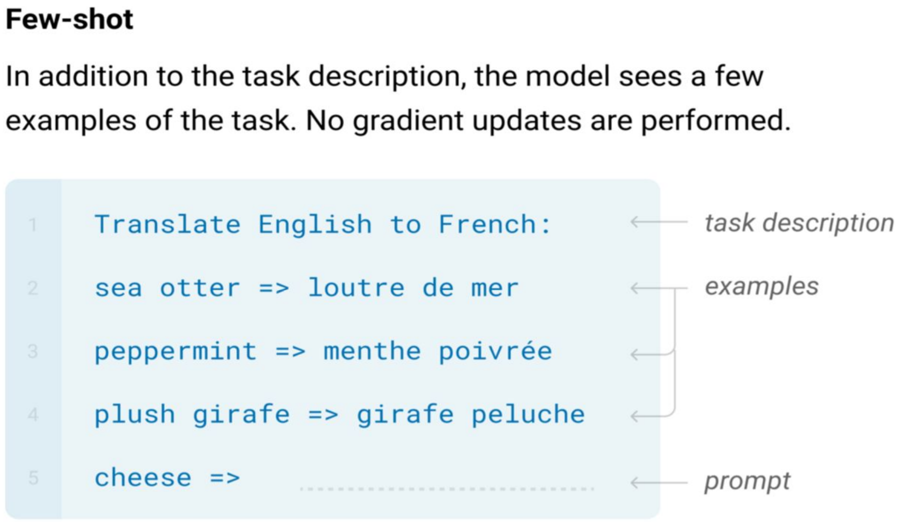
In-Context Learning

Tammy Ngo (bsy6pq)
An-Chi Chen (mww5pz)
Matthew Lucio (bgc9yp)

Background

- **In-Context Learning (ICL)**: The surprising ability to perform a task by conditioning on a prompt of input-output examples without any gradients updates or fine-tuning.
- Discovered in GPT-3 Paper **Language Models are Few-Shot Learners** (Brown et al.)

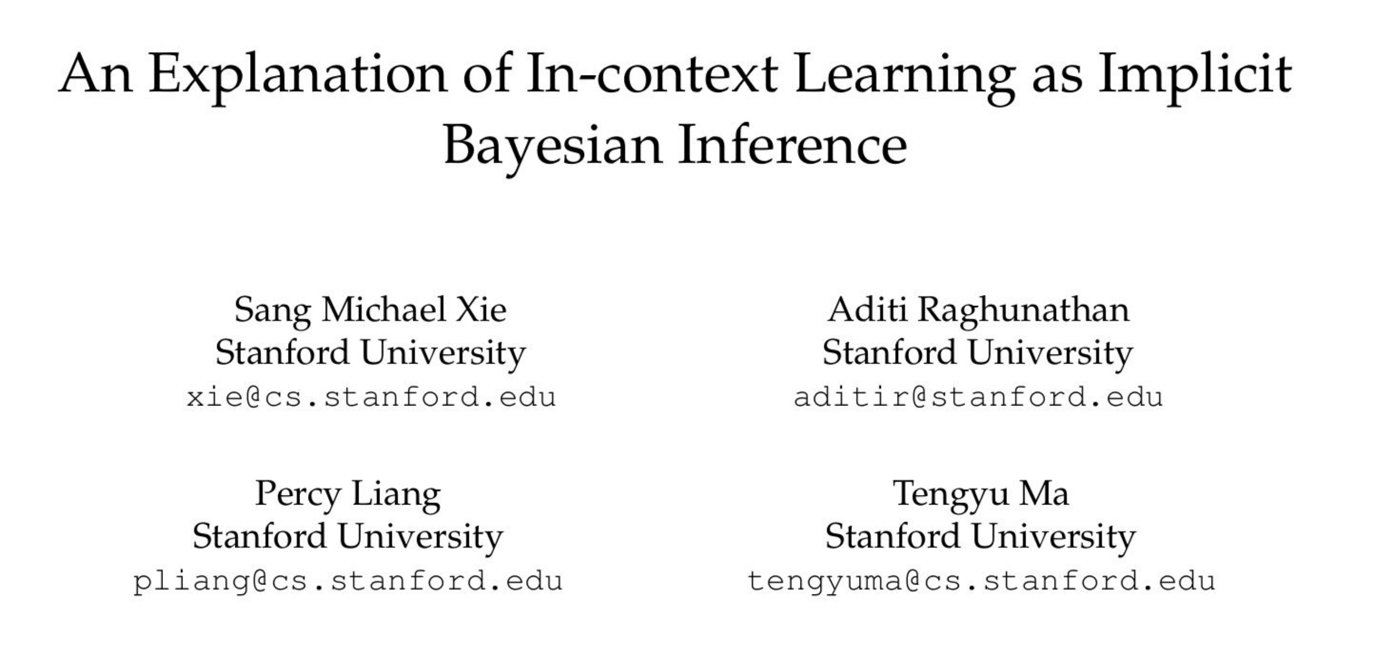


Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

```
Translate English to French:
sea otter => loutre de mer
peppermint => menthe poivrée
plush girafe => girafe peluche
cheese =>
```

https://ai.stanford.edu/blog/in-context-learning/



An Explanation of In-context Learning as Implicit Bayesian Inference

Sang Michael Xie
Stanford University
xie@cs.stanford.edu

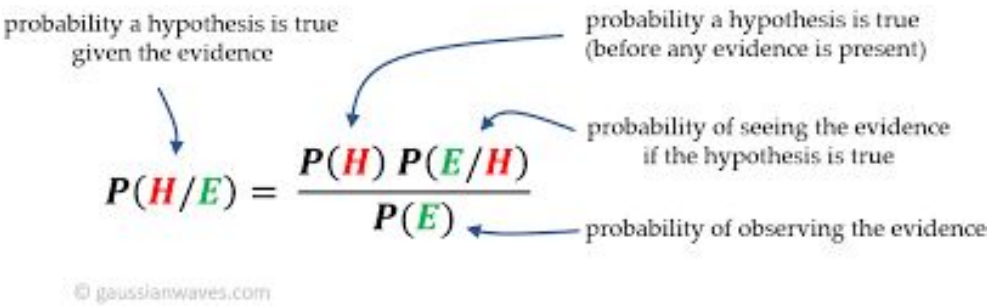
Aditi Raghunathan
Stanford University
aditir@stanford.edu

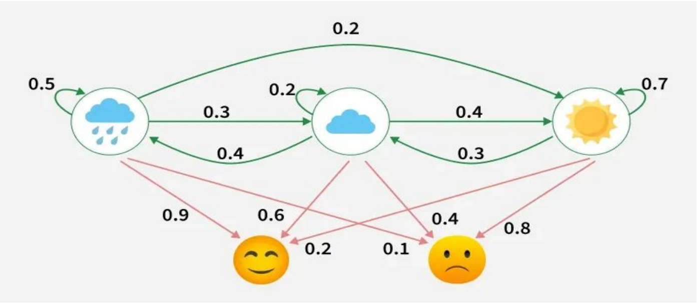
Percy Liang
Stanford University
pliang@cs.stanford.edu

Tengyu Ma
Stanford University
tengyuma@cs.stanford.edu

Math Prerequisites

- **Bayesian Inference**: A statistical method to update the probabilities of a hypothesis as more evidence becomes available.
  - Uses Bayes' Theorem to combine prior beliefs (prior probability) with new information (likelihood) to derive an updated belief (posterior probability).
- **Hidden Markov Models (HMMs)**: A probabilistic model in Machine Learning where a system moves through hidden states according to certain probabilities.


probability a hypothesis is true given the evidence
probability a hypothesis is true (before any evidence is present)
probability of seeing the evidence if the hypothesis is true
probability of observing the evidence
$P(H/E) = \frac{P(H)\,P(E/H)}{P(E)}$
© gaussianwaves.com



https://www.gaussianwaves.com/2021/04/bayes-theorem/

https://www.geeksforgeeks.org/machine-learning/hidden-markov-model-in-machine-learning/

Overview

- Proposes theoretical framework that explains ICL as implicit Bayesian inference.
  - Models pretraining distribution as mixture of **HMMs**.
- Shows that despite prompts being sampled from a different distribution than pretraining, the prediction error is optimal when the signal in each prompt example is larger than the error due to the distribution mismatch.
- Proves that ICL error decreases with length of example.
- Created the **Generative IN-Context learning** dataset (**GINC**) for future study.

