Counterfactual Input, Probing, Intervention- An Attempt To Interpret GPT-2's Zero-Shot Sequence Completion Ability

Rahul Chowdhury [1]
David Bau [1]

Abstract

Large Language Models have emergent behaviors and those emergent behaviors have caught more attention and investments into Large Language Models. Large Language Models have been observed to perform sequence completion task where given a sequence of numbers GPT can output numbers that match the original future sequence to some extent. Understanding the working of it would help us to use GPTs to perform numerical and analysis, make improvements in the way numbers should be prompted for better future prediction, and could also hint at the dynamics of transformers during inference. Basic interpretability techniques were used to understand GPT's knowledge about sequences and find a possible algorithm used by GPTs to perform sequence completion. Results indicate that GPTs have knowledge to distinguish different classes of functions as time series and they also indicate that predictions were largely based on very local information and induction heads might have been used to predict sequences that had cyclic structure.

1. Introduction

(Mirchandani et al., 2023) showed that Large Language Models can extrapolate a given sequence of numbers. This raises questions such as do Large Language Models understand time series, are they just copying patters over, are they doing both. Answers to these questions will give us more confidence to use Large Language Models to perform time series analysis zero shot and use them for extrapolation for everyday tasks. The contributions in this paper are as follows:

- **probing deep layers of GPT-2 for basic time series elements**
- **probing deep layers of GPT-2 for sinusoidal properties**
- **intervening layers of GPT-2 to understand the influence of tokens at different layers and at horizons of the sequence**
- **stimulating GPT-2 with counterfactual inputs to check for the presence of induction heads and the influence of tokens at different layers and at horizons of the sequence**
- **attempt to find a linear model of the system that GPT-2 might be using to make prediction**

2. Related Work

(Mirchandani et al., 2023) has discovered the ways Large Language Model can generalize zero-shot to many patterns they were not trained on. One of them is sequence completion. (Mirchandani et al., 2023) has attributed to in-context learning for zero-shot sequence completion task. But exactly what it does and how does it learn in-context were not discovered. If Large Language Models have any understanding and are able to discriminate between different functions were not discovered. How could the behavior of Large Language Models differ when prompted a cyclic sequence versus an acyclic ones, and do Large Language Models switch states when solving either of them or do they solve both of the problems with one unified algorithm were also not discussed. (Gruver et al., 2024)LLM-Time is a paper that explored ways to make a pre-trained LLMs fit for time-series forecasting. They showed how effective LLMs could be zero-shot without finetuning for time series prediction when made a modification in the way numbers are spaced for tokenization. Still, these papers do not provide evidence why they work well and lacks interpretability that is important for betterment of performances.

[*]Equal contribution [1]Northeastern University, Boston, USA. Correspondence to: Rahul Chowdhury <chowdhury.rah@northeastern.edu>, David Bau <davidbau@northeastern.edu>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

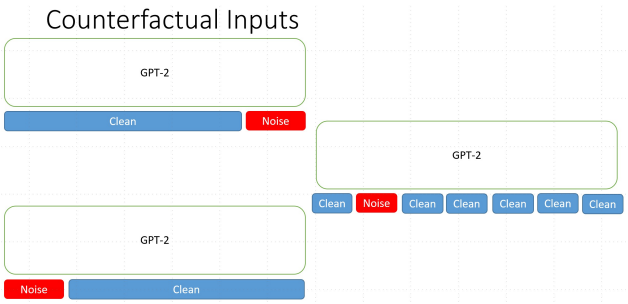

Figure 1. Three strategies of counterfactual inputs used to understand a GPT-2 as blackboxes

