonal component after first step, and v gets a quadratic decay
- Once the diagonal bias exists, v_2 gets higher gradient than all other positions



Theoretical analysis shows that the first step gradient for diagonal bias is $O(t)$ larger than the gradient bias for step 2, which could explain why step 2 takes a lot longer

Caveats: Hard to compute closed forms for $k > 2$, and dominance of v_2 for all losses

Edelman, Edelman, Goel, Malach, Tsilivis. The Evolution of Statistical Induction Heads: In-Context Learning Markov Chains. Under submission.

All within the last month or two 😱

USEFUL SETTING TO UNDERSTAND LLMS

IN-CONTEXT LANGUAGE LEARNING: ARCHITECTURES AND ALGORITHMS

Ekin Akyürek Bailin Wang Yoon Kim Jacob Andreas
MIT CSAIL
{akyurek, bailinw, yoonkim, jda}@mit.edu

Empirically find higher-order induction heads

Attention with Markov: A Framework for Principled Analysis of Transformers via Markov Chains

Ashok Vardhan Makkuva [*1] Marco Bondaschi [*1] Adway Girish [1] Alliot Nagle [2] Martin Jaggi [1] Hyeji Kim [†2]
Michael Gastpar [†1]

Loss landscape for data from single Markov chain

The Developmental Landscape of In-Context Learning

Jesse Hoogland [*1] George Wang [*1] Matthew Farrugia-Roberts [2] Liam Carroll [2] Susan Wei [3] Daniel Murfet [3]

Observe similar stages of learning in in-context linear regression

How Transformers Learn Causal Structure with Gradient Descent

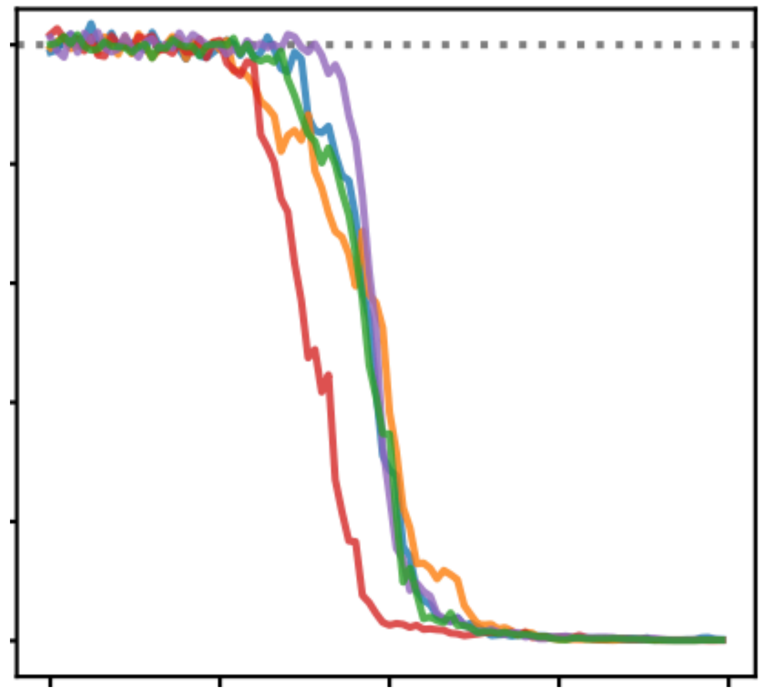
Eshaan Nichani, Alex Damian, and Jason D. Lee

