CONTEXT PARROTING: A SIMPLE BUT TOUGH-TO-BEAT BASELINE FOR FOUNDATION MODELS IN SCIENTIFIC MACHINE LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent time-series foundation models exhibit strong abilities to predict physical systems. These abilities include zero-shot forecasting, in which a model forecasts future states of a system given only a short trajectory as context, without knowledge of the underlying physics. Here, we show that foundation models often forecast through a simple parroting strategy, and when they are not parroting they exhibit some shared failure modes such as converging to the mean. As a result, a naive context parroting model that copies directly from the context scores higher than leading time-series foundation models on predicting a diverse range of dynamical systems, including low-dimensional chaos, turbulence, coupled oscillators, and electrocardiograms, at a tiny fraction of the computational cost. We draw a parallel between context parroting and induction heads, which explains recent works showing that large language models can often be repurposed for time series forecasting. Our dynamical systems perspective also ties the scaling between forecast accuracy and context length to the fractal dimension of the underlying chaotic attractor, providing insight into previously observed in-context neural scaling laws. By revealing the performance gaps and failure modes of current time-series foundation models, context parroting can guide the design of future foundation models and help identify in-context learning strategies beyond parroting.

1 INTRODUCTION

A key test of generalization in scientific machine learning (SciML) is zero-shot forecasting: the ability to forecast future states of a new physical system based on a short context trajectory. Prior SciML approaches primarily focus on developing specialized forecasting models trained specifically on the system that needs to be predicted (Brunton et al., 2016; Weinan, 2017; Chen et al., 2018; Pathak et al., 2018; Li et al., 2020; Chen & Tao, 2021; Jordan et al., 2021; Gauthier et al., 2021; Lim & Zohren, 2021; Karniadakis et al., 2021; Levine & Stuart, 2022; Mikhaeil et al., 2022; Brunton et al., 2022; Das et al., 2023; Krishnapriyan et al., 2023; Yang et al., 2024; Yu & Wang, 2024; Azizzadenesheli et al., 2024; Brenner et al., 2024a;b; Ricci et al., 2024; He et al., 2025; Cheng et al., 2025; Grigoryeva et al., 2025; Berry & Das, 2025). However, the generality of these models is limited by the amount of system-specific data available, motivating the recent development of time-series foundation models (Oreshkin et al., 2021; Garza & Mergenthaler-Canseco, 2023; Rasul et al., 2023; Jin et al., 2023; Zhou et al., 2023; Gruver et al., 2024; Dooley et al., 2024; Liu et al., 2024b; Woo et al., 2024; Ansari et al., 2024; Goswami et al., 2024; Das et al., 2024; Liang et al., 2024; Shi et al., 2025; Zhai et al., 2024; Liu et al., 2025b), which are trained on vast amounts of observed and simulated time series from diverse domains, and which can subsequently perform zero-shot forecasts for any time series—including those generated by previously-unseen dynamical systems. Interestingly, it was recently found that, when available historical data is limited, time-series foundation models outperform classical deep learning models in forecasting chaotic dynamical systems (Zhang & Gilpin, 2024).

What mechanisms do time-series foundation models use to make zero-shot forecasts, and why they are effective for dynamical systems not seen during pre-training? It was recently observed that one such foundation model, Chronos (Ansari et al., 2024), often employs an extremely simple strategy when forecasting chaotic systems (Zhang & Gilpin, 2024). The strategy, *context parroting*, scans the context for nearly repeating motifs and copies the part of the context following the best-matching motif as its prediction (Fig. 1). This can be viewed as a kind of "in-context nearest neighbor" algorithm, which is easy to implement during in-context computation (Garg et al., 2022). How good is context parroting as a zero-shot forecasting strategy? By comparing it with existing foundation models, what can we learn about current models' strengths and limitations?

Here, we compare context parroting with a diverse set of competitive baselines on the challenging task of forecasting chaotic systems. Our baselines include four state-of-the-art time-series foundation models: Chronos and Chronos Bolt (Ansari et al., 2024), TimesFM (Das et al., 2024), Time-MoE (Shi et al., 2025), and Moirai (Woo et al., 2024), as well as a recent foundation model specifically designed for dynamical systems: DynaMix (Hemmer & Durstewitz, 2025). In the Appendix, we also include two classical forecasting methods that are particularly effective in the small-data limit: AutoARIMA (Hyndman & Athanasopoulos, 2018) and simplex projection (Sugihara & May, 1990). The latter represents a classical nonlinear forecasting method conceptually resembling context parroting (Appendix H). We find that parroting outperforms all baselines (including the leading foundation models) in both zero-shot forecast accuracy and inference cost, especially for longer context windows. Our results suggest that current time-series foundation models do not fully utilize the information in the context data, and thus still have significant room for improvement when it comes to SciML tasks.

Our main contributions are:

1. Introduce context parroting as a simple but effective baseline for zero-shot forecasting of dynamical systems, which can guide the design of more informative benchmarks that cannot be solved by simple repetitions and help identify forecasting strategies beyond parroting
2. Show that context parroting outperforms leading time-series foundation models in predicting chaotic systems and reveal common failure modes of many existing foundation models, which can guide the design of better models in the future
3. Explain the in-context neural scaling law between forecast accuracy and context length, linking the scaling coefficient to the fractal dimension of the underlying chaotic attractor

2 RELATED WORK

Foundation models for science. Foundation models have recently been introduced for many scientific machine-learning tasks (Miller et al., 2024), including partial differential equations (Takamoto et al., 2022; Yang et al., 2023; Rahman et al., 2024; Subramanian et al., 2024; Herde et al., 2024; McCabe et al., 2024; Totounferoush et al., 2025), neuroscience (Cui et al., 2024; Caro et al., 2023; McKeen et al., 2024), and weather forecasting (Nguyen et al., 2023; Bodnar et al., 2024). However, most of these foundation models remain a black box, and they have not yet provided interpretable strategies for forecasting diverse physical and dynamical processes. Here, we analyze context parroting as a simple mechanism used by time-series foundation models, noting its strengths and weaknesses as a zero-shot forecasting strategy. This strategy, and the insights gained here, can potentially be applied to other scientific tasks.

In-context neural scaling laws. Neural scaling laws describe the relationship between the performance of a neural network and certain resources, such as model size, data size, or the amount of compute (Kaplan et al., 2020; Sorscher et al., 2022; Bahri et al., 2024; Yao et al., 2024). Such scaling laws allow practitioners to predict the performance of yet-to-be-trained models based on the available resources and allocate them strategically to optimize compute-adjusted accuracy (Hoffmann et al., 2022). When applying LLMs to forecast dynamical systems, Liu et al. (2024a) recently observed an in-context neural scaling law, in which the test loss decreases with the context length following a power law. Here, we show that this in-context neural scaling law can be reproduced when using context parroting to predict dynamical systems, and the scaling coefficient can be linked to an invariant property of the underlying dynamic process (the fractal dimension of the chaotic attractor). This finding shows that neural scaling laws are intrinsically linked to invariant properties of the process generating the data, and the theory can potentially be generalized to other models and tasks (e.g., can we estimate the "fractal dimension" of a language from the neural scaling laws of LLMs?).

In-context learning and induction heads. Induction heads are computational circuits that naturally emerge in simple transformers through training, and they have been hypothesized to underlie much of the in-context learning ability of foundation models (Elhage et al., 2021; Olsson et al., 2022;

Von Oswald et al., 2023; Reddy, 2023). In its simplest form, an induction head copies repeating tokens in the context to make predictions. For example, when presented with a token stream $[A][B]\ldots[A]$, an induction head will output $[B]$ as the next token. Prior works train transformers on minimal Markov chain grammars, and find that, during pretraining, models learn to identify increasingly higher-order k-grams, with different attention heads specializing in copying, lookup, and aggregation (Edelman et al., 2024; Chen et al., 2024a). These works imply that pretraining enables models to learn conditional distributions, allowing them to represent sequence distributions seen in the context (Lv et al., 2024; Chen et al., 2024b; Keskar et al., 2019; Zekri et al., 2024).

There is a clear parallel between context parroting and induction heads: both are essentially copy-and-paste operations, with context parroting involving the matching of not just one but multiple contiguous tokens. In fact, it is easy to imagine context parroting emerging naturally from combining multiple induction heads. This parallel can potentially explain the unreasonable effectiveness of applying language models trained on text to time-series tasks without fine-tuning or prompt engineering (Garza & Mergenthaler-Canseco, 2023; Jin et al., 2023; Zhou et al., 2023; Gruver et al., 2024; Liu et al., 2024a). The induction heads formed from training on natural language happen to be also effective for predicting time series and can be easily repurposed to implement strategies such as context parroting.

3 Context parroting as a zero-shot forecasting strategy

Overview of context parroting. In this section we motivate and introduce our baseline: context parroting. It was inspired by recent observations that Chronos often predicts chaotic systems by copying directly from the context (Zhang & Gilpin, 2024). An example of Chronos using parroting to forecast a partially-observed Lorenz system is shown in Fig. 1.

On a high level, context parroting uses the last D tokens of the context to query the remaining context. For whatever context sequence that most closely matches the query, the subsequent tokens in the context are copied and used as the forecast. Because the length of the motif D can be seen as the number of delayed states in a delay embedding from the lens of Takens' embedding theorem (Takens, 2006; Huke, 2006), we also refer to D as the embedding dimension and will use the terms embedding dimension and query length interchangeably. Interpreting D as the embedding dimension is convenient because context parroting can be seen as a nearest neighbor algorithm in the D-dimensional delay-embedded space. During the matching process, we exclude the last D motifs to avoid parroting too close to where the prediction starts. Framed in terms of induction heads, the query lookup acts as a *copy* head, the nearest-neighbor match is a *selector*, and the exact repetition is the *aggregation* operation (Chen et al., 2024a). We provide a pseudocode for context parroting in Algorithm 1.

Algorithm 1 Context Parroting

Input: Context trajectory $x_{1:L} = \{x_1, \ldots, x_L\}$, embedding dimension D (i.e., the length of the motif to match), and forecast length H.
Output: Forecast trajectory $x_{L+1:L+H} = \{x_{L+1}, \ldots, x_{L+H}\}$.

```
for all length-$D$ motif $s$: $x_{s-D+1:s}$ in the context $x_{1:L-D}$ do
    compute the Euclidean distance $d_s$ between motif $s$ and the last motif $x_{L-D+1:L}$
Find the best-matching motif, $s_{opt}$, with the smallest Euclidean distance
Set the first $L - s_{opt}$ predicted points to be $x_{L+1:2L-s_{opt}} = x_{s_{opt}+1:L}$ and repeat until the forecast length $H$ is reached
```

Relationship to classical nonlinear forecasting methods. We show in Appendix H.3 that, in various limits, context parroting is equivalent to two classical algorithms from nonlinear dynamics: the *simplex projection* technique and the *S-map* algorithm (Sugihara & May, 1990; Sugihara, 1994). Both approaches have their foundations in Takens' embedding theorem, which states that time-delayed low-dimensional observables derived from a nonlinear dynamical system can recover key geometric properties of the underlying high-dimensional attractor (Takens, 2006; Huke, 2006). However, unlike context parroting, which looks for the best matching motif, simplex projection tries to identify multiple matching motifs in the context and computes a weighted average as its forecast. This can potentially make simplex projection more sensitive to the choice of the embedding dimension

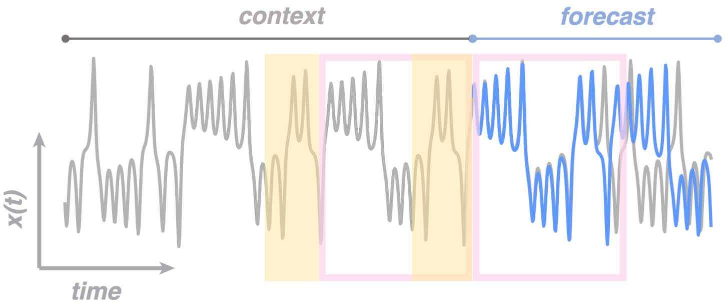

Figure 1: **Example of a foundation model forecasting chaotic dynamics with context parroting**. Here, the foundation model (Chronos) was asked to predict the x variable of the Lorenz system based on a short context trajectory with 512 data points. Blue is the prediction and gray is the ground truth. Chronos produced an accurate prediction by simply looking for a motif in the context similar to the motif immediately preceding the prediction (highlighted in yellow) and copying the evolution following the matching motif (highlighted by pink boxes). We distill this context parroting strategy into Algorithm 1 and compare it against time-series foundation models (including Chronos itself).

D, limiting the method to small embedding dimensions in practice (Chang et al., 2017). Other than simplex projection and S-map, there also exists other zero-shot forecasting strategies from nonlinear dynamics. For example, the Farmer-Sidorowich method (Farmer & Sidorowich, 1987) looks at multiple nearest neighbors in the context and builds a local linear model to make forecasts. An interesting future direction is to compare these methods from nonlinear dynamics with context parroting and time-series foundation models, which might inspire new zero-shot forecasting strategies.

4 Methods

Datasets. The `dysts` dataset provides a standardized benchmark of 135 low-dimensional chaotic systems, each defined by a set of ordinary differential equations between dimensionality three and six (Gilpin, 2021). The chaotic systems are drawn from different published papers and span fields such as neuroscience, climate science, fluid dynamics, and astrophysics. Every system is annotated with its largest Lyapunov exponent λ, an invariant characteristic of the underlying dynamics that quantifies the rate at which small perturbations grow over time. In chaotic systems, even minor errors rapidly compound over a characteristic timescale known as the Lyapunov time, defined as $\tau \equiv \lambda^{-1}$. To normalize the difficulty of predicting different chaotic systems (so results from the 135 systems can be meaningfully compared and combined), we generate trajectory data with a fixed sampling rate of 30 points per Lyapunov time, and also measure the forecast performance in terms of Lyapunov times. To show the relevance of our findings to a broad class of SciML tasks, later we also go beyond low-dimensional chaotic systems and simulated data by benchmarking on real-world datasets from ECG measurements and electronic circuits.

