Theoretical and Algorithmic Foundations of In-Context Learning Using Properly Trained Transformer Models

Presenter: Hongkang Li

FCRC Seminar

Authors: Hongkang Li (RPI), Meng Wang (RPI PI), Songtao Lu (IBM PI), Xiaodong Cui (IBM), Pin-Yu Chen (IBM PI)

Development of deep learning

Take the area of NLP as an example.

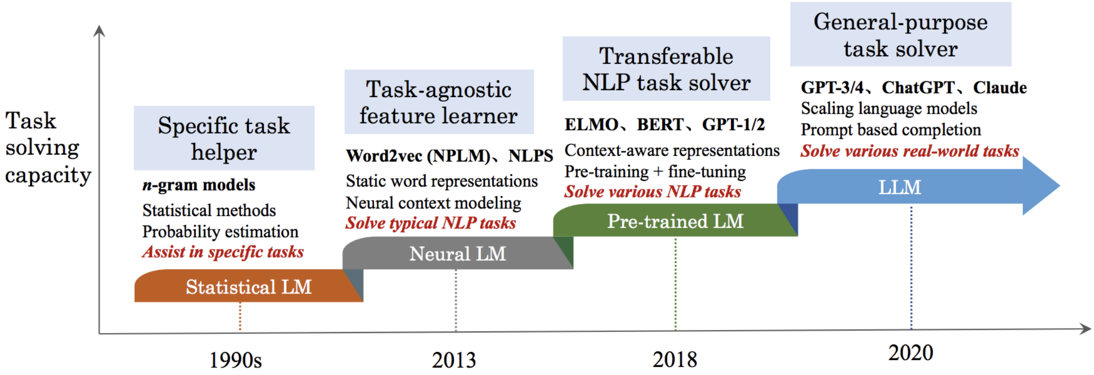

Figure 1: Deep Learning paradigm[1]

[1] source from [Zhao et al.23]

Large Language Model (LLM) and In-context learning (ICL)

- Transformer-based foundation models, e.g., ChatGPT, GPT-4, Sora, have achieved great empirical success in many areas.
- Large foundation models are able to implement **in-context learning (ICL)** and reasoning.

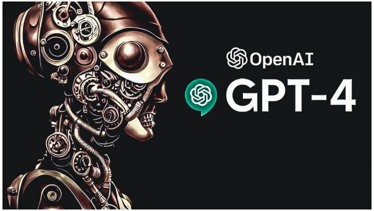

Figure 2: *GPT-4. Source from medium*

Figure 3: *Sora. Source from medium*

Large Language Model (LLM) and In-context learning (ICL)

- In-context learning makes predictions for new tasks on pre-trained LLM without fine-tuning the model.
- It is implemented by providing a few testing examples and necessary instructions as a prompt for the testing data.

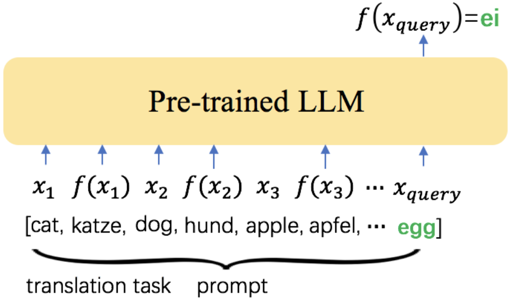

Figure 4: *Machine Translation with ICL*

Our focus

Despite the empirical success of ICL, one fundamental and theoretical question is less investigated, i.e.,

How can a Transformer be trained to perform ICL and generalize in and out of domain successfully and efficiently?

Related works

[Garg et al.22, Akyurek et al. 23] propose a framework for studying ICL on learning linear functions.

- Consider a prompt $P = (x_1, f(x_1), x_2, f(x_2), \cdots, x_{query})$. f is a linear function.
- We say a model M can in-context learn a function f with up to an ϵ error to predict $f(x_{query})$, if

$$\mathbb{E}_P[\ell(M(P), f(x_{query}))] \leq \epsilon. \tag{1}$$

- The model M parameterized by Θ is trained by minimizing the risk function

$$\min_{\Theta} \mathbb{E}_{P,f}[\ell(M_\Theta(P^i), f(x^i_{query}))]. \tag{2}$$

- They show that the trained Transformer is able to learn unseen linear functions from in-context examples with performance comparable to the optimal least squares estimator.