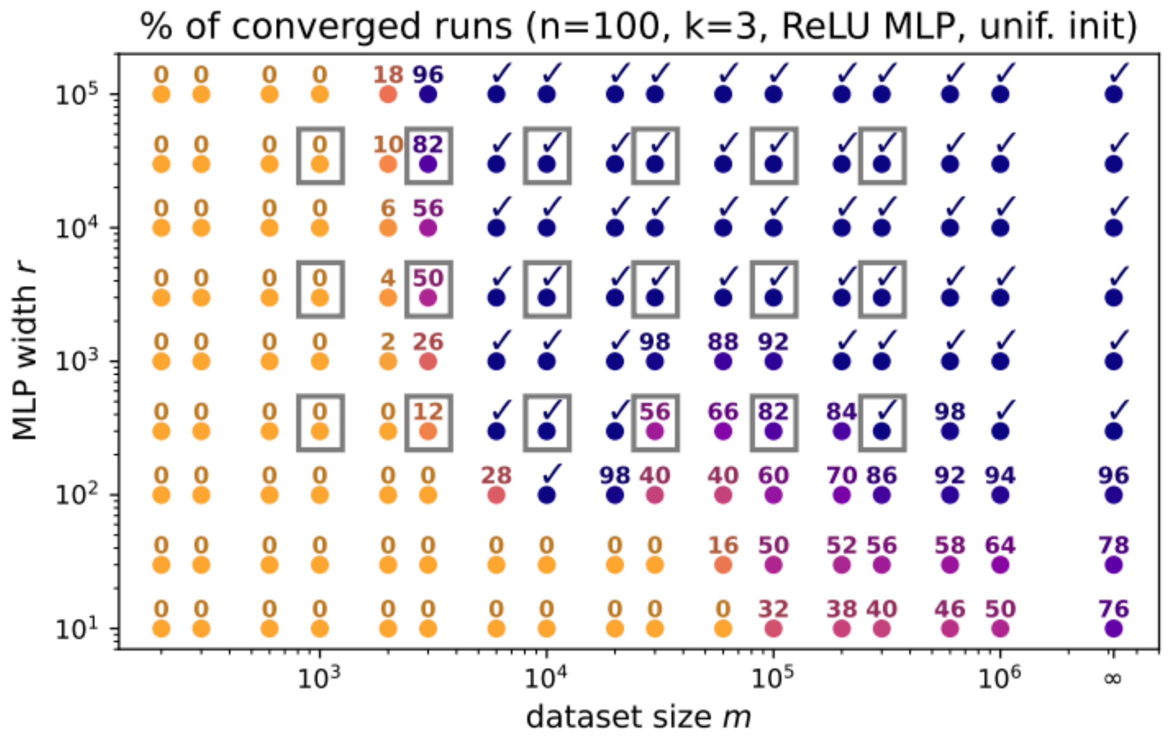
Show how Transformers learn general causal structures beyond Markov Chains