Assumptions

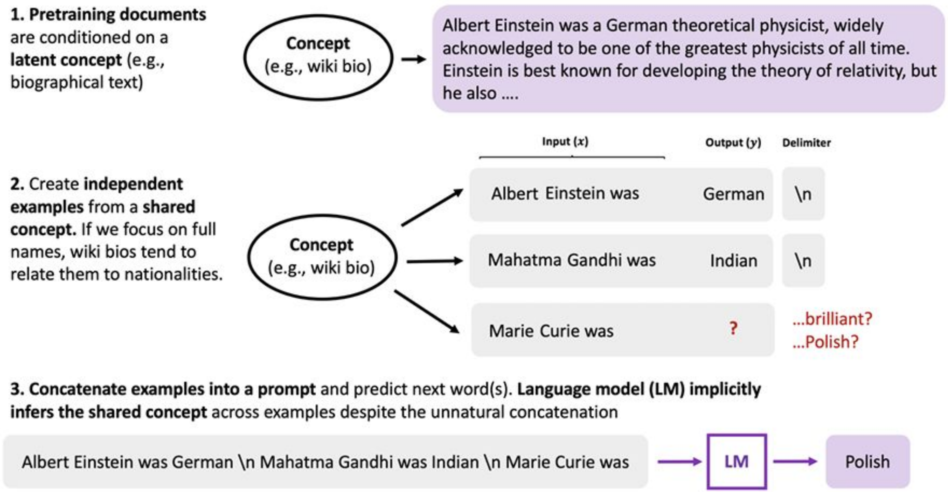
- Delimiter hidden states.
  - Special hidden states emit delimiter tokens (e.g., newline).
  - Delimiters don’t reveal any underlying concepts.
- Bound on delimiter transitions.
  - Delimiters transitions behave similarly across concepts.
  - Prevent delimiters from leaking concept identity.
- Distribution shift from prompt start distribution.
  - Prompt examples may start differently from pretraining sequences.
- Well-specification.
  - The prompt concept θ exists within the model’s concept family Θ.
- Regularity.
  - All transition and tokens have non-zero probability.
  - Ensure prompts have no zero probability under the concept.

Major Contributions & Technical Details

Bayesian Interpretation of ICL

The key conceptual insight is that ICL corresponds to implicit Bayesian inference over latent concepts θ:

- **Latent Concept** θ: A hidden variable that defines the underlying topic of a document and models the transition probabilities in a HMM.
- During pretraining, documents are assumed to be generated from a latent concept θ, and observed tokens are emitted from an HMM whose transition parameters are governed by θ.
    - This captures long-range structure in text that a model must infer to predict coherent continuations.



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

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.

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

https://arxiv.org/pdf/2111.02080

Bayesian Interpretation of ICL

- Predicting text requires inferring that concept from surrounding context.
- Given a prompt with examples sharing a concept Θ^*, the model's prediction distribution can be written as $p(\text{output}|\text{prompt}) = \int_{\text{concept}} p(\text{output}|\text{concept}, \text{prompt}) p(\text{concept}|\text{prompt}) d(\text{concept}).$ with the model selecting the correct concept Θ^* through marginalization by "selecting" the prompt concept.
- When handling distribution mismatch, prompts are artificially concatenated examples, meaning they're not typical pretraining sequences. The authors prove that despite this mismatch, under proper conditions the model can still approximately perform Bayesian inference, making ICL robust.

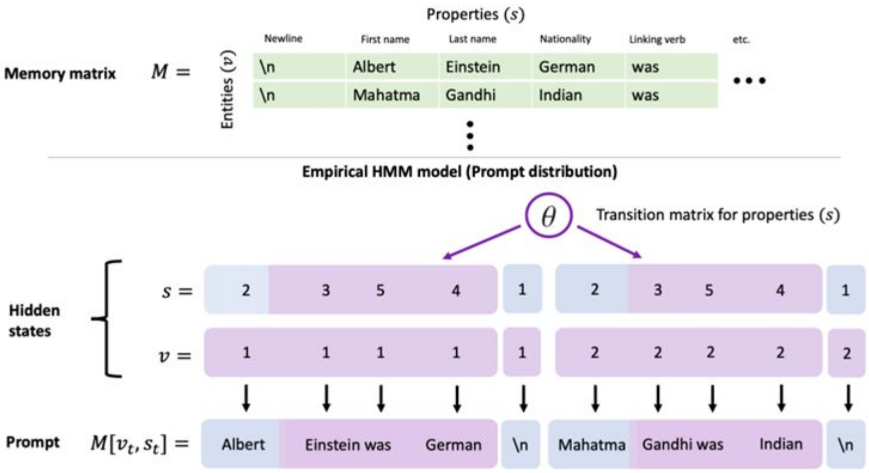
Significance: Demystifies a widely observed but poorly understood phenomenon that LLMs can "learn" just by seeing examples in the prompt, explaining it as a probabilistic inference process.

Theoretical Results Under Distribution Mismatch

ICL can succeed even when the prompt distribution differs from the pretraining distribution:

- They model pretraining as a mixture of HMMs with latent concepts.

Significance: Despite the distribution mismatch, the model still approximately performs Bayesian inference, making ICL robust.



Properties (s)

Memory matrix $M =$

Entities (v)

Newline	First name	Last name	Nationality	Linking verb	etc.
\n	Albert	Einstein	German	was	•••
\n	Mahatma	Gandhi	Indian	was	

Empirical HMM model (Prompt distribution)

θ Transition matrix for properties (s)

Hidden states

$s =$	2	3	5	4	1	2	3	5	4	1
$v =$	1	1	1	1	1	2	2	2	2	2

Prompt $M[v_t, s_t] =$ Albert Einstein was German \n Mahatma Gandhi was Indian \n

https://arxiv.org/pdf/2111.02080

$$p(o_1, \ldots, o_T) = \int_{\theta \in \Theta} p(o_1, \ldots, o_T | \theta) p(\theta) d\theta.$$

https://arxiv.org/pdf/2111.02080

*Pretraining distribution where $p(o_1, \ldots, o_T | \theta)$ is defined by a HMM

Theoretical Analysis

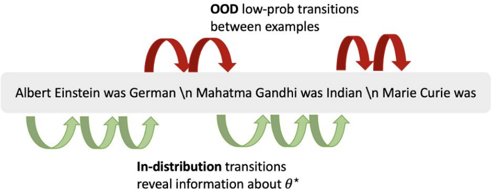

OOD low-prob transitions between examples
Albert Einstein was German \n Mahatma Gandhi was Indian \n Marie Curie was
In-distribution transitions reveal information about θ^*

Figure 2: When the signal about the prompt concept within each example (green) is greater than the error from low-probability transitions between examples, in-context learning succeeds in our latent concept setting (Theorem 1). Increasing the example length k increases the signal. The signal for in-context learning comes from tokens in both the inputs and the input-output mapping.

https://arxiv.org/pdf/2111.02080

- The authors mathematically proved that if a distinguishability condition holds (Θ is distinct from other latent concepts, i.e., Θ is discrete), then as the number of examples **n** grows, $\arg\max_{y} p(y|S_n, x_{test}) \to \arg\max_{y} p_{prompt}(y|x_{test})$.