Related works

A few works theoretically study the training dynamics and generalization of Transformers in implementing ICL.

- [Zhang et al.24, Wu et al.24] study linear regression tasks on $\{(x_n, f(x_n))\}_{n=1}^{N}$, where f is a linear function, using the prompt

$$E = \begin{pmatrix} x_1 & x_2 & \cdots & x_l & x_{query} \\ f(x_1) & f(x_2) & \cdots & f(x_l) & 0 \end{pmatrix} \in \mathbb{R}^{(d+1)\times(l+1)}. \tag{3}$$

  The training model they consider is a one-layer Transformer with linear attention,

$$F(E;\Theta) = E + W^{PV} E \cdot E^\top W^{KQ} E. \tag{4}$$

- [Zhang et al.24] further study the generalization when the data/task distribution shift exists; [Wu et al.24] characterize the required number of pretraining tasks for ICL.

Related works

- Given the prompt in (3), [Huang et al.23] explore a one-layer Transformer with softmax attention on learning linear regression tasks, i.e.,

$$F(E;\Theta) = \sum_{i=1}^{N} y_i \mathrm{softmax}(x_i^{\top} \Theta x_{query}) \tag{5}$$

- [Huang et al.23] consider x_i as orthogonal features, following the line of feature-learning analysis.
- [Huang et al.23] in-depth characterize the dynamics of the training process under cases of balanced and imbalanced prompt examples.

Our work and major contributions

Our recent work "Training Nonlinear Transformers for Efficient In-Context Learning: A Theoretical Learning and Generalization Analysis"[2] has the following contributions.

- A theoretical characterization of how to train Transformers with nonlinear attention and nonlinear MLP and to enhance their ICL capability.
- Expand the theoretical understanding of the mechanism of the ICL capability of Transformers.
- Theoretical justification of Magnitude-based Pruning in preserving ICL.

[2] https://arxiv.org/pdf/2402.15607.pdf

Our work and major contributions

Summary of contributions and comparisons with related works.

Theoretical Works	Nonlinear Attention	Nonlinear MLP	Training Analysis	Distribution -Shifted Data	Tasks
[Zhang et al.24]			✓	✓	linear regression
[Huang et al.23]	✓		✓		linear regression
[Wu et al.24]			✓		linear regression
Ours	✓	✓	✓	✓	classification

Table 1: Comparison with existing works about training analysis and generalization guarantee of ICL

Problem formulation

We study binary classification problems. Given the input $\boldsymbol{x}_{query}$, we aim to predict the label $f(\boldsymbol{x}_{query})$ for the task f. We conduct training with constructed prompts $\boldsymbol{P}$ on a model to enable ICL.

$$\boldsymbol{P} = \begin{pmatrix} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \cdots & \boldsymbol{x}_l & \boldsymbol{x}_{query} \\ \boldsymbol{y}_1 & \boldsymbol{y}_2 & \cdots & \boldsymbol{y}_l & 0 \end{pmatrix} := (\boldsymbol{p}_1, \boldsymbol{p}_2, \cdots, \boldsymbol{p}_{query}). \tag{6}$$

- $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ are context inputs and outputs, respectively.
- $\boldsymbol{y}_i = embedding(f(\boldsymbol{x}_i))$ is an embedding of $f(\boldsymbol{x}_i)$. $\boldsymbol{y}_i = \boldsymbol{q}$ if $f(\boldsymbol{x}_i) = +1$. $\boldsymbol{y}_i = -\boldsymbol{q}$ if $f(\boldsymbol{x}_i) = -1$.