(Garza & Mergenthaler-Canseco, 2023) Time-GPT paper is a pure fure foundational model trained with just time-series and is tasked to solve time-series tasks. (Rasul et al., 2023) Lag-Llama is another foundation model that were trained purely on time-series data. They showed how effective transformers are at solving sequence tasks. But they are not of our interests since it is not a general purpose machine. (Jin et al., 2023) Time-LLM is a closer attempt to make use of Large Language Models for time-series prediction but it trains an adapter to align time-series to language tokens. This shows that Large Language Models are capable of performing time-series prediction through its knowledge but either it lacked alignment or lack of knowledge of LLMs' about sequence of knowledge led them to train an adapter. (Akyürek et al., 2022) and (Kantamneni et al., 2024) are papers that attemped to interpret neural network's in-context learning behavior using toy-models. But they still did not investigate Large Language Model's ability to extrapolate sequences on the fly. This paper makes an attempt to understand what does Large Language Models trained on text understand about time-series. Do it use the just copy or it makes informed decisions along with extending a pattern it recognizes.

3. Methods

3.1. Probing For Basic Time Series Elements

The first step to understanding Large Language Model's is to probe Large Language Model's layers for basic time series elements. Can each of the layers of the Large Language Models' activation distinguish exponential signals from sinusoidal ones, and pure signals from a mixture of sinusoidal. This gives us the first motivation to peek into Large Language Model to figure out what they use to make a prediction, how much do they understand about time series and how do they use their understanding to extrapolate time series on the fly. This was probed by extracting the GPT-2's activations of basic sinusoids, exponentials and mixture, followed by training a linear classifier (logistic regression) and a separating hyperplane to cluster them into their classes.

3.2. Probing For Basic Time Series Elements

Once separability of basic time series elements was discovered, sensitivity of GPT-2 to key characteristics needed to be probed for. This was probed by extracting the GPT-2's activations of sinusoids that had just one frequency and a composition of multiple frequencies. Activations were used to train multiple linear classifiers to identify the frequencies that could be present in each of the sinusoids, and regressors were also trained to regress the peak of each of the test sinusoids.

3.3. Activation Patching

In order to understand the encoding at each layer of the GPT-2 and its reaction to noise-infused sequence where noise was added either from the last token towards the first and first token towards the last. GPT-2 with noise infused inputs had to be intervened at each layer from activations from the GPT-2 from the corresponding clean sequence. This was done to check if there are certain signals that makes each of the layer ignore the inputs and makes its final decision just based on the activation coming from a clean signal. This would signal if there are layers that makes decisions disregarding the input.

3.4. Token-wise Counterfactual Noisy Input

Instead of counterfactual unidirectional noisy token sequence, noise was added to each of the token position and GPT-2's reactions were recorded to that noise infused input token sequence. Fig 1 illustrate the counterfactual input type that are used to study the importance of different token for sequence prediction. The reactions of our interest were how much does the final token's prediction deviate from its token prediction from a clean sequence when introduced a noise, and how much does the final token gets influenced, copying behavior, by the noisy token at certain position. Four cases were tested: sine wave with rational frequency that completed a cycle, sine wave with irrational frequency that completed a cycle, sine wave with rational frequency that did not complete a cycle, and sine wave with irrational frequency that did not complete a cycle.

3.5. Finding the model of prediction

Given the results from the prior experiments, some trends could be observed and model of the system needed be guessed to form one of the possible hypothesis. One toy question was asked: could we find a simple linear model that could describe the GPT-2's predictive behavior? In order to find that out simple experiments were performed. A sequence length of 50 tokens were used for a case of 2 Hz

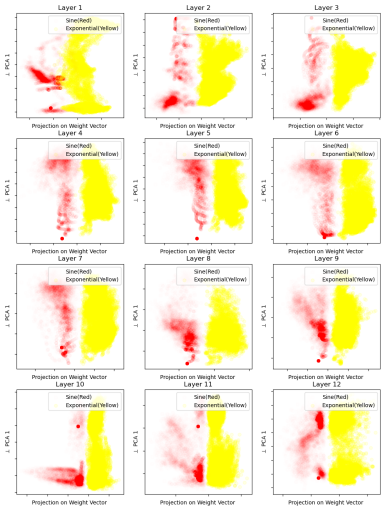

Figure 2. Linearly classifiable activation projection of sinusoids and exponential signals at each layer of GPT-2

