PROCEEDINGS A

rspa.royalsocietypublishing.org

Research

Article submitted to journal

Subject Areas:
Mechanical Engineering

Keywords:
Data-driven forecasting, Long-Short Term Memory, Gaussian Processes, T21 barotropic climate model, Lorenz 96

Author for correspondence:
Petros Koumoutsakos
e-mail: petros@ethz.ch

Data-Driven Forecasting of High-Dimensional Chaotic Systems with Long-Short Term Memory Networks

Pantelis R. Vlachas[1], Wonmin Byeon[1], Zhong Y. Wan[2], Themistoklis P. Sapsis[2], Petros Koumoutsakos[1]

[1]Chair of Computational Science, ETH Zurich, Clausiusstrasse 33, Zurich, CH-8092, Switzerland
[2]Department of Mechanical Engineering, Massachussetts Institute of Technology, 77 Massachusetts Ave., Cambridge, MA 02139, United States

We introduce a data-driven forecasting method for high dimensional, chaotic systems using Long-Short Term Memory (LSTM) recurrent neural networks. The proposed LSTM neural networks perform inference of high dimensional dynamical systems in their reduced order space and are shown to be an effective set of non-linear approximators of their attractor. We demonstrate the forecasting performance of the LSTM and compare it with Gaussian processes (GPs) in time series obtained from the Lorenz 96 system, the Kuramoto-Sivashinsky equation and a prototype climate model. The LSTM networks outperform the GPs in short-term forecasting accuracy in all applications considered. A hybrid architecture, extending the LSTM with a mean stochastic model (MSM-LSTM), is proposed to ensure convergence to the invariant measure. This novel hybrid method is fully data-driven and extends the forecasting capabilities of LSTM networks.

THE ROYAL SOCIETY PUBLISHING

1. Introduction

Natural systems, ranging from atmospheric climate and ocean circulation to organisms and cells, involve complex dynamics extending over multiple spatio-temporal scales. Centuries old efforts to comprehend and forecast the dynamics of such systems have spurred developments in large scale simulations, dimensionality reduction techniques and a multitude of forecasting methods. The goals of understanding and prediction have been complementing each other but have been hindered by the high dimensionality and chaotic behavior of these systems. In recent years we observe a convergence of these approaches due to advances in computing power, algorithmic innovations and the ample availability of data. A major beneficiary of this convergence are data-driven dimensionality reduction methods [2–7], model identification procedures [9–13] and forecasting techniques [14–19] that aim to provide precise short term predictions while capturing the long term statistics of these systems. Successful forecasting methods address the highly non-linear energy transfer mechanisms between modes not captured effectively by the dimensionality reduction methods.

The pioneering technique of analog forecasting proposed in [20] inspired a widespread research in non-parametric prediction approaches. Two dynamical system states are called analogues if they resemble one another on the basis of a specific criterion. This class of methods uses a training set of historical observations of the system. The system evolution is predicted using the evolution of the closest analogue from the training set corrected by an error term. This approach has led to promising results in practice [21] but the selection of the resemblance criterion to pick the optimal analogue is far from straightforward. Moreover, the geometrical association between the current state and the training set is not exploited. More recently [22], analog forecasting is performed using a weighted combination of data-points based on a localized kernel that quantifies the similarity of the new point and the weighted combination. This technique exploits the local geometry instead of selecting a single optimal analogue. Similar kernel-based methods, [23,24] use diffusion maps to globally parametrize a low dimensional manifold capturing the slower time scales. Moreover, non-trivial interpolation schemes are investigated in order to encode the system dynamics in this reduced order space as well as map them to the full space (lifting). Although the geometrical structure of the data is taken into account, the solution of an eigen-system with a size proportional to the training data is required, rendering the approach computationally expensive. In addition, the inherent uncertainty due to sparse observations in certain regions of the attractor introduces prediction errors which cannot be modeled in a deterministic context. In [25] a method based on Gaussian process regression (GPR) [26] was proposed for prediction and uncertainty quantification in the reduced order space. The technique is based on a training set that sparsely samples the attractor. Stochastic predictions exploit the geometrical relationship between the current state and the training set, assuming a Gaussian prior over the modeled latent variables. A key advantage of GPR is that uncertainty bounds can be analytically derived from the hyper-parameters of the framework. Moreover, in [25] a Mean Stochastic Model (MSM) is used for under-sampled regions of the attractor to ensure accurate modeling of the steady state in the long term regime. However the resulting inference and training have a quadratic cost in terms of the number of data samples $O(N^2)$. Some of the earlier approaches to capture the evolution of time series in chaotic systems using recurrent neural networks were developed during the inception of the Long-Short Term Memory networks (LSTM) [27]. However, to the best of our knowledge, these methods have been used only on low-dimensional chaotic systems [34]. Similarly, other machine learning algorithms such as Echo State Networks [36,37] and radial basis functions [38,39] have been successful, albeit only for low order dynamical systems.