Problem formulation

Learning model: a single-head, one-layer Transformer with a self-attention layer and a two-layer perceptron, i.e.,

$$F(\Psi; \boldsymbol{P}) = \boldsymbol{a}^\top \text{Relu}(\boldsymbol{W}_O \sum_{i=1}^{l} \boldsymbol{W}_V \boldsymbol{p}_i \cdot \text{attn}(\Psi; \boldsymbol{P}, i)), \tag{7}$$

$$\text{attn}(\Psi; \boldsymbol{P}, i) = \text{softmax}((\boldsymbol{W}_K \boldsymbol{p}_i)^\top \boldsymbol{W}_Q \boldsymbol{p}_{query})$$

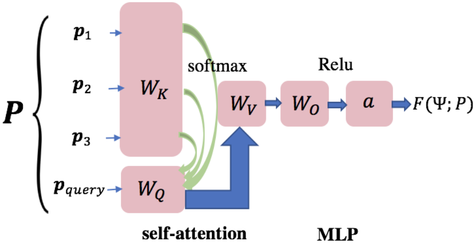

Figure 5: The Transformer network for learning

Problem formulation

Model training: The training is to solve the empirical risk minimization using N pairs of prompt and labels $\{\boldsymbol{P}^n, z^n\}_{n=1}^N$, $\Psi = \{\boldsymbol{W}_Q, \boldsymbol{W}_K, \boldsymbol{W}_V, \boldsymbol{W}_O, \boldsymbol{a}\}$,

$$\min_{\Psi} R_N(\Psi) := \frac{1}{N}\sum_{n=1}^{N} \ell(\Psi; \boldsymbol{P}^n, z^n) \tag{8}$$

- The query and context inputs are sampled from a distribution $\mathcal{D}$.
- The task f^n is sampled from a distribution $\mathcal{T}$. The training tasks form a set $\mathcal{T}_{tr} \subset \mathcal{T}$.
- $\ell(\Psi; \boldsymbol{P}^n, z^n) = \max\{0, 1 - z^n \cdot F(\Psi, \boldsymbol{P}^n)\}$ is the Hinge loss.
- The model is trained via stochastic gradient descent (SGD).

Problem formulation

Generalization: We introduce in-domain and out-of-domain generalization.

- In-domain generalization: No distribution shift between training and testing data. The generalization error is defined as

$$\mathop{\mathbb{E}}_{\boldsymbol{x}_{query} \sim \mathcal{D}, f \in \mathcal{T} \setminus \mathcal{T}_{tr}} [\ell(\Psi; \boldsymbol{P}, z)]. \tag{9}$$

- Out-of-domain generalization: The testing queries follow $\mathcal{D}' \neq \mathcal{D}$, and the testing tasks follow $\mathcal{T}' \neq \mathcal{T}$. The generalization error is defined as

$$\mathop{\mathbb{E}}_{\boldsymbol{x}_{query} \sim \mathcal{D}', f \in \mathcal{T}'} [\ell(\Psi; \boldsymbol{P}, z)]. \tag{10}$$

Problem formulation

Model pruning:

- Let $\mathcal{S} \in [m]$ be the index set of $\boldsymbol{W}_O$ neurons.
- Pruning neurons in $\mathcal{S}$: removing corresponding rows of the trained $\boldsymbol{W}_O$.

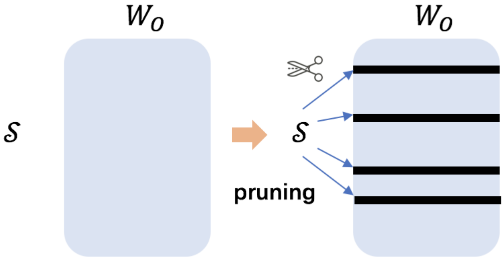

Figure 6: Pruning on $\boldsymbol{W}_O$.

Formulating data and tasks

In-domain data:

- $\{\boldsymbol{\mu}_j\}_{j=1}^{M_1}$: in-domain-relevant (IDR) pattern; $\{\boldsymbol{\nu}_j\}_{j=1}^{M_2}$: in-domain-irrelevant (IDI) pattern.
- IDR and IDI patterns are orthogonal.
- For a constant κ, each in-domain data

$$\boldsymbol{x} = \boldsymbol{\mu}_j + \kappa \boldsymbol{\nu}_k \tag{11}$$

In-domain tasks: A task based on $\boldsymbol{\mu}_a$ and $\boldsymbol{\mu}_b$ is defined as

- $f(\boldsymbol{x}) = +1$ (or -1) if the IDR pattern of $\boldsymbol{x}$ is $\boldsymbol{\mu}_a$ (or $\boldsymbol{\mu}_b$).
- $f(\boldsymbol{x})$ is randomly and equally chosen from $+1$ and -1 in other cases.

