STAT 992: Science of Large Language Models

Lecture 2: Emergent abilities, prompting, and in-context learning

Spring 2026

Yiqiao Zhong

Early surprises of LLMs

- **Qualitative change** when we keep scaling model sizes and data
- Proposed in [Emergent Abilities of Large Language Models]:

  An ability is emergent if it is not present in smaller models but is present in larger models.

- Phase transition, difficult to predict

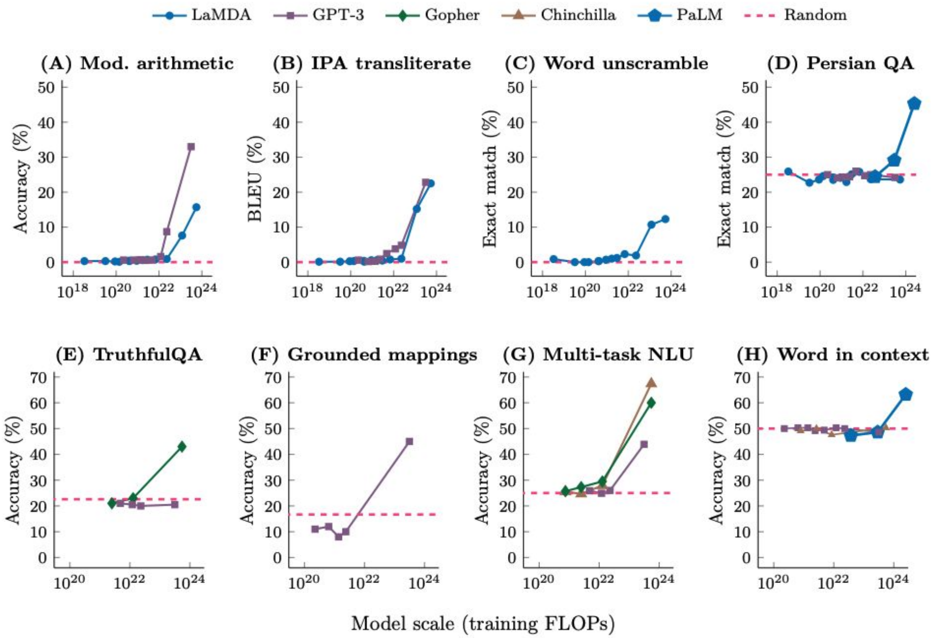


Figure 2: Eight examples of emergence in the few-shot prompting setting. Each point is a separate model.

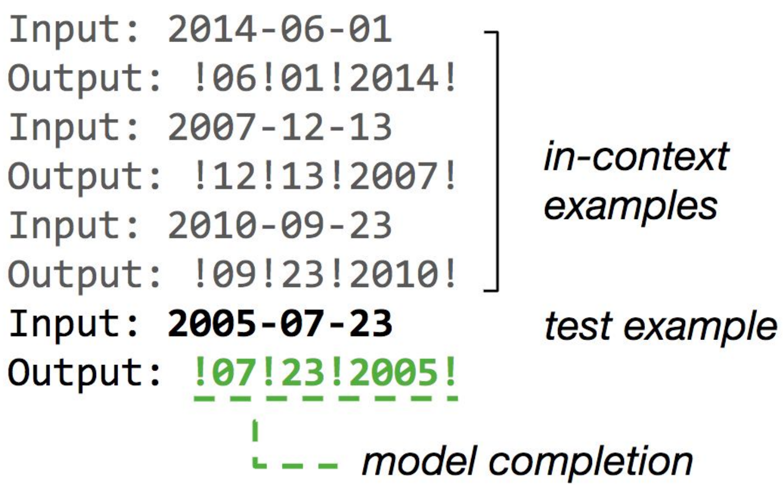
Prompting & in-context learning (ICL)

- Representative emergent abilities
- How do we adapt a pretrained language model $p_\theta(x_{t+1}|x_{1:t})$ for downstream tasks?
  - **Full fine-tuning** (classical ML/Stats approach): find $p_{\theta+\Delta\theta}(x_t|x_{1:t})$ by optimizing over $\Delta\theta$
  - **LoRA**: constrain the rank of the weight matrices in $\Delta\theta$
  - **Prompting**: find a good transformation of the input $x_{1:t} \to \tilde{x}$ and then use $p_\theta(x_{t+1}|\tilde{x})$ without updating the weights