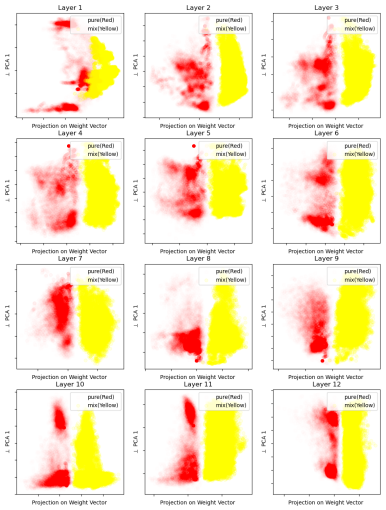

Figure 3. Linearly classifiable activation projection of mixture and pure sinusoids/ exponential signals at each layer of GPT-2

frequency sine wave and a case where the frequency was $\sqrt{5}$ Hz. Last three tokens were used to fit a linear model with polynomial of order n transformation using Ordinary Least Squares to find a polynomial that could be closest to the model's prediction.

4. Experiments

4.1. Probing for exponential, sinsuoids and mixture of both

After reaching near perfect accuracy for classifying sinusoid from exponential and vice versa, and then classifying mixture of sinusoids and exponentials from pure sinusoids and exponentials, the weight vector of the logistic regression was used along with the PCA1 perpendicularly projected on to the weight vector of the logistic regression. From Fig 2 it could be observed that GPT-2's activations could be linearly classified to detect exponentials from sinusoids and from Fig 3 it could be observed that the GPT-2's activations could be linearly classified to detect the mixture components from the pure waves.

4.2. Probing For Basic Time Series Elements

Classifiers trained at each layer to detect frequencies from pure sinusoids and sinusoids composed from two rational frequencies show very high accuracy from Table 1. There is one point to be noted that the accuracy decreases with higher frequencies. This could be due to the fact that there is lesser context points in higher frequencies than lower frequencies.

Regressors were also trained detect the peaks

4.3. Activation Patching

Table 3 and Table 4 show results of intervention where activations from a clean sequence were patched at each layer that received inputs that had a sequence of noise from the first token and clean tokens were added successively from the right most token. The first column where no activation patching was applied shows the mean absolute error of the predicted value against the ground truth and the successive columns show the mean absolute value from the predicted value against the intervening signals predictions. Table 5 and Table 6 show results of intervention where activations from a clean sequence were patched at each layer that received inputs that had a sequence of noise from the last token and clean tokens were added successively from the left most token. The first column where no activation patching was applied shows the mean absolute error of the predicted value against the ground truth and the successive columns show the mean absolute value from the predicted value against the intervening signals predictions. First column of Table 3 and Table where sine of frequency 2 Hz (an integer number frequency) and $\sqrt{5}$ Hz (an irrational number frequency) both had their loss reaching a value close to most of the sequence after introduction of 3 clean tokens. This shows the importance of first three tokens in predicting the next predicted value when the signals did not complete one full cycle. Sampling rate of these signals is 80 tokens per second, so a signal of 2 Hz completes one cycle at 40 tokens,

Table 1. Classification accuracies of three frequencies from waves containing pure and composition of different frequencies using activations of GPT-2 from each layer

LAYER	2 HZ	4 HZ	8 HZ
1	1.00	0.81	0.69
2	1.00	0.89	0.81
3	1.00	0.88	0.84
4	1.00	0.88	0.85
5	0.99	0.86	0.85
6	0.99	0.88	0.86
7	1.00	0.89	0.87
8	0.99	0.85	0.86
9	1.00	0.85	0.87
10	0.99	0.83	0.86
11	0.99	0.82	0.84
12	0.99	0.84	0.82

Table 2. Regression scores of peaks of waves composed of multiple frequencies using activations of GPT-2 from each layer