Formulating data and tasks

Out-of-domain data:

- $\{\boldsymbol{\mu}'_j\}_{j=1}^{M_1}$: out-of-domain-relevant (ODR) pattern; $\{\boldsymbol{\nu}'_j\}_{j=1}^{M_2}$: out-of-domain-irrelevant (ODI) pattern. ODR and ODI patterns are orthogonal.
- For a constant κ', each out-of-domain data

$$\boldsymbol{x} = \boldsymbol{\mu}'_j + \kappa' \boldsymbol{\nu}'_k \tag{12}$$

Out-of-domain tasks: A task based on $\boldsymbol{\mu}'_a$ and $\boldsymbol{\mu}'_b$ is defined as

- $f(\boldsymbol{x}) = +1$ (or -1) if the ODR pattern of $\boldsymbol{x}$ is $\boldsymbol{\mu}'_a$ (or $\boldsymbol{\mu}'_b$).
- $f(\boldsymbol{x})$ is randomly and equally chosen from $+1$ and -1 in other cases.

Formulating data and tasks

Prompt input selection:
For the training task based on $\boldsymbol{\mu}_a$ and $\boldsymbol{\mu}_b$,

- With a probability of $\alpha/2$, select examples of $\boldsymbol{\mu}_a$ and $\boldsymbol{\mu}_b$.
- With a probability of $(1-\alpha)/(M_1-2)$, select examples of other IDR patterns.

For the testing task based on $\boldsymbol{\mu}_a$ and $\boldsymbol{\mu}_b$ (or $\boldsymbol{\mu}'_a$ and $\boldsymbol{\mu}'_b$), assume at least $\alpha'/2$ fraction of context inputs contain the same IDR (or ODR) pattern as the query.

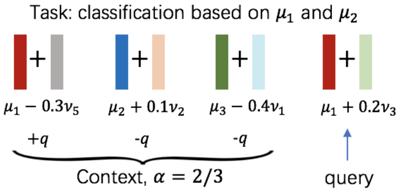

Figure 7: *Example of prompt,* $\alpha = 2/3$.

Main theoretical results

Theorem 1 (In-domain generalization)

For any $\epsilon > 0$, as long as

1. *the training tasks $\mathcal{T}_{tr}$ uniformly cover all the IDR patterns and labels with $|\mathcal{T}_{tr}|/|\mathcal{T}| \geq (M_1 - 1)^{-1/2}$, which means training a small fraction of the total tasks is sufficient,*
2. *the lengths of training and testing prompts $l_{tr} \geq \Omega(\alpha^{-1})$, $l_{ts} \geq \alpha'^{-1}$,*
3. *and the number of iterations $T = \Theta(\alpha^{-2/3})$,*

then with a high probability, the in-domain generalization error of the returned model is less than $\mathcal{O}(\epsilon)$.

Main theoretical results

Consider each ODR pattern as a linear combination of IDR patterns. Denote S_1 as the summation of the linear coefficients.

Theorem 2 (Out-of-domain generalization)

Suppose that the conditions (1) to (3) in Theorem 1 hold. If

- $S_1 \geq 1$,
- *each ODI pattern is in the subspace spanned by IDI patterns,*

then with a high probability, the out-of-domain generalization error of the returned model is less than $\mathcal{O}(\epsilon)$.

Main theoretical results

Theorem 3 (Model pruning)

- *There exists a constant fraction of MLP-layer neurons of* $\boldsymbol{W}_O$ *with large weights, while the remaining have small weights.*
- *Pruning all neurons with small weights leads to a generalization error* $\mathcal{O}(\epsilon + M_1^{-1/2})$*, which is almost the same as without pruning.*
- *Pruning an* R *fraction of neurons with large weights results in a generalization error greater than* $\Omega(R)$*.*

ICL mechanism by the trained transformer

Proposition 1

- $\boldsymbol{W}_Q^{(T)}$ *and* $\boldsymbol{W}_K^{(T)}$ *mainly project context inputs to the IDR or ODR pattern.*
- *After training, attention weights become concentrated on contexts that share the same IDR/ODR pattern as the query.*

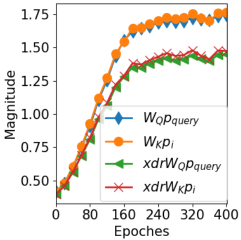

Figure 8: *The magnitude of the trained attention layer. xdr: IDR or ODR pattern of* p_{query}.