- Common input transformation for prompting: concatenating instruction tokens + providing few-shot demonstration (aka ICL) + question. Example—
  - **Instruction** = "*Classify the sentiment of this review as Positive or Negative*"
  - **Few-shot examples** = "*Tweet: 'I love the new updates!' -> Sentiment: Positive. Tweet: 'This app is so slow today.' -> Sentiment: Negative*"
  - **Question** = "*Tweet: 'The new feature is interesting, but hard to find.' -> Sentiment:*"

The empirical mystery in the GPT-3 age

- GPT-3 is closed-source with an API (no parameter update was allowed), lots of prompting experiments
- LLMs “solve” novel tasks using contexts
  - Following unnatural formats
  - Learning unnatural input-output mapping
- Out-of-distribution (OOD) generalization, but how?



Rong, [Extrapolating to Unnatural Language Processing with GPT-3's In-context Learning](), 2021

Science for emergence and ICL

Major scientific approaches

- “**C**omputer scientist” approach [C]
  - Start from benchmark models or SOTA models
  - Ablation experiments: applying perturbations to model components, training algorithms, or data
- “**P**hysicist” approach [P]
  - Well-controlled synthetic setting, training small transformer on arithmetic data
  - Focus on nontrivial phase transition, asymptotic analysis (often non-rigorous)
- “**M**athematician” approach [M]
  - Manageable, highly-simplified models and training algorithms
  - Typical assumptions: linear attention, one self-attention layer, no layer normalization, specific type of GD, etc
  - Focus on informative error bounds (optimization properties, generalization properties, etc)

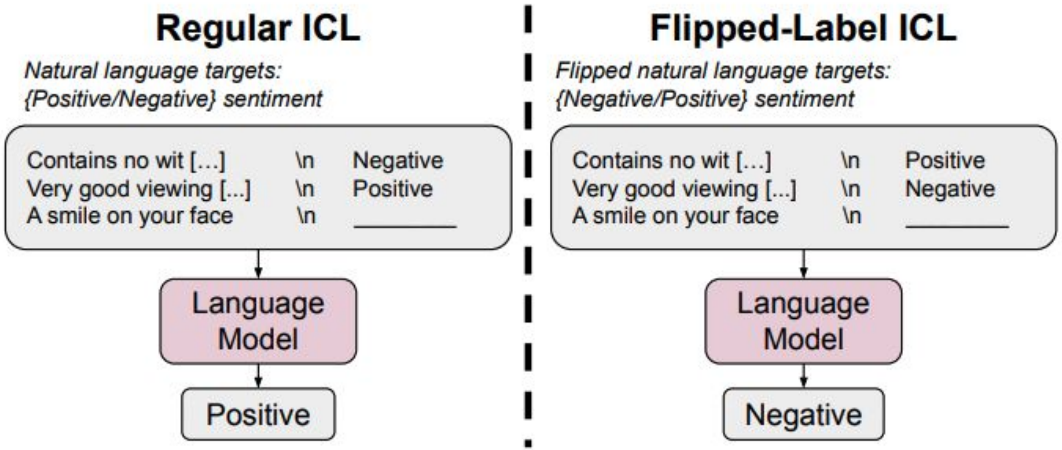
Some clusters of attempts for understanding

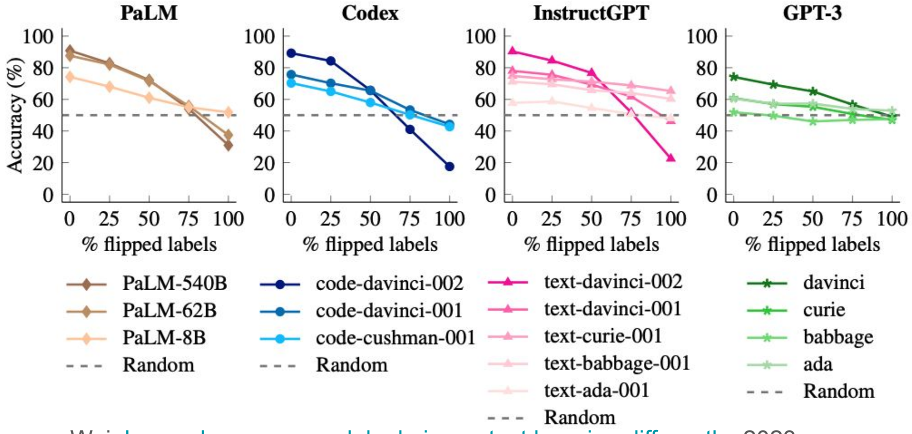
- LLM experiments [C]
- Grokking in modular arithmetic [PM]
- In-context (**IC**) linear regression [PM]
- Induction heads in copying tasks [CPM]