LAYER	2 AND 4 HZ	2 AND 8 HZ	4 AND 8 HZ
1	0.95	0.93	0.94
2	0.97	0.96	0.96
3	0.96	0.94	0.94
4	0.95	0.92	0.92
5	0.94	0.93	0.92
6	0.92	0.90	0.92
7	0.93	0.89	0.90
8	0.92	0.88	0.90
9	0.93	0.86	0.92
10	0.92	0.90	0.81
11	0.95	0.87	0.88
12	0.93	0.88	0.91

and a signal of $\sqrt{5}$ Hz completes a cycle after 35th token as the period of a sine wave with irrational frequency can only be approximated up to a certain point. When 2 Hz one full cycle the mean absolute error comes close to zero but that the wave with $\sqrt{5}$ Hz frequency does not follow suit. Interesting observation to note is that in Table 2 and Table 5, just the last noisy token can enough to ruin the prediction, giving rise to a high mean absolute error. Another Interesting observation to note that in Table 1 and Table 3 with full noise intervention could ignore the input fully and gave out the identical response from 10th layer onward and this points to the presence of induction heads getting triggered with well defined period.

4.4. Token-wise Counterfactual Noisy Input

Counter-factual noisy input was presented at each position one at a time and the corresponding output was recorded. Mean absolute error was calculated to measure the deviation of the output due to induction of one noisy token from the output of the original sequence. Mean absolute error was

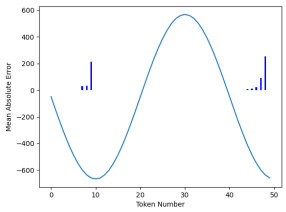

Figure 4. Reaction of counterfactual noisy input at each token position of length 50 tokens of 2 Hz sine wave

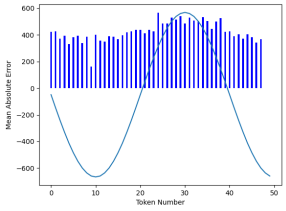

Figure 5. Reaction of counterfactual noisy input at each token position of length 50 tokens of 2 Hz sine wave

again calculated but another time to measure how much the output token matched the noisy input to evaluate GPT's sensitivity to the value of that particular token position. Four cases were tested where sinusoids of integer frequency (2 Hz) and irrational frequency ($\sqrt{5}$ Hz) completed one full cycle and where both did not complete one full cycle. Figure 4 shows that when an integer frequency wave form completes a cycle it pays attention chiefly to the three last neighboring tokens and the three tokens from the beginning up to the phase that it is on after it completes a full cycle. On top of that in Figure 5 it could be seen that 10th token has higher chance of being copied at the output. But if its cycle is not completed in Figure 8 GPT-2 focuses mainly on three contiguous last tokens. Sine wave with irrational frequency ($\sqrt{5}$ Hz) mainly focuses on last three contiguous token in both full cycles and incomplete case as shown in Figure 6 and Figure 10 with prominent copying tendency in incomplete cycle as shown in Figure 11 but not when cycle completes as shown in Figure 7. These all point out that there could be multiple algorithms at play to solve zero shot sequence completion problem present in GPT-2.

4.5. Finding the model of prediction

Linear models of polynomials of order n on waves of 50th token length for an integer frequency wave and irrational frequency were fitted to find some evidence of a specific model that GPT-2 could be using. Length 50 was selected since both waves completes a full cycle. Fig 12 shows that 2 Hz shows that a polynomial of order 1 which is a simple linear combination of last three tokens could be used

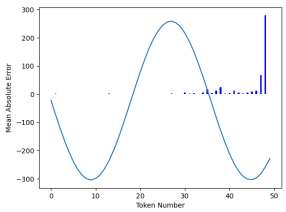

Figure 6. Reaction of counterfactual noisy input at each token position of length 50 tokens of $\sqrt{5}$ Hz sine wave

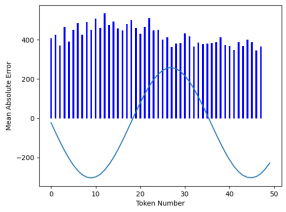

Figure 7. Reaction of counterfactual noisy input at each token position of length 50 tokens of $\sqrt{5}$ Hz sine wave