Models. For time-series foundation models, we select Chronos$_{\text{base}}$ (200M parameters), its variant Chronos-Bolt$_{\text{base}}$ (205M parameters), Time-MoE$_{\text{large}}$ (200M parameters), TimesFM-2.0 (500M parameters) and Moirai-2.0$_{\text{small}}$ (11M parameters) (Das et al., 2024; Ansari et al., 2024; 2025; Shi et al., 2025; Liu et al., 2025a). All of these models are pretrained on massive amounts of real-world time series data (hundreds of billions of data points), which are often complemented by synthetic data to improve generalization. We also consider DynaMix, a foundation model pretrained on chaotic dynamical systems (Hemmer & Durstewitz, 2025). These models encompass a wide array of design choices: Time-MoE, TimesFM-2.0, and Moirai-2.0 are decoder-only architectures, Chronos is an encoder-decoder architecture, and DynaMix is an almost-linear RNN trained via teacher forcing (Brenner et al., 2024a). Chronos and Time-MoE use pointwise tokenization, DynaMix implicitly tokenizes pointwise, while TimesFM-2.0 and Moirai-2.0 use patching. Time-MoE, DynaMix, and TimesFM-2.0 by default give point forecasts, whereas Chronos, Chronos-Bolt, and Moirai-2.0 provide probabilistic forecasts with uncertainty quantification. For these models, we use the median prediction when evaluating forecast errors. An important parameter for all foundation models is the maximum context length $L_{\max}$, which varies from 512 data points (Chronos), 1680 (Moirai-2.0), 2048 (TimesFM-2.0), 4096 (Time-MoE), to arbitrary for DynaMix due to its recurrent formulation.

Pipelines. To evaluate different models' ability to zero-shot forecast dynamical systems, we generate a chaotic trajectory of length 10^5 for each of the 135 chaotic systems in `dysts`, with a granularity of 30 data points per Lyapunov time. Each trajectory is normalized to have zero mean and unit standard deviation. For a given context length L, we randomly pick a length-L segment from the chaotic trajectory and provide it to the model as the context. The model's task is to predict the next 300 data points (equivalent to 10 Lyapunov times) solely based on the context. We ask the model to make a univariate forecast on each dimension independently, which is then evaluated separately for each dimension. To obtain reliable statistics, we aggregate the results over all 135 chaotic systems, all dimensions, and 20 random initial conditions for each system.

Metrics. In line with previous research (Hyndman & Koehler, 2006; Makridakis et al., 2022; Gilpin, 2021; 2023), we assess forecasting performance using a diverse set of complementary metrics.

Symmetric Mean Absolute Percentage Error (sMAPE).

$$\text{sMAPE}(\mathbf{x}, \hat{\mathbf{x}}) \equiv 2\frac{100}{T}\sum_{t=1}^{T}\frac{|\mathbf{x}_t - \hat{\mathbf{x}}_t|}{|\mathbf{x}_t| + |\hat{\mathbf{x}}_t|},$$

where the sequence $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T$ denotes the ground truth, and $\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \ldots, \hat{\mathbf{x}}_T$ are the corresponding predictions made by the model. To provide some context to help interpret the sMAPE value, we note that predicting the mean of white noise would give you an sMAPE around 200.

Valid Prediction Time (VPT). This metric identifies the latest time step t_f before which the sMAPE remains below a predefined threshold ϵ, as described in Vlachas et al. (2020). Formally:

$$\text{VPT} \equiv \text{argmax}_{t_f}\{t_f | \text{sMAPE}(\mathbf{x}_t, \hat{\mathbf{x}}_t) < \epsilon,\ \forall t < t_f\}.$$

We use $\epsilon = 30$, consistent with prior work (Vlachas et al., 2020; Gilpin, 2023; Zhang & Gilpin, 2024). For chaotic systems, consistently achieving a VPT over one Lyapunov time is considered impressive (Gilpin, 2023; Zhang & Gilpin, 2024).

Other than sMAPE and VPT, we also show benchmark results using *Mean Square Error* (MSE) and *Mean Absolute Error* (MAE), two other metrics commonly used in the time series literature.

For chaotic dynamical systems, point forecasts will inevitably fail due to the exponential rate of error accumulation. It is thus equally important for a forecasting model to preserve the long-term statistical and geometric properties of the chaotic attractors, such as Lyapunov exponents and the attractor dimension. Here, we compare the structure of true and predicted attractors by calculating the KL Divergence between their distributions.

Kullback–Leibler Divergence between Attractors (D_{stsp}).

$$D_{\text{stsp}} \equiv D_{\text{KL}}(P \| Q) = \sum_{x \in \mathcal{X}} P(x) \log \frac{P(x)}{Q(x)},$$

where P and Q represent the true and the predicted attractor, respectively. When estimating D_{stsp}, we follow the methodology in Hess et al. (2023); Göring et al. (2024). Specifically, we place Gaussian kernels at each point in the true and predicted trajectories and estimate the KL divergence between these Gaussian mixtures using a sampling-based approximation (Hershey & Olsen, 2007).

In the appendix, we also measure attractor reconstruction accuracy using *Fractal Dimension* and *Lyapunov Exponents*. The *correlation dimension* estimates the fractal dimension from a time series by calculating the scaling of the number of attractor points that fall within a given radius of each point (Grassberger & Procaccia, 1983). This quantity is among the few invariant quantities of a dynamical system that can be non-parametrically estimated from a time series; however, the estimates can still be unstable and brittle depending on the fitting procedure (Clauset et al., 2009).

5 Results

5.1 Context parroting versus foundation models

Here, we compare context parroting and foundation models in their ability to predict chaotic dynamics. Figure 2 shows each model's forecasting error (measured by sMAPE) as well as their accuracy in attractor reconstruction (measured by KL Divergence). It is clear that context parroting is better than all foundation models tested here in both metrics. In Fig. 6, we show that this remains true when benchmarked against MSE and MAE. The results for fractal dimension accuracy are shown in Fig. 7, where parroting and Chronos significantly outperform the rest of the foundation models. We also explore the effects of observational noise and data granularity on the forecast performance in Tables 5 to 14. Over the wide range of noise intensity and data granularity we tested, parroting is always the best or the second best according to all metrics (VPT, MAE, MSE, fractal dimension accuracy, and attractor KL Divergence).

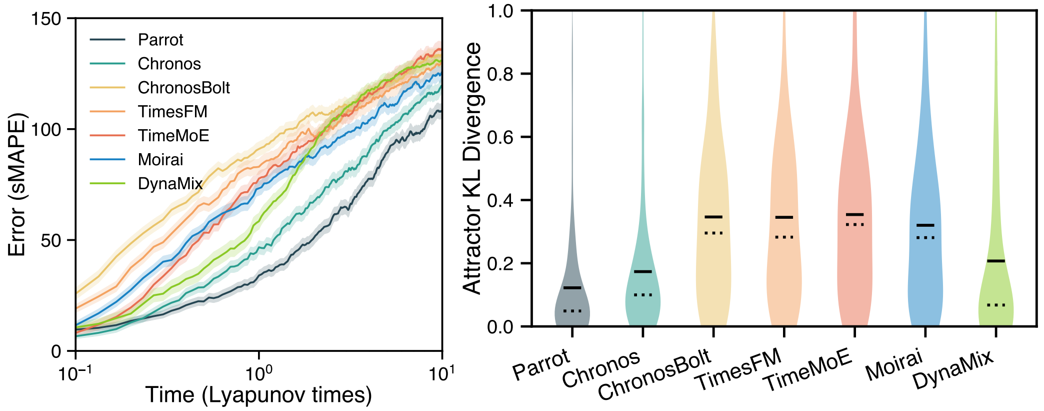

Figure 2: **Context parroting outperforms foundation models in zero-shot forecasting for both short-term point-wise accuracy and long-term attractor reconstruction**. Left: Forecast error of each model as a function of the forecast horizon. The context length is set to 512 for all models. Right: KL Divergence between the predicted attractors and the true attractors (smaller is better). Solid lines represent mean and dotted lines represent median. All results are obtained from 135 chaotic systems in the `dysts` database, with 20 trajectories from random initial conditions for each system.

Among foundation models, Chronos is the best performer in predicting chaotic systems, which is not surprising given that it utilizes parroting as a main forecasting strategy (Zhang & Gilpin, 2024). Chronos's tendency to context parrot arises from its distinct architecture as a language model that implicitly quantizes time series. As a result, Chronos is trained using cross-entropy loss, which incentivizes preservation of k-gram frequencies and encourages the generation of diverse forecast samples consistent with the dynamical system's underlying measure (Yu et al., 2025). In contrast, TimeMoE and TimesFM are trained using mean squared error loss. As a result, these models lose diversity and forecast the mean at long forecast horizons (i.e., they tend to underestimate the oscillations). Some representative forecasts from the foundation models are shown in Fig. 5, which shows that regressing to the mean is a common failure mode for many foundation models on chaotic systems.

Moreover, as we demonstrate in Fig. 8, the inference cost of context parroting is negligible compared to foundation models (not to mention the substantial GPU time needed to pre-train them). For example, there is an over six orders of magnitude computational gap between Chronos and context parroting for all context lengths. Combined with the fact that the performance of parroting is not sensitive to the choice of the embedding dimension D (Fig. 10), these results establish context parroting as a simple but effective baseline for zero-shot forecasting of dynamical systems.

Figure 3 further explores the effect of context length on forecast accuracy. We find that longer context windows generally lead to better performance for both context parroting and Chronos. However, the longest context length Chronos can effectively utilize is 512 data points. This limit is determined by Chronos's maximum context window chosen at pre-training. To be able to utilize longer context,

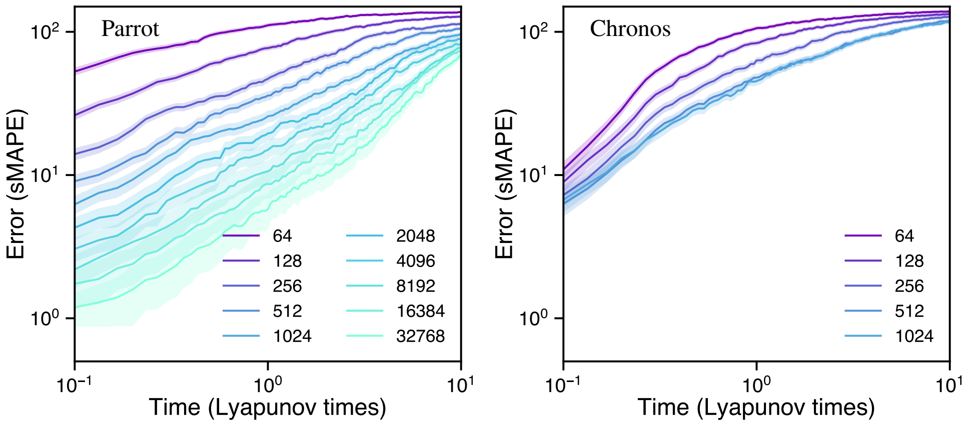

Figure 3: **Parroting can better utilize longer context data while Chronos does better for shorter contexts**. Each line represents the forecast error for a different context length. The performance of Chronos saturates once the context length exceeds its designed upper limit of 512 data points, whereas the accuracy of context parroting keeps improving for longer context windows. Here we set the embedding dimension $D = 10$ for the parroting algorithm.

Chronos must be retrained from scratch with much more data and compute. Context parroting, on the other hand, is happy to utilize context data of any length. In Fig. 9, we show VPT as a function of the context length for context parroting and the foundation models. Again, parroting is the clear winner for medium to long context lengths, easily reaching an average VPT of over 5 Lyapunov times when given sufficient context.

Interestingly, Chronos outperforms context parroting on short contexts, which points to additional zero-shot learning strategies beyond parroting employed by Chronos. This is perhaps not surprising given that at short context length, the time series becomes effectively nonstationary, which is the strength of time-series foundation models. For example, Chronos is great at continuing the local trend in the context, which can be a more effective strategy than parroting when the length of the context is limited. Moreover, even when restricted to parroting, the $\sim \mathcal{O}(L^2)$ operations performed by attention heads in transformers like Chronos have, in principle, sufficient computational complexity to dynamically choose the optimal embedding dimension D for each individual time series, giving attention an advantage over parroting algorithms with a fixed D, which have the $\sim \mathcal{O}(D\,L)$ complexity of nearest-neighbor search. It would be interesting to explicitly identify the mechanisms that enable Chronos to outperform parroting in the short-context regime.

5.2 In-context neural scaling law

Liu et al. (2024a) reported an in-context neural scaling law for LLMs applied to dynamical systems, in which the one-step forecast error decreases algebraically with context length. However, it is unclear where this scaling law came from or why LLMs trained on text can be effective for time series without fine tuning. Here, we show that context parroting naturally gives rise to the same in-context scaling law and provides geometric insights into its origin. Given the similarity between parroting and the induction heads implemented by LLMs (Olsson et al., 2022), the geometric explanation we develop next for context parroting can conceivably be applied to LLMs and partially explain the observations in Liu et al. (2024a).