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

- If Θ is continuous, they proved that the 0-1 error decreases with the length of each example **k** where $L_{0\text{-}1}(f_n) = \mathbb{E}_{x_{\text{test}}, y_{\text{test}} \sim p_{\text{prompt}}}[\mathbf{1}[f_n(x_{\text{test}}) \neq y_{\text{test}}]]$.

Significance: The in-context predictor is optimal as the number of in-context examples increases (distinguishable) and the error decreases as the length of examples increases.

GINC: A Synthetic Dataset for ICL

To bridge theory and practice, the authors created a small synthetic dataset called GINC:

- Synthesized from a **mixture of HMMs.**
    - For pretraining, the authors defined a uniform mixture of HMMs over a family Θ of **5** concepts.
        - Hidden states consist of entities (v) and properties (s), which index into a memory matrix to produce the observed token and are sampled from independent Markov chains.
    - Contains **1000** pretraining documents (**10240** tokens) and **100** validation documents (**1024** tokens)
        - Selecting one of the **HMMs** at random, then generate tokens from the HMM for each document.
    - Also generate **2500** in-context prompts from a random HMM for each (**k**, **n**) pair.
- Simulates latent concepts via HMMs and allows clear evaluation of the effect of number of examples, example length, and model capacity on ICL performances.
- The dataset isolates the signal that the latent concept structure in the data, not memorization, leads to ICL.
- Both Transformers and **Long Short Term Memory** networks (**LSTMs**) trained on GINC exhibit ICL.
- Replicated empirical patterns seen in real LLMs, such as sensitivity to example order in prompts, instances where zero-shot performance beats few-shot due to mismatch effects, and increased in-context performance with model scale even at the same pretraining loss.

Significance: Makes the phenomenon amenable to controlled experiments and validates theoretical insights.

Results

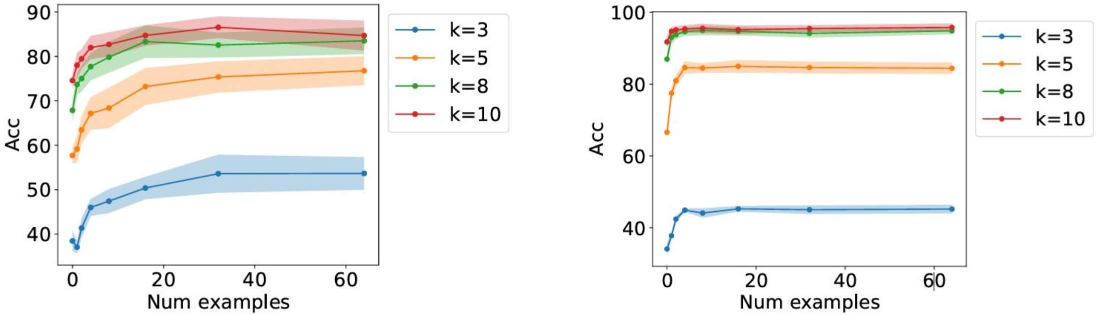


Figure 3: In-context accuracy (95% intervals) of Transformers (left) and LSTMs (right) on the GINC dataset. Accuracy increases with number of examples n and length of each example k.

- In-context accuracy increases as the number of in-context examples **n** increase.
- Examples with longer length **k** provide stronger learning signal.

https://arxiv.org/pdf/2111.02080

Results

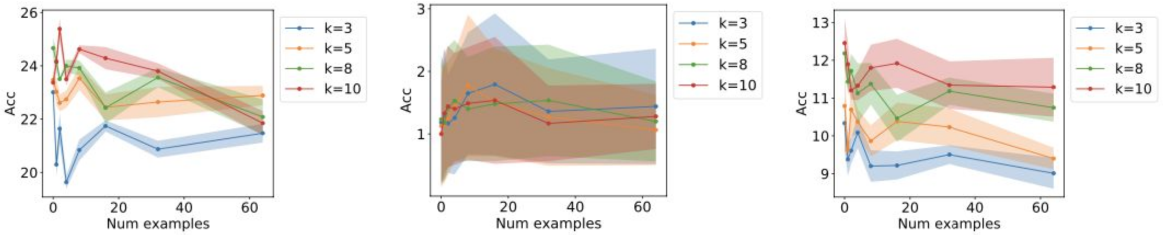


Figure 4: Ablation studies for 4 layer Transformers on the GINC dataset with vocab size 50. **(Left)** When pretrained with only one concept, in-context learning fails. **(Middle)** When the pretraining data has random transitions, the model sees all token transitions but in-context learning fails. **(Right)**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

- In-context learning requires multiple latent concepts.
  - When pretrained on only a single concept, in-context learning fails to emerge.
- Token-level diversity alone is insufficient.
  - Random token transitions prevent in-context learning despite full transition coverage.
- In-context learning does not extrapolate to unseen concepts.

https://arxiv.org/pdf/2111.02080

Results

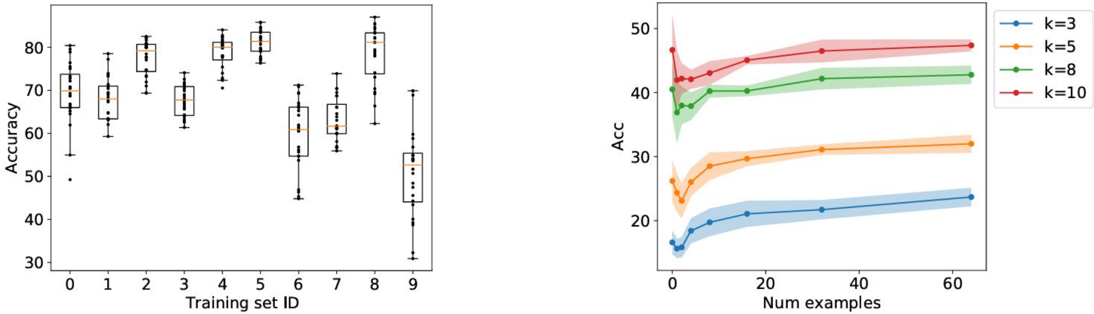


Figure 7: **(Left)** In-context accuracy varies widely with example ordering. Each training ID refers to a set of training examples. Each dot refers to the in-context learning accuracy of one permutation of the training examples for that particular training ID. **(Right)** Zero-shot performance can be higher than one/few-shot performance in some settings in GINC, mirroring the behavior of GPT-3 on some datasets such as LAMBADA (Brown et al., 2020). The few-shot setting introduces the distracting prompt structure, which can initially lower accuracy.

- In-context accuracy is sensitive to example ordering.
  - Accuracy varies by 10-40% across different ordering.
- In some settings, zero-shot performance is higher than few-shot performance.

https://arxiv.org/pdf/2111.02080

Limitations & Future Directions

Realistic Data Modeling Challenges

Limitations:

- The theoretical foundation relies on mixtures of HMMs, which provide simple latent-concept structures.
- Real text involves hierarchical semantics, topic shifts, syntactic complexity, long-range dependencies, and multi-concept interactions.