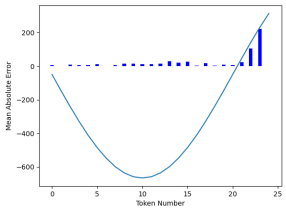

Figure 8. Reaction of counterfactual noisy input at each token position of length 25 tokens of 2 Hz sine wave

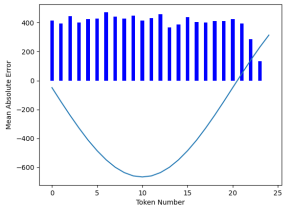

Figure 9. Reaction of counterfactual noisy input at each token position of length 25 tokens of 2 Hz sine wave

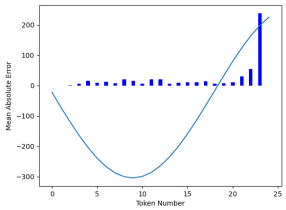

Figure 10. Reaction of counterfactual noisy input at each token position of length 25 tokens of $\sqrt{5}$ Hz sine wave

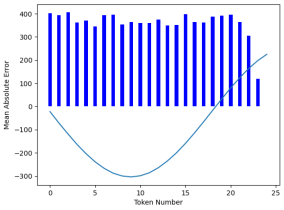

Figure 11. Reaction of counterfactual noisy input at each token position of length 25 tokens of $\sqrt{5}$ Hz sine Wave

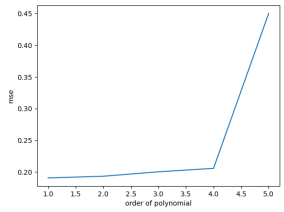

Figure 12. Experiments showing mse of polynomial fitting using last three tokens for 2 Hz waveform

to predict the 50th token with good accuracy of around mse 3. But when different linear models were tested for frequency ($\sqrt{5}$ Hz, even very high order polynomial could not approximate the model that GPT-2 is using for predicting the 50th token from last three tokens. (Note: All of the experiments were performed using GPT-2 small with the help of nnsight)

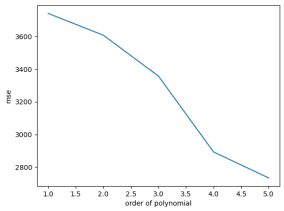

Figure 13. Experiments showing mse of polynomial fitting using last three tokens for $\sqrt{5}$ Hz waveform

Counterfactual Input Sequence Length	Intervening Layer												
	Nill	1	2	3	4	5	6	7	8	9	10	11	12
0	251	5970	4519	785	2902	800	1080	13946	2969	976	0	0	0
1	187	407	227	234	272	5315	2225	820	3	0	0	0	0
2	89	1661	201	220	280	1967	164	2775	0	3	0	0	0
3	56	561	199	207	268	563	805	3011	3	0	0	0	0
4	46	1678	190	199	271	1967	798	3016	3	0	0	0	0
5	39	551	190	199	259	898	798	3012	3	0	0	0	0
6	38	549	186	196	250	902	795	2570	4	0	0	0	0
7	38	548	186	198	250	889	795	386	4	0	0	0	0
8	36	1694	183	195	277	897	788	379	4	0	0	0	0
9	34	2032	182	192	559	892	783	805	0	0	0	0	0
10	34	2032	182	192	586	895	784	795	4	0	0	0	0
15	44	2018	187	202	308	880	531	536	4	0	0	0	0
20	52	1435	245	250	738	1391	1315	852	413	0	0	0	0
25	59	999	244	250	374	2819	1328	832	413	0	0	0	0
30	57	662	244	255	336	2823	846	831	413	0	0	0	0
35	52	1938	243	252	729	1067	843	821	413	0	0	0	0
40	53	661	242	246	352	1377	832	808	413	0	0	0	0
60	1	457	158	140	197	771	763	760	413	0	0	0	0
80	0	436	182	133	138	764	758	413	413	0	0	0	0
120	0	413	166	150	442	413	413	413	0	0	0	0	0
160	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 3. Steering GPT-2 with counterfactual activation and input on an original sequence input of a sine wave of frequency 2 Hz