The left panel in Fig. 4 shows the power law relation between one-step forecast error (measured by sMAPE) and context length for the parroting method. Longer context lengths improve predictions because more context data allows the algorithm to find better matching motifs, and a closely-matched motif allows the parroted sequence to shadow the ground truth for longer. The overlap between the matching motifs can be measured by their Euclidean distance. For length-D motifs, this is equivalent to embedding the context trajectory in a D-dimensional delay-embedded space (i.e., mapping x_s to $x_{s-D+1:s}$) and finding the distance between the embedded last context point $x_{L-D+1:L}$ and its nearest embedded neighbor. The right panel in Fig. 4 shows the improving overlap explicitly, with the

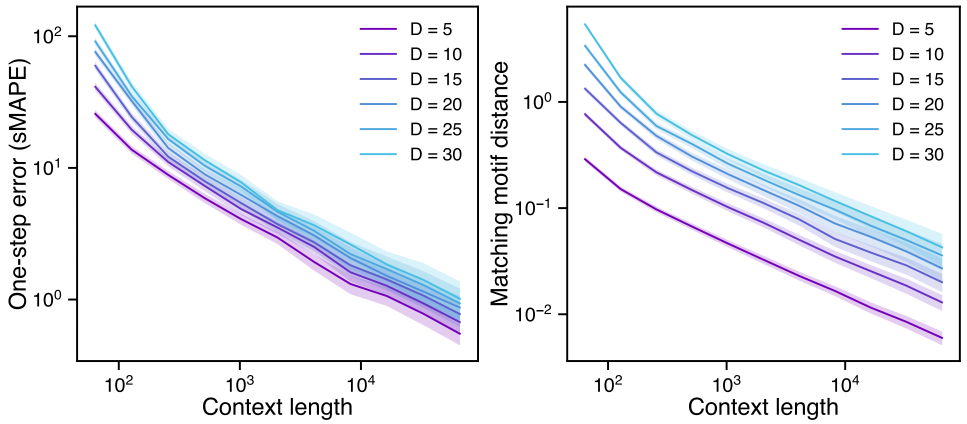

Figure 4: **Scaling laws with context length.** Left: One-step forecast error versus context length. The scaling follows a power law for all embedding dimensions D considered. Smaller D is more accurate here because of the one-step forecast error. Larger D can be more accurate for longer forecasting horizons. Right: Euclidean distance between the last context motif $x_{L-D+1:L}$ and its closest match, as a function of context length. Again, the scaling follows a power law for all D. The forecast accuracy is directly tied to the motif distance: Smaller distances translate into better predictions. As the number of context data points is increased, it becomes more and more likely that a context motif will land in the vicinity of the last context motif, and the fractal dimension of the chaotic attractor determines the rate of approach. In principle, infinite context length should give context parroting infinite accuracy for any deterministic system with a well-defined attractor.

distance between the matching motifs decreasing algebraically with context length. It is easy to see that matching motif distance should map linearly to the expected one-step forecast error, so the power law for the matching motif distance implies the power law for one-step forecast error, as confirmed by Fig. 4. Although we focus on deterministic ODEs in Fig. 4, we note that the same power law scaling is expected to hold for discrete maps and for systems subject to weak noise. Explaining the power law in the case of stochastic systems, such as Markov chains with randomly generated transition matrices (Liu et al., 2024a), is a promising direction for future research.

Why then does the matching motif distance follow a power law with context length, $\ell \propto L^{-\alpha}$? We can link the scaling coefficient α to the fractal dimension of the chaotic attractor. The fractal dimension of an attractor, as estimated by the correlation dimension d_{cor}, is defined as

$$d_{\text{cor}} \equiv \lim_{\epsilon,\epsilon' \to 0^+} \frac{\ln\left[\frac{C(\epsilon)}{C(\epsilon')}\right]}{\ln\left(\frac{\epsilon}{\epsilon'}\right)},$$

where $C(\epsilon)$ is the number of point pairs in the attractor that are within a given radius ϵ from each other. In other words, if we plot $C(\epsilon)$ against ϵ on a log-log plot, d_{cor} would be the slope of the plotted line. Due to the ergodic property of chaotic attractors, the context trajectory can be seen as a random sample of the attractor. Longer context trajectory contains more samples, and the expected distance between two context points in a delay-embedded space decreases with context length as $L^{-1/d_{\text{cor}}}$. For example, for a two-dimensional attractor, the distance between two random points on the attractor will decrease as $1/\sqrt{L}$. Fractal dimension thus measures the speed at which the minimum embedding distance between points on an attractor can be reduced by including more samples, and higher dimensionality requires more points to reduce the distance to the same extent. Mathematically, we thus expect $\alpha = 1/d_{\text{cor}}$. A similar scaling law has been derived for the Farmer-Sidorowich forecasting method from the nonlinear dynamics community (Farmer & Sidorowich, 1987). Despite the theoretical correspondence, however, numerically the fractal dimension is challenging to accurately estimate from finite time series, due to instabilities in estimating the scaling coefficient α as the data resolution and fitting conditions vary (Clauset et al., 2009; Grassberger & Procaccia, 1983). Nonetheless, in Fig. 11, we observe relatively strong correlation between d_{cor} and $1/\alpha$, supporting our theoretical argument above.

5.3 SciML tasks beyond low-dimensional chaotic systems

So far we focused on low-dimensional chaotic systems from the `dysts` dataset, which enabled systematic comparison between different forecasting models with standardized benchmarks. Here, we show that parroting also outperforms foundation models on a broader class of SciML tasks, including real-world datasets of current scientific interest. Our datasets are: (1) the von Karman vortex street at Reynolds number $\mathrm{Re} = 900$, a standard problem in fluid dynamics representing a flow exhibiting intermittency. We generated time series corresponding to the top PCA modes, in order to capture global structure; (2) electrocardiogram recordings (via the QT Database in PhysioNet); (3) 28 coupled electronic circuits measured experimentally from Vera-Ávila et al. (2020)); and (4) 23 Kuramoto oscillators coupled through frustrated and nonreciprocal interactions, recently studied in León & Pazó (2025). These are all high-dimensional systems, two generated from simulations and two measured in the real world. For the metrics, we use MAE and MSE to measure pointwise forecast accuracy, and KL Divergence to measure the accuracy in attractor reconstruction. The results are summarized below. Parroting is the only model that ranks in the top three for all tasks and all metrics. Other metrics, such as valid prediction time and fractal dimension accuracy, give similar rankings.

Table 1: Performance comparison (**MAE @ 50 steps**, mean ± standard deviation) of forecasting models across SciML tasks. **Bold = best**, *italic = second and third best*.

Task	Parrot	DynaMix	Chronos	Chronos Bolt	TimesFM	TimeMoE	Moirai
Turbulence	*0.403±0.210*	0.505±0.247	0.431±0.237	0.567±0.247	0.510±0.174	*0.394±0.172*	**0.382±0.189**
ECG	**0.624±0.315**	0.777±0.241	0.873±0.422	0.752±0.279	*0.723±0.259*	0.799±0.158	*0.684±0.237*
Circuit	**0.083±0.050**	0.425±0.172	*0.111±0.065*	0.349±0.120	*0.196±0.090*	0.206±0.102	0.213±0.093
Kuramoto	**0.004±0.001**	0.076±0.002	0.072±0.029	0.961±0.084	0.624±0.061	*0.070±0.011*	**0.004±0.001**

Table 2: Performance comparison (**MSE @ 50 steps**) of forecasting models across SciML tasks. **Bold = best**, *italic = second and third best*.

Task	Parrot	DynaMix	Chronos	Chronos Bolt	TimesFM	TimeMoE	Moirai
Turbulence	*0.322±0.333*	0.490±0.4530	0.380±0.408	0.531±0.447	0.403±0.262	**0.278±0.268**	**0.278±0.267**
ECG	*0.916±0.630*	1.063±0.488	1.461±1.097	0.950±0.581	0.940±0.530	*0.893±0.287*	**0.851±0.488**
Circuit	**0.012±0.016**	0.297±0.294	*0.024±0.030*	0.181±0.122	*0.065±0.056*	0.076±0.080	0.075±0.060
Kuramoto	**0.001±0.002**	0.006±0.001	0.009±0.007	1.296±0.188	0.512±0.096	*0.008±0.002*	**0.001±0.001**

Table 3: Performance comparison (**KL Divergence between predicted and true attractors**) of forecasting models across SciML tasks. **Bold = best**, *italic = second and third best*.

Task	Parrot	DynaMix	Chronos	Chronos Bolt	TimesFM	TimeMoE	Moirai
Turbulence	*0.028±0.044*	**0.005±0.008**	0.041±0.046	0.048±0.058	0.111±0.072	0.070±0.058	*0.030±0.041*
ECG	**0.065±0.089**	*0.099±0.104*	0.403±0.367	0.253±0.185	0.220±0.153	*0.188±0.094*	0.276±0.311
Circuit	*0.572±0.082*	2.940±0.528	*0.630±0.118*	1.710±0.255	**0.383±0.087**	0.816±0.200	0.848±0.155
Kuramoto	**0.001±0.001**	1.010±0.150	0.537±0.087	3.116±0.202	4.489±0.363	*0.076±0.040*	*0.010±0.011*

6 Conclusion and future directions

We find that a simple forecast strategy—context parroting—outperforms leading foundation models on dynamical systems forecasting, a critical task in scientific machine learning. This surprising finding exposes a limitation of current time-series foundation models as general-purpose time-series forecasters and highlights the need to further scale them or to fine-tune them for specific domains. It also suggests that accurately measuring the performance of foundation models can be difficult for scientific machine learning tasks, because strategies like parroting can effectively game both short- and long-term accuracy metrics.

Finding a simple but effective baseline for a challenging task can encourage rethinking of the status quo, motivating the development of better model architectures (Arora et al., 2017). For example, context parroting formalizes an explicit baseline to compare against in the time-series domain and can help discover beyond-parroting strategies. Identifying in-context learning strategies beyond parroting can spur the development of next-generation foundation models and contribute to the debate on whether (or to what extent) large language models are stochastic parrots (Bender et al., 2021; Mitchell & Krakauer, 2023; Arora & Goyal, 2023; McCoy et al., 2024).

An interesting future direction is to generalize context parroting to deal with non-stationary time series while keeping the simplicity and efficiency of the method. Context parroting assumes the existence of a stationary underlying measure; for an ergodic deterministic system this implies that conditional probabilities of timepoints are stationary up to any order (Appendix H). However, newer foundation models readily handle simple nonstationarity like baseline drift, implying that a modified parroting strategy may be possible in-context (Das et al., 2024). A promising avenue involves combining parroting with a probabilistic model, such as a Gaussian Process, to account for nonstationary trends. Additionally, the diversity of long-term forecasts can be improved by allowing stochastic selection among multiple candidate forecasts (e.g., from different matching motifs). Once generalized, the non-stationary parroting method can replace Naive and Seasonal Naive to serve as a more informative baseline for the zero-shot forecasting of general time series (weather, traffic, finance, etc.).

Finally, we want to emphasize that we are not proposing to replace time-series foundation models with context parroting. Instead, the value of parroting is as a simple baseline that can reveal the gaps in current foundation models and guide the design of new ones. When foundation models under perform relative to context parroting, it reveals that they haven't learned to fully utilize the context data. For example, a common failure mode we observed across a range of leading foundation models (TimesFM, TimeMoE, Chronos Bolt) is that they tend to underestimate oscillations in the dynamics and the predictions often quickly converge to the mean (Fig. 5). Being aware of context parroting as a baseline also guides the design of more informative benchmarks in the future. For example, for dynamical systems in `dysts`, one can generate test context data that are impossible to parrot by using only initial conditions that lead to trajectories without nearly repeating motifs.

7 Reproducibility Statement

A Python implementation of the context parroting algorithm and the benchmarks are available at `https://anonymous.4open.science/r/parroting-4D26`.

References

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *arXiv:2403.07815*, 2024.

Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, et al. Chronos-2: From univariate to universal forecasting. *arXiv preprint arXiv:2510.15821*, 2025.

Sanjeev Arora and Anirudh Goyal. A theory for emergence of complex skills in language models. *arXiv:2307.15936*, 2023.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. A simple but tough-to-beat baseline for sentence embeddings. In *International conference on learning representations*, 2017.

Kamyar Azizzadenesheli, Nikola Kovachki, Zongyi Li, Miguel Liu-Schiaffini, Jean Kossaifi, and Anima Anandkumar. Neural operators for accelerating scientific simulations and design. *Nat. Rev. Phys.*, pp. 1–9, 2024.

Yasaman Bahri, Ethan Dyer, Jared Kaplan, Jaehoon Lee, and Utkarsh Sharma. Explaining neural scaling laws. *Proc. Natl. Acad. Sci. U.S.A.*, 121(27):e2311878121, 2024.

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pp. 610–623, 2021.

Tyrus Berry and Suddhasattwa Das. Limits of learning dynamical systems. *SIAM Review*, 67(1): 107–137, 2025.

Cristian Bodnar, Wessel P Bruinsma, Ana Lucic, Megan Stanley, Johannes Brandstetter, Patrick Garvan, Maik Riechert, Jonathan Weyn, Haiyu Dong, Anna Vaughan, et al. Aurora: A foundation model of the atmosphere. *arXiv:2405.13063*, 2024.

Richard C Bradley. Basic properties of strong mixing conditions. a survey and some open questions. *Probability Surveys*, 2:107–144, 2005.

Manuel Brenner, Florian Hess, Jonas M Mikhaeil, Leonard F Bereska, Zahra Monfared, Po-Chen Kuo, and Daniel Durstewitz. Tractable dendritic rnns for reconstructing nonlinear dynamical systems. In *International conference on machine learning*, pp. 2292–2320. Pmlr, 2022.

Manuel Brenner, Christoph Jürgen Hemmer, Zahra Monfared, and Daniel Durstewitz. Almost-linear rnns yield highly interpretable symbolic codes in dynamical systems reconstruction. *Advances in Neural Information Processing Systems*, 37:36829–36868, 2024a.

