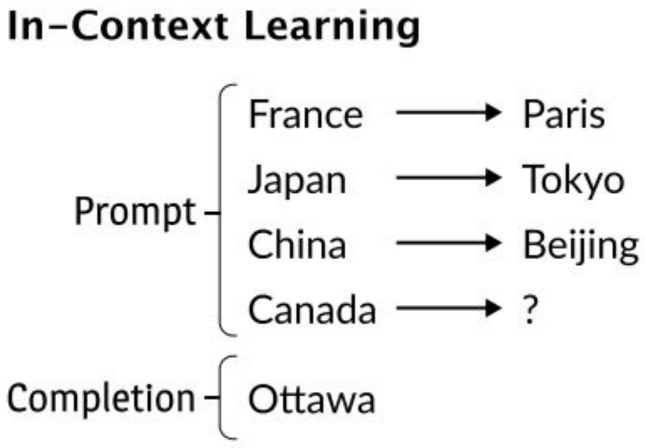
Understanding In-Context Learning in Transformers and LLMs by Learning to Learn Discrete Functions

Satwik Bhattamishra[▲] Arkil Patel[■] Phil Blunsom[▲¶] Varun Kanade[▲]

[▲]University of Oxford [■]Mila and McGill [¶]Cohere

In-Context Learning (ICL)

- A language model observes a sequence of labelled examples as part of context from a novel task and then makes a prediction on a new examples without updating its weights



Prior work: Formal framework for ICL

- Recent works [1-3] have adopted a stylised setup to understand ICL
- The models are trained from scratch on a set of prompts using a meta-learning objective
- Within the stylised framework, recent works have found that Transformers can learn various classes of real-valued functions in-context

[1] Garg et. al. What can transformers learn in-context? a case study of simple function classes. Neurips 2022
[2] von Oswald et. al. Transformers learn in-context by gradient descent. ICML 2023
[3] Bai et. al. Transformers as statisticians: Provable in-context learning with in-context algorithm selection. Neurips 2023

Our Work

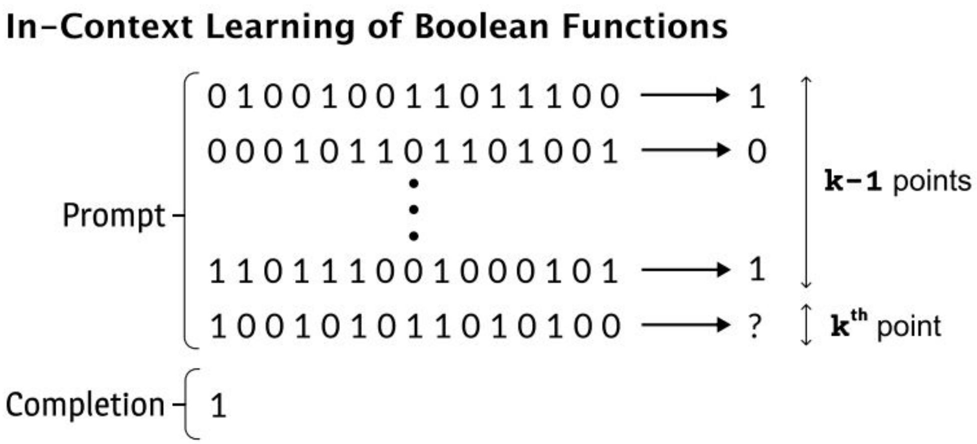
- Transformer's ability to ICL Boolean functions
- Comparison with other recently proposed architectures
- Ability to in-context learn with curated sequence of informative examples
- Ability of LLMs used in practice to act as learning algorithms

Why Boolean functions?

- Learnability of function classes is well understood which helps in designing the experimental framework
- We explore 10 different classes of Boolean functions of varying difficulty, e.g. in terms of VC dimension or noise sensitivity
- The discrete nature of inputs allows us to directly test LLMs as well

Setup

The model receives a prompt $P_k = (\mathbf{x}_1, y_1, \ldots, \mathbf{x}_{k-1}, y_{k-1}, \mathbf{x}_k)$ and the goal is to accurately predict the label for the input $\mathbf{x}_k$



Example

$$\begin{array}{lcccccc}
x_1 & 0 & 1 & 1 & 0 & 1 & 0 \\
x_2 & 0 & 0 & 1 & 0 & 0 & 0 \\
 & & & & \vdots & & \\
x_m & 1 & 0 & 1 & 0 & 1 & 1
\end{array}$$