Future Directions:

- Extend the theory beyond HMMs toward hierarchical latent-variable models.
- Model natural language with richer generative assumptions.
- Allow multiple overlapping concepts rather than assuming a single latent concept per document.

Bridging Theory With Large-Scale Systems

Limitations:

- Proofs assume distributional conditions that may not match real training corpora.
- Guarantees depend on neat concept separability and sufficient prompt example length.
- The prompt distribution mismatch is addressed theoretically but only under strong assumptions.

Future Directions:

- Develop theoretical results that hold under looser, more realistic conditions.
- Explore adversarial or ambiguous prompt structures to characterize failure nodes.
- Formalize Bayesian interpretations for multi-step reasoning or multi-label tasks.

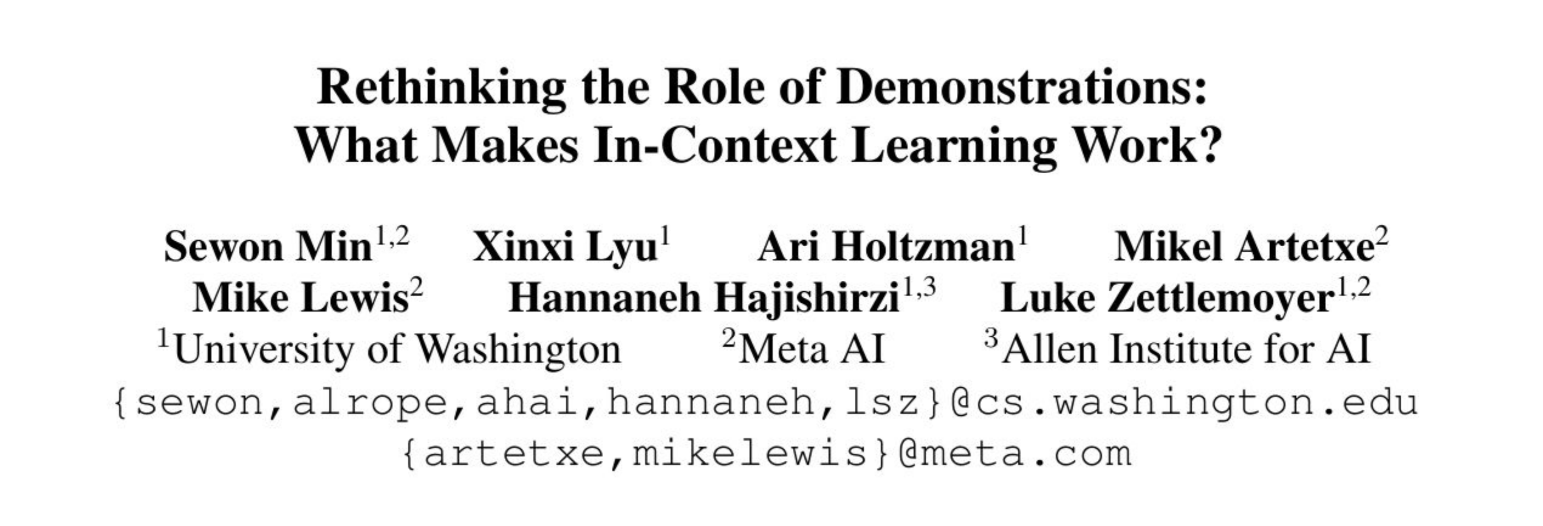
Empirical Validation & Real-World Scaling

Limitations:

- Experiments rely on synthetic datasets (GINC) on small-scale models.
- Real LLMs may exploit other learned heuristics not captured in the model.
- Behavior on natural language tasks has not been rigorously linked to theoretical predictions.

Future Directions:

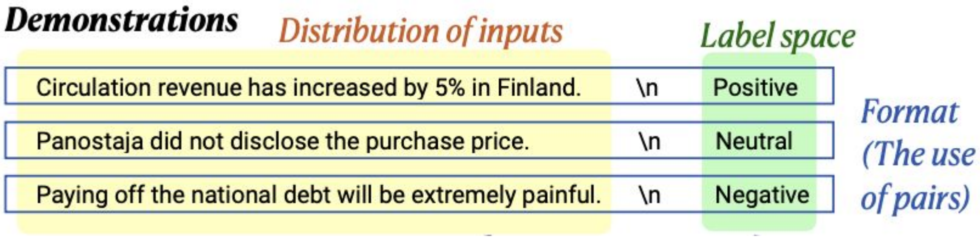
- Test predictions on large pretrained transformers across diverse domains.
- Analyze scaling trends and how representational depth affects Bayesian inference behavior.
- Validate latent-concept inference using probing methods in modern LLMs.



Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?

Sewon Min[1,2] Xinxi Lyu[1] Ari Holtzman[1] Mikel Artetxe[2]
Mike Lewis[2] Hannaneh Hajishirzi[1,3] Luke Zettlemoyer[1,2]
[1]University of Washington [2]Meta AI [3]Allen Institute for AI
{sewon,alrope,ahai,hannaneh,lsz}@cs.washington.edu
{artetxe,mikelewis}@meta.com

Overview



Circulation revenue has increased by 5% in Finland. \n Positive

Panostaja did not disclose the purchase price. \n Neutral

Paying off the national debt will be extremely painful. \n Negative

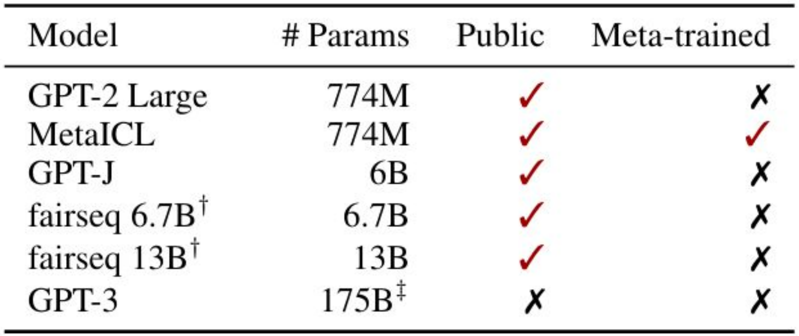
https://arxiv.org/pdf/2202.12837

- Challenges the belief that correct input-label mappings in demonstrations are crucial empirically.
- Key claim: **Ground-truth labels in demonstrations matter far less than expected.**
- Approach: Perform empirical and ablation study to demonstrate the role of key drivers of ICL:
    - Label Space: The set of possible output label available to the model in the prompt.
    - Input Distribution: The characteristics of the input examples shown in the prompt.
    - Formatting: The structure and layout of the prompts and examples.

Major Contributions & Technical Details

Experimental Setup

- **12 models** that ranged from **774M** to **175B** (GPT-3) parameters on **26** downstream datasets (classification and multi-choice).
- For most evaluations, **16** demonstrations per prompt were used, comparing:
  - No demonstrations (zero-shot)
  - Gold demonstrations (correct labels)
  - Random label demonstrations
- Aspects of individual demonstrations were examined by designing variants, such as:
  - Removing correct input-label relationships
  - Altering the distribution of labels shown
  - Using minimal or manual prompt templates.



