In-Context Learning

REFORM Reading Group

March 12, 2025

Credits: I am mostly just the messenger

Many many slides shamelessly stolen from SK

Many paper summaries taken from ST

Many slides taken from SC

Thank you!

What is in-context learning?

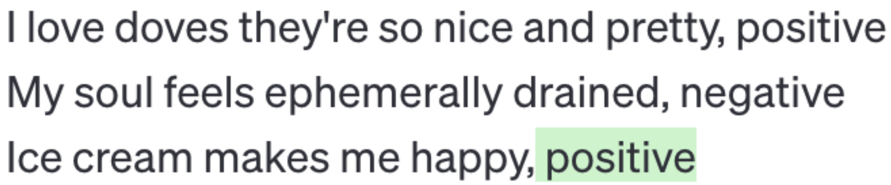
Goal: To perform some mapping from input ⇒ output with few examples

What is in-context learning?

Goal: To perform some mapping from input ⇒ output with few examples

What is in-context learning?

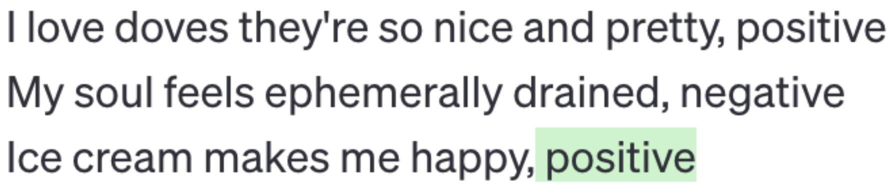
Goal: To perform some mapping from input ⇒ output with few examples



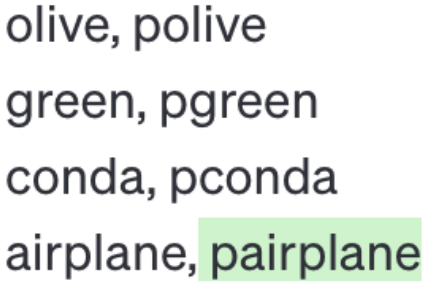
I love doves they're so nice and pretty, positive
My soul feels ephemerally drained, negative
Ice cream makes me happy, positive

What is in-context learning?

Goal: To perform some mapping from input ⇒ output with few examples



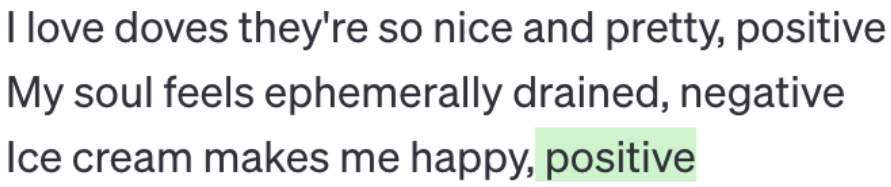
I love doves they're so nice and pretty, positive
My soul feels ephemerally drained, negative
Ice cream makes me happy, positive



olive, polive
green, pgreen
conda, pconda
airplane, pairplane

What is in-context learning?

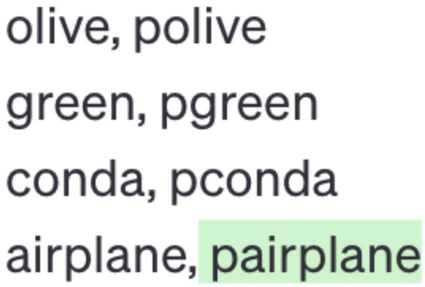
Goal: To perform some mapping from input ⇒ output with few examples



I love doves they're so nice and pretty, positive

My soul feels ephemerally drained, negative

Ice cream makes me happy, positive

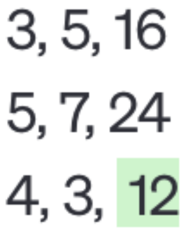


olive, polive

green, pgreen

conda, pconda

airplane, pairplane



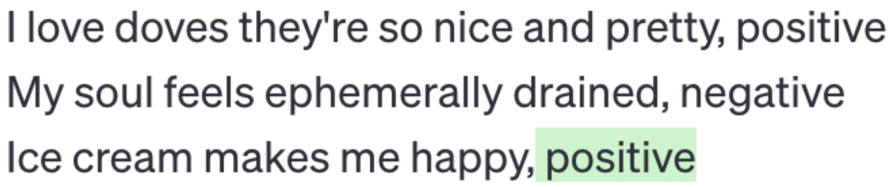
3, 5, 16

5, 7, 24

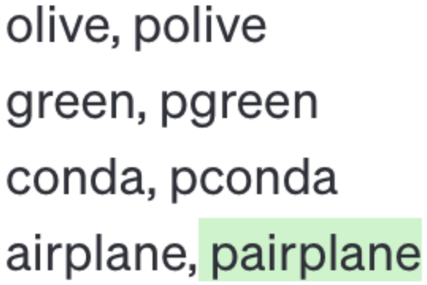
4, 3, 12

What is in-context learning?

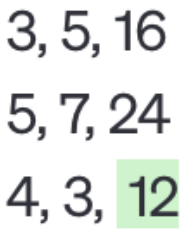
Goal: To perform some mapping from input ⇒ output with few examples



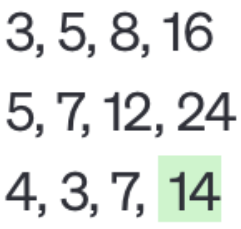
I love doves they're so nice and pretty, positive
My soul feels ephemerally drained, negative
Ice cream makes me happy, positive



olive, polive
green, pgreen
conda, pconda
airplane, pairplane



3, 5, 16
5, 7, 24
4, 3, 12

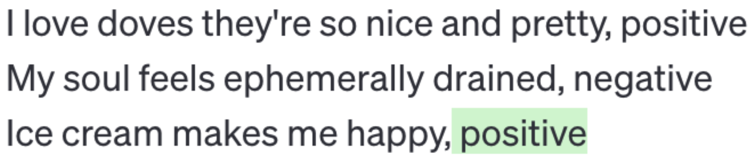


3, 5, 8, 16
5, 7, 12, 24
4, 3, 7, 14

Two broad forms:

Task Recognition: Has seen task before, figures out it needs to do that task

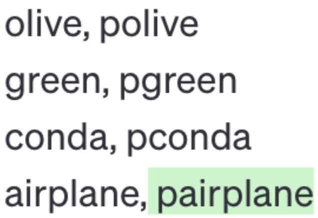
Task recognition toy model: HMM’s



I love doves they're so nice and pretty, positive
My soul feels ephemerally drained, negative
Ice cream makes me happy, positive