In this work, we propose LSTM based methods that exploit information of the recent history of the reduced order state to predict the high-dimensional dynamics. Time-series data are used to train the model while no knowledge of the underlying system equations is required. Inspired by Taken's theorem [40] an embedding space is constructed using time delayed versions of the

reduced order variable. The proposed method tries to identify an approximate forecasting rule globally for the reduced order space. In contrast to GPR [25], the method has a deterministic output while its training cost scales linearly with the number of training samples and it exhibits an $\mathcal{O}(\infty)$ inference computational cost. Moreover, following [25], LSTM is combined with a MSM, to cope with attractor regions that are not captured in the training set. In attractor regions, under-represented in the training set, the MSM is used to guarantee convergence to the invariant measure and avoid an exponential growth of the prediction error. The effectiveness of the proposed hybrid method in accurate short term prediction and capturing the long-term behavior is shown in the Lorenz 96 system and the Kuramoto-Sivashisky system. Finally the method is also tested on predictions of a prototypical climate model.

The structure of the paper is as follows: In Section 2 we explain how the LSTM can be employed for modeling and prediction of a reference dynamical system and a blended LSTM-MSM technique is introduced. In Section 3 three other state of the art methods, GPR, MSM and the hybrid GPR-MSM scheme are presented and two comparison metrics are defined. The proposed LSTM technique and its LST-MSM extension are benchmarked in three complex chaotic systems in Section 4. In Section 5 we discuss the computational complexity of training and inference in LSTM. Finally, Section 6 offers a summary and discusses future research directions.

2. Long-Short Term Memory (LSTM) Recurrent Neural Networks

The LSTM was introduced in order to regularize the training of recurrent neural networks (RNNs) [27]. RNNs contain loops that allow information to be passed between consecutive temporal steps (see Figure 1) and can be expressed as:

$$\mathbf{h}_t = \sigma_h\big(\mathbf{W}_{hi}\mathbf{i}_t + \mathbf{W}_{hh}\mathbf{h}_{t-1} + b_h\big), \tag{2.1}$$

$$\mathbf{o}_t = \sigma_o\big(\mathbf{W}_{oh}\mathbf{h}_t + b_o\big) \tag{2.2}$$

where $\mathbf{i}_t$, $\mathbf{o}_t$ and $\mathbf{h}_t$ are the input, the output and the hidden state of the RNN at time step t, while D represents a delay block and $\mathbf{W}_{hi}$, $\mathbf{W}_{hh}$, $\mathbf{W}_{oh}$ are the input-to-hidden, hidden-to-hidden and hidden-to-output weight matrices. Moreover, σ_h and σ_o are the hidden and output activation functions, while b_h and b_o are the respective biases. Temporal dependencies are captured by the hidden-to-hidden weight matrix $\mathbf{W}_{hh}$, which couples two consecutive hidden states together. The RNN can be viewed in its unfolded form in Figure 2. In many practical applications, RNNs

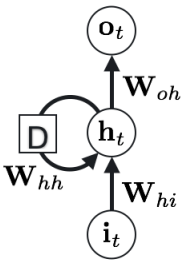

Figure 1: RNN

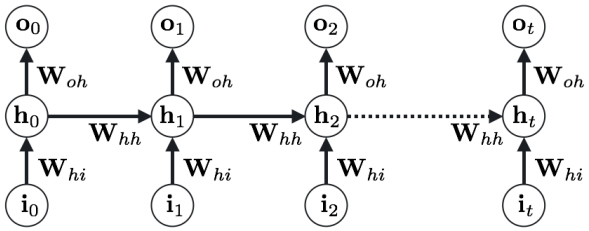

Figure 2: RNN unfolded in time

suffer from the vanishing (or exploding) gradient problem and have failed to capture long term dependencies [41,42]. Today the RNNs owe their renaissance largely to the LSTM, that copes effectively with the aforementioned problem using *gates*. The LSTM has been successfully applied in sequence modeling [32], speech recognition [28–30], hand-writing recognition [31] and language translation [33].