Manuel Brenner, Elias Weber, Georgia Koppe, and Daniel Durstewitz. Learning interpretable hierarchical dynamical systems models from time series data. *arXiv preprint arXiv:2410.04814*, 2024b.

Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc. Natl. Acad. Sci. U.S.A.*, 113(15): 3932–3937, 2016.

Steven L Brunton, Marko Budisić, Eurika Kaiser, and J Nathan Kutz. Modern Koopman theory for dynamical systems. *SIAM Rev.*, 64(2):229–340, 2022.

Josue Ortega Caro, Antonio H de O Fonseca, Christopher Averill, Syed A Rizvi, Matteo Rosati, James L Cross, Prateek Mittal, Emanuele Zappala, Daniel Levine, Rahul M Dhodapkar, et al. Brainlm: A foundation model for brain activity recordings. *bioRxiv*, pp. 2023–09, 2023.

Chun-Wei Chang, Masayuki Ushio, and Chih-hao Hsieh. Empirical dynamic modeling for beginners. *Ecological research*, 32:785–796, 2017.

Renyi Chen and Molei Tao. Data-driven prediction of general hamiltonian dynamics via learning exactly-symplectic maps. In *International Conference on Machine Learning*, pp. 1717–1727. PMLR, 2021.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *NeurIPS*, 31, 2018.

Siyu Chen, Heejune Sheen, Tianhao Wang, and Zhuoran Yang. Unveiling induction heads: Provable training dynamics and feature learning in transformers. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024a.

Yanda Chen, Chen Zhao, Zhou Yu, Kathleen McKeown, and He He. Parallel structures in pre-training data yield in-context learning. In *62nd Annual Meeting of the Association for Computational Linguistics, ACL 2024*, pp. 8582–8592. Association for Computational Linguistics (ACL), 2024b.

Xiaoyuan Cheng, Yi He, Yiming Yang, Xiao Xue, Sibo Chen, Daniel Giles, Xiaohang Tang, and Yukun Hu. Learning choas in a linear way. *arXiv preprint arXiv:2503.14702*, 2025.

Aaron Clauset, Cosma Rohilla Shalizi, and Mark EJ Newman. Power-law distributions in empirical data. *SIAM review*, 51(4):661–703, 2009.

Wenhui Cui, Woojae Jeong, Philipp Thölke, Takfarinas Medani, Karim Jerbi, Anand A Joshi, and Richard M Leahy. Neuro-gpt: Towards a foundation model for eeg. In *2024 IEEE International Symposium on Biomedical Imaging (ISBI)*, pp. 1–5. IEEE, 2024.

Abhimanyu Das, Weihao Kong, Andrew Leach, Shaan Mathur, Rajat Sen, and Rose Yu. Long-term forecasting with tide: Time-series dense encoder. *arXiv:2304.08424*, 2023.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *Forty-first International Conference on Machine Learning*, 2024.

Samuel Dooley, Gurnoor Singh Khurana, Chirag Mohapatra, Siddartha V Naidu, and Colin White. Forecastpfn: Synthetically-trained zero-shot forecasting. *Advances in Neural Information Processing Systems*, 36, 2024.

Ezra Edelman, Nikolaos Tsilivis, Benjamin Edelman, Eran Malach, and Surbhi Goel. The evolution of statistical induction heads: In-context learning markov chains. *Advances in Neural Information Processing Systems*, 37:64273–64311, 2024.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 1(1):12, 2021.

Jianqing Fan and Qiwei Yao. *Nonlinear time series: nonparametric and parametric methods*. Springer Science & Business Media, 2008.

J Doyne Farmer and John J Sidorowich. Predicting chaotic time series. *Physical review letters*, 59(8): 845, 1987.

Shivam Garg, Dimitris Tsipras, Percy S Liang, and Gregory Valiant. What can transformers learn in-context? a case study of simple function classes. *Advances in neural information processing systems*, 35:30583–30598, 2022.

Azul Garza and Max Mergenthaler-Canseco. Timegpt-1. *arXiv:2310.03589*, 2023.

Daniel J Gauthier, Erik Bollt, Aaron Griffith, and Wendson AS Barbosa. Next generation reservoir computing. *Nat. Commun.*, 12:5564, 2021.

William Gilpin. Chaos as an interpretable benchmark for forecasting and data-driven modelling. *NeurIPS*, 34, 2021.

William Gilpin. Model scale versus domain knowledge in statistical forecasting of chaotic systems. *Phys. Rev. Research*, 5(4):043252, 2023.

Niclas Göring, Florian Hess, Manuel Brenner, Zahra Monfared, and Daniel Durstewitz. Out-of-domain generalization in dynamical systems reconstruction. *arXiv:2402.18377*, 2024.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. Moment: A family of open time-series foundation models. *arXiv:2402.03885*, 2024.

Peter Grassberger and Itamar Procaccia. Measuring the strangeness of strange attractors. *Physica D: nonlinear phenomena*, 9(1-2):189–208, 1983.

Lyudmila Grigoryeva, Hannah Lim Jing Ting, and Juan-Pablo Ortega. Infinite-dimensional next-generation reservoir computing. *Physical Review E*, 111(3):035305, 2025.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G Wilson. Large language models are zero-shot time series forecasters. *Advances in Neural Information Processing Systems*, 36, 2024.

Yi He, Yiming Yang, Xiaoyuan Cheng, Hai Wang, Xiao Xue, Boli Chen, and Yukun Hu. Chaos meets attention: Transformers for large-scale dynamical prediction. *arXiv preprint arXiv:2504.20858*, 2025.

Christoph Jürgen Hemmer and Daniel Durstewitz. True zero-shot inference of dynamical systems preserving long-term statistics. *Advances in Neural Information Processing Systems*, 38, 2025.

Maximilian Herde, Bogdan Raonić, Tobias Rohner, Roger Käppeli, Roberto Molinaro, Emmanuel de Bézenac, and Siddhartha Mishra. Poseidon: Efficient foundation models for pdes. *arXiv:2405.19101*, 2024.

John R Hershey and Peder A Olsen. Approximating the kullback leibler divergence between gaussian mixture models. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, volume 4, pp. IV–317. IEEE, 2007.

Florian Hess, Zahra Monfared, Manuel Brenner, and Daniel Durstewitz. Generalized teacher forcing for learning chaotic dynamics. In *International Conference on Machine Learning*, pp. 13017–13049. PMLR, 2023.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv:2203.15556*, 2022.

Jeremy P Huke. Embedding nonlinear dynamical systems: A guide to takens' theorem. 2006.

Rob J Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

Rob J Hyndman and Anne B Koehler. Another look at measures of forecast accuracy. *International journal of forecasting*, 22(4):679–688, 2006.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. Time-LLM: Time series forecasting by reprogramming large language models. *arXiv:2310.01728*, 2023.

Ian D Jordan, Piotr Aleksander Sokół, and Il Memming Park. Gated recurrent units viewed through the lens of continuous time dynamical systems. *Frontiers in computational neuroscience*, 15: 678158, 2021.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv:2001.08361*, 2020.

George Em Karniadakis, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nat. Rev. Phys.*, 3(6):422–440, 2021.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*, 2019.

Aditi S Krishnapriyan, Alejandro F Queiruga, N Benjamin Erichson, and Michael W Mahoney. Learning continuous models for continuous physics. *Communications Physics*, 6(1):319, 2023.

Iván León and Diego Pazó. Dynamics and chaotic properties of the fully disordered kuramoto model. *Chaos*, 35(7), 2025.

Matthew Levine and Andrew Stuart. A framework for machine learning of model error in dynamical systems. *Commun. Am. Math. Soc.*, 2(07):283–344, 2022.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv:2010.08895*, 2020.

Yuxuan Liang, Haomin Wen, Yuqi Nie, Yushan Jiang, Ming Jin, Dongjin Song, Shirui Pan, and Qingsong Wen. Foundation models for time series analysis: A tutorial and survey. *arXiv:2403.14735*, 2024.

Bryan Lim and Stefan Zohren. Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, 379(2194):20200209, 2021.

Chenghao Liu, Taha Aksu, Juncheng Liu, Xu Liu, Hanshu Yan, Quang Pham, Doyen Sahoo, Caiming Xiong, Silvio Savarese, and Junnan Li. Moirai 2.0: When less is more for time series forecasting. *arXiv:2511.11698*, 2025a.

Toni JB Liu, Nicolas Boullé, Raphaël Sarfati, and Christopher J Earls. Llms learn governing principles of dynamical systems, revealing an in-context neural scaling law. *arXiv:2402.00795*, 2024a.

Yong Liu, Guo Qin, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Autotimes: Autoregressive time series forecasters via large language models. *arXiv:2402.02370*, 2024b.

Yong Liu, Guo Qin, Zhiyuan Shi, Zhi Chen, Caiyin Yang, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Sundial: A family of highly capable time series foundation models. *arXiv:2502.00816*, 2025b.

Ang Lv, Ruobing Xie, Xingwu Sun, Zhanhui Kang, and Rui Yan. Language models" grok" to copy. *arXiv preprint arXiv:2409.09281*, 2024.

Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. The m5 competition: Background, organization, and implementation. *International Journal of Forecasting*, 38(4): 1325–1336, 2022.

Michael McCabe, Bruno Régaldo-Saint Blancard, Liam Parker, Ruben Ohana, Miles Cranmer, Alberto Bietti, Michael Eickenberg, Siavash Golkar, Geraud Krawezik, Francois Lanusse, et al. Multiple physics pretraining for spatiotemporal surrogate models. *Advances in Neural Information Processing Systems*, 37:119301–119335, 2024.

R Thomas McCoy, Shunyu Yao, Dan Friedman, Mathew D Hardy, and Thomas L Griffiths. Embers of autoregression show how large language models are shaped by the problem they are trained to solve. *Proc. Natl. Acad. Sci. U.S.A.*, 121(41):e2322420121, 2024.

Kaden McKeen, Laura Oliva, Sameer Masood, Augustin Toma, Barry Rubin, and Bo Wang. Ecg-fm: An open electrocardiogram foundation model. *arXiv preprint arXiv:2408.05178*, 2024.

Jonas Mikhaeil, Zahra Monfared, and Daniel Durstewitz. On the difficulty of learning chaotic dynamics with rnns. *Advances in Neural Information Processing Systems*, 35:11297–11312, 2022.

John A Miller, Mohammed Aldosari, Farah Saeed, Nasid Habib Barna, Subas Rana, I Budak Arpinar, and Ninghao Liu. A survey of deep learning and foundation models for time series forecasting. *arXiv:2401.13912*, 2024.

Melanie Mitchell and David C Krakauer. The debate over understanding in ai's large language models. *Proc. Natl. Acad. Sci. U.S.A.*, 120(13):e2215907120, 2023.

Tung Nguyen, Johannes Brandstetter, Ashish Kapoor, Jayesh K Gupta, and Aditya Grover. Climax: A foundation model for weather and climate. *arXiv:2301.10343*, 2023.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. In-context learning and induction heads. *arXiv:2209.11895*, 2022.

Boris N Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. Meta-learning framework with applications to zero-shot time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 9242–9250, 2021.

Jaideep Pathak, Brian Hunt, Michelle Girvan, Zhixin Lu, and Edward Ott. Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach. *Phys. Rev. Lett.*, 120(2):024102, 2018.

Md Ashiqur Rahman, Robert Joseph George, Mogab Elleithy, Daniel Leibovici, Zongyi Li, Boris Bonev, Colin White, Julius Berner, Raymond A Yeh, Jean Kossaifi, et al. Pretraining codomain attention neural operators for solving multiphysics pdes. *arXiv:2403.12553*, 2024.

Kashif Rasul, Arjun Ashok, Andrew Robert Williams, Arian Khorasani, George Adamopoulos, Rishika Bhagwatkar, Marin Bilos, Hena Ghonia, Nadhir Vincent Hassen, Anderson Schneider, et al. Lag-llama: Towards foundation models for time series forecasting. *arXiv:2310.08278*, 2023.

Gautam Reddy. The mechanistic basis of data dependence and abrupt learning in an in-context classification task. *arXiv:2312.03002*, 2023.

Matthew Ricci, Guy Pelc, Zoe Piran, Noa Moriel, and Mor Nitzan. Trendy: Temporal regression of effective non-linear dynamics. *arXiv preprint arXiv:2412.03496*, 2024.

Michael T Rosenstein, James J Collins, and Carlo J De Luca. A practical method for calculating largest lyapunov exponents from small data sets. *Physica D: Nonlinear Phenomena*, 65(1-2): 117–134, 1993.

Xiaoming Shi, Shiyu Wang, Yuqi Nie, Dianqi Li, Zhou Ye, Qingsong Wen, and Ming Jin. Time-moe: Billion-scale time series foundation models with mixture of experts. In *The Thirteenth International Conference on Learning Representations*, 2025.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. *Advances in Neural Information Processing Systems*, 35:19523–19536, 2022.

Shashank Subramanian, Peter Harrington, Kurt Keutzer, Wahid Bhimji, Dmitriy Morozov, Michael W Mahoney, and Amir Gholami. Towards foundation models for scientific machine learning: Characterizing scaling and transfer behavior. *Advances in Neural Information Processing Systems*, 36, 2024.

George Sugihara. Nonlinear forecasting for the classification of natural time series. *Philosophical Transactions of the Royal Society of London. Series A: Physical and Engineering Sciences*, 348 (1688):477–495, 1994.

George Sugihara and Robert M May. Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series. *Nature*, 344(6268):734–741, 1990.

Makoto Takamoto, Timothy Praditia, Raphael Leiteritz, Daniel MacKinlay, Francesco Alesiani, Dirk Pflüger, and Mathias Niepert. Pdebench: An extensive benchmark for scientific machine learning. *Advances in Neural Information Processing Systems*, 35:1596–1611, 2022.