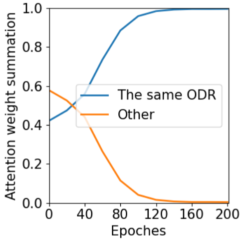

Figure 9: *The attention weight summation*

ICL mechanism by the trained transformer

Proposition 2

- *The feature embedding of rows of $\boldsymbol{W}_O^{(T)}\boldsymbol{W}_V^{(T)}$ approximate $\bar{\mu}$, i.e., the average of IDR patterns.*
- *The label embedding of rows $\boldsymbol{W}_O^{(T)}\boldsymbol{W}_V^{(T)}$ approximate $\boldsymbol{q}$ for positive neurons and $-\boldsymbol{q}$ for negative neurons.*

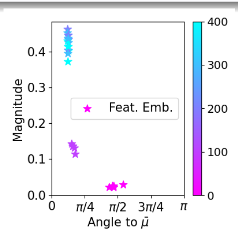

Figure 10: ***The feature embedding of $W_O W_V$. bar: iteration***

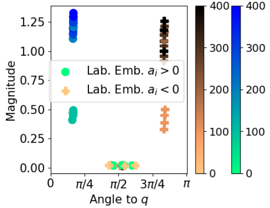

Figure 11: ***The label embedding of $W_O W_V$. bars: iterations***

Results of multi-layer Transformers (3-layer).

- Each attention layer selects contexts with the same IDR pattern as the query.

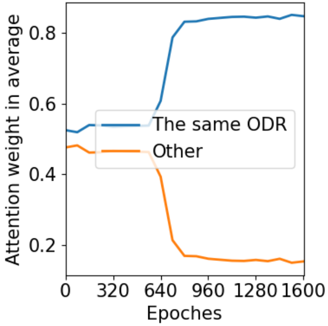

Figure 12: Layer 1 self-attention

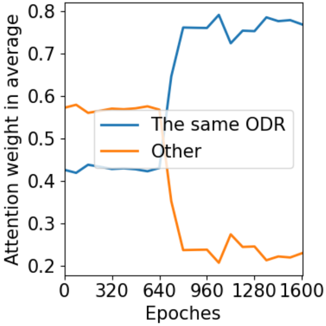

Figure 13: Layer 2 self-attention

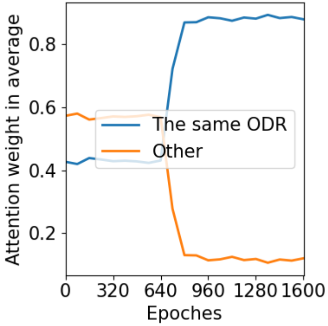

Figure 14: Layer 3 self-attention

ICL mechanism by the trained transformer

Results of multi-layer Transformers (3-layer).

- The magnitude of the majority of neurons increases along the training.
- The angle changes still hold for one of the layers.

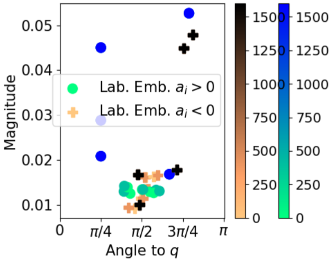

Figure 15: Layer 1 self-attention

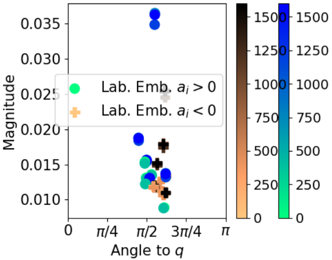

Figure 16: Layer 2 self-attention

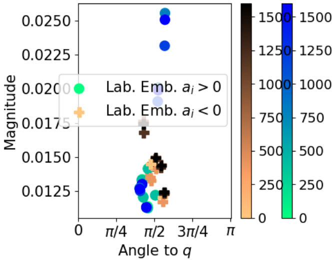

Figure 17: Layer 3 self-attention

Numerical experiments

Verifying the sufficient conditions for out-of-domain generalization.