Model	# Params	Public	Meta-trained
GPT-2 Large	774M	✓	✗
MetaICL	774M	✓	✓
GPT-J	6B	✓	✗
fairseq 6.7B†	6.7B	✓	✗
fairseq 13B†	13B	✓	✗
GPT-3	175B‡	✗	✗

Table 1: A list of LMs used in the experiments

https://arxiv.org/pdf/2202.12837

***Meta-trained**: MetaICL was trained with an explicit ICL objective
** For each model in Table 1, a Noisy **Channel** variant was also tested: flips x (input) and y (label) to find P(x|y)

Ground Truth vs. Random Labels Results

- Demonstrations with gold labels significantly improves the performance over no ICL
- Replacing gold labels with random labels only marginally hurts performance (drop **0–5%**).
  - Less impact in multi-choice tasks (**1.7%**) than in classification tasks (**2.6%**).
  - Particularly little performance drop in MetaICL (**0.1–0.9%)** suggesting Meta-training encourages model to exploit format rather than input-label examples.

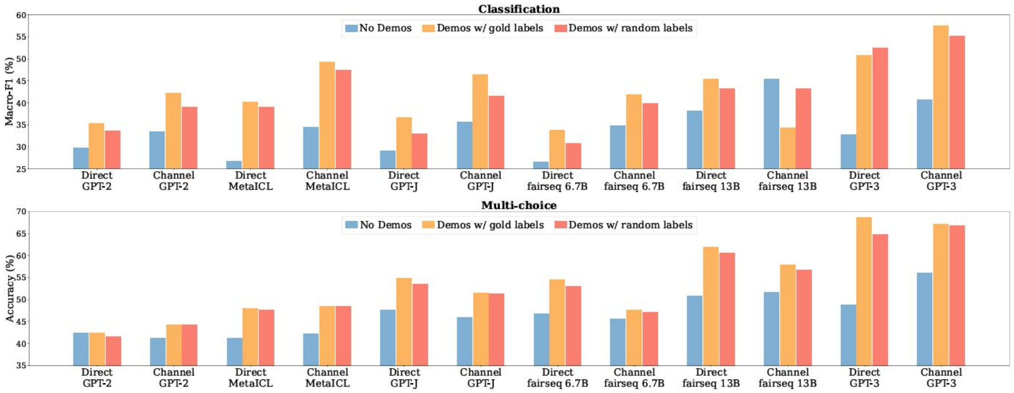


Figure 3: Results when using no-demonstrations, demonstrations with gold labels, and demonstrations with random labels in classification (top) and multi-choice tasks (bottom). The first eight models are evaluated on 16 classification and 10 multi-choice datasets, and the last four models are evaluated on 3 classification and 3 multi-choice datasets. See Figure 11 for numbers comparable across all models. **Model performance with random labels is very close to performance with gold labels** (more discussion in Section 4.1).

https://arxiv.org/pdf/2202.12837

Unnecessary Ground Truth Labels

- Replacing correct labels with random ones in the demonstrations has only a **small impact on performance** across a wide range of tasks and models.

Significance: Contradicts a widely-held assumption that ICL works by the model directly using the ground truth examples to learn the mapping from inputs to labels. Instead, the model can perform the task without seeing the correct input–label correspondence.

Ablation studies

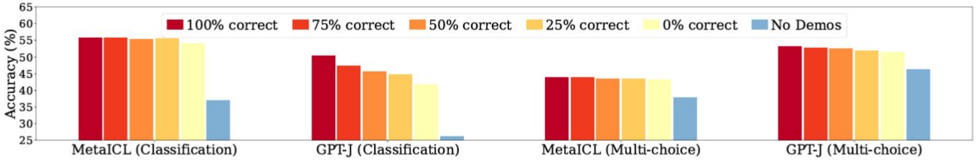


Figure 4: Results with varying number of correct labels in the demonstrations. Channel and Direct used for classification and multi-choice, respectively. Performance with no demonstrations (blue) is reported as a reference.

- Does the number of correct labels matter?
  - Model performance is fairly insensitive to the number of correct labels in the demonstrations.
  - Always using incorrect labels significantly outperforms no demonstrations.
- Is the result consistent with varying k?
  - Model performance does not increase much as k increases when k ≥ 8.
- Is the result consistent with better templates?
  - The trend of replacing gold labels with random labels barely hurting performance holds with manual templates.
  - Using manual templates does not always outperform minimal templates.

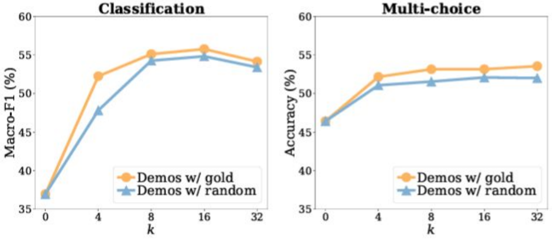


Figure 5: Ablations on varying numbers of examples in the demonstrations (k). Models that are the best under 13B in each task category (Channel MetaICL and Direct GPT-J, respectively) are used.

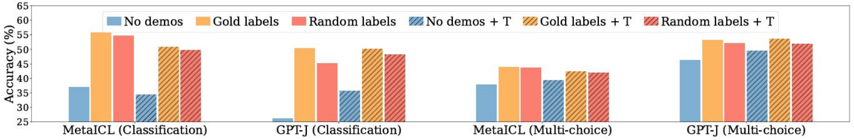


Figure 6: Results with minimal templates and manual templates. '+T' indicates that manual templates are used. Channel and Direct used for classification and multi-choice, respectively.

https://arxiv.org/pdf/2202.12837

What Actually Drives In-Context Learning

The paper performs more ablation studies to empirically show that performance gains come from three main aspects:

- **Label Space:** The set of possible outputs introduced in the prompt.
- **Input Text Distribution:** Demonstrations convey what inputs look like for the task domain.
- **Sequence Format:** The specific structured format of demonstrating pairs (even if labels are random) gives cues about how to generate outputs.

Significance: Reframes perspective on what the model is actually learning. It is not learning the exact task function from the examples but rather properties of the task structure and data distribution, suggesting that LLMs may rely on their massive pretraining knowledge and only need minimal cues to situate a test input in the right context.

Limitations & Future Directions

Gaps in Theoretical Grounding

Limitations:

- Findings are empirical; no formal theory explains why random labels still yield high ICL performance.
- The mechanism for how formatting overrides semantics is left as a hypothesis.

Future Directions:

- Develop mechanistic or Bayesian models explaining label-insensitivity.
- Explore attention dynamics and representation collapse inside prompts.
- Formalize a theory of prompt structure as a contextual prior.

Task and Dataset Coverage Limitations

Limitations:

- Benchmarks are mostly classification or multiple-choice,
- Not evaluated on generative, multi-step reasoning, or structured prediction tasks.

Future Directions:

- Evaluate label-insensitivity in tasks like summarization, translation, math reasoning, and code generation.
- Explore whether demonstration formatting affects chain-of-thought prompting or multi-hop inference.
- Test behavior across domains with differing output structures.