Task Learning: Has never seen task, learns the pattern

Task learning toy model: Linear regression



olive, polive
green, pgreen
conda, pconda
airplane, pairplane

Style 1: Task Recognition

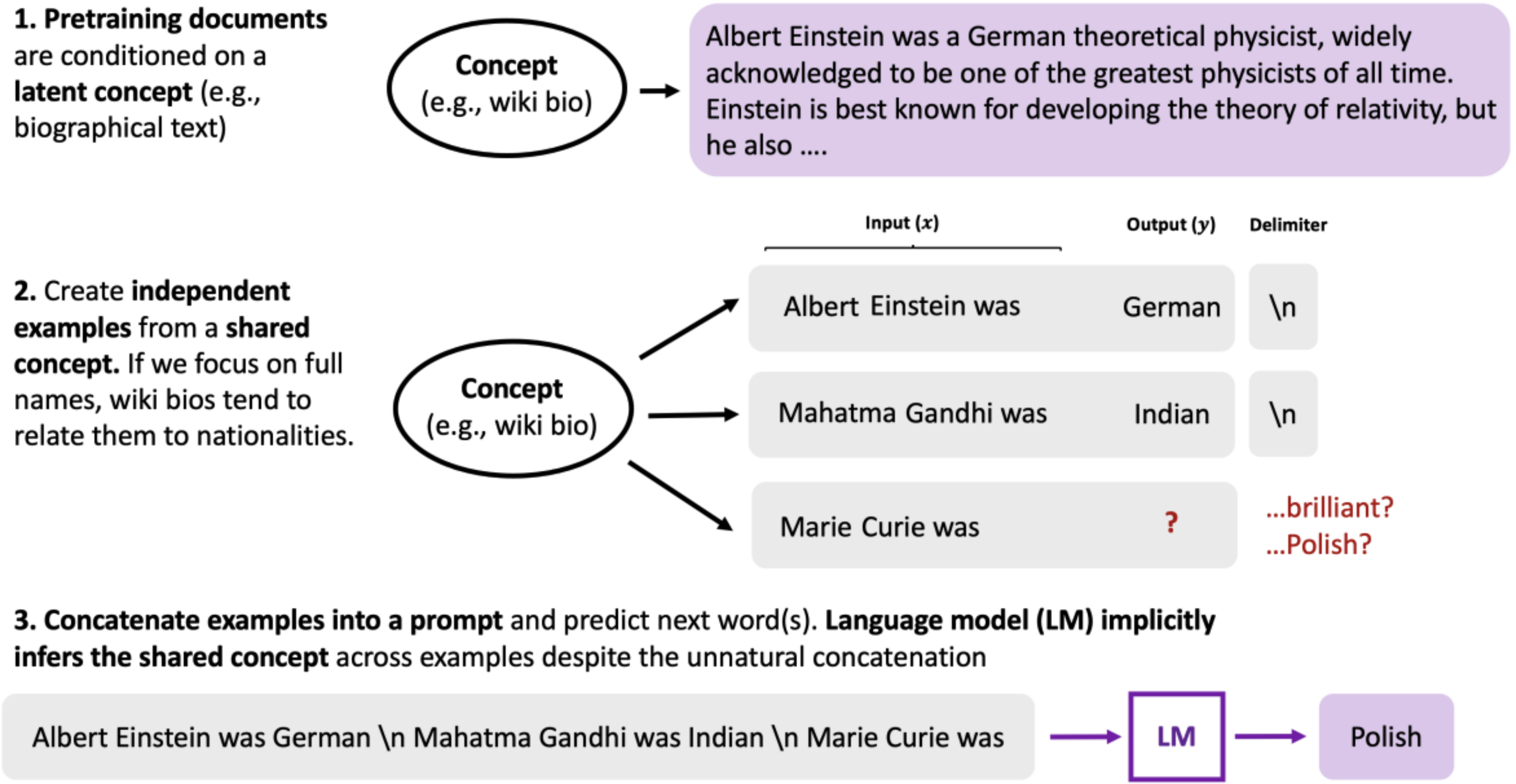
Implicit Inference Hypothesis

Consider an implicit concept θ which represents the “task”

Assume pretraining documents are generated in the following manner

- Sample a concept from a prior
- Sample a document from HMM with this parameter

Example: if the concept is sentiment analysis, then given a movie review, the model will produce the sentiment of the movie review



1. Pretraining documents are conditioned on a **latent concept** (e.g., biographical text)

Concept (e.g., wiki bio) → Albert Einstein was a German theoretical physicist, widely acknowledged to be one of the greatest physicists of all time. Einstein is best known for developing the theory of relativity, but he also ….

2. Create **independent examples** from a **shared concept.** If we focus on full names, wiki bios tend to relate them to nationalities.

Concept (e.g., wiki bio)

Input (x)	Output (y)	Delimiter
Albert Einstein was	German	\n
Mahatma Gandhi was	Indian	\n
Marie Curie was	?	…brilliant? …Polish?

3. Concatenate examples into a prompt and predict next word(s). **Language model (LM) implicitly infers the shared concept** across examples despite the unnatural concatenation

Albert Einstein was German \n Mahatma Gandhi was Indian \n Marie Curie was → LM → Polish

In-context learning as implicit Bayesian inference

“An Explanation of In-context Learning as Implicit Bayesian Inference” [Xie, Raghunathan, Liang, and Ma, 2022]

LLMs aren’t *explicitly* trained to perform in-context learning. So, how do they do it?

Hypothesis:

1. During pre-training, model is forced to learn latent concepts that span multiple sentences/paragraphs

2. So, in-context learning arises from learning *shared prompt concept* across examples

In-context learning as implicit Bayesian inference

Hypothesis: During pre-training, model is forced to learn latent concepts that span multiple sentences/paragraphs. So, in-context learning arises from learning *shared prompt concept* across examples.

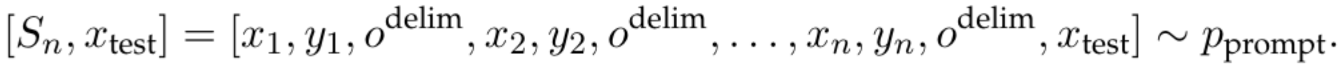
$$p(\text{output}|\text{prompt}) = \int_{\text{concept}} p(\text{output}|\text{concept}, \text{prompt})p(\text{concept}|\text{prompt})d(\text{concept}).$$

If $p(\text{concept} \,|\, \text{prompt})$ concentrates on the prompt concept with more examples, then the LM learns via marginalization by “selecting” the prompt concept

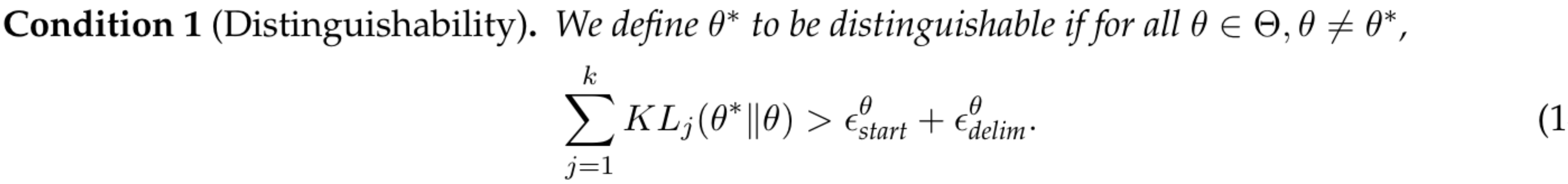
Thus, in-context learning —> LM implicitly performing Bayesian inference.

Proving hypothesis in HMM setting

Use HMM to model concepts and output generation



$$[S_n, x_{\text{test}}] = [x_1, y_1, o^{\text{delim}}, x_2, y_2, o^{\text{delim}}, \ldots, x_n, y_n, o^{\text{delim}}, x_{\text{test}}] \sim p_{\text{prompt}}.$$

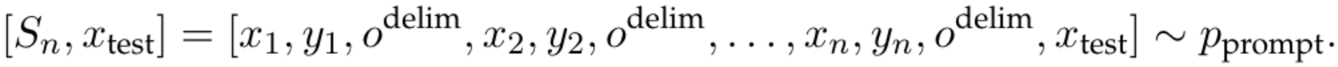


Condition 1 (Distinguishability). *We define θ^* to be distinguishable if for all $\theta \in \Theta, \theta \neq \theta^*$,*

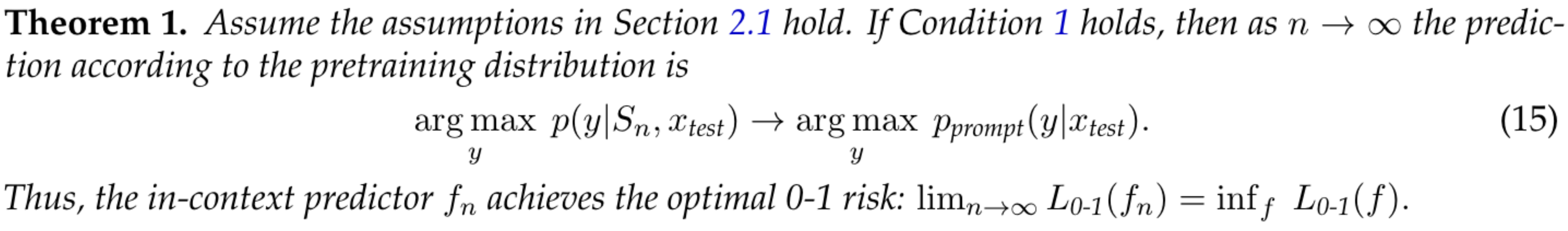
$$\sum_{j=1}^{k} KL_j(\theta^* \| \theta) > \epsilon^{\theta}_{start} + \epsilon^{\theta}_{delim}. \quad (1$$

Proving hypothesis in HMM setting

Use HMM to model concepts and output generation



$$[S_n, x_{\text{test}}] = [x_1, y_1, o^{\text{delim}}, x_2, y_2, o^{\text{delim}}, \ldots, x_n, y_n, o^{\text{delim}}, x_{\text{test}}] \sim p_{\text{prompt}}.$$