Floris Takens. Detecting strange attractors in turbulence. In *Dynamical Systems and Turbulence, Warwick 1980: proceedings of a symposium held at the University of Warwick 1979/80*, pp. 366–381. Springer, 2006.

Kunio Takezawa. *Introduction to nonparametric regression*. John Wiley & Sons, 2005.

Amin Totounferoush, Serge Kotchourko, Michael W Mahoney, and Steffen Staab. Paving the way for scientific foundation models: enhancing generalization and robustness in pdes with constraint-aware pre-training. *arXiv preprint arXiv:2503.19081*, 2025.

Yao-Hung Hubert Tsai, Shaojie Bai, Makoto Yamada, Louis-Philippe Morency, and Ruslan Salakhutdinov. Transformer dissection: An unified understanding for transformer's attention via the lens of kernel. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2019.

VP Vera-Ávila, Ricardo Sevilla-Escoboza, AA Lozano-Sánchez, RR Rivera-Durón, and Javier M Buldú. Experimental datasets of networks of nonlinear oscillators: Structure and dynamics during the path to synchronization. *Data in brief*, 28:105012, 2020.

Pantelis R Vlachas, Jaideep Pathak, Brian R Hunt, Themistoklis P Sapsis, Michelle Girvan, Edward Ott, and Petros Koumoutsakos. Backpropagation algorithms and reservoir computing in recurrent neural networks for the forecasting of complex spatiotemporal dynamics. *Neural. Netw.*, 126: 191–217, 2020.

Johannes Von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *International Conference on Machine Learning*, pp. 35151–35174. PMLR, 2023.

P Walters. An introduction to ergodic theory. *Springer-Verlag*, 1982.

E Weinan. A proposal on machine learning via dynamical systems. *Commun. Math. Stat.*, 1(5):1–11, 2017.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. *arXiv:2402.02592*, 2024.

Liu Yang, Siting Liu, Tingwei Meng, and Stanley J Osher. In-context operator learning with data prompts for differential equation problems. *Proc. Natl. Acad. Sci. U.S.A.*, 120(39):e2310142120, 2023.

Lu Yang, Xiuwen Sun, Boumediene Hamzi, Houman Owhadi, and Naiming Xie. Learning dynamical systems from data: A simple cross-validation perspective, part v: Sparse kernel flows for 132 chaotic dynamical systems. *Physica D: Nonlinear Phenomena*, 460:134070, 2024.

Qingren Yao, Chao-Han Huck Yang, Renhe Jiang, Yuxuan Liang, Ming Jin, and Shirui Pan. Towards neural scaling laws for time series foundation models. *arXiv:2410.12360*, 2024.

Annan Yu, Danielle C Maddix, Boran Han, Xiyuan Zhang, Abdul Fatir Ansari, Oleksandr Shchur, Christos Faloutsos, Andrew Gordon Wilson, Michael W Mahoney, and Yuyang Wang. Understanding the implicit biases of design choices for time series foundation models. *arXiv preprint arXiv:2510.19236*, 2025.

Rose Yu and Rui Wang. Learning dynamical systems from data: An introduction to physics-guided deep learning. *Proc. Natl. Acad. Sci. U.S.A.*, 121(27):e2311808121, 2024.

Oussama Zekri, Ambroise Odonnat, Abdelhakim Benechehab, Linus Bleistein, Nicolas Boullé, and Ievgen Redko. Large language models as markov chains. *arXiv preprint arXiv:2410.02724*, 2024.

Zheng-Meng Zhai, Jun-Yin Huang, Benjamin D Stern, and Ying-Cheng Lai. Reconstructing dynamics from sparse observations with no training on target system. *arXiv:2410.21222*, 2024.

Yuanzhao Zhang and William Gilpin. Zero-shot forecasting of chaotic systems. *arXiv:2409.15771*, 2024.

Tian Zhou, Peisong Niu, Liang Sun, Rong Jin, et al. One fits all: Power general time series analysis by pretrained lm. *Advances in neural information processing systems*, 36:43322–43355, 2023.

A Sample predictions from foundation models

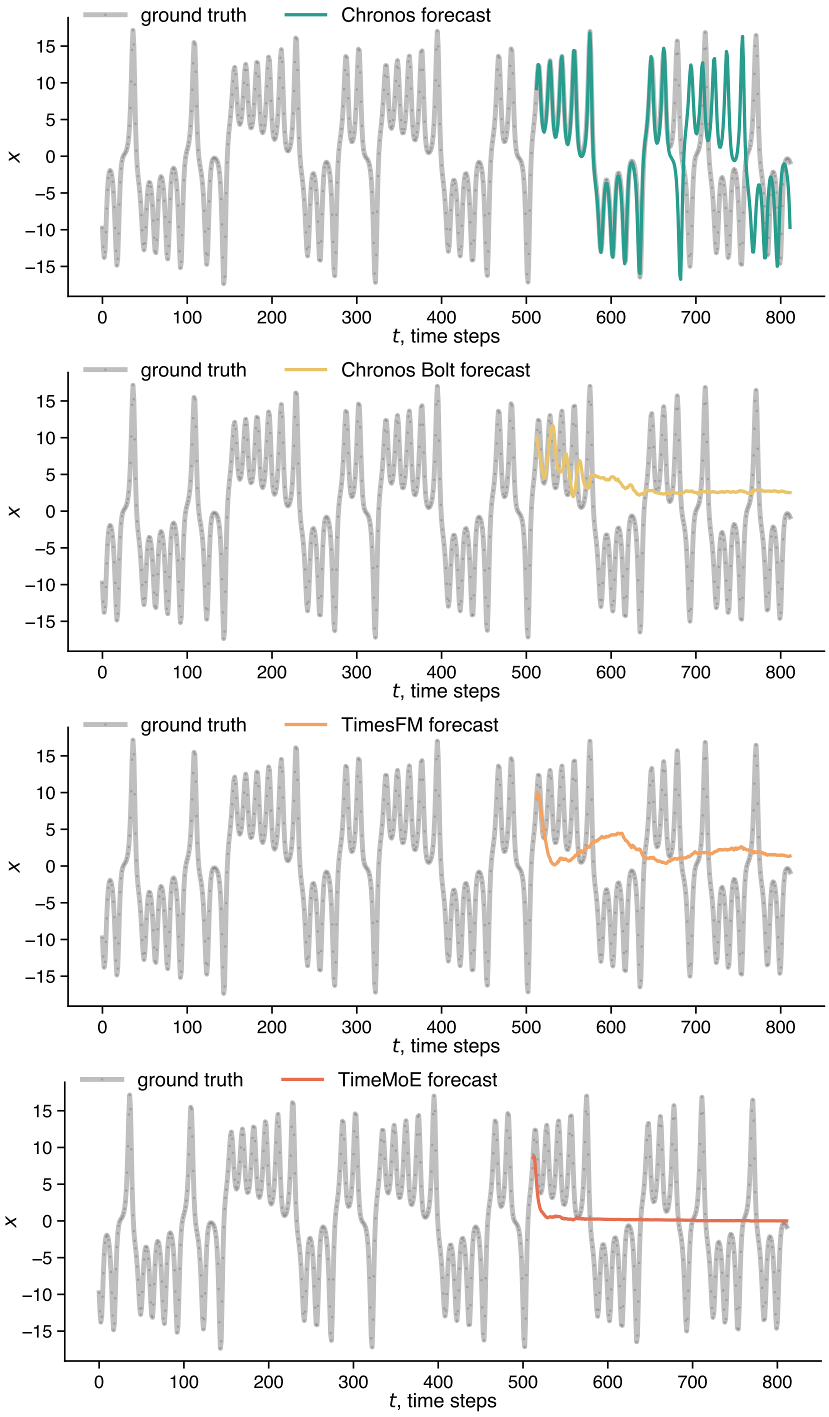

Figure 5: **Example forecasts on a chaotic system from foundation models reveal common failure modes**. This is the same task as presented in Fig. 1 (predicting the x variable of the Lorenz system based on a short context trajectory with 512 data points). Chronos does extremely well with a parroting strategy. The other models perform comparatively poorly and all exhibit a tendency to underestimate the oscillations (e.g., by quickly converging towards the mean). This is a general trend that we consistently observe across different chaotic systems and initial conditions.

B Benchmarking with other metrics

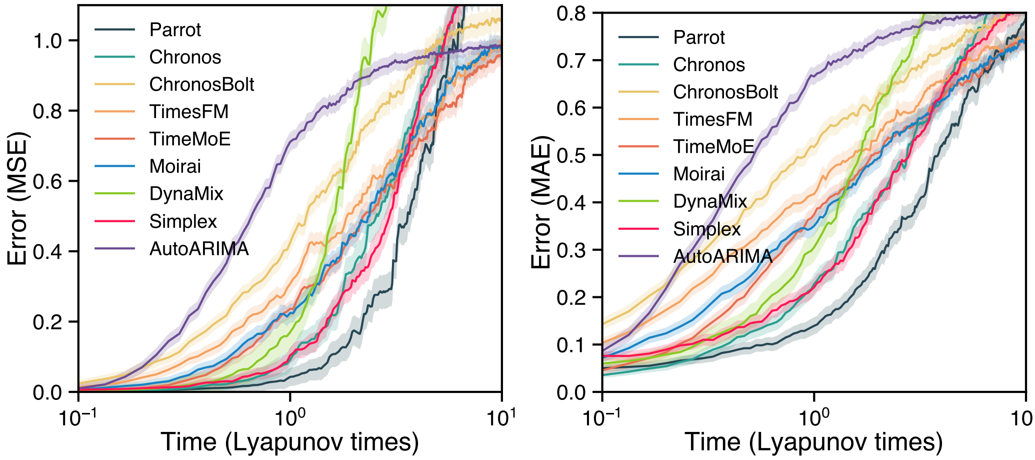

Figure 6: **Context parroting outperforms foundation models in zero-shot forecasting**. Same setup as in Fig. 2, but with the forecast error measured by MSE (left) and MAE (right). On top of the foundation models, we also include two classical forecasting methods in the comparison: simplex projection (Sugihara & May, 1990) and AutoARIMA (Hyndman & Athanasopoulos, 2018).

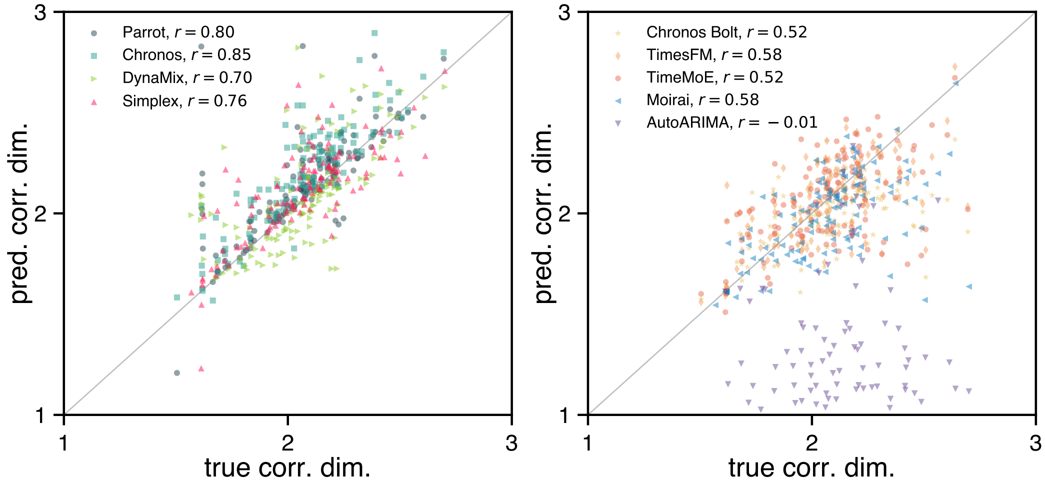

Figure 7: **Fractal dimension accuracy for parroting and foundation models**. Each point represents the predicted fractal dimension of a chaotic attractor by a model (median of 20 predictions from random initial conditions). The accuracy is measured by the Spearman correlation r between the 135 predicted fractal dimensions and the true fractal dimensions.

C Effects of embedding dimension D

Fig. 10 investigates how the choice of the embedding dimension D affects the performance of context parroting. Overall, the valid prediction time stays consistent over a wide range of embedding dimension D. For short context windows, there is a slight advantage to small D. For long context windows, larger embedding dimensions are marginally better. This observation suggests potential improvements in the future that choose D adaptively based on factors such as context length.

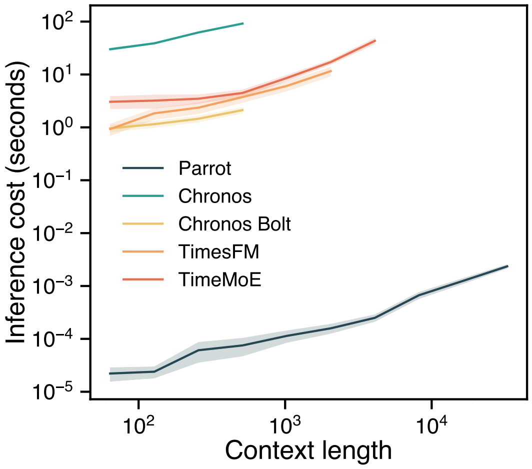

Figure 8: **Context parroting is computationally more efficient than foundation models.** There is generally a gap of five or six orders of magnitude between context parroting and foundation models. For each foundation model, context lengths from 64 to the maximum context window are considered. All inferences are performed with CPUs, and the forecast horizon is fixed to 300 steps.