Example: Conjunctions

$$z_2 \wedge \bar{z}_4 \wedge z_5$$

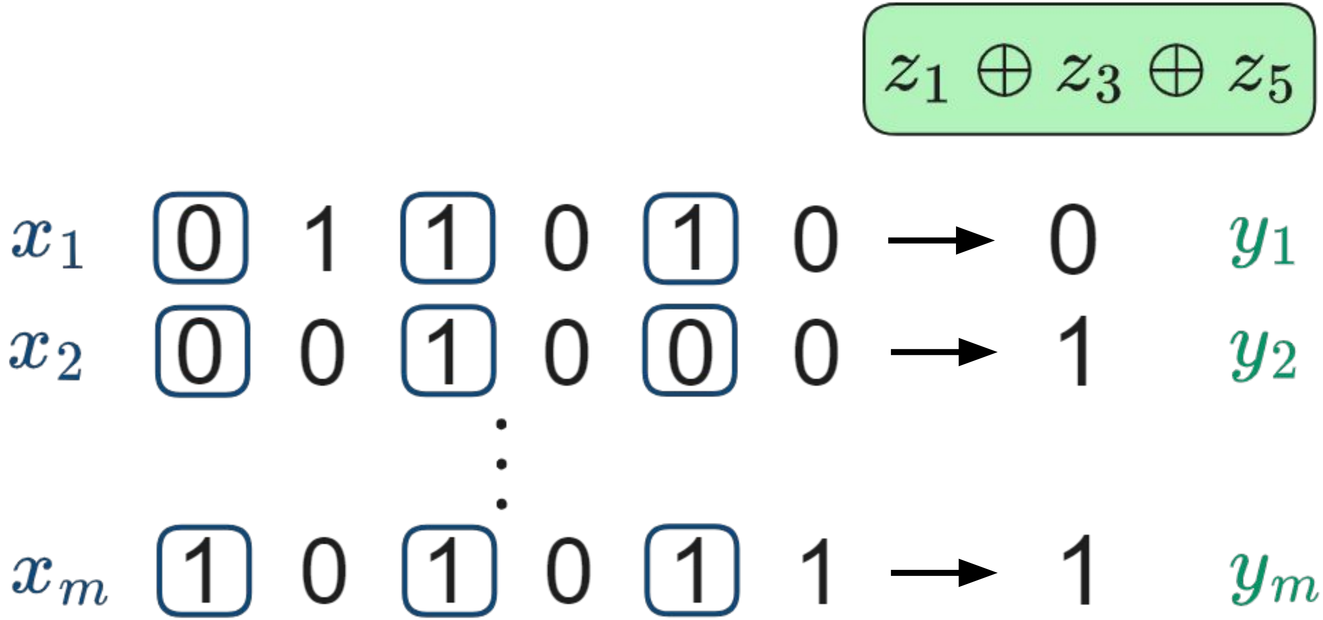
$$x_1 \quad 0 \ \boxed{1} \ 1 \ \boxed{0} \ \boxed{1} \ 0 \longrightarrow 1 \quad y_1$$

$$x_2 \quad 0 \ \boxed{0} \ 1 \ \boxed{0} \ \boxed{0} \ 0 \longrightarrow 0 \quad y_2$$