Counterfactual Input Sequence Length	Intervening Layer												
	Nill	1	2	3	4	5	6	7	8	9	10	11	12
0	251	252	258	254	3308	260	310	12808	3458	880	523	825	0
1	209	290	290	295	3646	782	486	7867	72	21	555	825	0
2	62	157	172	218	4016	553	291	7682	37	15	552	824	0
3	38	999	163	183	3594	610	683	2242	55	37	548	825	0
4	21	1009	155	163	3576	620	621	2242	34	15	552	825	0
5	22	1005	153	165	3581	568	203	2240	29	18	549	824	0
6	22	1006	153	164	3580	550	596	951	27	22	523	825	0
7	21	1006	148	163	3580	106	587	901	26	17	526	824	0
8	20	1002	149	160	3573	202	542	890	24	17	526	824	0
9	21	1002	149	158	3568	199	983	898	21	18	523	824	0
10	22	1001	148	160	3554	198	526	883	24	15	526	824	0
15	22	1431	635	646	4088	668	600	847	467	48	526	491	0
20	27	2230	1112	642	996	675	1055	550	464	47	527	491	0
25	25	1730	600	642	994	610	1041	98	17	46	832	489	0
30	27	2206	1108	640	1091	1061	1034	543	465	49	527	490	0
35	26	1735	632	644	1094	606	1038	537	465	45	527	489	0
40	23	1399	632	643	698	1653	1030	571	21	13	492	491	0
60	21	777	1123	146	499	98	484	83	14	10	492	489	0
80	18	754	831	143	3580	593	476	533	14	9	491	489	0
120	19	715	855	830	802	718	712	83	17	13	3	489	0
160	18	0	0	0	0	0	0	0	0	0	0	0	0

Table 4. Steering GPT-2 with counterfactual activation and input on an original sequence input of a sine wave of frequency $\sqrt{5}$ Hz

Counterfactual Input Sequence Length	Intervening Layer												
	Nill	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	248	248	248	232	114	41	22	0	0	0	0	0	0
2	251	251	251	249	185	109	46	1	0	0	0	0	0
3	251	250	250	248	201	497	444	0	0	0	0	0	0
4	251	250	250	261	271	524	462	0	0	0	0	0	0
5	251	250	250	340	284	508	452	0	0	0	0	0	0
6	251	251	250	345	296	500	42	0	0	0	0	0	0
7	251	251	250	342	294	516	53	0	0	0	0	0	0
8	251	250	282	312	233	552	63	0	0	0	0	0	0
9	251	252	286	273	213	517	54	0	0	0	0	0	0
10	251	253	287	298	209	488	38	0	0	0	0	0	0
15	251	252	345	292	225	60	29	0	0	0	0	0	0
20	251	251	293	284	640	486	29	0	0	0	0	0	0
25	251	251	285	297	921	435	13	0	0	0	0	0	0
30	251	252	303	285	562	425	3	0	0	0	0	0	0
35	251	253	293	252	567	427	1	0	0	0	0	0	0
40	251	252	273	252	314	717	114	44	1	1	0	0	0
60	251	248	262	251	210	442	15	0	0	0	0	0	0
80	251	292	256	252	292	541	101	83	1	0	0	0	0
120	251	234	249	252	331	621	169	157	0	0	0	0	0
160	251	252	264	251	282	251	251	251	27	15	0	0	0

Table 5. Steering GPT-2 with counterfactual activation and input on an noisy sequence input of a sine wave of frequency 2 Hz