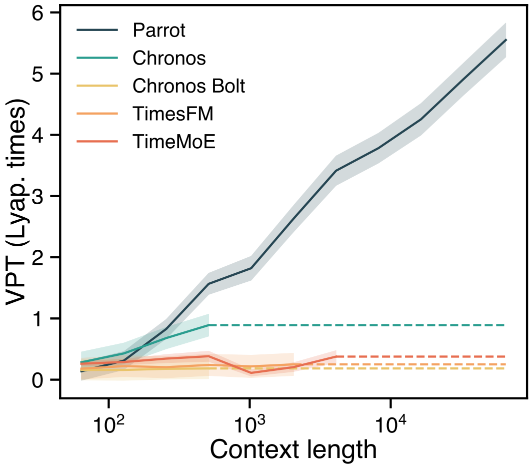

Figure 9: **Valid prediction time as a function of the context length**. Dashed lines indicate context lengths exceeding the maximum context window of the model. Results are averaged over 135 chaotic systems in the dysts database, with 20 trajectories from random initial conditions for each system.

D Predicting scaling coefficient with the attractor dimension

Figure 11 shows the relation between the scaling coefficient of the in-context neural scaling law and the correlation dimension of the chaotic attractor.

E Effect of context length on invariant properties

In Figure 12, we explore how the accuracy of four representative zero-shot models changes as the context length varies over an order of magnitude. We observe that all models exhibit monotonic scaling, consistent with additional samples from the attractor enabled by larger context acting to improve the stability of reconstructed attractors.

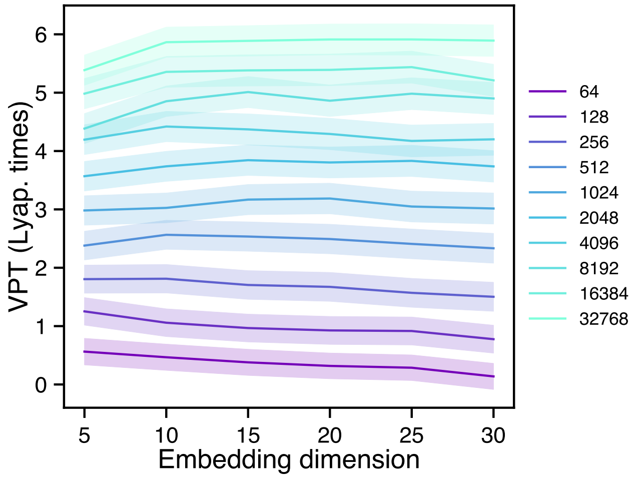

Figure 10: **Effect of the embedding dimension D on the forecast accuracy of context parroting.** Each curve represents a different context length. Results are averaged over 135 chaotic systems in the `dysts` database, with 100 trajectories from random initial conditions for each system.

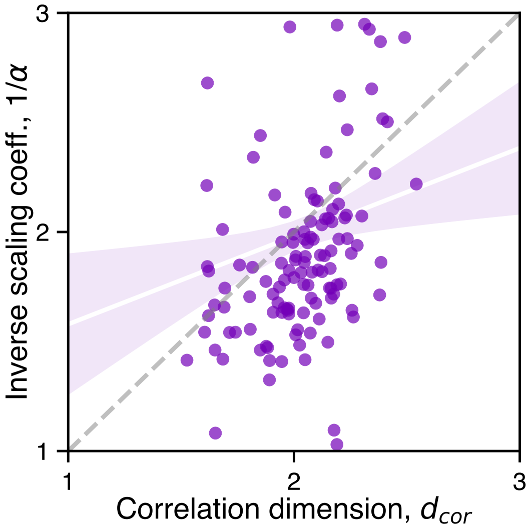

Figure 11: **Inverse scaling coefficient $1/\alpha$ versus fractal dimension d_{cor} of the chaotic attractor.** We estimate α from the ℓ vs L plot obtained with context parroting at $D = 10$, where L ranges from 2^6 to 2^{16}. This is done separately for each chaotic system. Thus, each dot represents one of the 135 chaotic systems included in our benchmark. The Spearman rank-order correlation coefficient is 0.51 ± 0.08 (bootstrapped standard error), and a linear regression with standard error range is underset (shaded interval). The dashed line corresponds to an exact $1:1$ scaling.

F Effect of forecast horizon on invariant properties

We next test the performance of parroting for long forecast horizons. We fix the context length $L = 512$ and then generate forecasts of length $H = 10000 - 512 = 9488$ (equivalent to over 316 Lyapunov Times). Table 4 shows the results of generating forecasts using the best-performing models from our shorter-horizon experiments. For each model, we evaluate its global accuracy by calculating (1) the correlation between the fractal dimension of the long forecast, and an estimate generated from the ground truth; (2) the correlation between the largest Lyapunov exponent of the long forecast, and an estimate generated from the ground truth; and (3) the attractor KL-divergence between the long forecast and ground truth (Grassberger & Procaccia, 1983; Rosenstein et al., 1993; Hess et al.,

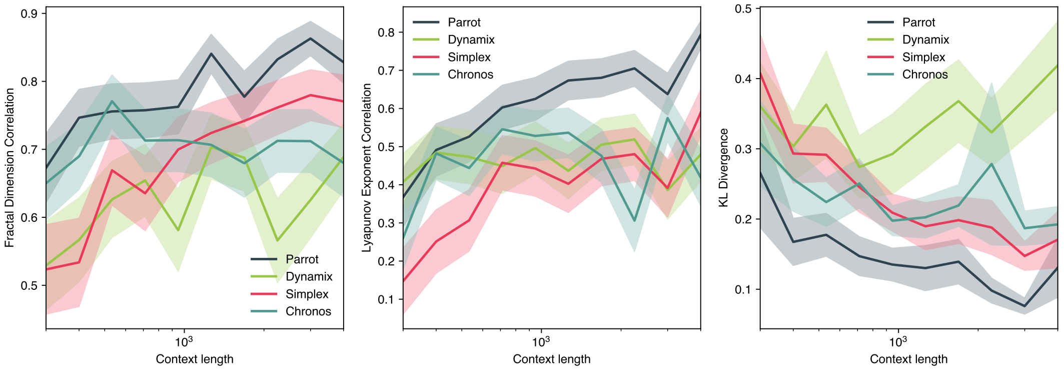

Figure 12: **Dependence of invariant properties on context length.** (Left) The Spearman correlation between the fractal dimension of the true trajectory and predicted trajectory as the context length varies. (Middle) The Spearman correlation between the estimated maximum Lyapunov exponent of the true trajectory and predicted trajectory as the context length varies. (Right) The estimated KL divergence between Gaussian mixtures fit to the true trajectory and predicted trajectory, as the context length varies. Note that Chronos has a finite context length, and so its performance saturates due to architectural constraints, and not intrinsic limitations. For this plot, forecasts are generated for 300 timepoints past the context. For the same figure with longer forecasts, see Figure 13.

2023). We find that context parroting and DynaMix both perform well, indicating that DynaMix is a general-purpose time-series foundation model with a unique ability to capture the long-term climate of chaotic systems (Hemmer & Durstewitz, 2025).

Metric	**Parrot**	**Chronos**	**Dynamix**	**Simplex**
Attractor KL Divergence	**0.412 ± 0.141**	0.679 ± 0.101	*0.508 ± 0.147*	0.546 ± 0.140
Fractal Dimension Correlation	**0.723 ± 0.042**	0.120 ± 0.118	*0.521 ± 0.057*	0.341 ± 0.072
Largest Lyapunov Correlation	*0.343 ± 0.018*	0.269 ± 0.114	**0.466 ± 0.071**	*0.343 ± 0.085*

Table 4: KL Divergence and correlation of invariant properties between predicted and true attractors for different models for long forecast horizons. Error bars are standard deviation across all attractors for the KL Divergence, and uncertainty bounds based on the p-value for correlations. Bold = best, italic = second best.

G Effects of noise and sampling rate

We add Gaussian noise to normalized chaotic trajectories and repeat the original experiments (a noise level of 0.1 translates to 10% perturbation on each data point on average). The results are consistent across different orders of magnitude in noise, and parroting is consistently the best or the second best in all experiments.

Noise level	**Parrot**	**Chronos**	**Chronos Bolt**	**TimesFM**	**TimeMoE**
10^{-3}	**2.17±0.19**	*1.68±0.18*	0.79±0.18	1.07±0.19	0.92±0.15
10^{-2}	**2.10±0.18**	*1.65±0.18*	0.79±0.18	1.05±0.19	0.92±0.16
10^{-1}	**1.04±0.14**	*0.89±0.15*	0.71±0.17	*0.89±0.17*	0.66±0.11

Table 5: **Valid prediction time** across noise levels (higher is better). Bold = best, italic = second best. Shading highlights best (dark) and second best (light).

In the main text, we set an intermediate granularity of 30 points per Lyapunov time. Below we compare it with results obtained for granularities of 10 points per Lyapunov time and 50 points per

Noise level	Parrot	Chronos	Chronos Bolt	TimesFM	TimeMoE
10^{-3}	**0.233±0.221**	*0.297±0.243*	0.491±0.223	0.440±0.228	0.377±0.211
10^{-2}	**0.235±0.220**	*0.311±0.245*	0.492±0.223	0.441±0.228	0.383±0.211
10^{-1}	**0.286±0.209**	*0.366±0.235*	0.509±0.219	0.455±0.224	0.415±0.216

Table 6: **MAE @ 1 Lyapunov Time** across noise levels (lower is better). Bold = best, italic = second best. Shading highlights best (dark) and second best (light).

Noise level	Parrot	Chronos	Chronos Bolt	TimesFM	TimeMoE
10^{-3}	**0.183±0.339**	*0.268±0.367*	0.473±0.314	0.394±0.307	0.315±0.273
10^{-2}	**0.185±0.340**	*0.282±0.361*	0.474±0.314	0.394±0.306	0.318±0.274
10^{-1}	**0.220±0.346**	*0.328±0.373*	0.489±0.314	0.407±0.302	0.349±0.285

Table 7: **MSE @ 1 Lyapunov Time** across noise levels (lower is better). Bold = best, italic = second best. Shading highlights best (dark) and second best (light).

Noise level	Parrot	Chronos	Chronos Bolt	TimesFM	TimeMoE
10^{-3}	*0.73*	**0.85**	0.52	0.58	0.52
10^{-2}	*0.63*	**0.77**	0.51	0.55	0.45
10^{-1}	**0.59**	*0.57*	0.37	0.49	0.16

Table 8: **Fractal dimension accuracy** (Spearman correlation) across noise levels (higher is better). Bold = best, italic = second best. Shading highlights best (dark) and second best (light).

Noise level	Parrot	Chronos	Chronos Bolt	TimesFM	TimeMoE
10^{-3}	**0.113±0.205**	*0.173±0.209*	0.346±0.297	0.345±0.298	0.354±0.290
10^{-2}	**0.115±0.201**	*0.189±0.244*	0.344±0.292	0.356±0.298	0.344±0.285
10^{-1}	**0.141±0.207**	*0.218±0.263*	0.382±0.314	0.389±0.306	0.433±0.306

Table 9: **KL Divergence** between predicted and true attractors across noise levels (lower is better). Bold = best, italic = second best. Shading highlights best (dark) and second best (light).

Lyapunov time. Granularity does not strongly affect the results or relative model ranking. Parroting is either the best or the second best in all experiments. This makes sense, as we would expect changing granularity to have a similar effect as rescaling of time (although with bigger or smaller gaps between data points). For example, if we use finer granularity by a factor of 2, then we would need to double the context length to get the same lookback window.

Granularity	Parrot	Chronos	Chronos Bolt	TimesFM	TimeMoE
10	**4.70±0.57**	*3.93±0.59*	1.55±0.50	1.92±0.54	1.43±0.26
30	**2.15±0.19**	*1.68±0.18*	0.79±0.18	1.07±0.19	0.92±0.15
50	**1.41±0.11**	*1.12±0.11*	0.55±0.10	0.79±0.11	0.54±0.05

Table 10: **Valid prediction time** across different granularities (higher is better). Bold = best, italic = second best.

H Theoretical Properties of Context Parroting

H.1 Overview

Mathematical Formulation. Context parroting corresponds to a continuous 1-nearest-neighbor search over sequences of length D in the context of length L. It thus corresponds to a limit of a

Granularity	Parrot	Chronos	Chronos Bolt	TimesFM	TimeMoE
10	**0.219±0.204**	*0.316±0.256*	0.567±0.226	0.481±0.218	0.414±0.232
30	**0.233±0.221**	*0.297±0.243*	0.491±0.223	0.440±0.228	0.377±0.211
50	**0.270±0.226**	*0.329±0.241*	0.527±0.224	0.448±0.235	0.412±0.193

Table 11: **MAE @ 1 Lyapunov Time** across different granularities (lower is better). Bold = best, italic = second best.

Granularity	Parrot	Chronos	Chronos Bolt	TimesFM	TimeMoE
10	**0.163±0.311**	*0.291±0.364*	0.565±0.314	0.429±0.293	0.349±0.300
30	**0.164±0.295**	*0.268±0.367*	0.473±0.314	0.394±0.307	0.315±0.273
50	**0.224±0.347**	*0.310±0.377*	0.536±0.341	0.426±0.331	0.352±0.272

Table 12: **MSE @ 1 Lyapunov Time** across different granularities (lower is better). Bold = best, italic = second best.

Granularity	Parrot	Chronos	Chronos Bolt	TimesFM	TimeMoE
10	**0.87**	*0.82*	0.34	0.39	0.36
30	*0.80*	**0.85**	0.52	0.58	0.52
50	**0.89**	*0.86*	0.41	0.60	0.56

Table 13: **Fractal dimension accuracy** (Spearman correlation) across different granularities (higher is better). Bold = best, italic = second best.