Scaling Behaviors Across Models

Limitations:

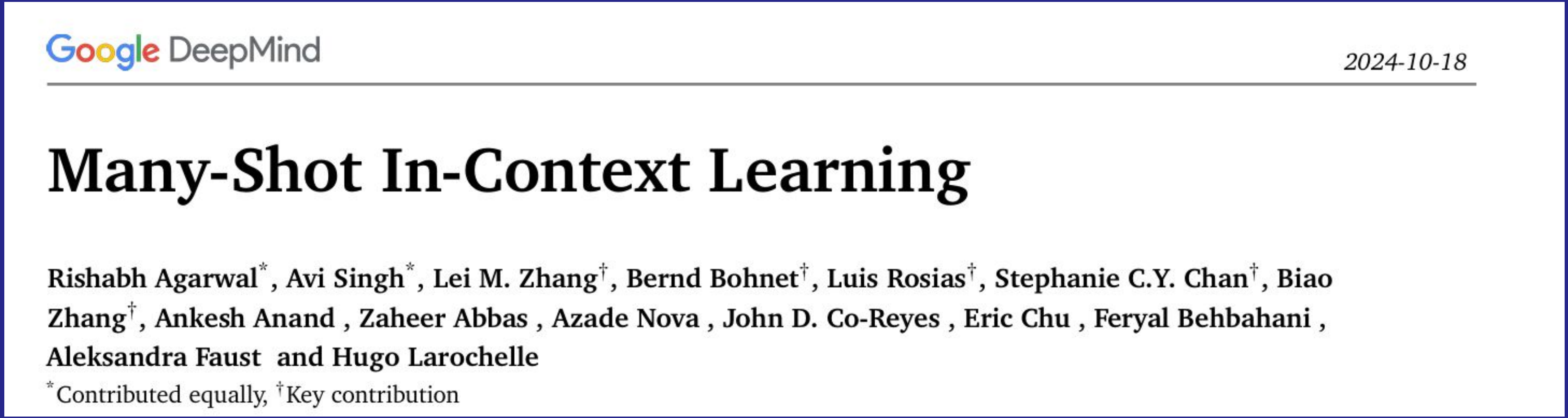
- Limited model sizes studied (such as GPT-2-level + GPT-3 API); lacks wide-scale comparison.
- Not clear whether large models become more or less sensitive to random labels.

Future Directions:

- Establish scaling laws for prompt sensitivity and label-dependence.
- Evaluate behavior across multilingual or domain-specialized LLMs.
- Study whether model size amplifies or mitigates formatting dominance.

“Min et al. [41] found that replacing the ground truth labels in in-context examples with random labels barely affected final performance. Further investigations by Yoo et al. [69] and Kossen et al. [30] found that this finding does not necessarily hold across tasks and model sizes. In particular, Kossen et al. [30], Lin and Lee [36] showed that LLMs can indeed learn input-output relationships via in-context learning, but require more examples in order to do so well.”

https://arxiv.org/pdf/2404.11018



Google DeepMind

2024-10-18

Many-Shot In-Context Learning

Rishabh Agarwal[*], Avi Singh[*], Lei M. Zhang[†], Bernd Bohnet[†], Luis Rosias[†], Stephanie C.Y. Chan[†], Biao Zhang[†], Ankesh Anand , Zaheer Abbas , Azade Nova , John D. Co-Reyes , Eric Chu , Feryal Behbahani , Aleksandra Faust and Hugo Larochelle

[*] Contributed equally, [†] Key contribution

Overview

- As context windows grew larger, ICL could become **many-shot learning**.
  - Hundreds to thousands of shots (input-output examples provided at inference without weight updates).
- Many-shot ICL can:

  1. Outperform few-shot ICL.
  2. Override pretraining biases.
  3. Rival supervised fine-tuning in some tasks.
- Approach:
  - Scale the number of in-context examples.
  - Evaluate across diverse tasks (reasoning, translation, planning, classification).
  - Introduce: **Reinforced ICL** (model-generated rationales) and **Unsupervised ICL** (inputs only, no demonstration).

Major Contributions & Technical Details

Growing Context Windows

- Evaluated Gemini 1.5 Pro with **1 million token** context length.
- Tasks studied:
  - Math problem solving.
  - Question answering.
  - Summarization.
  - Algorithmic reasoning.
  - Code verification.
  - Machine translation.
  - Planning.
  - Sentiment analysis.
- Certain tasks worsened in performance after a certain number of shots.
  - Code verification.
  - Planning.

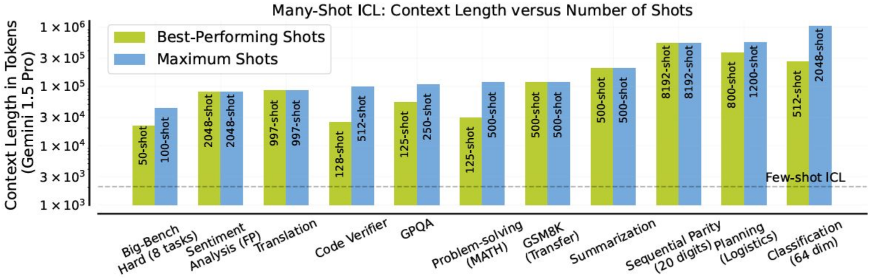


Figure 2 | **Context Length** for best-performing and the maximum number of shots tested for each task. The horizontal dashed line shows the context length of GPT-3 (2048 tokens), which is representative of typical few-shot prompts tested in the LLM literature. For several tasks, we observed the best-performing shots correspond to the maximum number of shots we tested, which was often limited by the number of available examples for in-context learning. On some tasks (e.g., code verifier, planning), we did observe slight performance deterioration beyond a certain number of shots.

https://arxiv.org/pdf/2404.11018

Analysis of Many-Shot ICL

- First systematic analysis of ICL with hundreds or thousands of examples (many-shot).
- Reveals that giving models more context examples can meaningfully improve their task performance.
- Many-shot settings overcome the biases learned during pretraining and enables models to learn high-dimensional functions with numerical inputs, tasks where few-shot ICL often fails.

Significance: Larger context windows (now available in newer LLMs) are not just useful for longer inputs, but for improving how effectively models can learn from examples at inference time.

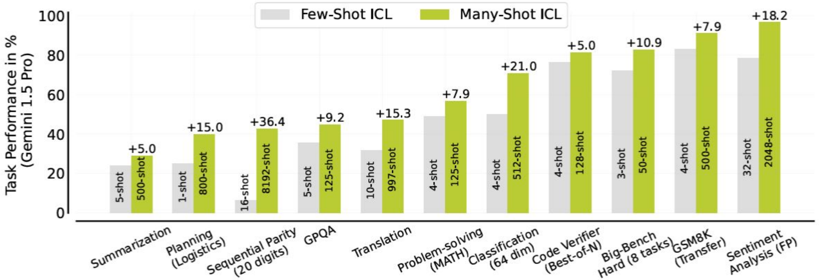


Figure 1 | **Many-shot vs Few-Shot In-Context Learning** (ICL) across several tasks. Many-shot ICL consistently outperforms few-shot ICL, particularly on difficult non-natural language tasks. Optimal number of shots for many-shot ICL are shown inside the bar for each task. For few-shot ICL, we either use typical number of shots used on a benchmark, for example, 4-shot for MATH, or the longest prompt among the ones we tested with less than the GPT-3 context length of 2048 tokens. Reasoning-oriented tasks, namely MATH, GSM8K, BBH, and GPQA use chain-of-thought rationales. For translation, we report performance on English to Bemba, summarization uses XLSum, MATH corresponds to the MATH500 test set, and sentiment analysis results are reported with semantically-unrelated labels. See §2, §3, and §4 for more details.