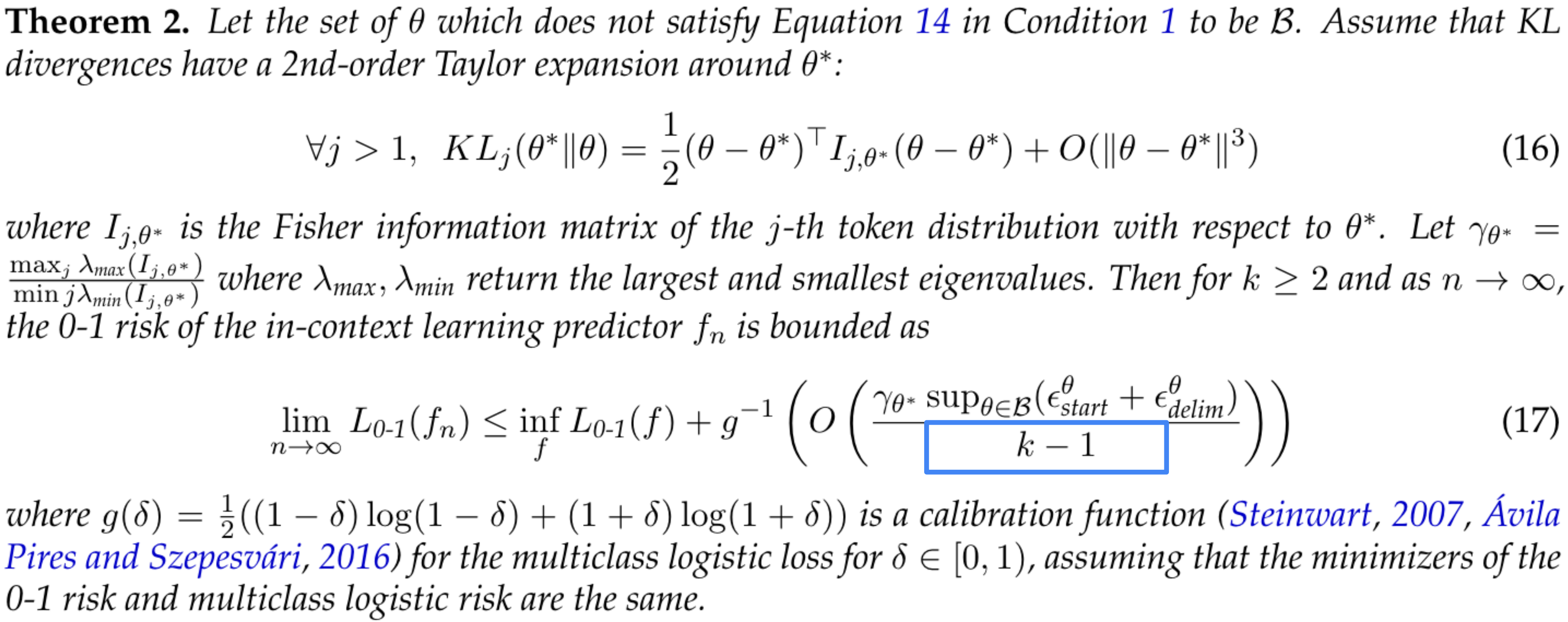


Theorem 1. *Assume the assumptions in Section 2.1 hold. If Condition 1 holds, then as $n \to \infty$ the prediction according to the pretraining distribution is*

$$\arg\max_y \; p(y|S_n, x_{test}) \to \arg\max_y \; p_{prompt}(y|x_{test}). \tag{15}$$

Thus, the in-context predictor f_n achieves the optimal 0-1 risk: $\lim_{n\to\infty} L_{0\text{-}1}(f_n) = \inf_f \; L_{0\text{-}1}(f)$.

Also show error decrease approximately inversely in example length k



Theorem 2. *Let the set of θ which does not satisfy Equation 14 in Condition 1 to be $\mathcal{B}$. Assume that KL divergences have a 2nd-order Taylor expansion around θ^*:*

$$\forall j > 1, \;\; KL_j(\theta^* \| \theta) = \frac{1}{2}(\theta - \theta^*)^\top I_{j,\theta^*}(\theta - \theta^*) + O(\|\theta - \theta^*\|^3) \tag{16}$$

where I_{j,θ^*} is the Fisher information matrix of the j-th token distribution with respect to θ^*. Let $\gamma_{\theta^*} = \frac{\max_j \lambda_{max}(I_{j,\theta^*})}{\min j \lambda_{min}(I_{j,\theta^*})}$ where $\lambda_{max}, \lambda_{min}$ return the largest and smallest eigenvalues. Then for $k \geq 2$ and as $n \to \infty$, the 0-1 risk of the in-context learning predictor f_n is bounded as

$$\lim_{n \to \infty} L_{0\text{-}1}(f_n) \leq \inf_f L_{0\text{-}1}(f) + g^{-1}\left(O\left(\frac{\gamma_{\theta^*} \sup_{\theta \in \mathcal{B}}(\epsilon^\theta_{start} + \epsilon^\theta_{delim})}{\boxed{k-1}}\right)\right) \tag{17}$$

where $g(\delta) = \frac{1}{2}((1-\delta)\log(1-\delta) + (1+\delta)\log(1+\delta))$ is a calibration function (Steinwart, 2007, Ávila Pires and Szepesvári, 2016) for the multiclass logistic loss for $\delta \in [0, 1)$, assuming that the minimizers of the 0-1 risk and multiclass logistic risk are the same.

Model of in-context learning holds

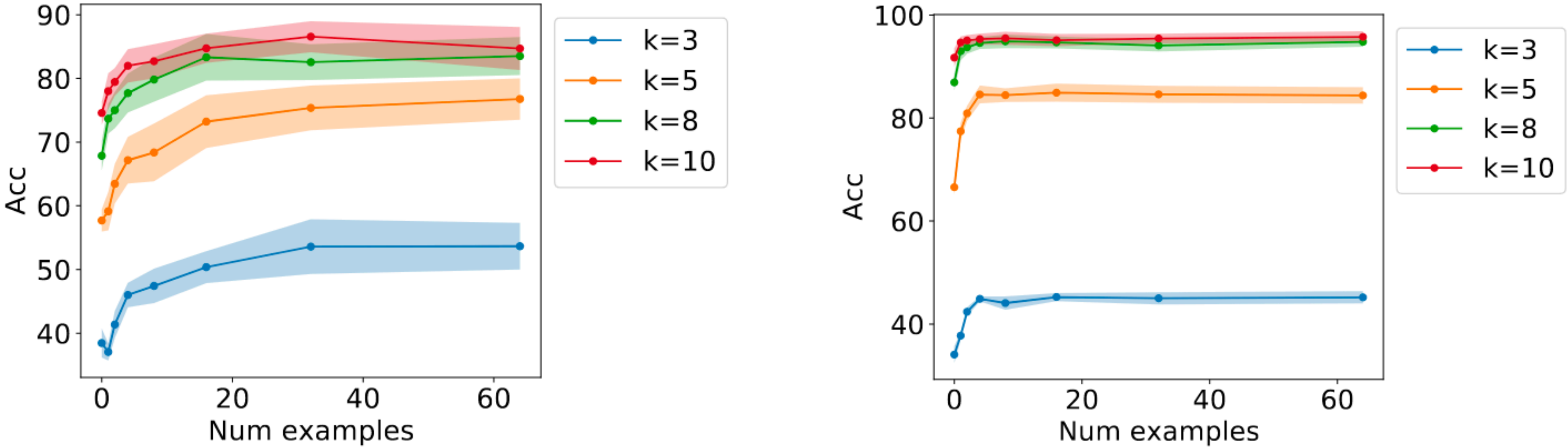


Figure 3: In-context accuracy (95% intervals) of Transformers (left) and LSTMs (right) on the GINC dataset. Accuracy increases with number of examples n and length of each example k.