	Core Concept	**Perspective**	**Setting**	**What it Explains?**
LLM Experiments	Probing models with flipped labels or corrupted formats.	**Behavioral**	Informative prompting on **pretrained LLMs**.	**Task inference:** Prompting retrieves tasks or learns new tasks
Grokking	Sudden change in memorization & generalization properties	**Emergence**	Toy models. Mostly **train from scratch**.	**Phase Transitions:** How generalization emerges from training
IC Linear Regression	Learning input-output mapping in context	**Algorithmic**	Toy models. Mostly **train from scratch**.	**Implicit meta-algorithm:** ICL emulate gradient descent in context
Induction Heads	Internal mechanism for solving copying [A][B]...[A] —> [B]	**Mechanistic**	**Both** (Toy models and Pretrained LLMs).	**Internal mechanism:** how do transformers encode copying ability

LLM experiments

- Use unnatural or counterfactual IC examples in the prompt
- Conflicting features
  - Prioritize semantic features won't predict flipped labels
  - Prioritize format/abstract features will predict flipped labels
- Similar to [Stroop effect](#) in psychology
- Finding: large model scales favor predicting flipped labels

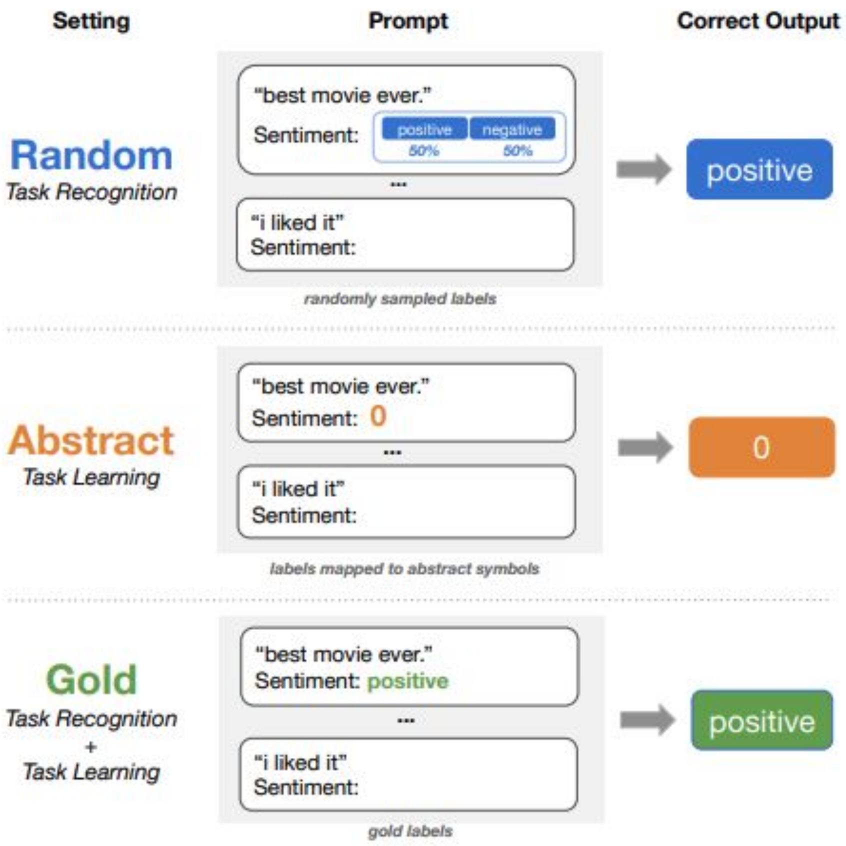




Wei, [Larger language models do in-context learning differently](#), 2023

LLM experiments