The equations of the LSTM are

$$g_t^f = \sigma_f\big(\mathbf{W}_f[\mathbf{h}_{t-1}, \mathbf{i}_t] + b_f\big) \tag{2.3}$$

$$g_t^i = \sigma_i\big(\mathbf{W}_i[\mathbf{h}_{t-1}, \mathbf{i}_t] + b_i\big) \tag{2.4}$$

$$\tilde{C}_t = tanh\big(\mathbf{W}_C[\mathbf{h}_{t-1}, \mathbf{i}_t] + b_C\big) \tag{2.5}$$

$$C_t = g_t^f C_{t-1} + g_t^i \tilde{C}_t \tag{2.6}$$

$$g_t^o = \sigma_h\big(\mathbf{W}_h[\mathbf{h}_{t-1}, \mathbf{i}_t] + b_h\big) \tag{2.7}$$

$$\mathbf{h}_t = g_t^o\, tanh(C_t), \tag{2.8}$$

where g_t^f, g_t^i and g_t^o are the gate signals (forget, input and output gates), $\mathbf{i}_t$ is the input, $\mathbf{h}_t$ is the hidden state, $\mathbf{C}_t$ is the cell state, while $\mathbf{W}_f$, b_f, $\mathbf{W}_i$, b_i, $\mathbf{W}_C$, b_C, $\mathbf{W}_h$ and b_h are weight matrices and biases of appropriate dimensions. The activation functions σ_f, σ_i and σ_h are sigmoids. For a more detailed explanation on the LSTM architecture refer to [27]. The hidden state $\mathbf{h}_t \in \mathbb{R}^h$, with h the number of hidden units. In practice we want the output to have a specific dimension d_o. For this reason, a trivial fully connected final layer without activation function is added

$$\mathbf{o}_t = \mathbf{W}_{oh}\mathbf{h}_t, \tag{2.9}$$

with $\mathbf{W}_{oh} \in \mathbb{R}^{d_o \times h}$. In the following we refer to the LSTM hidden and cell states ($\mathbf{h}_t$ and C_t) jointly as *LSTM states*.

In this work, we consider the reduced order problem where the system state is projected in the reduced order space. Moreover, the system is considered to be autonomous, while $\Delta z_t = \frac{dz_t}{dt}$ is the system state derivative at time step t. The LSTM model is trained using time series data from the system to predict the state derivative $\mathbf{o}_t \hat{=} \Delta z_t = \frac{dz_t}{dt}$ at time t, using delayed versions of the reference reduced model state z_t. It is a solely data-driven approach and no explicit information regarding the form of the underlying equations is required.

(a) Training and inference

The available time series data are divided into two separate sets, the training dataset and the validation dataset, i.e. z_t^{train}, Δz_t^{train}, $t \in \{1, \cdots, N_{train}\}$, and z_t^{val}, Δz_t^{val}, $t \in \{1, \cdots, N_{val}\}$. N_{train} and N_{val} are the number of training and validation samples respectively. This data is stacked in batches as

$$\underbrace{\mathbf{I}_t^{train} = \begin{pmatrix} z_{t+d-1}^{train} \\ z_{t+d-2}^{train} \\ \vdots \\ z_t^{train} \end{pmatrix}}_{\text{Input batch}}, \quad \underbrace{\mathbf{o}_t^{train} = \Delta z_{t+d-1}^{train}}_{\text{Output batch}}, \tag{2.10}$$

for $t \in \{1, 2, \ldots, N_{train} - d + 1\}$, in order to form the training (and validation) input and output of the LSTM. These training batches are used to optimize the parameters of the LSTM (weights and biases) in order to learn the mapping $\mathbf{I}_t \to \mathbf{o}_t$.

The training proceeds by optimising the network weights iteratively for each batch (*training of one epoch*). The training loss function is a weighted version of the root mean square error, i.e. $loss = \sqrt{\frac{1}{d_o}\sum_{i=1}^{d_o} w_i \left(\mathbf{o}_t^{train,\,i} - \mathbf{o}_t^i\right)^2}$ where d_o is the dimension of the output of the LSTM, and the weights w_i are selected according to the significance of each output component, e.g. energy of each component. Moreover, the LSTM is trained using truncated Back-propagation Through Time (BPTT) [35]. The BPTT is truncated after layer d. As a consequence, the LSTM is trained to predict the derivative at time t using information from the previous d time steps.