Model of in-context learning holds

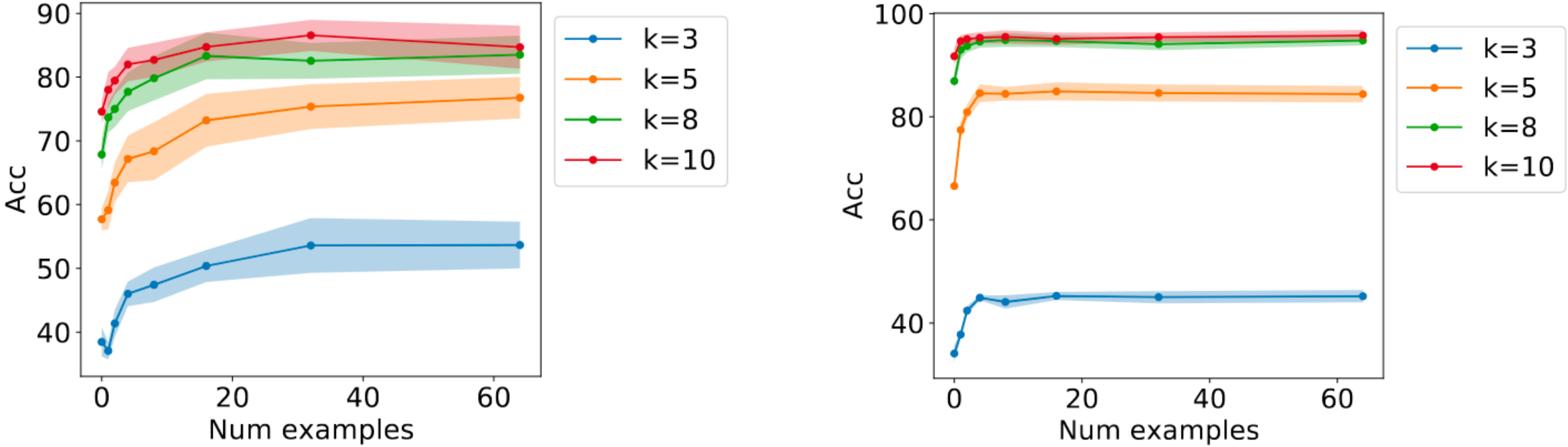


Figure 3: In-context accuracy (95% intervals) of Transformers (left) and LSTMs (right) on the GINC dataset. Accuracy increases with number of examples n and length of each example k.

In-context learning fails when model might expect it

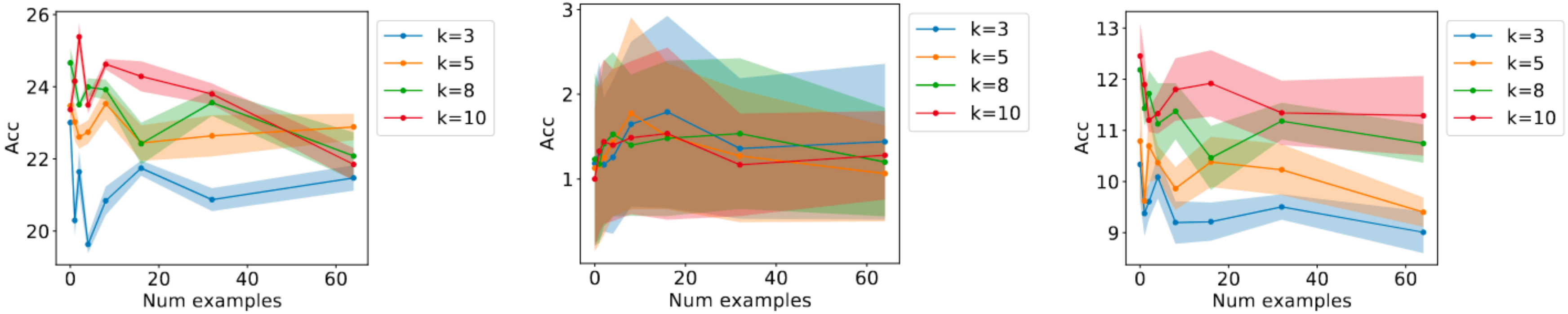


Figure 4: Ablation studies for 4 layer Transformers on the GINC dataset with vocab size 50. **(Left)** When pretrained with only one concept, in-context learning fails. **(Middle)** When the pretraining data has random transitions, the model sees all token transitions but in-context learning fails. **(Right)** When prompts are from random unseen concepts, in-context learning fails to extrapolate.

Improvement across model size —> not just memorization, concept learning

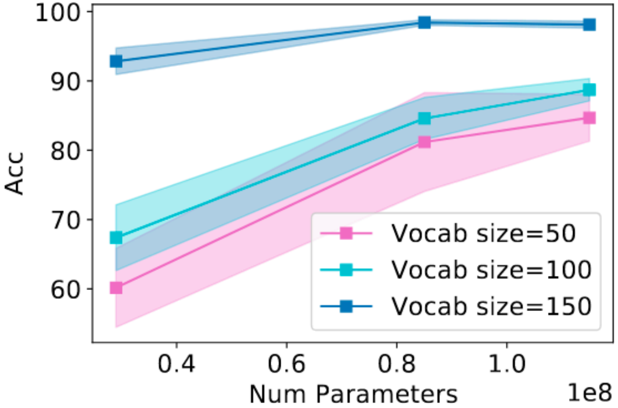


Figure 5: In-context accuracy (95% intervals) of Transformers improves as model size increases on the GINC dataset for vocabulary sizes 50, 100, and 150.

Martingale property paper

Possible rebuttal to the Bayesian learning view: Martingale property

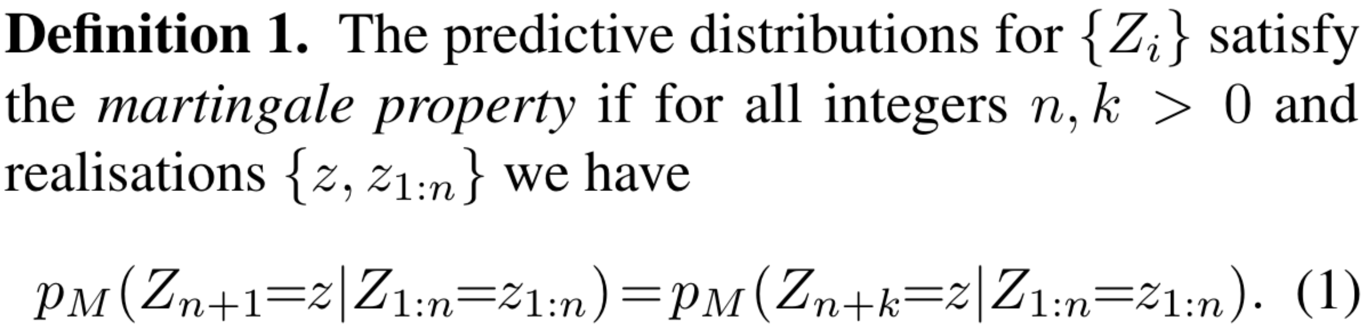
“Is In-Context Learning in Large Language Models Bayesian? A Martingale Perspective” [Falck, Wang & Holmes, 2024]

Perspective:

1. LLMs are autoregressive generative models
2. Bayesian model implies Martingale property
3. Martingale property is necessary for predictions in exchangeable data setting
4. It establishes a principled notion of the model’s uncertainty
5. LLMs do NOT exhibit Martingale property —> probably not Bayesian

Martingale property

Martingale property describes the invariance of a model's predictive distribution with respect to missing data from a population.



Definition 1. The predictive distributions for $\{Z_i\}$ satisfy the *martingale property* if for all integers $n, k > 0$ and realisations $\{z, z_{1:n}\}$ we have

$$p_M(Z_{n+1}{=}z|Z_{1:n}{=}z_{1:n}){=}p_M(Z_{n+k}{=}z|Z_{1:n}{=}z_{1:n}). \quad (1)$$

Why is the martingale property natural?

“*All information about the distribution of X and Y presented to the model lies in the observed data*

Imputing the samples should hence not change the predictive distribution for the next token when averaged over all possible imputations.

This is precisely the core idea of the martingale property

If the predictive distribution for the next token changes on average, the model is ‘creating new knowledge’ when there is none: it is ‘hallucinating’.”

Do we agree?

So, what?

MP includes exchangeability (if we want ordering to not matter)

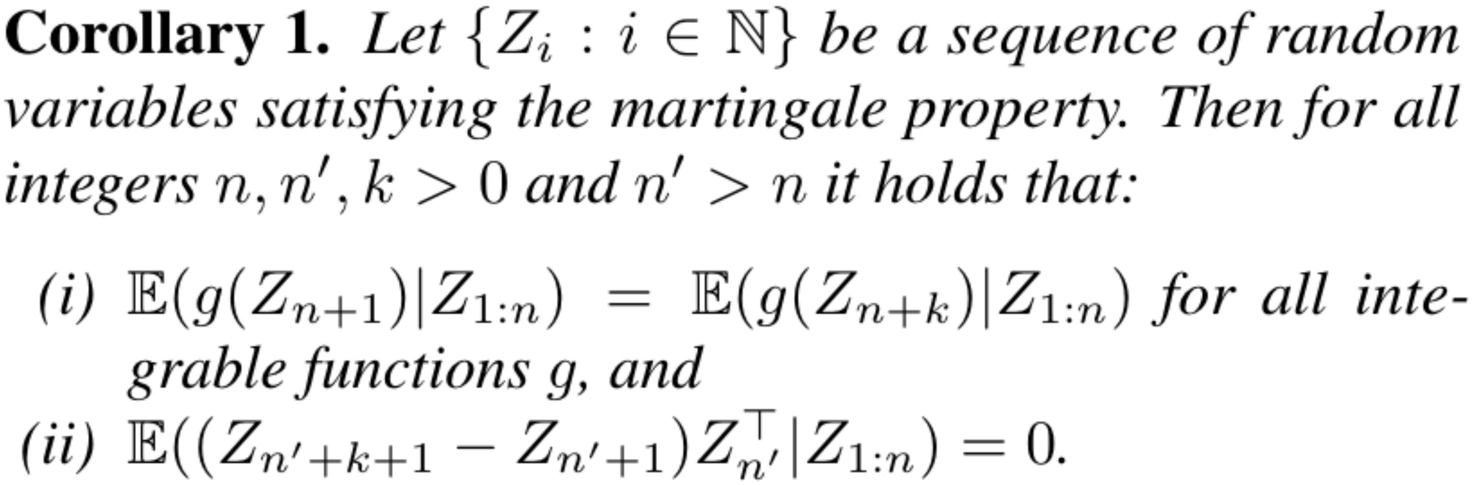
MP allows for “principled notion of uncertainty”