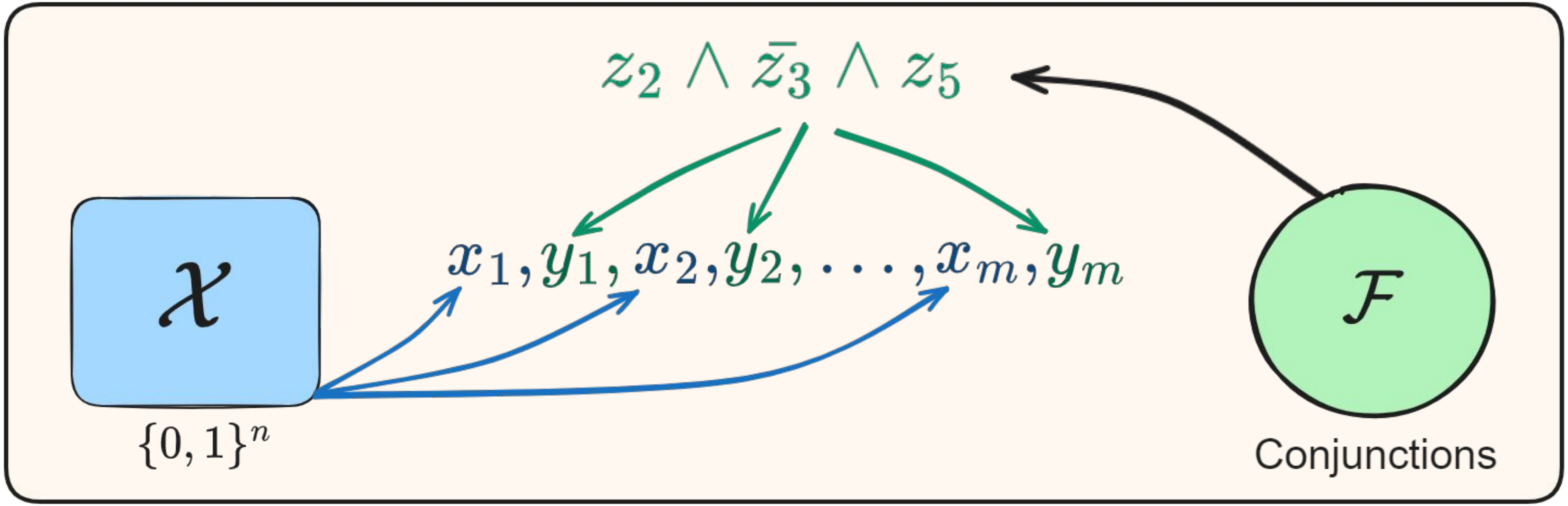
$$\vdots$$

$$x_m \quad 1 \ \boxed{0} \ 1 \ \boxed{0} \ \boxed{1} \ 1 \longrightarrow 0 \quad y_m$$

Example: Parities

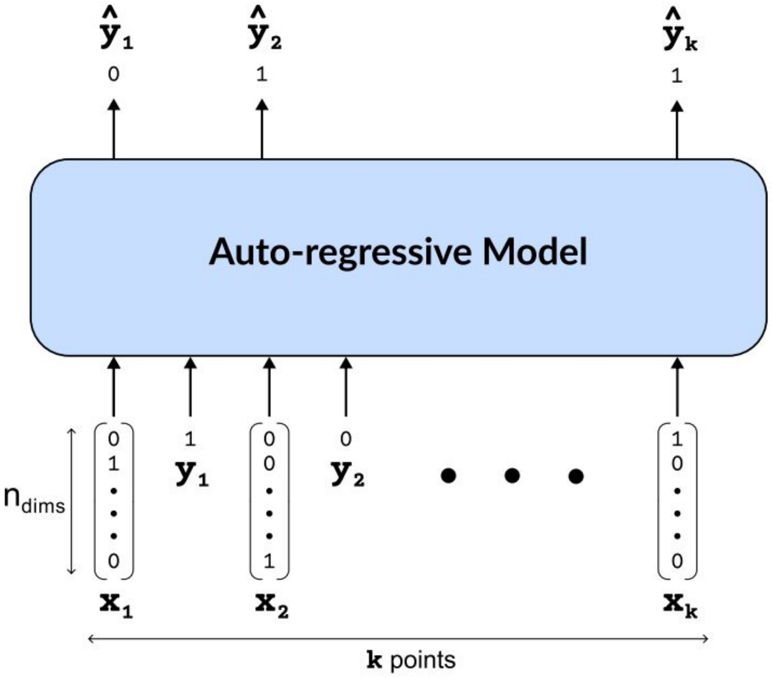


Generating Examples



Setup

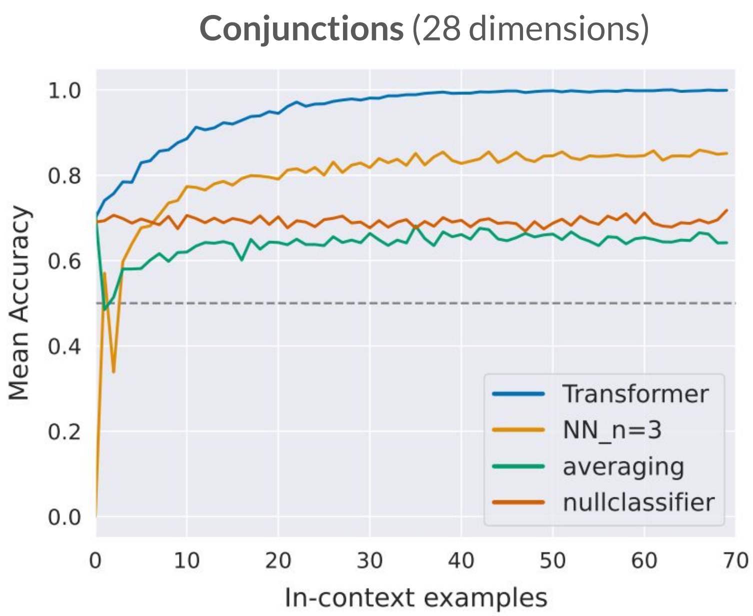
- Each prompt is created *m* examples and a function
- Model is trained from scratch on a set of prompts to predict the labels
- We explore various classes of Boolean functions such as Conjunctions, DNFs, Parities, etc