https://arxiv.org/pdf/2404.11018

New Novel ICL Methods

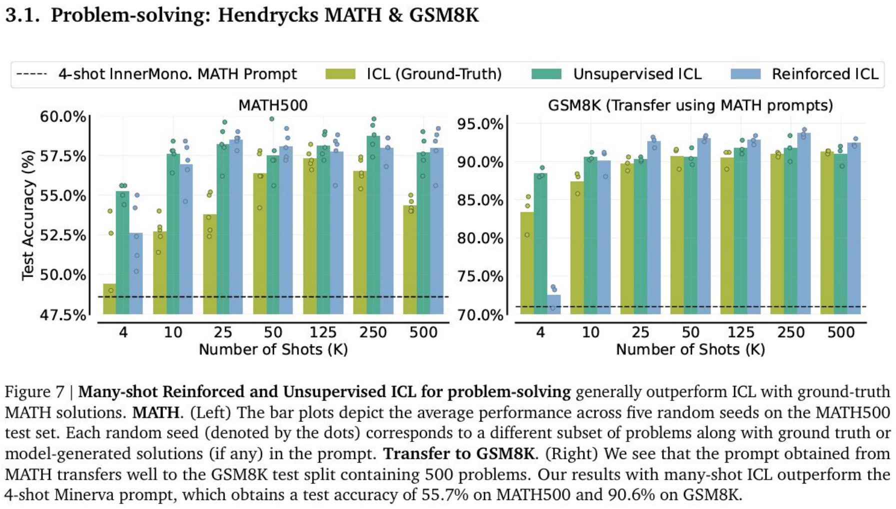
Two novel approaches for many-shot ICL have been developed when human-generated demonstrations are limited:

- **Reinforced ICL:** Uses the model’s own generated rationales (chain-of-thought explanations) as examples to effectively boost performance.
- **Unsupervised ICL:** Removes rationales from prompts entirely by prompting only with inputs (with a few input-output pairs to specify the format).

Significance: They reduce reliance on costly human-generated datasets while still enabling many-shot improvements, a practical advance for applying ICL in real-world settings where labeled data is scarce.

Reinforced and Unsupervised ICL Results

- On MATH500 (test set), both Reinforced and Unsupervised ICL outperforms ICL with ground-truth solutions in both the few-shot and many-shot regime.
- For ICL, performance improves up to a certain point (**125**), and then declines.
- Reinforced ICL performance also improves, but then plateaus (**25**).
- Comparable or superior performance with Unsupervised ICL.



3.1. Problem-solving: Hendrycks MATH & GSM8K

Figure 7 | **Many-shot Reinforced and Unsupervised ICL for problem-solving** generally outperform ICL with ground-truth MATH solutions. **MATH.** (Left) The bar plots depict the average performance across five random seeds on the MATH500 test set. Each random seed (denoted by the dots) corresponds to a different subset of problems along with ground truth or model-generated solutions (if any) in the prompt. **Transfer to GSM8K.** (Right) We see that the prompt obtained from MATH transfers well to the GSM8K test split containing 500 problems. Our results with many-shot ICL outperform the 4-shot Minerva prompt, which obtains a test accuracy of 55.7% on MATH500 and 90.6% on GSM8K.

https://arxiv.org/pdf/2404.11018

Many-Shot ICL Can Overcome Pre-training Biases

- Test whether many-shot ICl can overcome biases learned during pretraining when label semantics conflict with prior knowledge.
  - Flipped labels: Rotates semantic labels (e.g., positive → negative).
  - Abstract labels: Replaces sentiment labels with non-semantic symbols (A/B/C).
- With few-shot ICL, performance drops under flipped or abstract labels.
- With many-shot ICL, accuracy and confidence recover and approach default label performance.
- Many-shot ICL enables the model to overcome pretraining biases and infer the task from in-context examples.

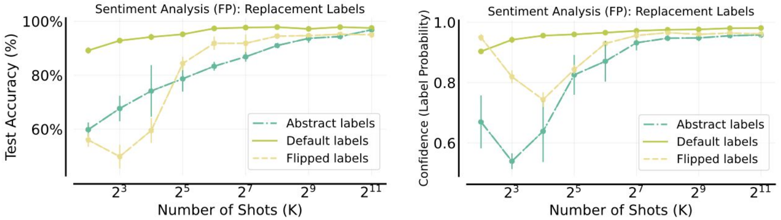


Figure 10 | **Overcoming Pre-Training Bias with Many-Shot ICL.** (Left) **Many-shot ICL overcomes label flips**: Test accuracy for sentiment analysis typically improves with more training shots. Flipped and abstract labels eventually approaching the performance of default labels. (Right) **Confidence shift in overcoming bias**. For flipped and abstract labels, model confidence in its predicted sentiment labels initially drops, then sharply increases with more training shots to similar value, suggesting a period of overcoming pre-training bias.

https://arxiv.org/pdf/2404.11018

Many-Shot Learning for High-Dimensional Functions

- Test many-shot ICL's ability to learn abstract mathematical functions with numerical inputs to test generality to unseen tasks.
  - Binary Linear Classification (find best hyperplane to linearly separate data).
  - Sequential Parity (even or odd number of 1's).
- Performance surpasses a GPT-2 Medium sized transformer trained on 20x more examples.
- Many-shot ICL can implement nearest-neighbour search over inputs and computations analogous to gradient descent.

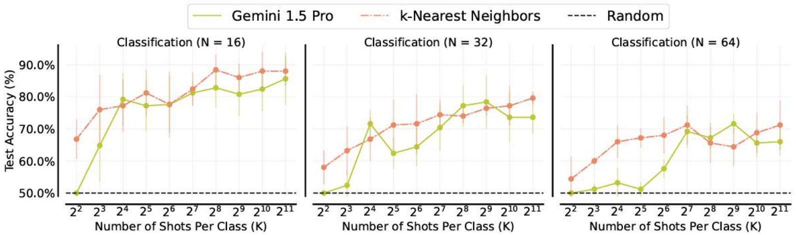
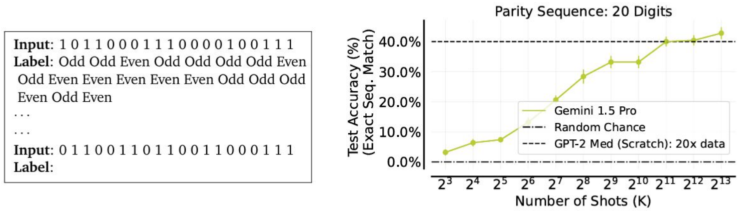


Figure 11 | **In-Context Classification**. Test accuracy for 16, 32 and 64 dimensional linear classification problems, averaged across 5 randomly-generated datasets with 25 points per class for each dataset (250 evaluation points total). As we increase the number of shots, the accuracy improves and approximately tracks the performance of the nearest-neighbor baseline trained from scratch on the same data. We use the default implementation of *k*-nearest neighbours (with *k* = 5) from scikit-learn [48]. See Figure A.7 for an example prompt.



Input: 1 0 1 1 0 0 0 1 1 1 0 0 0 0 1 0 0 1 1 1
Label: Odd Odd Even Odd Odd Odd Odd Even Odd Even Even Even Even Even Odd Odd Odd Even Odd Even
. . .
. . .
Input: 0 1 1 0 0 1 1 0 1 1 0 0 1 1 0 0 0 1 1 1
Label:

Figure 12 | **Learning Sequential Parity Function In-context**. We report test accuracy over 200 unseen inputs, averaged across 3 seeds. Error bars denote standard error of the mean. **Task Prompt**. (Left) Example prompt with input and output labels of the 20-digit Sequential Parity Function. **Test accuracy** (Right) Many-shot ICL performance improves almost monotonically with the number of shots, surpassing performance of GPT-2 Medium sized transformer trained from scratch for 1 forward-backward pass per example on 20× more data.

https://arxiv.org/pdf/2404.11018

Many-Shot ICL vs Supervised Fine-Tuning

- Many-shot ICL does not require any training like for fine-tuning, however it has a larger inference cost.
    - Can be reduced with KV caching.
    - Might be available off-the-shelf with context caching.
- For machine translation tasks, SFT and ICL performance is quite close for Bemba, while SFT has a slight edge for Kurdish.
- Many-shot ICL can be a viable alternative to SFT for some tasks.