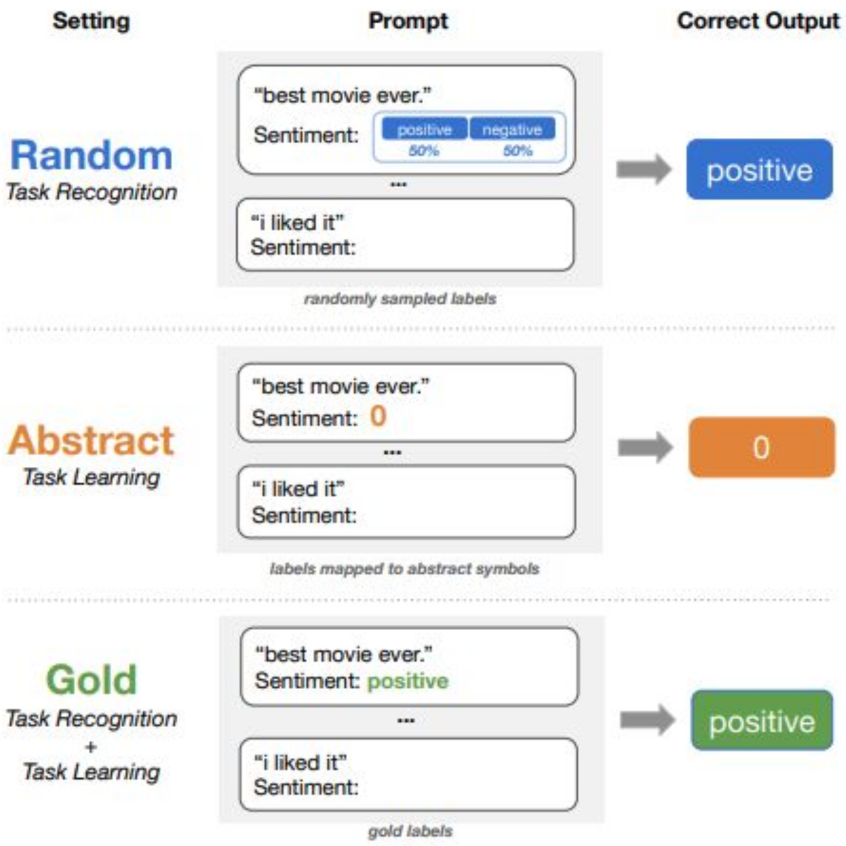
- Two distinct mechanisms coexist in LLMs
  - Task recognition / task retrieval
  - Task learning
- Models can achieve non-trivial performance with task recognition
- Model scales improve task learning
- Empirical evidence for a novel memorization vs generalization tradeoff



Pan, What In-Context Learning "Learns" In-Context: Disentangling Task Recognition and Task Learning, 2023

LLM experiments

- Two distinct mechanisms coexist in LLMs
  - Task recognition / task retrieval
  - Task learning
- Models can achieve non-trivial performance with task recognition
- Model scales improve task learning
- Empirical evidence for a novel memorization vs generalization tradeoff



Pan, What In-Context Learning "Learns" In-Context: Disentangling Task Recognition and Task Learning, 2023

LLM experiments

- An explanation for ICL for task retrieval: the model is doing Bayesian inference over the context [M]
- More IC examples → Posterior distribution concentrates on the right latent concept (e.g., sentiment classification)

$$p(\text{output}|\text{prompt}) = \int_{\text{concept}} p(\text{output}|\text{concept}, \text{prompt})p(\text{concept}|\text{prompt})d(\text{concept}).$$

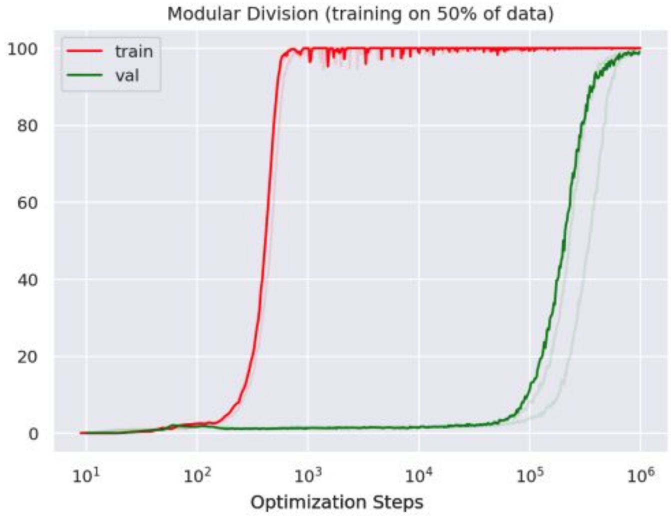
Xie, An Explanation of In-context Learning as Implicit Bayesian Inference, 2022