Results

In-Context Learning Boolean Functions

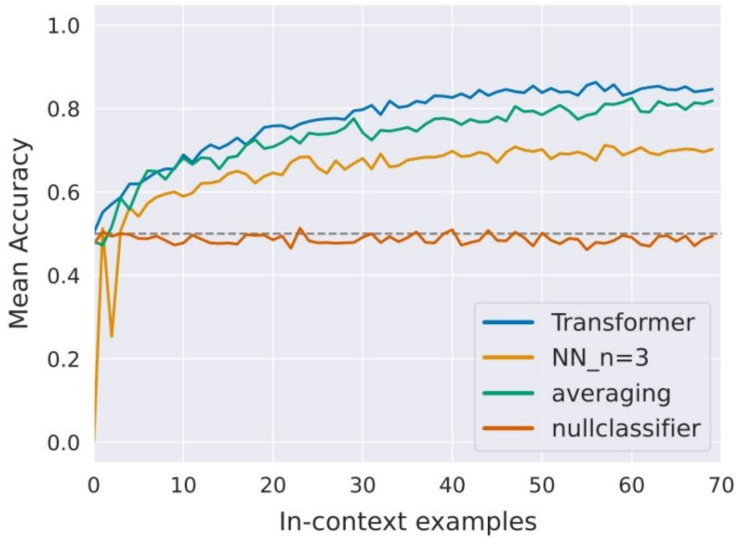
- Evaluate the performance of models on 10 different classes of varying complexity
- Transformers perform in near-optimal manner for some problems, but their performance becomes suboptimal on more complex classes



In-Context Learning Boolean Functions

- Evaluate the performance of models on 10 different classes of varying complexity
- Transformers perform in near-optimal manner for some problems, but their performance becomes suboptimal on more complex classes

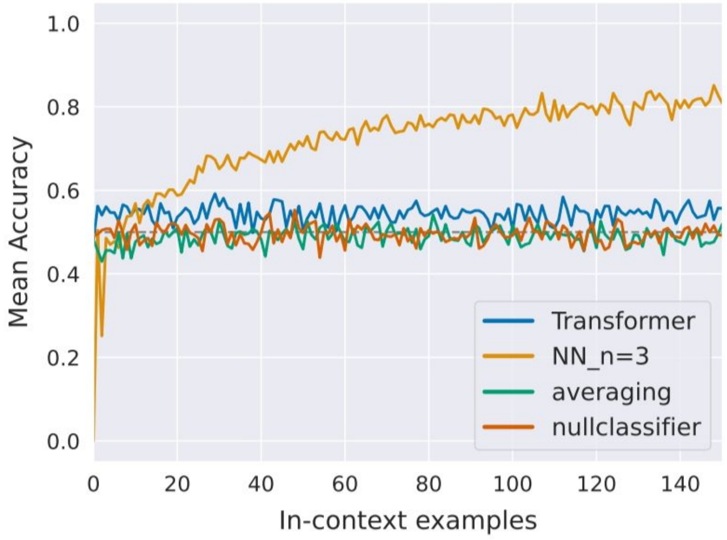
Linear Classifiers (20 dimensions)