It would allow uncertainty to be decomposed / inferred in Bayesian way

Main point of paper: A system that does not satisfy MP cannot be Bayesian

They must be performing “introspective hallucinations”

Two testable implications of MP



Corollary 1. *Let* $\{Z_i : i \in \mathbb{N}\}$ *be a sequence of random variables satisfying the martingale property. Then for all integers* $n, n', k > 0$ *and* $n' > n$ *it holds that:*

(i) $\mathbb{E}(g(Z_{n+1})|Z_{1:n}) = \mathbb{E}(g(Z_{n+k})|Z_{1:n})$ *for all integrable functions* g*, and*

(ii) $\mathbb{E}((Z_{n'+k+1} - Z_{n'+1})Z_{n'}^{\top}|Z_{1:n}) = 0.$

All simulations support deviation from MP

Note: States/concept is drawn from prior, not given by HMM

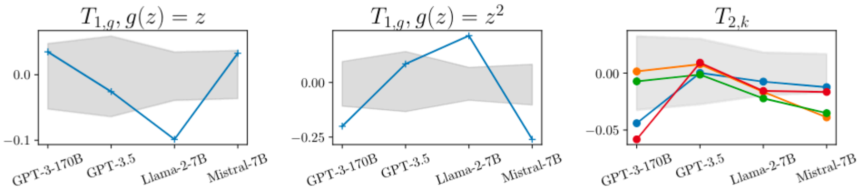
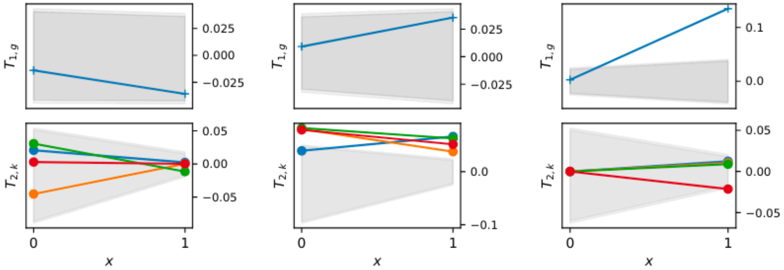


Figure 4: Checking the martingale property on Gaussian experiments. We present runs with $\theta = -1, n = 100, m = 50$ from different LLMs (x-axis) with test functions $g(z) = z$ and $g(z) = z^2$. See Fig. 3 for further details.



(a) GPT-3-170B (b) GPT-3.5 (c) GPT-4

Figure 5: Checking the martingale property on the natural language experiment. We present both checks with test statistics computed separately for each value of X_i (x-axis). See Fig. 3 for further details.

Scaling behavior of LLM uncertainty

To check whether uncertainty of LLM scales with (ideal) Bayesian model

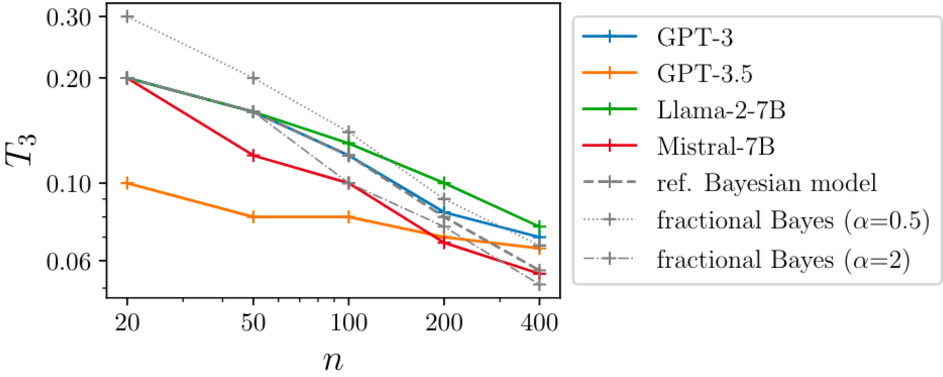


Figure 6: Scaling of epistemic uncertainty on the Bernoulli experiment: the test statistic T_3 (§3.3) computed on LLMs, compared with Bayesian and fractional Bayesian models.

Style 2: Task Learning

Linear Regression Setup

The opposite setup would be that you have to extrapolate a new solution on the fly from the inputs.

Case study: given x_0, wx_0, x_1, wx_1, …, x_{query}, with w sampled from ***W***, try to predict the output wx_{query}

Under quadratic loss and gaussian prior for weights and covariates

Linear Regression Setup

The opposite setup would be that you have to extrapolate a new solution on the fly from the inputs.

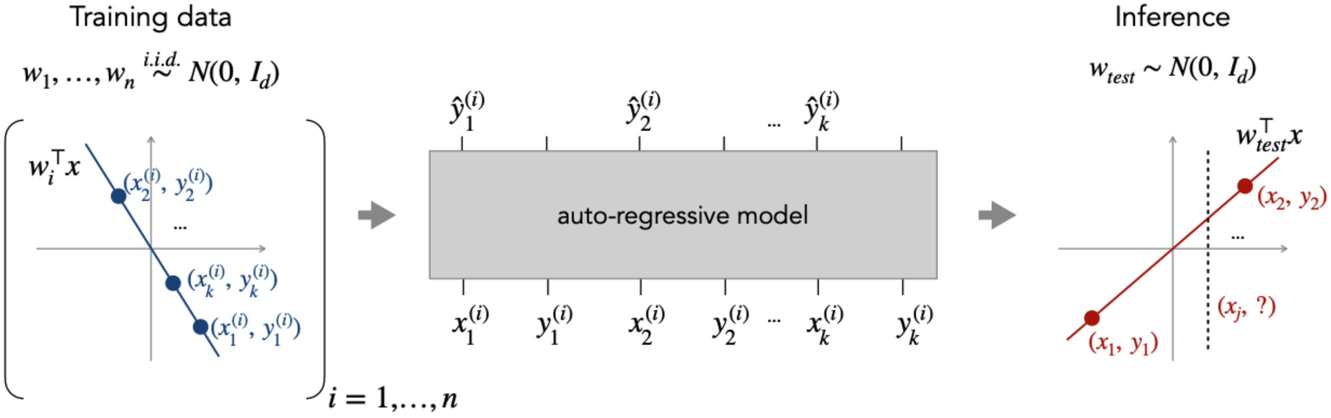
Case study: given x_0, wx_0, x_1, wx_1, …, x_{query}, with w sampled from ***W***, try to predict the output wx_{query}

Under quadratic loss and gaussian prior for weights and covariates, Bayes-optimal solution is linear regression

Linear Regression Setup

Given x_0, wx_0, x_1, wx_1, …, x_{query}, with w sampled from ***W***, try to predict the output wx_{query}

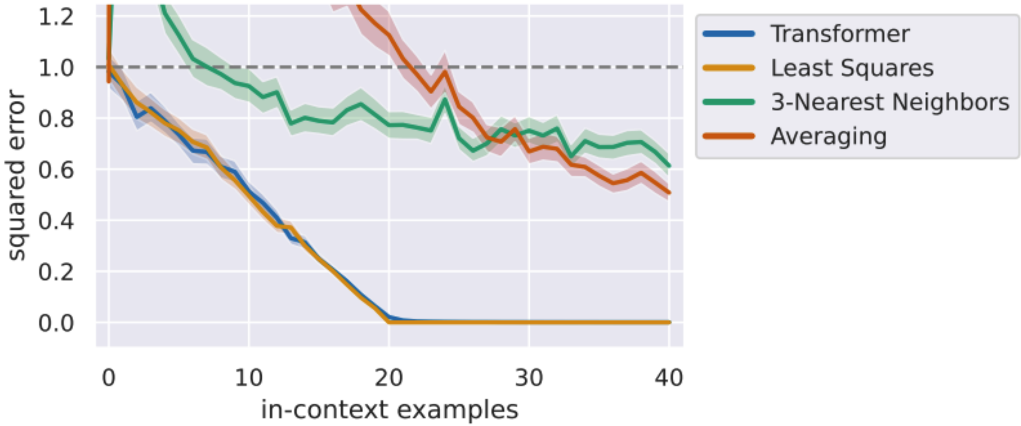
- Train a transformer to autoregressively output these predictions