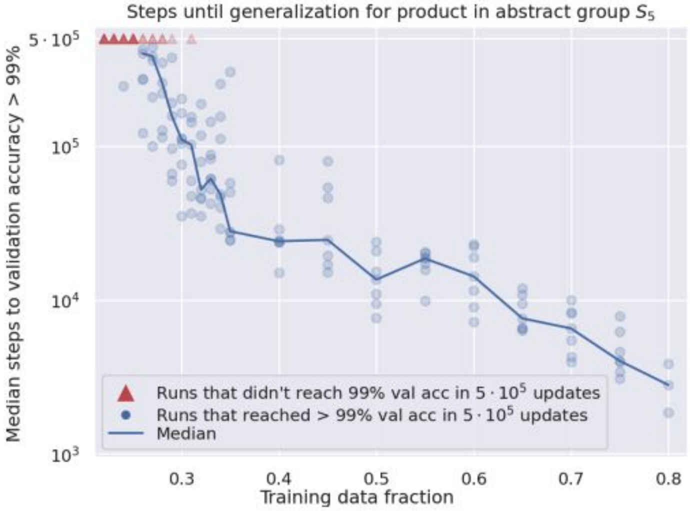
- It does not explain why two mechanisms—task retrieval and taks learning—coexist, how are they encoded by the model, why they emerge (especially task learning) under model scaling
- In later lectures, we will see the two mechanisms are mostly attributable to FFN and self-attention respectively

Grokking in modular arithmetic

- Motivation: transformers learn certain discrete / math structures at scale, why?
- Training smaller transformers from scratch on arithmetic data, e.g.,

$$a \times b = c \mod 97$$

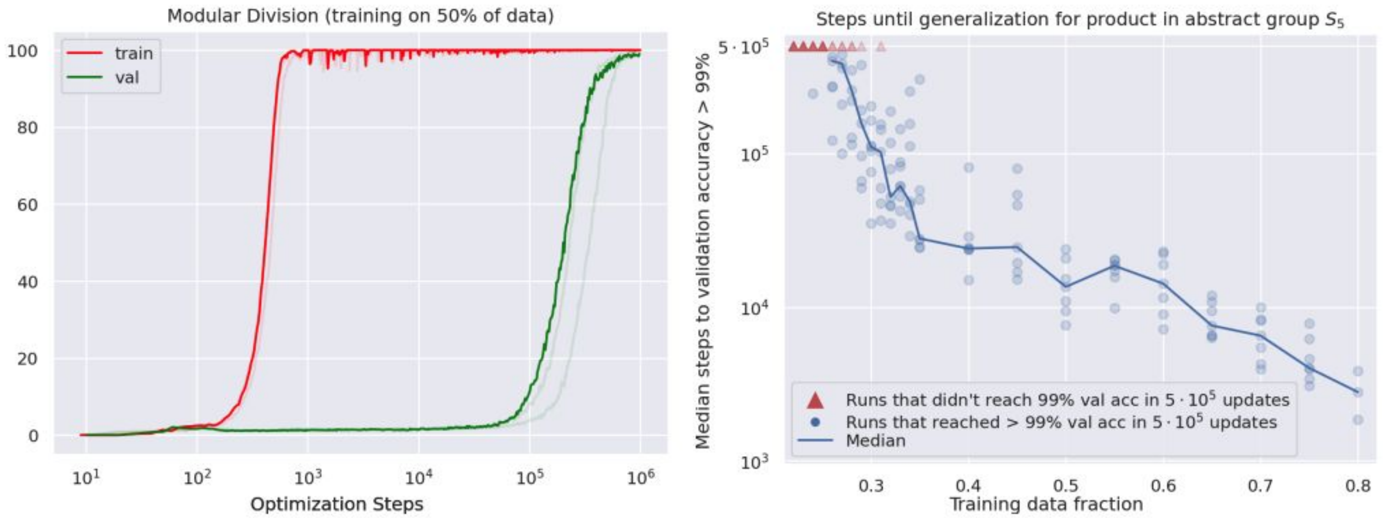




OpenAI, Grokking: Generalization Beyond Overfitting on Small Algorithmic Datasets, 2022