Granularity	Parrot	Chronos	Chronos Bolt	TimesFM	TimeMoE
10	**0.087±0.137**	*0.127±0.173*	0.573±0.307	0.444±0.326	0.467±0.368
30	**0.122±0.194**	*0.173±0.209*	0.346±0.297	0.345±0.298	0.354±0.290
50	**0.137±0.207**	*0.230±0.256*	0.406±0.305	0.361±0.323	0.370±0.338

Table 14: **KL Divergence** between predicted and true attractors across different granularities (lower is better). Bold = best, italic = second best.

Nadaraya–Watson model of the time series,

$$\hat{p}(\mathbf{y} \mid \mathbf{q}) = \frac{\sum_{j=D}^{L-H} K_\sigma\left(\mathbf{q},\, \mathbf{x}_{j-(D-1):j}\right)\; K_\sigma\left(\mathbf{y},\, \mathbf{x}_{j+1:j+H}\right)}{\sum_{j=D}^{L-H} K_\sigma\left(\mathbf{q},\, \mathbf{x}_{j-(D-1):j}\right)}, \tag{1}$$

where the query $\mathbf{q}$ represents the length-D motif immediately preceding the end of the context window. $\mathbf{y}$ represents a length-H forecast of subsequent values. The forecast sequence $\mathbf{y}$ has probability $\hat{p}$ conditioned on the query. The symmetric kernel $K_\sigma(\mathbf{u}, \mathbf{v}) = \sigma^{-d} K\big((\mathbf{u} - \mathbf{v})/h\sigma\big)$ has bandwidth σ in dimension $d = D \cdot \dim(x_t)$. Assuming mean-squared error as a distance function in sequence space, we use a Gaussian kernel

$$K_\sigma(\mathbf{u}, \mathbf{v}) = \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{\|\mathbf{u} - \mathbf{v}\|^2}{2\sigma^2}\right)$$

We set the second kernel on $\mathbf{y}$ in Eq. 1 to a delta function, in order to output predictions that exactly match sequences from the context, rather than nearby sequences in a least-squares sense. We write the conditional mean predictor

$$\hat{\mathbf{y}}(\mathbf{q}) = \sum_{j=D}^{L-H} w(\mathbf{q}, \mathbf{x}_{j-(D-1):j})\, \mathbf{x}_{j+1:j+H}, \quad w(\mathbf{q}, \mathbf{z}) \equiv \frac{K_\sigma\left(\mathbf{q},\, \mathbf{z}\right)}{\sum_{j=D}^{L-H} K_\sigma\left(\mathbf{q},\, \mathbf{x}_{j-(D-1):j}\right)}. \tag{2}$$

Context parroting corresponds to the 1-nearest-neighbor limit $\sigma \to 0$.

Context parroting preserves attractor properties at long context lengths. In Appendix H.4, we derive the following proposition,

$$\lim_{L\to\infty} \mathbb{E}_p[F(\mathbf{y})|\mathbf{q}] = \mathbb{E}_\mu[F(\mathbf{x})]$$

where L is the context length for an Nadaraya–Watson estimator p, $F(\mathbf{y})$ is an estimate from a forecast sequence $\mathbf{y}$ of a property F of an ergodic dynamical system, which has an invariant value $\mathbb{E}_\mu[F(\mathbf{x})]$ when calculated over the full attractor with underlying measure μ. The query $\mathbf{q}$ is an arbitrary sequence of consecutive timepoints from the dynamical system. This proposition states that, when the context is sufficiently long, context parroting of an ergodic system preserves invariant values of the underlying dynamics. Context parroting thus represents an effective baseline for dynamical systems forecasting, because, in the limit of long context, it will preserve global properties like conditional distributions of values, Lyapunov exponents, or entropy production rates.

H.2 Discrete-token parroting

For fully-discrete tokens, a D^{th} order Markov chain fit to the context has the form

$$p(\mathbf{y}|\mathbf{q}) = \frac{\#\{j : (x_{j-(D-1):j} = \mathbf{q}) \wedge (x_{j+1:j+H} = \mathbf{y})}{\sum_{y'} \#\{i : (x_{j-(D-1):j} = \mathbf{q}) \wedge (x_{j+1:j+H} = \mathbf{y}')} \tag{3}$$

where the overall context has length L, and the Markov chain conditions the $H < L$ future tokens on the $D < L$ preceding tokens. The index j runs over all contiguous sequences of length $D + H$ in the context, $j \in \{D-1, D, ..., L-H-2, L-H-1\}$. The vector $\mathbf{q} \in \mathbb{R}^D$ represents the query, and the vector $\mathbf{y} \in \mathbb{R}^H$ represents the prediction in response to this query. Eq. 3 simply counts the number of token sequences of length $D + H$ that start with a given sequence of D query tokens. A maximum-likelihood estimator derived from this model always samples the highest-likelihood sequence $\mathbf{y}$,

$$\hat{\mathbf{y}}_{\text{MLE}}(\mathbf{q}) = \text{argmax}_{\mathbf{y}} \log p(\mathbf{y}|\mathbf{q})$$

However, this estimator may be unstable due to the appearance of queries $\mathbf{q}$ not seen in the context, motivating the use of *token smoothing*, in which Eq. 3 is replaced by the distribution

$$p(\mathbf{y}|\mathbf{q}) = \frac{\#\{j : (x_{j-(D-1):j} = \mathbf{q}) \wedge (x_{j+1:j+H} = \mathbf{y}) + \alpha}{\sum_{\mathbf{y}'} \left(\#\{i : (x_{j-(D-1):j} = \mathbf{q}) \wedge (x_{j+1:j+H} = \mathbf{y}') + \alpha\right)} \tag{4}$$

with increasing values of the parameter α causing predictions to converge to a uniform sample over possible predictions $\mathbf{y}$. The parameter value $\alpha = 0$ reduces to no smoothing, while $\alpha = 0.5$ corresponds to the Jeffreys prior and $\alpha = 1$ corresponds to Laplace's rule of succession.

H.3 Continuous-token parroting

A more general time series model treats tokens as continuous-valued. Some time series foundation models like Chronos use binning to discretize time series values, allowing the direct use of discrete-token architectures (Ansari et al., 2024). However, many time series models assume effective continuity in token values, and we favor a continuous formulation in order to highlight connections to dynamical systems theory.

To model continuous-valued tokens directly, we replace the discrete count in §H.2 with a kernel-weighted estimate over all past subsequences. Let $\{\mathbf{x}_t\}$ denote a univariate or multivariate time series. For context length L and prediction horizon H, the Nadaraya–Watson estimate of the conditional density is

$$\hat{p}(\mathbf{y} \mid \mathbf{q}) = \frac{\sum_{j=D}^{L-H} K_h\left(\mathbf{q},\, \mathbf{x}_{j-(D-1):j}\right)\; K_h\left(\mathbf{y},\, \mathbf{x}_{j+1:j+H}\right)}{\sum_{j=D}^{L-H} K_h\left(\mathbf{q},\, \mathbf{x}_{j-(D-1):j}\right)}, \tag{5}$$

where $K_h(\mathbf{u}, \mathbf{v}) = h^{-d} K\big((\mathbf{u} - \mathbf{v})/h\big)$ is a symmetric kernel with bandwidth h in dimension $d = D \cdot \dim(\mathbf{x}_t)$ for the first kernel, and $d = H \cdot \dim(\mathbf{x}_t)$ for the second kernel. Assuming mean-squared error as a distance function in sequence space, we use a Gaussian kernel

$$K_h(\mathbf{u}, \mathbf{v}) = \frac{1}{(2\pi h^2)^{d/2}} \exp\left(-\frac{\|\mathbf{u} - \mathbf{v}\|^2}{2h^2}\right)$$

In practice, we drop the second kernel on $\mathbf{y}$ in Eq. 5 in order to output a prediction that exactly matches sequences from the context, rather than nearby sequences in a least-squares sense. We thus write the conditional mean predictor

$$\hat{\mathbf{y}}(\mathbf{q}) = \sum_{j=D}^{L-H} w(\mathbf{q}, \mathbf{x}_{j-(D-1):j})\, \mathbf{x}_{j+1:j+H} \tag{6}$$

where we have isolated a term corresponding to the weight of each sequence,

$$w(\mathbf{q}, \mathbf{z}) \equiv \frac{K_h\left(\mathbf{q},\, \mathbf{z}\right)}{\sum_{j=D}^{L-H} K_h\left(\mathbf{q},\, \mathbf{x}_{j-(D-1):j}\right)}.$$

Nearest-neighbor and global average limits. The bandwidth h plays the role of a smoothing parameter (analogous to α in Eq. 4). As $h \to 0$ the scheme approximates a single-nearest neighbor parrot, while as $h \to \infty$ it converges to a global average over all sequences.

Connection to attention. If one takes

$$K(\mathbf{u}, \mathbf{v}) = \exp\left(\mathbf{u}^\top \mathbf{v}/\tau\right),$$

then Eq. 6 recovers a simplified form of softmax-attention, with the temperature hyperparameter τ controlling smoothness. In this view, the continuous parroting scheme is a kernel-regression analogue of discrete k–gram smoothing (Tsai et al., 2019).

k-nearest-neighbor limit. We define a set Top$k$ corresponding to a subset of the possible values of the index $j \in \{D, D+1, ..., L-H-1, L-H\}$. The k elements of Top$k$ correspond to the indices j that produce the k largest values of $w(\mathbf{q}, \mathbf{x}_{j-(D-1):j})$ across all values of j. We compute a simple average of these k closest matches

$$\hat{\mathbf{y}}(\mathbf{q}) = \frac{1}{k} \sum_{j \in \text{Top}k} w(\mathbf{q}, \mathbf{x}_{j-(D-1):j})\, \mathbf{x}_{j+1:j+H} \tag{7}$$

yielding a k–nearest-neighbors parroting scheme. As k increases, this estimator interpolates between exact parroting ($k = 1$) and global average ($k \to L$).

Simplex projection. Simplex projection, a classical forecasting method in nonlinear dynamics, corresponds to the special case $H = 1$ (single step prediction), $k = D + 1$ in Eq. 7. The condition $k = D + 1$ represents the minimal number of affinely independent neighbors needed to triangulate a point in a D-dimensional space (Sugihara & May, 1990).

In simplex projection, the query $\mathbf{q}$ is interpreted as a time-delay embedding of the time series observable $\mathbf{x}$. Takens' theorem argues that, under mild conditions, a finite number of time delay embeddings of an observable drawn from a deterministic ergodic system will be diffeomorphic (smoothly mappable) to the full-state dynamics (Takens, 2006). Because simplex projection uses only neighbor identities, and not absolute distances, to weight context points, a delay embedding is sufficient to calculate the appropriate weights.

S-map forecasts. Another common nonlinear forecasting technique retains all terms in the sum Eq. 6 , but instead performs a nonlinear weighting of the form

$$K_\theta(\mathbf{u}, \mathbf{v}) = \exp(-\theta \|\mathbf{u} - \mathbf{v}\|/\bar{d})$$

where the scale parameter $\bar{d}$ is determined by the distribution of distances among queries and points in the context. In practice, this parameter is often set to the mean pairwise distance among all sequences of length D in the context. The optimal value of the hyperparameter θ increases as the underlying dynamics become more strongly nonlinear Sugihara (1994). We note that, in the classical formulation of the S-map, a locally-linear model is fit based on all sequences of length $D + H$ seen in the context, while here we use the Nadaraya–Watson estimator in order to emphasize connections to modern kernel regression.

H.4 Invariants of motion

For ergodic dynamical systems in continuous time, there exists a natural measure $\mu(\mathbf{x})$ such that, for certain observables $F(\mathbf{x})$, the following condition almost surely holds,

$$\mathbb{E}_\mu[F] \equiv \lim_{T\to\infty} \frac{1}{T} \int_0^T F(\mathbf{x}) = \int F(\mathbf{x}) d\mu(\mathbf{x}) = \text{constant}$$

where the second equality arises from the Birkhoff ergodic theorem (Walters, 1982).

We use the following convention for expectation values of sequences and single tokens; the expectation $\mathbb{E}_\mu[\mathbf{x}_{t:t+T}]$ refers to the expected value of the sequence $\mathbf{x}_{t:t+T}$ given pointwise measure μ. We note that, for deterministic dynamical systems, once a given point is sampled on the attractor with measure $\mu(\mathbf{x}_t)$, subsequent points have delta function conditional probability on the first point. Thus, we use the convention $\mu(\mathbf{x}_t) = \mu(\mathbf{x}_{t:t+T})$ and we use the measure to refer to both the probability of a given timepoint, or a sequence of arbitrary length originating from that timepoint.

Proposition. Under appropriate kernel conditions,

$$\lim_{L\to\infty} \mathbb{E}_p[F(\mathbf{y})|\mathbf{q}] = \mathbb{E}_\mu[F(\mathbf{x})]$$

where L is the context length for a Nadaraya–Watson estimator p, $F(\mathbf{y})$ is an estimate on a sequence $\mathbf{y}$ of an invariant property of an ergodic dynamical system with measure μ, and $\mathbf{q}$ is an arbitrary sequence of consecutive timepoints from the dynamical system. This proposition states that, when the context is sufficiently long, a Nadaraya–Watson estimator of an ergodic system preserves the invariant values of the underlying dynamics.