- $S_1 \geq 1$ is needed for a desired out-of-domain generalization.
- The required length of testing prompts decreases as α' increases.

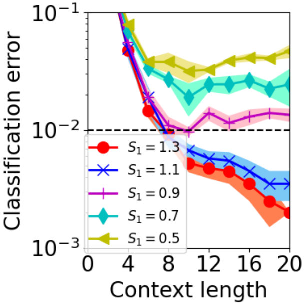

Figure 18: Out-of-domain ICL classification error on GPT-2 with different S_1

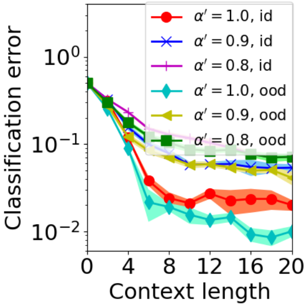

Figure 19: Out-of-domain ICL classification error on GPT-2 with different α'

Numerical experiments

Comparing ICL on a one-layer Transformer with other machine learning algorithms.

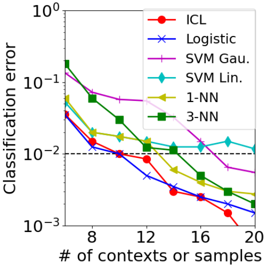

Figure 20: Binary classification performance of using different algorithms, $\alpha' = 0.8$

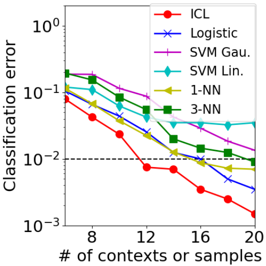

Figure 21: Binary classification performance of using different algorithms, $\alpha' = 0.6$

- Logistic: logistic regression; SVM Gau.: SVM with Gaussian kernel; SVM Lin.: SVM with linear kernel; 1-NN: 1-nearest neighbor; 3-NN: 3-nearest neighbor.

Magnitude-based model pruning for out-of-domain ICL inference.

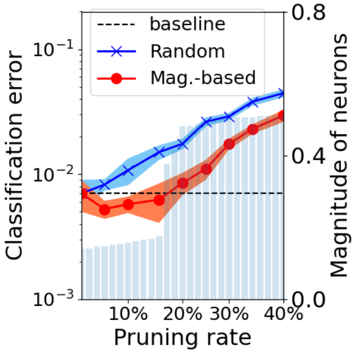

Figure 22: *Out-of-domain classification error with model pruning of the trained W_O and the magnitude of W_O neurons.*

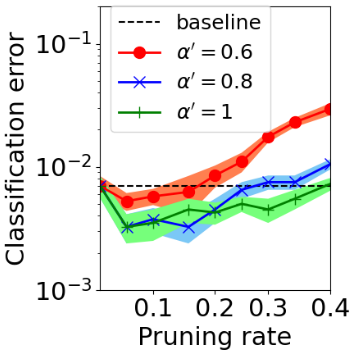

Figure 23: *Out-of-domain classification error with different α'*

Summary

- This work provides theoretical analyses of the training dynamics of Transformers with nonlinear attention and nonlinear MLP, and the resulting ICL capability for new tasks with possible data shift.

- This work also provides a theoretical justification for magnitude-based pruning to reduce inference costs while maintaining the ICL capability.

- This work provably characterizes the mechanism of ICL implemented by a single-head, one-layer Transformer.

Further exploration in LLM reasoning ability

Reasoning problems

Arithmetic Reasoning (AR) $(+\,-\,\times\div\ldots)$	Symbolic Reasoning (SR)	Commonsense Reasoning (CR)
Q: If there are 3 cars in the parking lot and 2 more cars arrive, how many cars are in the parking lot? A: The answer is **5**	Q: Take the last letters of the words in "Elon Musk" and concatenate them A: The answer is **nk.**	Q: What home entertainment equipment requires cable? Answer Choices: (a) radio shack (b) substation (c) television (d) cabinet A: The answer is **(c).**

Can Transformer-based LLM solve reasoning problems?