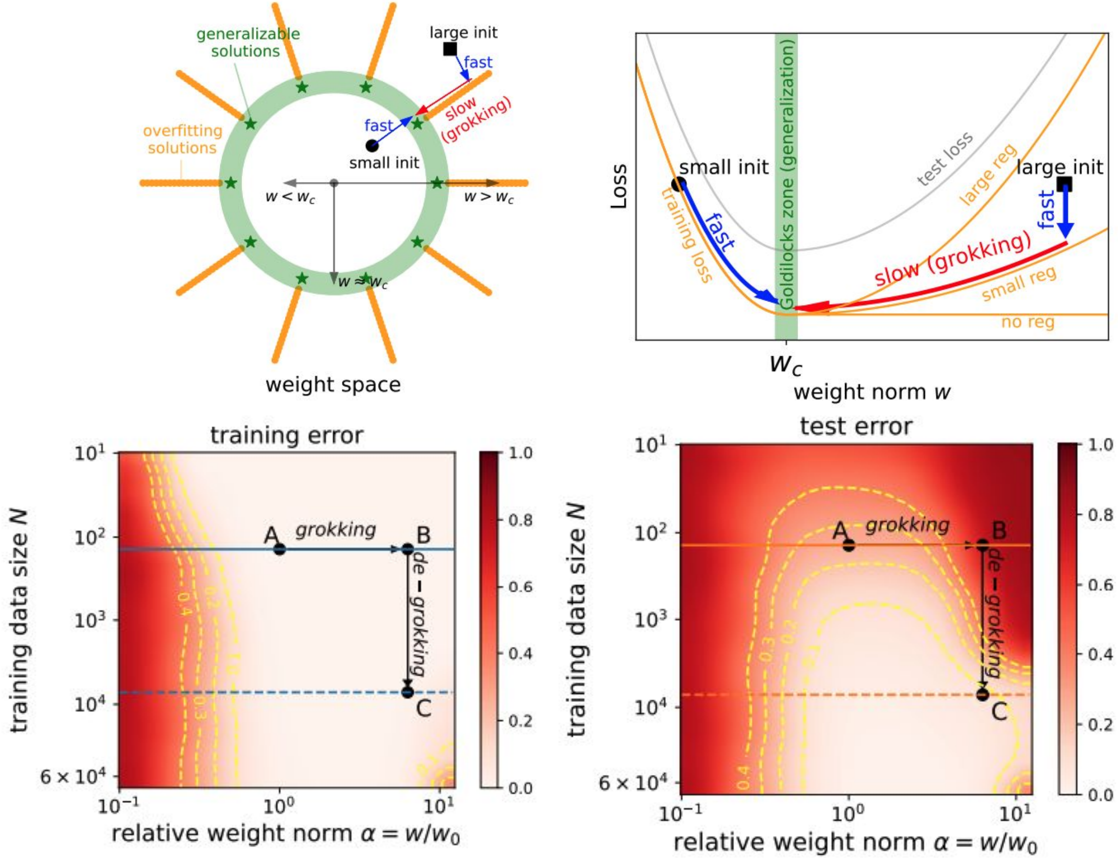
Grokking in modular arithmetic

- Finding 1: Phase change thresholds: interpolating training data much earlier than generalization
- Finding 2: Small training data size means much more training steps required



Grokking in modular arithmetic

- Explanation 1: Loss landscape is affected by multiple factors
    - Small vs large initialization
    - Sample size
    - Regularization
- Overfitting solutions consist of almost flat regions, thus slow at generalization
- [Existing theory]() [M] already compared kernel learning regime vs NTK regime