In-Context Learning Boolean Functions

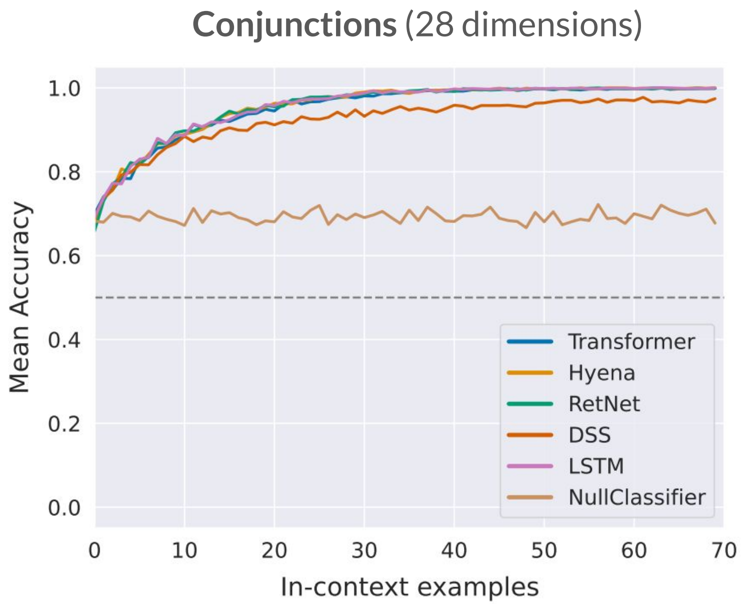
- Evaluate the performance of models on 10 different classes of varying complexity
- Transformers perform in near-optimal manner for some problems, but their performance becomes suboptimal on more complex classes

Sparse Parities (10 dimensions)



In-Context Learning Boolean Functions

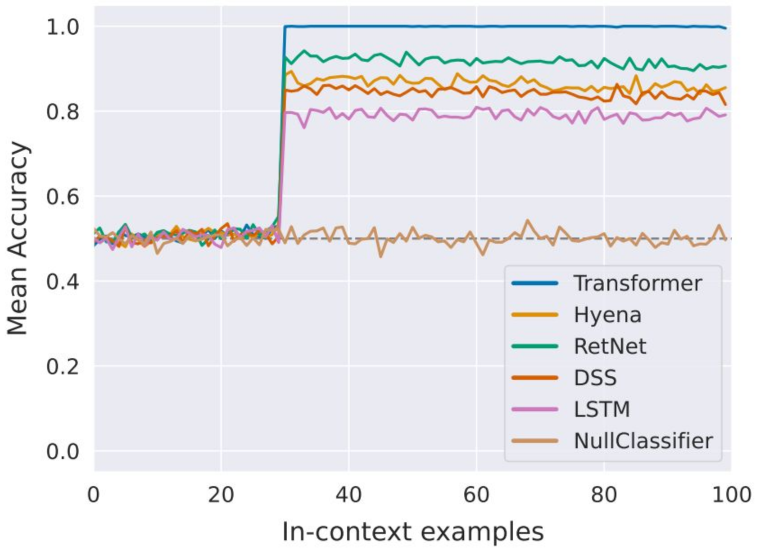
- Compare with attention-free models such as LSTMs, Hyena, RetNet and Diagonal state space models
- Attention-free models match Transformer's performance on most tasks but perform relatively worse on a few tasks



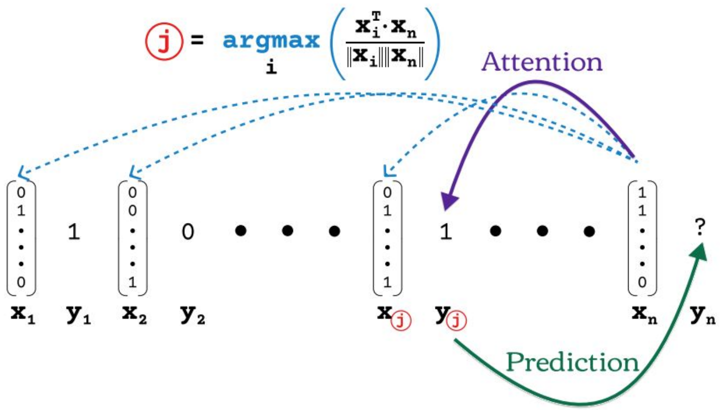
In-Context Learning Boolean Functions

- Compare with attention-free models such as LSTMs, Hyena, RetNet and Diagonal state space models
- Attention-free models match Transformer’s performance on most tasks but perform relatively worse on a few tasks