Counterfactual Input Sequence Length	Intervening Layer												
	Nill	1	2	3	4	5	6	7	8	9	10	11	12
0	18	0	0	0	0	0	0	0	0	0	0	0	0
1	231	234	1825	1809	140	81	66	20	14	8	3	0	0
2	251	249	249	250	235	170	115	29	20	6	6	0	0
3	251	249	249	265	1839	1800	123	16	15	339	2	0	0
4	251	249	249	269	1829	197	172	28	20	6	3	0	0
5	251	249	254	313	244	199	174	12	10	3	4	0	0
6	251	250	261	317	249	216	189	24	17	6	6	2	0
7	251	250	266	309	250	247	211	27	20	6	6	2	0
8	251	249	293	303	250	249	210	20	18	5	2	2	0
9	251	251	299	289	244	686	640	21	20	3	4	1	0
10	251	252	307	295	244	232	641	32	18	5	492	2	0
15	251	251	353	253	280	150	107	19	10	3	490	2	0
20	251	250	303	258	239	463	439	20	12	491	490	0	0
25	251	250	297	258	299	459	85	17	10	491	490	0	0
30	251	250	327	253	217	433	71	16	3	490	521	0	0
35	251	251	280	254	238	473	437	360	15	830	521	0	0
40	251	250	269	255	730	93	74	366	9	491	490	0	0
60	251	252	263	252	690	98	81	366	833	827	521	0	0
80	251	290	264	251	293	546	2141	473	833	827	522	0	0
120	251	261	254	251	1204	917	572	1315	528	492	525	0	0
160	251	303	261	251	282	250	250	246	127	63	530	3	0

Table 6. Steering GPT-2 with counterfactual activation and input on an noisy sequence input of a sine wave of frequency $\sqrt{5}$ Hz

5. Conclusion

GPT-2's remarkable ability to complete sequence had many wonder if they were simple following one algorithm to complete the sequences and most papers used a blanket statement of in-context learning to explain its behaviors. Evidence shows otherwise. Evidence shows that GPT-2 is sensitive to wave properties like frequency, phase and amplitude and are wary about different functions that could represent a time-series. Evidence and experimental results point out the importance of last three tokens for predicting the next one, but if these last three tokens could give us a linear model to emulate GPT-2's predictive mechanism remains to be seen. But from the toy experiments it could be guessed that there could be one out of many possible algorithms that could be linearly modeled with last three tokens under unique condition like if the wave completed one full cycle and if the wave is simple coming from a integer number of frequencies. Finally, in-context learning through induction could not be ruled out since experiments confirm that when integer frequency sine wave is queried GPT-2 has a tendency to be sensitive to three last tokens in the past corresponding to the phase after full cycle and evidence strongly points out it copies token input from those token and use them to make prediction about the next one along with the immediate last tokens. GPT certainly uses multiple algorithms to multiple algorithms to solve sequence completion problems, and the number and modes are yet to be discovered.

References

Akyürek, E., Schuurmans, D., Andreas, J., Ma, T., and Zhou, D. What learning algorithm is in-context learning? investigations with linear models. *arXiv preprint arXiv:2211.15661*, 2022.

Garza, A. and Mergenthaler-Canseco, M. Timegpt-1. *arXiv preprint arXiv:2310.03589*, 2023.

Gruver, N., Finzi, M., Qiu, S., and Wilson, A. G. Large language models are zero-shot time series forecasters. *Advances in Neural Information Processing Systems*, 36, 2024.

Jin, M., Wang, S., Ma, L., Chu, Z., Zhang, J. Y., Shi, X., Chen, P.-Y., Liang, Y., Li, Y.-F., Pan, S., et al. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*, 2023.

Kantamneni, S., Liu, Z., and Tegmark, M. How do transformers" do" physics? investigating the simple harmonic oscillator. *arXiv preprint arXiv:2405.17209*, 2024.

Mirchandani, S., Xia, F., Florence, P., Ichter, B., Driess, D., Arenas, M. G., Rao, K., Sadigh, D., and Zeng, A. Large language models as general pattern machines. *arXiv preprint arXiv:2307.04721*, 2023.

Rasul, K., Ashok, A., Williams, A. R., Khorasani, A., Adamopoulos, G., Bhagwatkar, R., Biloš, M., Ghonia, H., Hassen, N., Schneider, A., et al. Lag-llama: Towards foundation models for time series forecasting. In *R0-FoMo: Robustness of Few-shot and Zero-shot Learning in Large Foundation Models*, 2023.