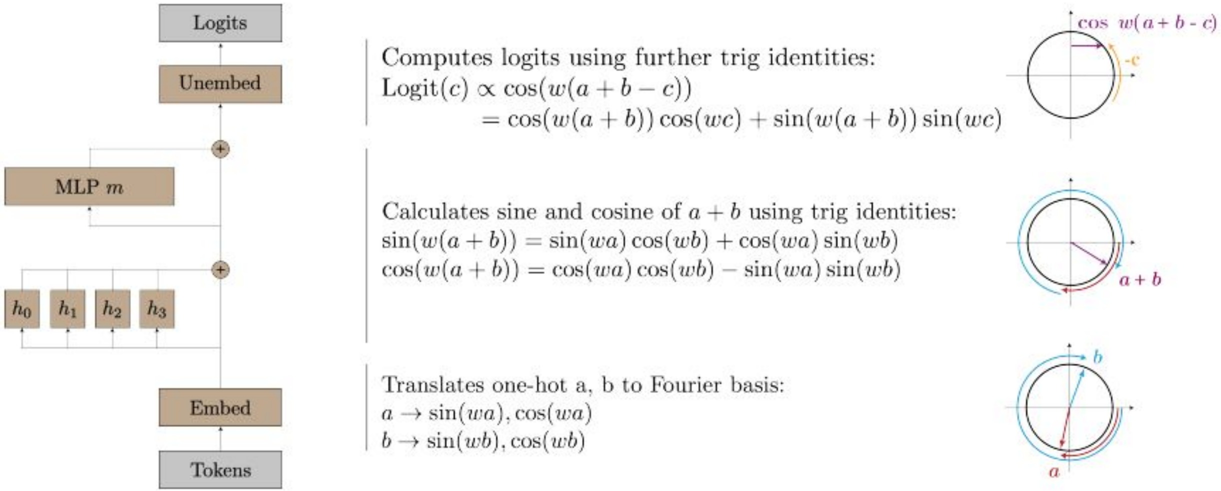


Liu, [Omnigrok: Grokking Beyond Algorithmic](), 2023

Grokking in modular arithmetic

- Explanation 2: mechanistic interpretability (internal representation)
- Model learns to implement algorithms (based on fourier frequency for modular arithmetic) as training progresses

- Circuits (certain model components) are interpretable sub-rules for solving a task
- Further theoretical analysis [M] built upon the finding



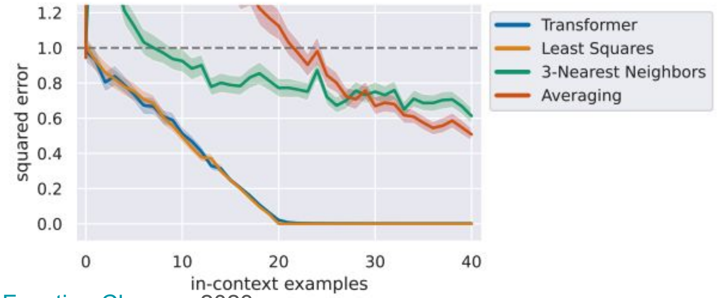
Nanda, Progress Measures for Grokking Via Mechanistic Interpretability, 2023

In-context linear regression

- Motivation: A clean setup of ICL without entanglement of natural languages?
- Learning mapping in context
  - IC linear regression (most studied)
  - IC nonparametric regression

$$\underbrace{\text{maison} \rightarrow \text{house, chat} \rightarrow \text{cat, chien} \rightarrow}_{\text{prompt}} \ \underbrace{\text{dog}}_{\text{completion}} . \qquad P = (x_1, f(x_1), \ldots, x_{k+1}, f(x_{k+1}))$$

- f is a sequence-specific linear function sampled from certain distribution, i.e., the coefficient vector of f is first sampled, then sample IC input-output pairs
- Finding 1: training transformers from scratch yields ICL with near-optimal acc
- Finding 2: somewhat generalize to new function (unseen f during training)



Garg, What Can Transformers Learn In-Context? A Case Study of Simple Function Classes, 2023

In-context linear regression

- Explanation: linear self-attention emulates gradient descent [P]
- One self-attention layer learns a gradient step to update the residual stream
- Loss function $$L(W) = \frac{1}{2N}\sum_{i=1}^{N} \|W x_i - y_i\|^2.$$
- Gradient step $$\Delta W = \sum_i \mathbf{e}_i \otimes \mathbf{x}'_i,$$ re-organize

$$\begin{aligned}
\mathcal{F}(\mathbf{x}) &= (W_0 + \Delta W)\,\mathbf{x} \\
&= W_0\mathbf{x} + \Delta W\mathbf{x} \\
&= W_0\mathbf{x} + \sum_i \left(\mathbf{e}_i \otimes \mathbf{x}'_i\right)\mathbf{x} \\
&= W_0\mathbf{x} + \sum_i \mathbf{e}_i \left(\mathbf{x}_i'^{T}\mathbf{x}\right) \\
&= W_0\mathbf{x} + \operatorname{LinearAttn}\left(E, X', \mathbf{x}\right),
\end{aligned}$$