Derivation. We start with the definition of the dynamical average,

$$\mathbb{E}_\mu[F] = \int F(\mathbf{x}) d\mu(\mathbf{x})$$

Inserting Eq. 5 into this expression,

$$\mathbb{E}_\mu[F(\mathbf{y})|\mathbf{q}] = \frac{\sum_{j=D}^{L-H} K_h\left(\mathbf{q},\, \mathbf{x}_{j-(D-1):j}\right) \int F(\mathbf{y}) K_h\left(\mathbf{y},\, \mathbf{x}_{j+1:j+H}\right) d\mu(\mathbf{y})}{\sum_{j=D}^{L-H} K_h\left(\mathbf{q},\, \mathbf{x}_{j-(D-1):j}\right)},$$

We multiply both the numerator and denominator by $1/L$ and take the limit $L \to \infty$, in order to convert the summations to expectations,

$$\lim_{L\to\infty} \mathbb{E}_\mu[F(\mathbf{y})|\mathbf{q}] = \frac{\mathbb{E}_\mu\left[K_h(\mathbf{q}, \mathbf{x}_\leftarrow) \int F(\mathbf{y}) K_h\left(\mathbf{y}, \mathbf{x}_\rightarrow\right) d\mu(\mathbf{y})\right]}{\mathbb{E}_\mu[K_h(\mathbf{q}, \mathbf{x}_\leftarrow)]},$$

where $\mathbf{x}_\leftarrow$ represents the first D points of random lookback window of length $D + H$ sampled from the underlying dynamical system, while $\mathbf{x}_\rightarrow$ denotes the next H timepoints. In practice, this corresponds to a time series of $D + H$ points generated by simulating the dynamics starting at a point on the attractor randomly-sampled according to the measure μ.

If we take the limit $h \to 0$ (exact matching), then the kernel K_h becomes a delta function, yielding

$$\lim_{h\to 0} \lim_{L\to\infty} \mathbb{E}_\mu[F(\mathbf{y})|\mathbf{q}] = \mathbb{E}_\mu[F(\mathbf{x}_\rightarrow)|\mathbf{x}_\leftarrow = \mathbf{q}]$$

If the measure μ is ergodic, then the conditional expectation of an invariant F given any query $\mathbf{q}$ is simply its unconditional expectation,

$$\lim_{h\to 0} \lim_{L\to\infty} \mathbb{E}_\mu[F(\mathbf{y})|\mathbf{q}] = \mathbb{E}_\mu[F(\mathbf{x})]$$

H.5 Scaling laws limiting prediction of stochastic systems

For a stochastic time series $\mathbf{x}_{1:T}$ with autocorrelation given by

$$|\text{Corr}(\mathbf{x}_t, \mathbf{x}_{t+\tau})| \le Ce^{-\alpha\tau}, \quad \alpha > 0$$

with C representing a proportionality constant, the expected mean squared error of a forecast scales as

$$\mathbb{E}\left[\|\hat{\mathbf{y}}-\mathbf{y}\|^2\right] \sim e^{-\alpha L}, \quad L \to \infty.$$

Thus, under exponential decay of correlations (mixing), the amount of information about future states in a length-L context window saturates exponentially quickly Bradley (2005). Thus, forecasts derived from increasingly large context windows converge exponentially quickly to optimal conditional forecasts under the invariant measure μ.

Under standard smoothness conditions Fan & Yao (2008); Takezawa (2005), the forecast error also exhibits a standard bias-variance tradeoff of the form

$$\mathbb{E}\left[\|\hat{\mathbf{y}}-\mathbf{y}\|^2\right] = \mathcal{O}(h^4) + \mathcal{O}\left(\frac{1}{Ch^{L+H}}\right).$$

The optimal width of the kernel thus scales as,

$$h_{\text{opt}} \sim C^{-1/(4+L+H)}$$

I Stability of forecasts at long prediction horizons

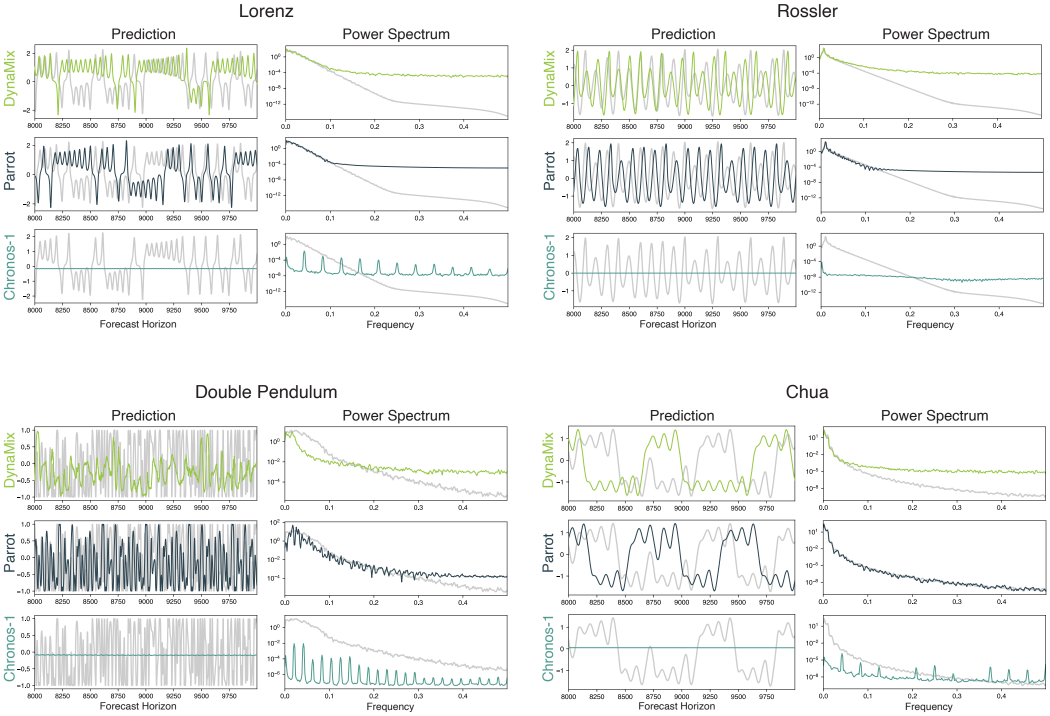

Figure 13: **Properties of forecast models at long forecast horizons.** Predictions of four distinct chaotic systems using various forecast models. Forecasts are generated for $10,000$ points past the end of the context, and the last 2000 timepoints are shown. The power spectrum is estimated using Welch's method on the last 5000 timepoints of a $10,000$ timepoint prediction.

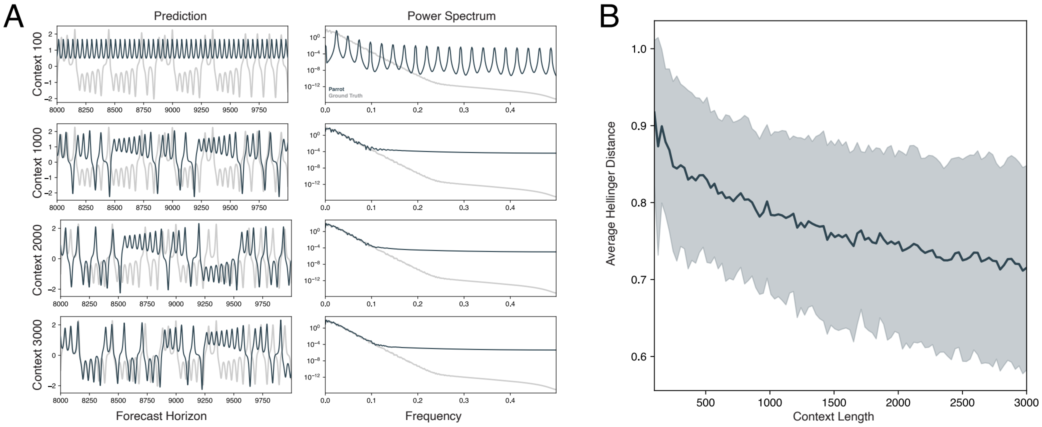

Figure 14: **Attractor properties improve with context length.** (A) Predictions of a single chaotic system, the Lorenz attractor, by the parroting model as the context length increases. Forecasts are generated for $10,000$ points past the end of the context, and the last 2000 timepoints are shown. The power spectrum is estimated using Welch's method on the last 5000 timepoints of a $10,000$ timepoint prediction. (B) The average Hellinger distance between the true and predicted power spectrum as a function of context length, averaged over 129 distinct chaotic systems (including the Lorenz attractor). Error bars correspond to standard deviations. The averaged Hellinger distance is introduced as a long-term metric for chaotic systems in Mikhaeil et al. (2022) and Brenner et al. (2022).

Metric	DynaMix	Parrot
Average Hellinger Distance	0.595 ± 0.166	0.591 ± 0.198
KL Divergence	0.624 ± 0.128	0.469 ± 0.162

Table 15: **Global attractor fidelity metrics calculated at long forecast horizons.** For these experiments, the context length is 2000 timepoints, and the prediction horizon is $10,000$ timepoints. Values correspond to mean and standard errors across 129 distinct chaotic systems.

Invariant Property	DynaMix	Parrot
Largest Lyapunov Exponent	0.278 ± 0.100	0.328 ± 0.097
Fractal Dimension	0.441 ± 0.073	0.832 ± 0.028

Table 16: **Correlation of forecast properties with invariant properties.** For these experiments, the context length is 2000 timepoints, and the prediction horizon is $10,000$ timepoints. Values correspond to Pearson correlations and standard errors (Fisher's transformation) across 129 distinct chaotic systems.

J Stability of distributional metrics against hyperparameter choices

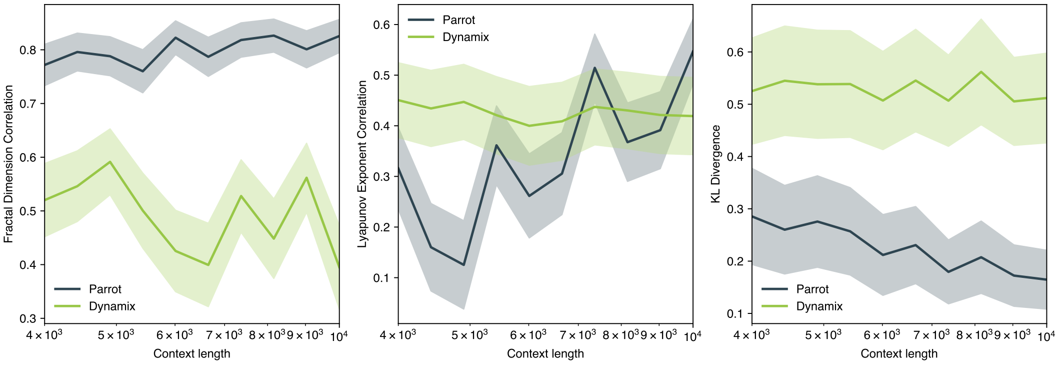

Figure 15: **Properties of forecast models at long forecast horizons as context length varies.** Three measures of long-term forecast properties at very long rollouts (10, 000 timepoints) as the context length varies. For the fractal dimension and Lyapunov exponents, values correspond to Pearson correlations and standard errors (Fisher's transformation) across 129 distinct chaotic systems. For the KL Divergence, values correspond to mean and standard errors across 129 distinct chaotic systems.

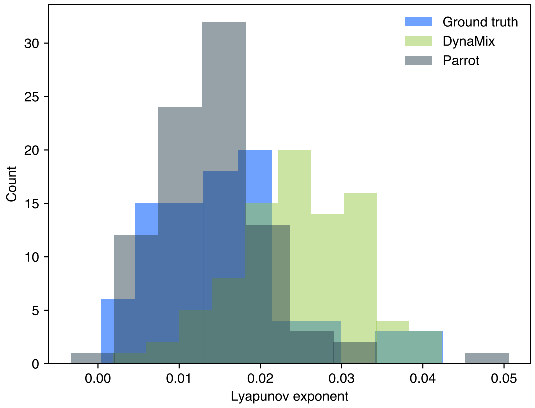

Figure 16: **Histograms of Lyapunov Exponents.** We estimate the Lyapunov exponents from the ground-truth time series, as well as from long rollouts from parroting and DynaMix. These rollouts are generated with a context length of 2000 and a prediction horizon of 10000, and correspond to estimates from all distinct chaotic systems in `dysts`.

Kernel Radius	DynaMix	Parrot
0.01	6.921 ± 0.594	4.194 ± 0.474
0.1	2.786 ± 0.361	1.589 ± 0.281
1.0	0.624 ± 0.128	0.469 ± 0.162
10.0†	-0.098 ± 0.076	-0.075 ± 0.078

Table 17: **Variation of KL divergence at long forecast horizons, for different values of the Gaussian kernel.** For these experiments, the context length is 2000 timepoints, and the prediction horizon is 10, 000 timepoints. Values correspond to mean and standard errors across 129 distinct chaotic systems. †Note: sampling-based KL divergence calculations can fluctuate below zero when the center of the estimate is close to zero. To avoid introducing directional bias, we do not enforce non-negativity with clipping or renormalization, and so these values should be interpreted as being near zero.

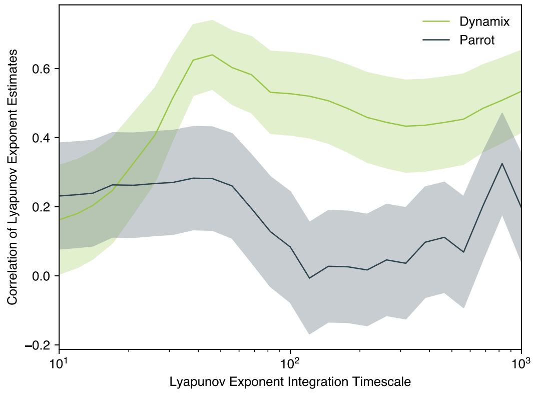

Figure 17: **Dependence of Lyapunov exponent accuracy on estimation algorithm integration time.** We estimate the Lyapunov exponents from the ground-truth time series, as well as from long rollouts from parroting and DynaMix, for varying integration timescales in the Rosenstein algorithm (Rosenstein et al., 1993). These rollouts are generated with a context length of 2000 and a prediction horizon of 10000, and correspond to estimates from all distinct chaotic systems in `dysts`.