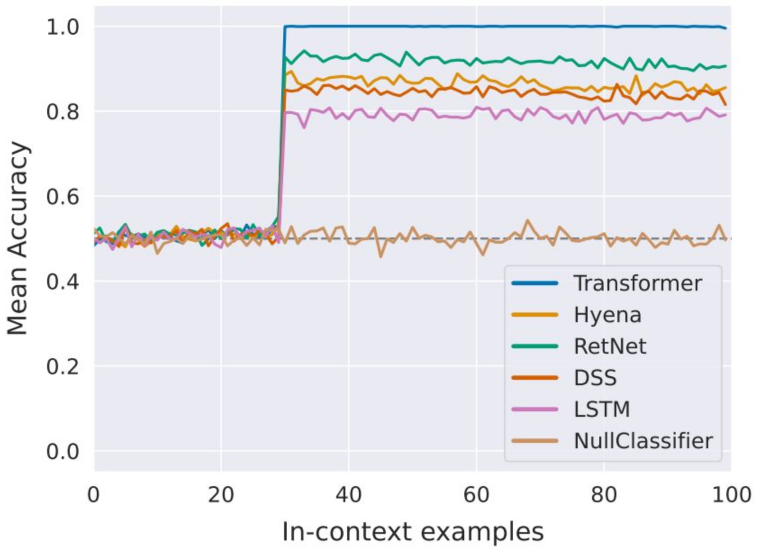
Nearest Neighbour (10 dimensions)



In-Context Learning Boolean Functions



Nearest Neighbour (10 dimensions)



Learning with Teaching Sequences

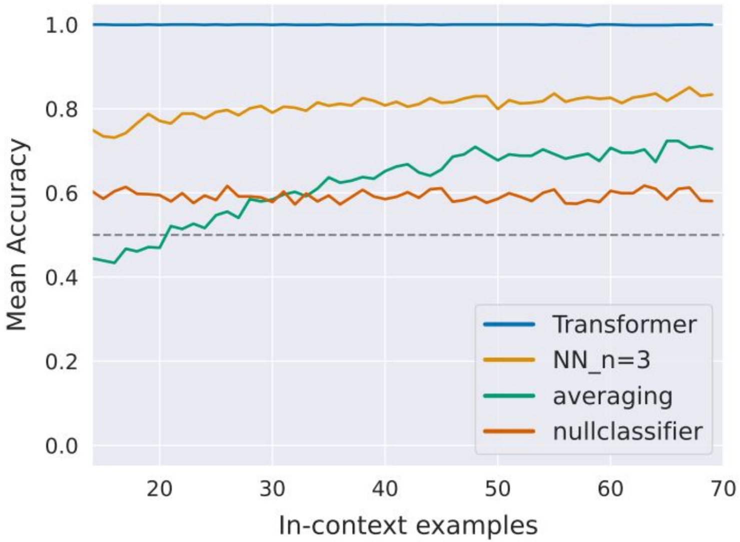
- Can models ICL more efficiently if provided with more informative prompts?
- Teaching sequence: A sequence of labelled examples which are sufficient to exactly identify the target function

$$\underbrace{t_1, y_1, \ldots, t_k, y_k,}_{\text{Teaching Sequence}} \underbrace{x_{k+1}, y_{k+1}, \ldots, x_m, y_m}_{\text{Random examples}}$$