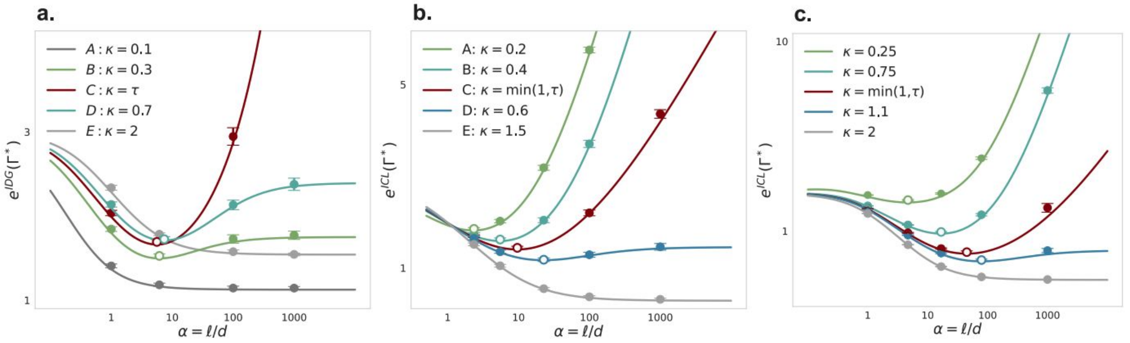
- Theory about training dynamics [M]
  - Explicit formula under simplifying assumption

Oswald, Transformers Learn In-Context by Gradient Descent, 2023
Dai, Why Can GPT Learn In-Context? Language Models Implicitly Perform Gradient Descent as Meta-Optimizers, 2023

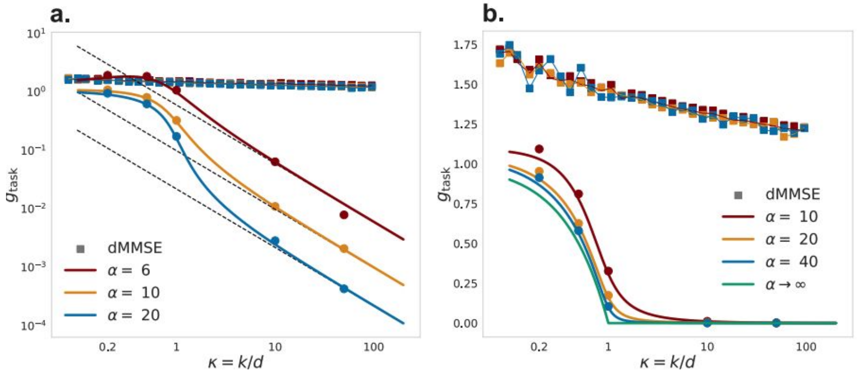
In-context linear regression

- Comprehensive theory (PNAS paper) for one-layer linear self-attention [M]
- Formalizes and analyzes two solutions (task-retrieval solution, task-learning solution)
- Emphasis on the critical role of **task diversity** in phase transition of the two mechanisms

D. ICL and IDG error curves can have non-monotonic dependence on context length

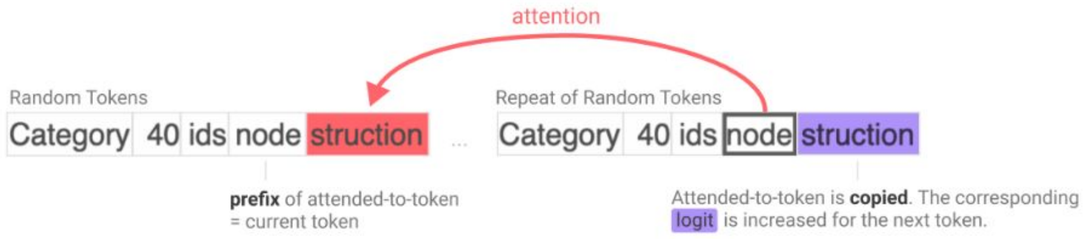


E. Learning transition with increasing pretraining task diversity



Induction heads in copying tasks

- Both verified on large-scale LLMs and synthetic settings [CP] limited [M]
- ICL is attributed to the copying ability [A] [B] … [A] → [B]
- Pioneered by Anthropic
  - Model internal attention pattern
  - A clear interpretable mechanism how copying is encoded by self-attention
  - One abstract (non-knowledge) ability critical to matching format, solving math



- Detailed analysis in the next lecture

Do we reach consensus,
or do puzzles remain?

Open problems & research ideas

- Ambiguity in the definition of emergent abilities? What really is emergence / phase transitions? Critique: "Are Emergent Abilities of Large Language Models a Mirage?"
- Model, data diversity, and training steps may all have impact, suggested by the PNAS theory paper. But analysis is limited.
- Self-attention is viewed as meta-algorithm components capable of implementing certain rules (mechanistic analysis), yet reverse engineering is hard
- In LLMs, the effects of training data is very poorly understood, since it is very expensive to pretrain the model