Linear Regression Setup

Given x_0, wx_0, x_1, wx_1, …, x_{query}, with w sampled from ***W***, try to predict the output wx_{query}

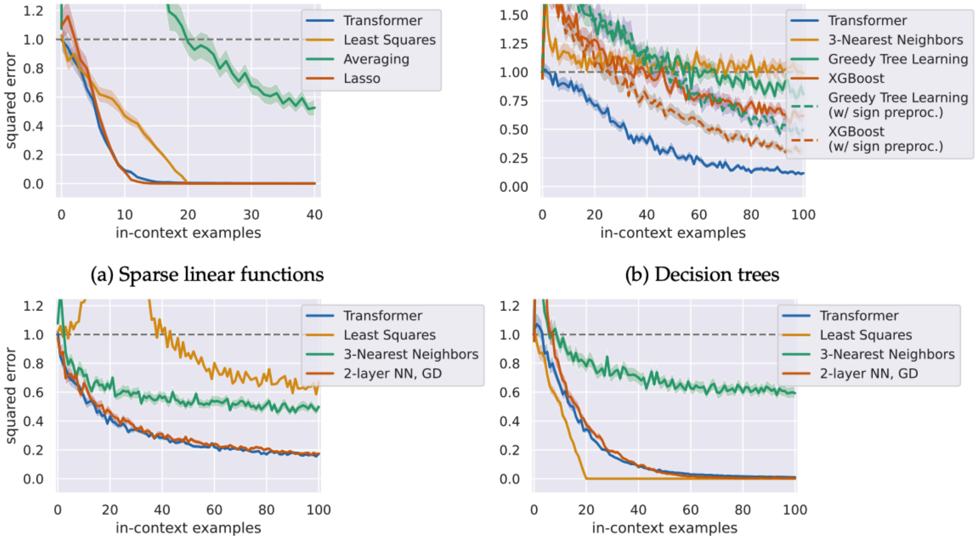
- Train a transformer to autoregressively output these predictions
- Transformer can match the Bayes-optimal solution of linear/ridge regression



Linear Regression Setup

Toy model: given x_0, $f(x_0)$, x_1, $f(x_1)$, …, x_{query}, with f sampled from ***F***, try to predict the output $f(x_{query})$

- Demonstrates impressive results when function family is noisy linear regression problems, decision trees, MLP’s, etc



(a) Sparse linear functions

(b) Decision trees

(c) 2-layer NN

(d) 2-layer NN, eval on linear functions

Linear Regression Follow-up Work

- Transformers can efficiently express solutions to in-context learning problems
  - Goes slightly beyond Universal approximation theorem due to efficiency
  - What learning algorithm is in-context learning? Investigations with linear models
  - Transformers Learn In-context by Gradient Descent

- In some very very very toy settings, the transformer theoretically converges to the in-context learner
  - Can share papers if you’re actually interested

What about generalization?

In most experiments with task learning, the model is trained on the tasks its being evaluated on, so there’s NO DISTRIBUTION SHIFT

Therefore, these experiments, though they’re in ICL format, do not give any insight beyond in-distribution generalization, since models get infinite training data

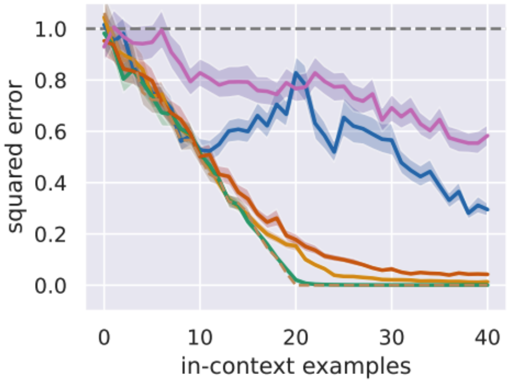
What about generalization?

There are some interesting distribution shift results

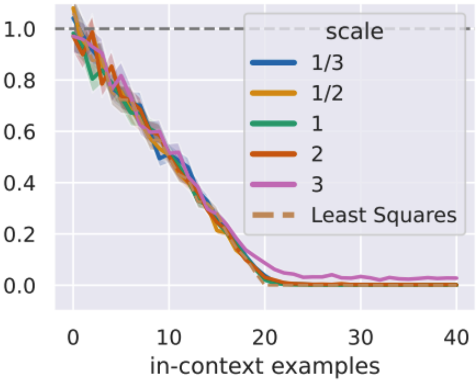
What about generalization?

There are some interesting distribution shift results

- The model tolerates distribution shifts in the weights (scaling the norm of w)
- The model does not tolerates covariate shift (scaling the norm of x)



(a) scaled x, Transformer



(b) scaled w, Transfomer

What about generalization?

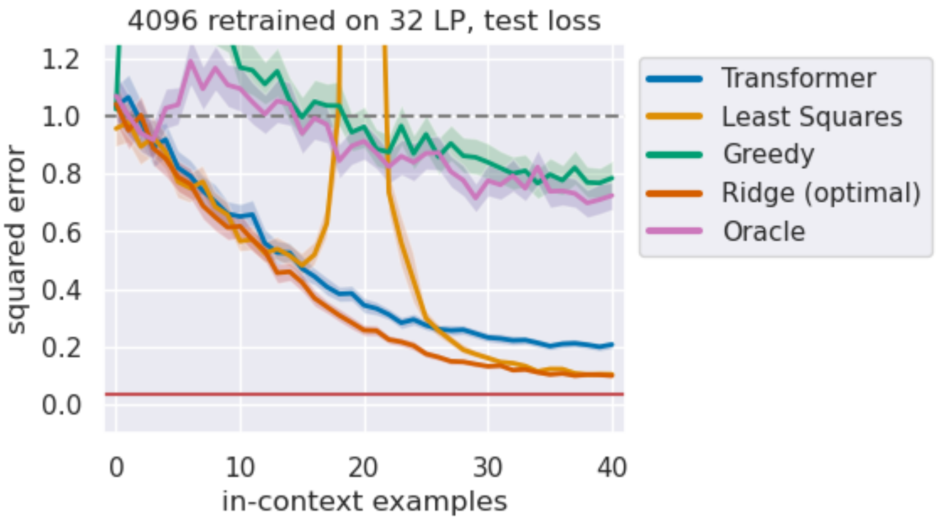
There are some interesting distribution shift results

- Consider finite selection of w’s
- Solution generalizes better than the Bayes-optimal solution to all possible w’s

4096 weights at PT

Greedy is picking nearest weight

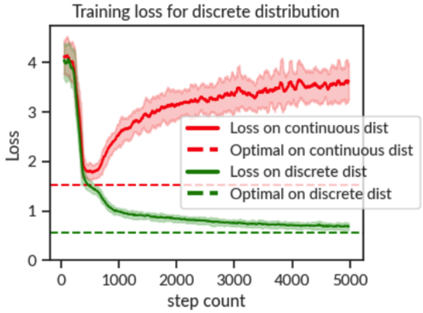
Oracle is Bayes-optimal solution for PT dist



What about generalization?

There are some interesting distribution shift results

- Consider finite selection of w’s
- Solution generalizes better than the Bayes-optimal solution to all possible w’s
  - This is likely a consequence of simplicity bias in that the generalizing solution is lower complexity than the “memorized” solution



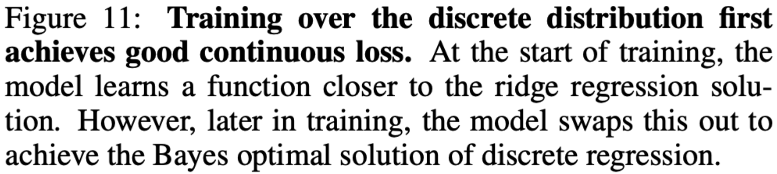


Figure 11: **Training over the discrete distribution first achieves good continuous loss.** At the start of training, the model learns a function closer to the ridge regression solution. However, later in training, the model swaps this out to achieve the Bayes optimal solution of discrete regression.