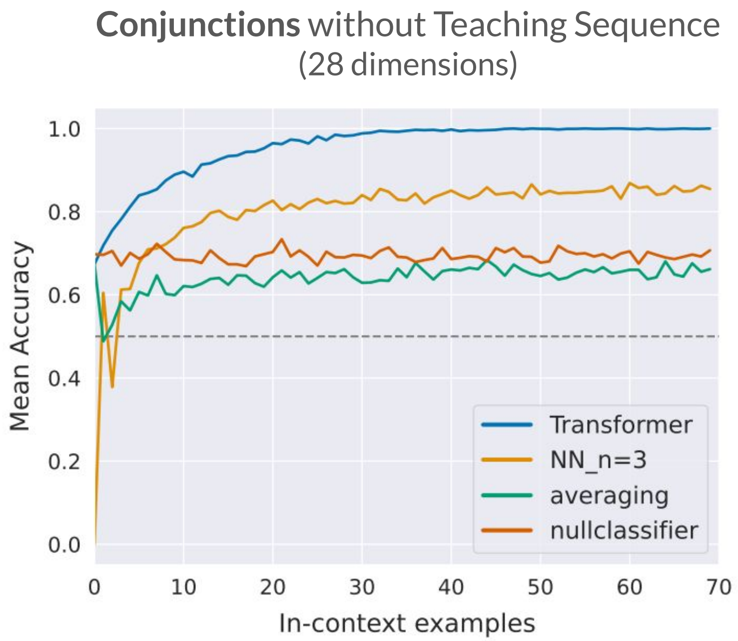
Learning with Teaching Sequences

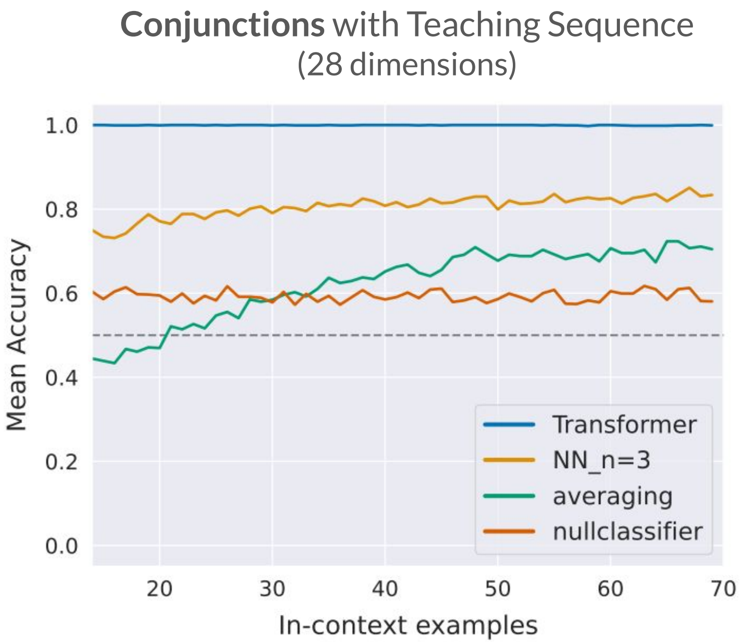
- Teaching sequence: A sequence of labelled examples which are sufficient to exactly identify the target function
- Instead of providing random examples, the prompt contains the teaching sequence followed by random examples

Conjunctions with Teaching Sequence
(28 dimensions)



Learning with Teaching Sequences





Investigations with LLMs

Investigations with LLMs used in practice

- Goal of this setup was to understand in-context learning (ICL) but how relevant is it for ICL with LLMs used in practice?
- Can pretrained models predict accurately by implementing learning algorithms or do they simply index from tasks seen during pretraining?

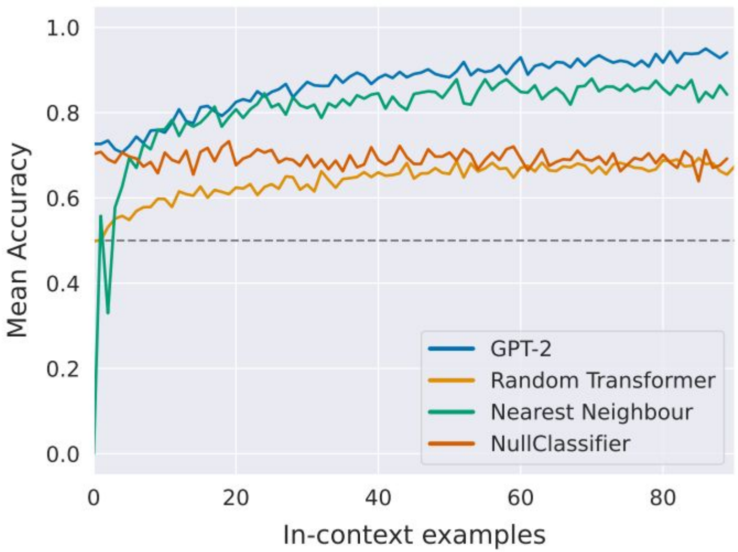
Frozen GPT Experiments

- Take a GPT-2 model with learnable input and output layers while all the weights of the Transformer model are frozen
- The model (input/output layers) is trained in the same way as earlier
- Baseline: A randomly initialized Transformer model with learnable input and output layers

Frozen GPT Experiments

- Take a GPT-2 model with learnable input and output layers while all the weights of the Transformer model are frozen
- Find that they are competitive with nearest neighbour on Conjunctions task and can implement the nearest neighbour algorithm

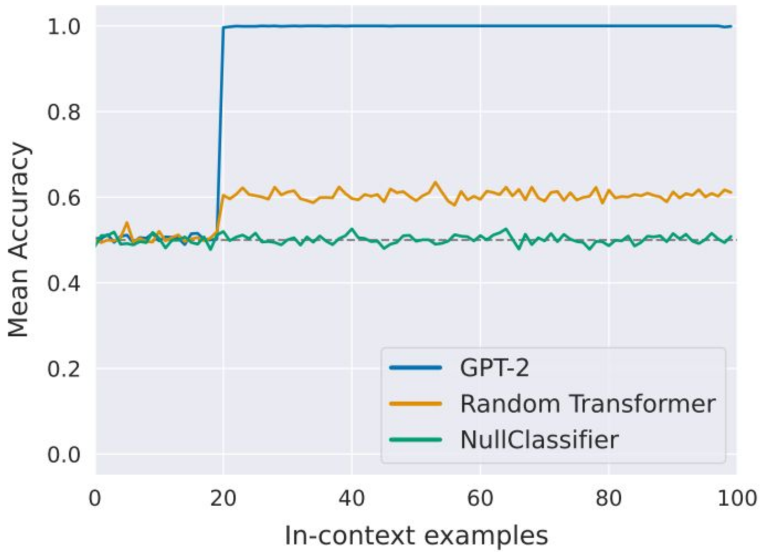
GPT2 - Conjunctions (28 dimensions)