Chain-of-Thought (COT)

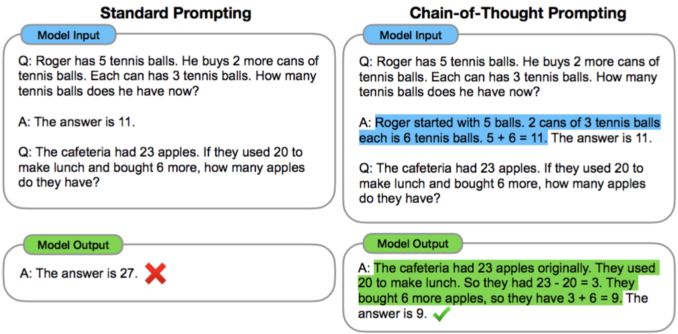

Figure 24: Few-shot COT [Wet et al.22]

Relationship with ICL: prompting multiple steps of reasoning.

Further exploration in LLM reasoning ability

Existing works focus on the expressive power of Transformer in implementing COT.

- [Li et el.23]: COT=Filtering+ICL.
- [Zhang et al.23, Li et al.23]: Transformers can be constructed to solve many reasoning problems via COT.
- [Yang et al.24]: Linear Transformers can be more efficient than softmax Transformers in some dynamic programming tasks.

Problems to solve:

- How can a Transformer be trained to perform COT?
- When is COT better than ICL?
- Generalization with Data/Task distribution shift.
- Linear Transformer vs Softmax Transformer.

Thank you!

Q & A

Wayne Xin Zhao, Kun Zhou*, Junyi Li*, Tianyi Tang, Xiaolei Wang, et al.
A Survey of Large Language Models
https://arxiv.org/pdf/2303.18223.pdf

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, et al.
Language Models are Few-Shot Learners
OpenAI.

Shivam Garg, Dimitris Tsipras, Percy Liang, Gregory Valiant
What Can Transformers Learn In-Context? A Case Study of Simple Function Classes.
In *Advances in Neural Information Processing Systems 2022.*

Ekin Akyurek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, Denny Zhou
What learning algorithm is in-context learning? Investigations with linear models
In *International conference on Learning Representations 2023.*

Ruiqi Zhang, Spencer Frei, Peter L. Bartlett
Trained transformers learn linear models in-context
In *Journal of Machine Learning Research*

Jingfeng Wu, Difan Zou, Zixiang Chen, Vladimir Braverman, Quanquan Gu, Peter L. Bartlett
How many pretraining tasks are needed for in-context learning of linear regression?
In *International conference on Learning Representations 2024.*

Yu Huang, Yuan Cheng, Yingbin Liang
In-context convergence of transformers.
In *NeurIPS 2023 Workshop on Mathematics of Modern Machine Learning.*

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter et al.
Chain-of-Thought Prompting Elicits Reasoning in Large Language Models
In *Neurips 2022.*

Yingcong Li, Kartik Sreenivasan, Angeliki Giannou, Dimitris Papailiopoulos, Samet Oymak
Dissecting Chain-of-Thought: Compositionality through In-Context Filtering and Learning
In *Neurips 2023.*

Zhiyuan Li, Hong Liu, Denny Zhou, Tengyu Ma
Chain of Thought Empowers Transformers to Solve Inherently Serial Problems

In *International conference on Learning Representations 2024.*

Guhao Feng, Bohang Zhang, Yuntian Gu, Haotian Ye, Di He, Liwei Wang
Towards Revealing the Mystery behind Chain of Thought: A Theoretical Perspective
In *Neurips 2023.*

Kai Yang, Jan Ackermann, Zhenyu He, Guhao Feng, Bohang Zhang, Yunzhen Feng, Qiwei Ye, Di He, and Liwei Wang.
Do efficient transformers really save computation?
https://arxiv.org/pdf/2402.13934.pdf