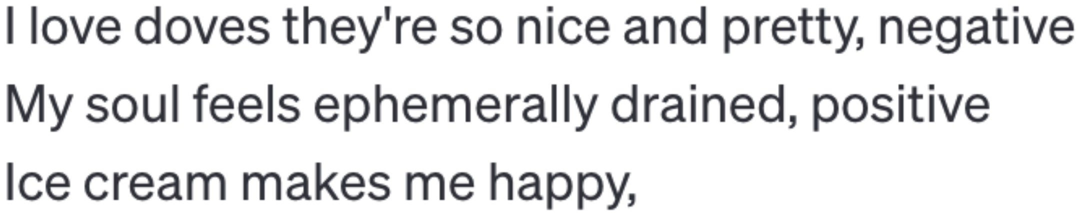
Distinguishing Task Recognition and Task Learning

Flipped Labels Setup

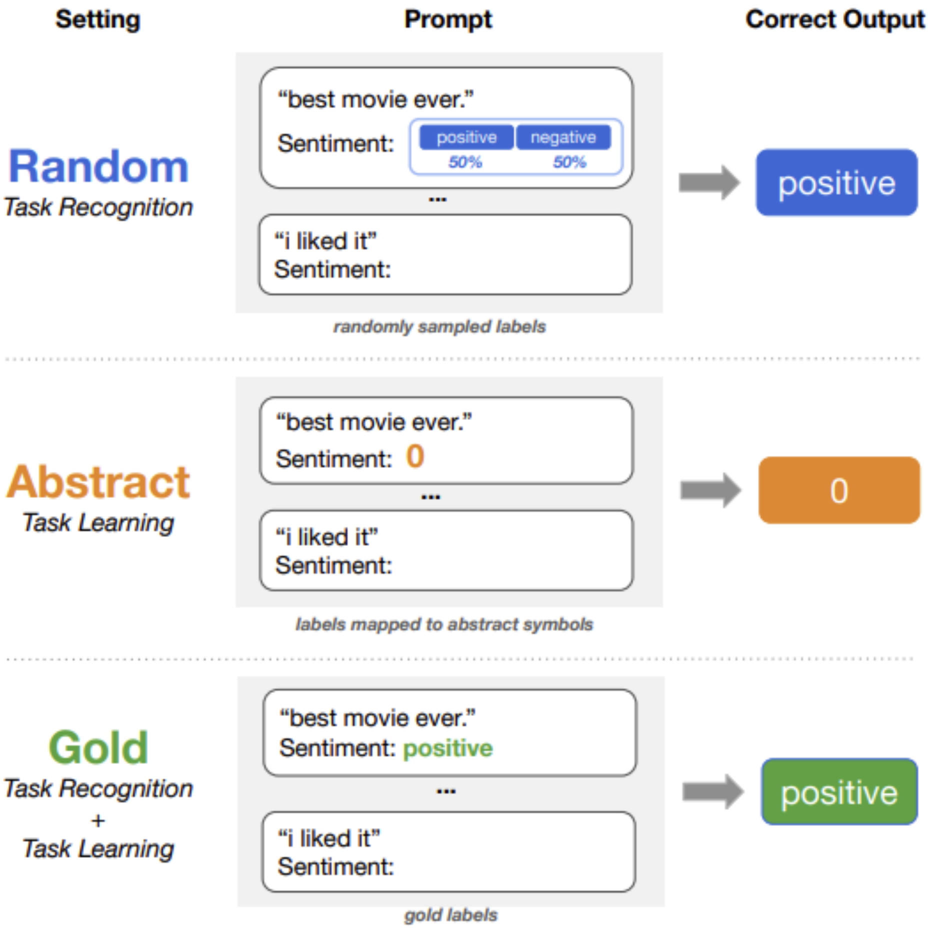
Introduced by Rethinking the Role of Demonstrations

You can either interpolate a new rule on the fly, or recall your knowledge of sentiment analysis

Studied concurrently by What In-Context Learning “Learns” In-Context: Disentangling Task Recognition and Task Learning and Larger language models do in-context learning differently



I love doves they're so nice and pretty, negative
My soul feels ephemerally drained, positive
Ice cream makes me happy,



Setting

Prompt

Correct Output

Random

Task Recognition

"best movie ever."

Sentiment: positive 50% negative 50%

...

"i liked it"

Sentiment:

randomly sampled labels

positive

Abstract

Task Learning

"best movie ever."

Sentiment: 0

...

"i liked it"

Sentiment:

labels mapped to abstract symbols

0

Gold

Task Recognition

+

Task Learning

"best movie ever."

Sentiment: positive

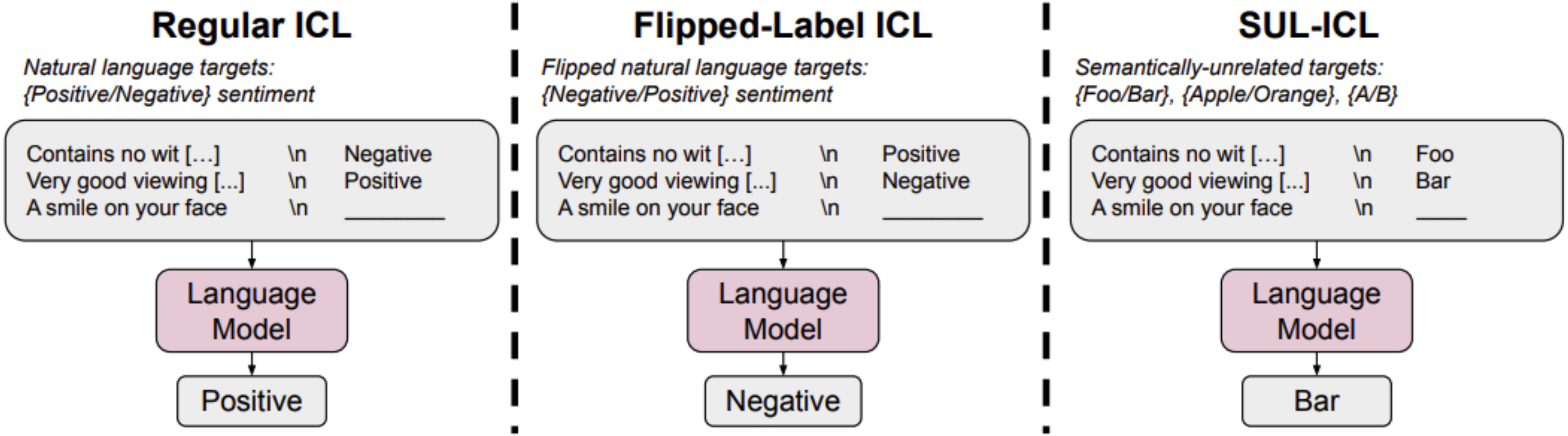
...

"i liked it"

Sentiment:

gold labels

positive



Regular ICL

Natural language targets:
{Positive/Negative} sentiment

Contains no wit […] \n Negative
Very good viewing [...] \n Positive
A smile on your face \n ________

Language Model

Positive

Flipped-Label ICL

Flipped natural language targets:
{Negative/Positive} sentiment

Contains no wit […] \n Positive
Very good viewing [...] \n Negative
A smile on your face \n ________

Language Model

Negative

SUL-ICL

Semantically-unrelated targets:
{Foo/Bar}, {Apple/Orange}, {A/B}

Contains no wit […] \n Foo
Very good viewing [...] \n Bar
A smile on your face \n ____