Frozen GPT Experiments

- Take a GPT-2 model with learnable input and output layers while all the weights of the Transformer model are frozen
- Find that they are competitive with nearest neighbour on Conjunctions task and can implement the nearest neighbour algorithm

GPT2 - Nearest Neighbour (10 dimensions)



Can LLMs learn from in-context examples alone?

- Since we are working with discrete inputs, we can also directly evaluate LLMs such as LLaMA-2, GPT-4
- None of the parameters are modified and the original embeddings are used for tokens 0 and 1
- Goal is to test whether LLMs can learn solely from in-context examples

Few-shot Learning

- In practice, LLMs may rely on tasks already seen during pretraining
- Can LLMs learn from in-context examples alone?

The movie was great! → 1 (Positive)

The food was bad. → 0 (Negative)

The book was interesting → 1 (Positive)

The weather was nice → ?

Direct Evaluation with LLMs

You are given some examples of inputs and their corresponding labels. You need to learn the underlying boolean function represented by these input-label examples. Predict the label (either 0 or 1) for the final input.

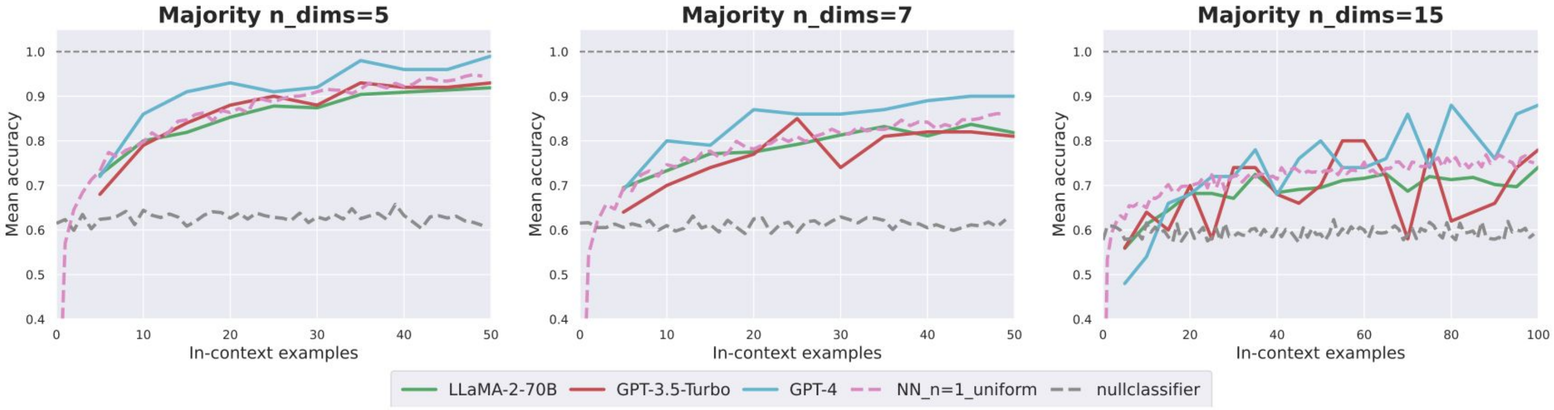
Input: 0 0 0 1 0
Label: 0
Input: 1 0 0 0 1
Label: 0
Input: 0 0 0 0 1
Label: 0

(. . . more exemplars . . .)

Input: 1 1 1 0 1
Label: 1
Input: 1 1 1 0 0
Label: 0
Input: 0 1 1 0 0
Label:

Direct Evaluation with LLMs

- Since we sample functions from a large combinatorial space, it is virtually guaranteed that LLMs are not pretrained on the same set of functions
- Find that LLMs perform as good as or better than Nearest neighbour baseline up to dimensions 7 on tasks such as Conjunctions, Majority, etc



finis.