TODAY: PARITIES AND MARKOV CHAINS

Sparse-parities and Feature Learning

with Boaz Barak, Ben Edelman, Sham Kakade, Eran Malach & Cyril Zhang

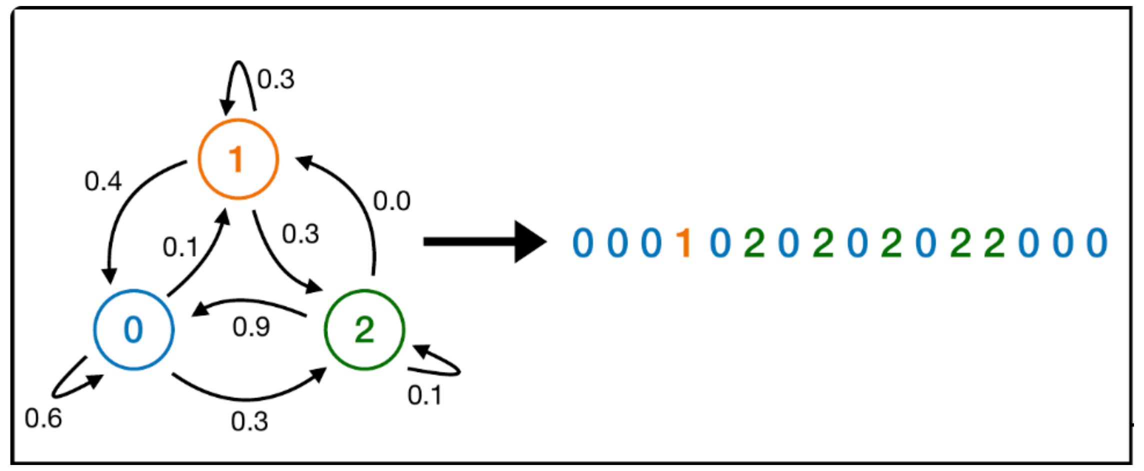


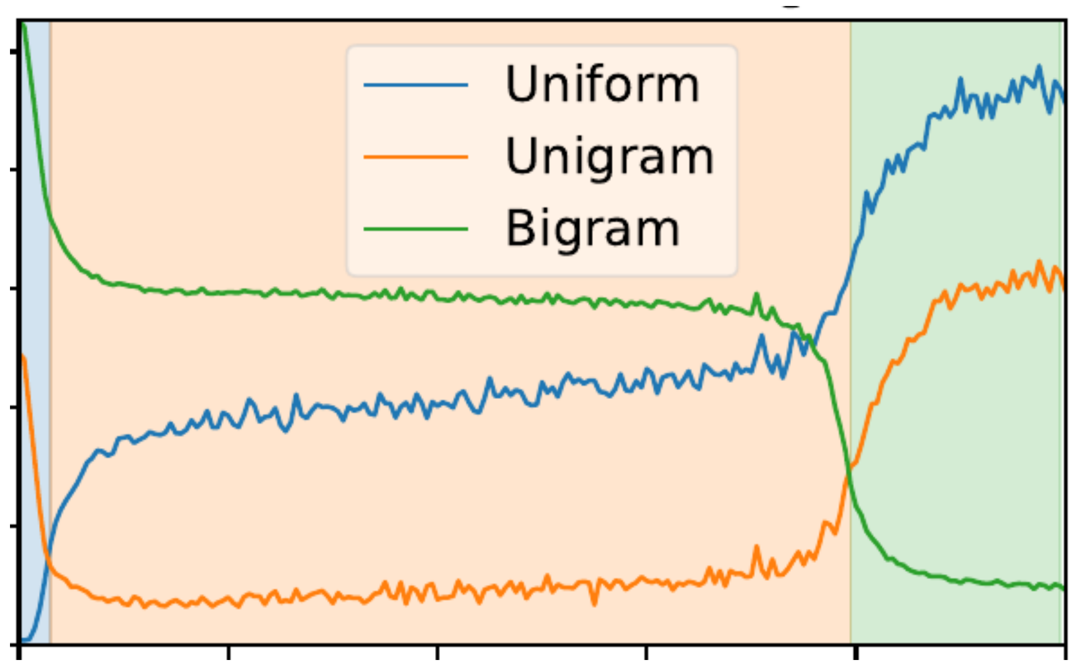


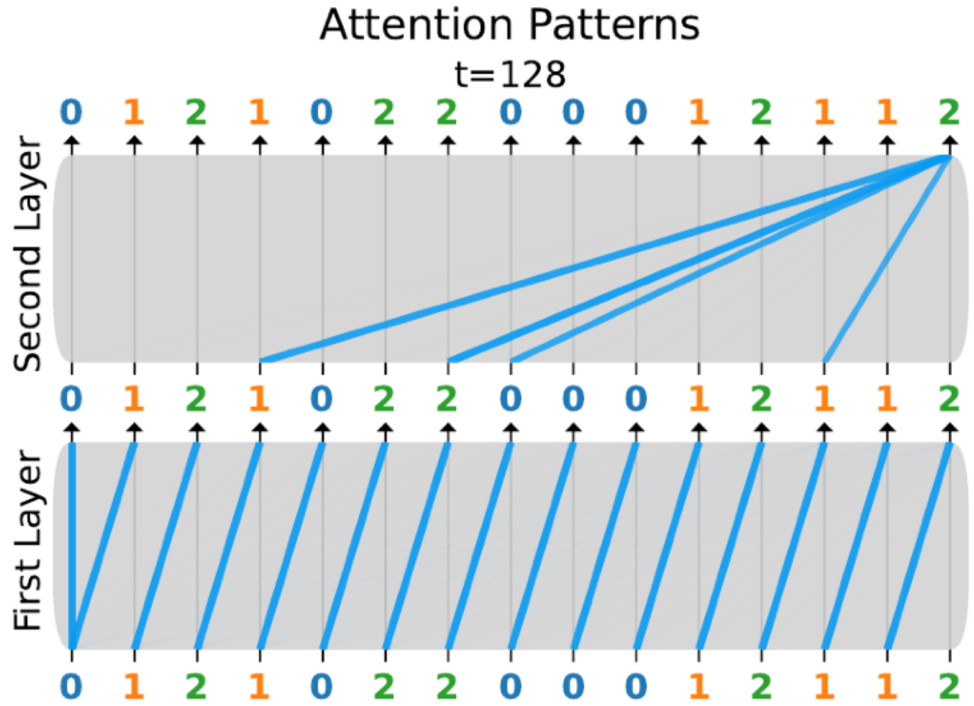
Slide credits shared with Cyril Zhang

Markov Chains and Induction Heads

with Ben Edelman, Ezra Edelman, Eran Malach & Nikos Tsilivis







Slide credits shared with Ben Edelman

LOOKING AHEAD

Synthetic controlled setup as a playground to probe:

- dynamics of feature learning
- algorithmic learning
- emergent phenomena
- …

LEGO [Zhang et al.'22]

PVRs [Zhang et al.'21]

DFAs (Dyck, …) [Yao et al.'21]

Math (modulo arithmetic) [Power et al'21]

Learning to Learn Simple Function Classes [Garg et al'22]

Outcomes: Architectural modifications, evaluation methods, data importance measures, quantification of unexpected behaviors, …

Many interesting optimization questions in these non-convex dynamics!