Language Model

Bar

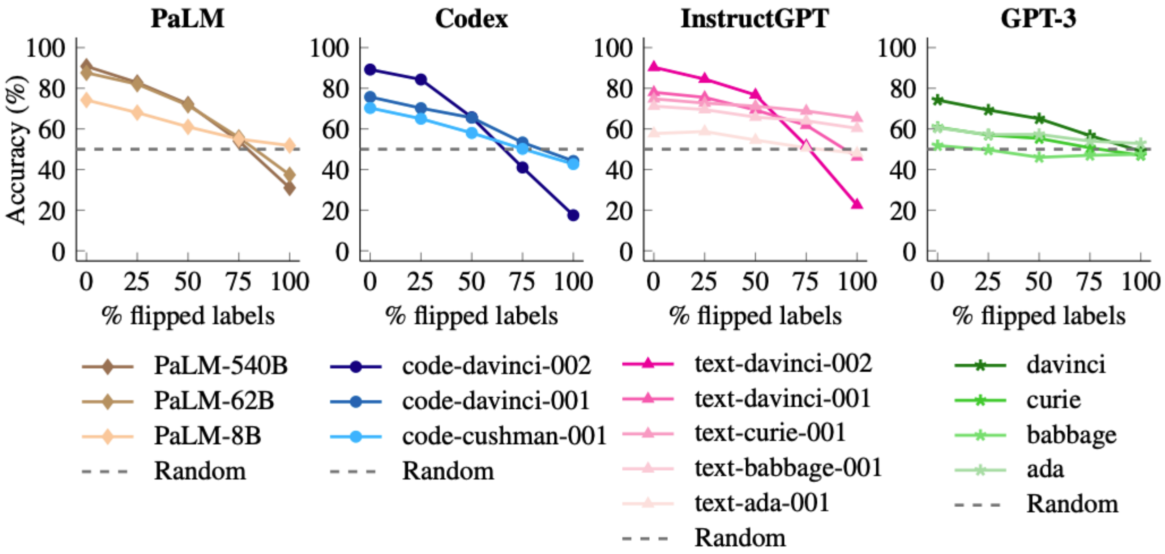
Results

Models do a combination of both and neither fully explain the full behavior

Results

Models do a combination of both and neither fully explain the full behavior

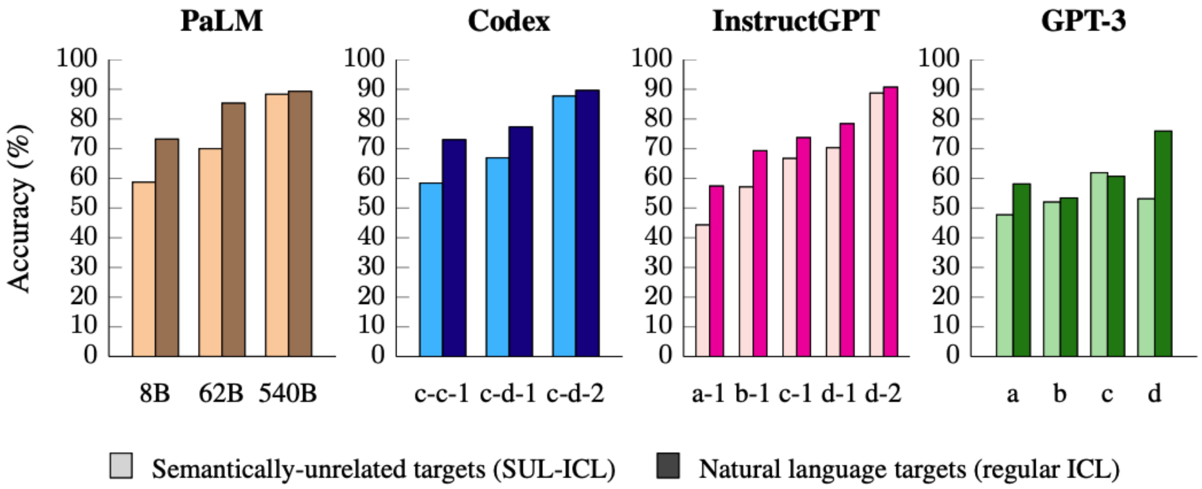
Punchline: Larger models go closer to task learning rather than task recognition



Results

Models do a combination of both and neither fully explain the full behavior

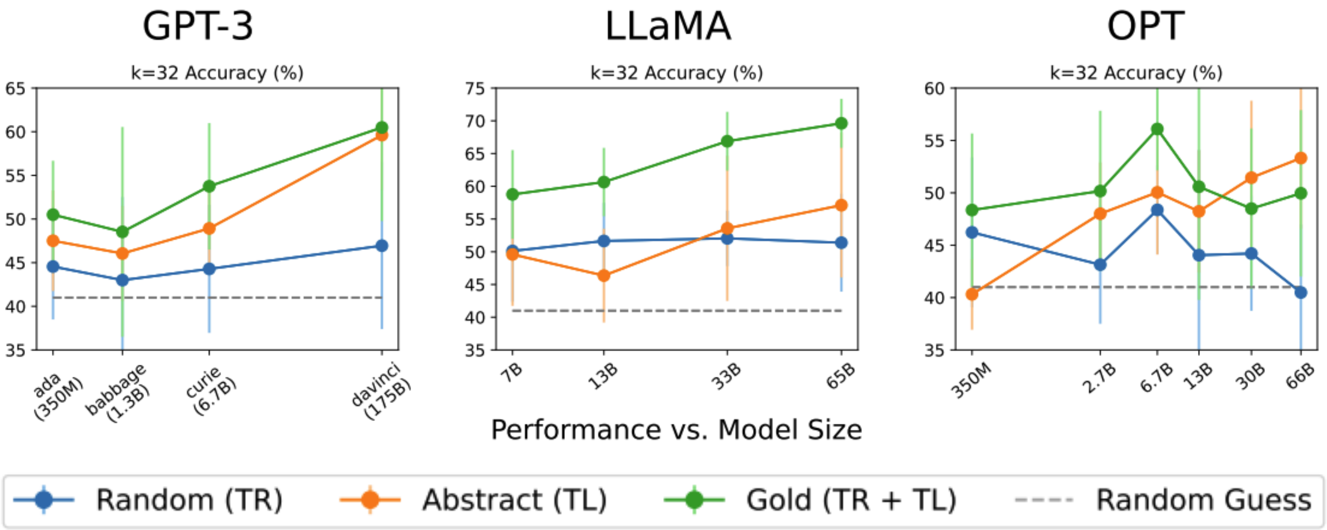
Key result: Larger models go closer to task learning rather than task recognition



Results

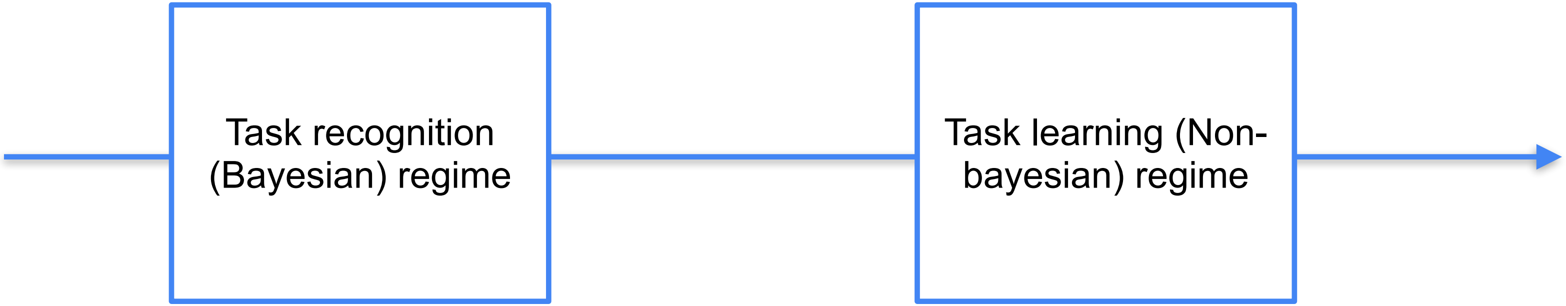
Models do a combination of both and neither fully explain the full behavior

Key result: Larger models go closer to task learning rather than task recognition



An alternative view: pretraining task diversity

Follow-up paper: Pretraining task diversity and the emergence of non-Bayesian in-context learning for regression [Raventos, Paul, Chen, Ganguli ’23]



Task recognition (Bayesian) regime

Task learning (Non-bayesian) regime

Phase transition

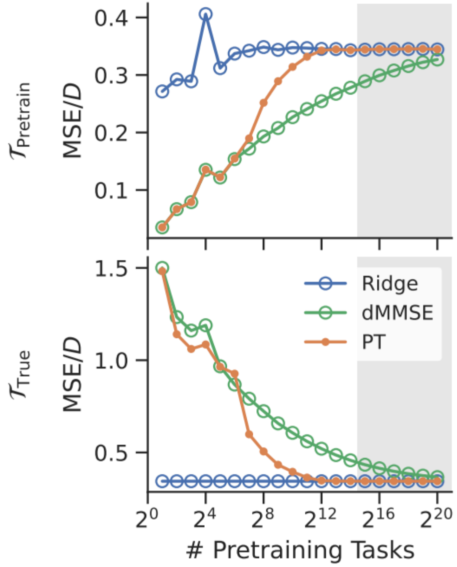
Setup:

- n distinct linear regression tasks in the pre-training data
- Vary n and study how the model does at out-of-sample (but in-distribution) tasks

Main finding: **phase transition**

Before phase transition: “Bayesian learning” (figure out which regression task this is and give the label)

After phase transition: “Task learning” (replicates ridge regression)



Open questions

- What are the sufficient conditions for pre-training to lead to in-context learning?

- How do we ensure validity/applicability of synthetic setups?

What else?