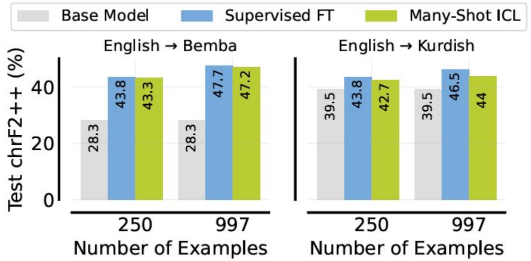


Figure 13 | **Comparing SFT with Many-Shot ICL** on low-resource translation. We plot mean performance across 3 seeds. The standard deviation is between 0.1% to 0.5%. Base model corresponds to 1-shot performance of Gemini 1.5 Pro.

https://arxiv.org/pdf/2404.11018

Model Comparison Results

- Compare Many-shot ICL for Gemini 1.5 Pro with GPT-4 Turbo (**128K** context length) , Claude-3-Opus (**200K** context length), and Gemini 1.5 Flash (smaller context length)
- Frontier LLMs exhibit varying degree of many-shot ICL capability.
- Even smaller LLMs can benefit from many-shot ICL and outperform LLMs with stronger few-shot performance with enough shots.

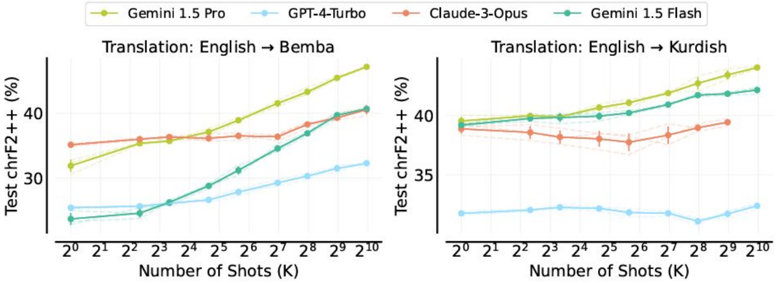


Figure 15 | Many-shot ICL with **GPT-4-Turbo** and **Claude-3-Opus** [3] on low-resource machine translation (§2.1).

https://arxiv.org/pdf/2404.11018

The Gains Behind Many-shot ICL

- Test whether gains from many-shot ICL arise from additional information or longer context length.
  - Distinct examples: Uses many unique examples to increase information content.
  - Repeated examples: Repeated 25 examples several times to form long prompts
- With repeated examples, performance remains nearly the same despite much longer prompts.
- With distinct examples, performance improves substantially as the number of shots increase.
- Many-shot ICL gains primarily stem from additional information rather than increased context length.

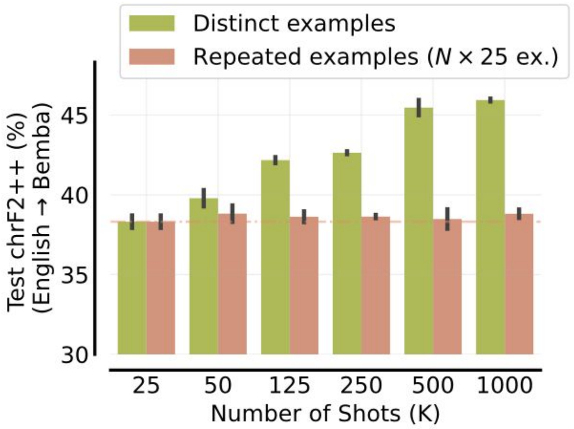


Figure 16 | **Many-shot performance with distinct examples vs repeating the same 25 examples *N* times** on low-resource MT. Bars show avg. perf with std across 3 seeds. Most of the benefit of many-shot ICL stems from adding new information.

https://arxiv.org/pdf/2404.11018

Multi-Shot Sensitivity to Example Ordering

- Few-shot ICL is known to be sensitive to example ordering.
- Authors test if this holds true for multi-shot ICL
  - Evaluate **10** orderings of **50** in-context training examples and evaluate performance on **500** test set examples.
- Multi-shot ICL is also affected by ordering.
- Could optimize prompts using frameworks such as DSPy but these results emphasize the challenge of ensuring reliable results .

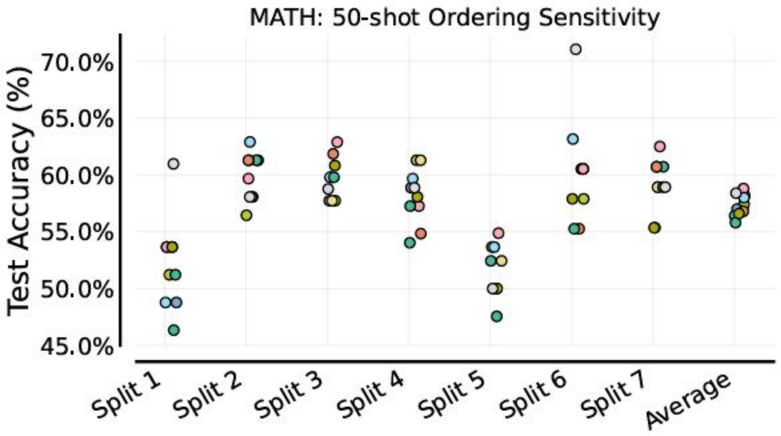


Figure 17 | **Many-Shot Sensitivity To Example Ordering.** Each colored data point represents a different random ordering of 50 in-context examples provided to Gemini 1.5 Pro.

https://arxiv.org/pdf/2404.11018

Limitations &
Future Directions

Challenges in Constructing Many-Shot Prompts

Limitations:

- Many-shot ICL often assumes access to large curated demonstration sets.
- Even reinforced ICL still requires initial high-quality seeds.

Future Directions:

- Develop automated demonstration generation pipelines.
- Combine ICL with retrieval systems to reduce the need for manual curation.

Context Windows’ Architectural Limits

Limitations:

- Current models have fixed context windows limiting how many examples can be used.
- KV cache constraints can bottleneck long prompts.
- Attention mechanisms may degrade with context length.

Future Directions:

- Explore architectures optimized for long contexts.
- Incorporate memory-augmented systems or retrieval-augmented transformers.
- Develop efficient compression or summarization methods for large demonstration sets.

Model Transferability

Limitations:

- Testing mainly limited to Gemini 1.5 Pro model.
- Experiments primarily focus on NLP reasoning, math, and synthetic numerical tasks.

Future Directions:

- Evaluate on a variety of models.
- Test many-shot performance across multimodal benchmarks.
- Explore generalization to RL-style environments or interactive tasks.

Summary

Title	**An Explanation of In-context Learning as Implicit Bayesian Inference**	**Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?**	**Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?**
Year	2021	2022	2024
Main idea	Mathematical and theoretical foundation of ICL	Ground truth demonstrations matter less than expected	Scaling ICL to many-shot
Key Results	ICL performs implicit Bayesian inference over latent concepts via HMMs	Demonstrations should specify label space, format and input distribution rather than exact input-label mappings	Many-shot ICL enables real task learning with fine-tuning-level performance
Main Limitation	Many assumptions that may not match real-world models and datasets	Does not generalize across tasks and model sizes	Mostly evaluated on one model (Gemini 1.5 Pro)

Sources

- https://arxiv.org/pdf/2111.02080
- https://arxiv.org/abs/2202.12837
- https://arxiv.org/abs/2404.11018
- https://ai.stanford.edu/blog/in-context-learning/
- https://www.geeksforgeeks.org/machine-learning/hidden-markov-model-in-machine-learning/

Questions?