An important issue is how to select the hidden state dimension h and how to initialize the *LSTM states* at the truncation layer d. A small h reduces the expressive capabilities of the LSTM

and deteriorates inference performance. On the other hand, a big h leads to fast overfitting, an upturn in the generalization error and increased computational cost of training. For this reason, h has to be tuned depending on the observed data (training and validation). For the truncation layer d, there are two alternatives, namely *stateless* and *statefull* LSTM. In *stateless* LSTM the *LSTM states* at layer d are initialized to zero. As a consequence, the LSTM can only capture dependencies up to d previous time steps. In the second variant, the *statefull* LSTM, the state is always propagated for p time steps in the future and then reinitialized to zero, to help the LSTM capture longer dependencies. In this work, the systems considered exhibit chaotic behavior and the dependencies are inherently short term, as the states in two time steps that differ significantly can be considered statistically independent. For this reason, the short temporal dependencies can be captured without propagating the hidden state for a long horizon. As a consequence, we consider only the *stateless* variant $p = 0$. We also applied *statefull* LSTM without any significant improvement so we omit the results for brevity. Optimization during training is performed using the Adam stochastic optimization method [1] with an adaptive learning rate (initial learning rate $\eta = 0.0001$). Training is stopped when convergence of the training error is detected or the maximum or 100 epochs is reached. The LSTM model with the smallest validation error is considered to avoid over-fitting.

The trained LSTM model can be used to forecast the system state in the next time steps in an iterative fashion. The history of the system up to time step d, i.e. $z_1^{true}, \ldots, z_d^{true}$, is assumed to be known. We initialize the *LSTM states* with h_0 and C_0 and we use the trained LSTM to predict the derivative Δz_d^{pred}. By integrating the derivative with a reference time difference dt and initial condition z_d^{true} the value z_{d+1}^{pred} is obtained. This value is used for the next prediction in an iterative fashion as illustrated in Figure 3. In *statefull* LSTM, initial values for h_0 and C_0 can be obtained by *teacher forcing* the LSTM for a few time steps propagating values from the known history and ignoring the outputs. In *stateless* LSTM, h_0 and C_0 are initialized with zero vectors.

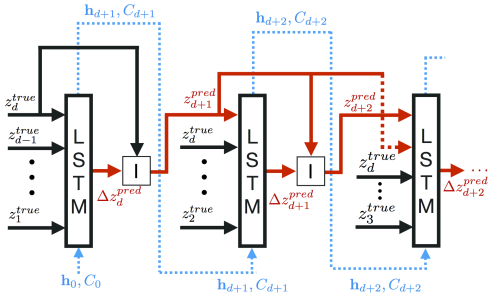

Figure 3: Iterative prediction using LSTM

(b) Mean Stochastic Model (MSM) and Hybrid LSTM-MSM

The MSM is a powerful data-driven method used to quantify uncertainty and perform forecasts in turbulent systems with high intrinsic attractor dimensionality [25,43]. It is parametrized a priori to capture global statistical information of the attractor by design, while its computationally complexity is very low compared to LSTM or GPR. The concept behind MSM is to model each component of the state z^i independently with an Ornstein-Uhnelbeck (OU) process that captures the energy spectrum and the damping time scales of the statistical equilibrium. The process takes the following form

$$dz^i = c_i z^i dz + \xi_i dW_i, \tag{2.11}$$

where c_i, ξ_i are parameters fitted to the centered training data and W_i is a wiener process. In the statistical steady state the mean, energy and damping time scale of the process are given by

$$\mu_i = \mathbb{E}[z^i] = 0, \quad E_i = \mathbb{E}[z^i(z^i)*] = -\frac{\xi^2}{2c_i}, \quad T_i = -\frac{1}{c_i}. \tag{2.12}$$

In order to fit the model parameters c_i, ξ_i we directly estimate the variance $\mathbb{E}[z^i(z^i)^*]$ from the time series training data and the decorrelation time using

$$T_i = \frac{1}{\mathbb{E}[z^i(z^i)^*]} \int_0^\infty \mathbb{E}[z^i(t)(z^i)^*(t+\tau)d\tau. \tag{2.13}$$

After computing these two quantities we replace in (2.12) and solve with respect to c_i and ξ_i. Since the MSM is modelled a priori to mimic the global statistical behavior of the attractor, forecasts made with MSM can never escape. This is not the case with LSTM and GPR, as prediction errors accumulate and iterative forecasts escape the attractor fast due to the chaotic dynamics, although short term predictions are accurate. This problem has been addressed with respect to GPR in [25]. In order to cope effectively with this problem we introduce a hybrid LSTM-MSM technique that prevents forecasts from diverging from the attractor.

The state dependent decision rule for forecasting in LSTM-MSM is given by

$$\Delta z_t = \begin{cases} (\Delta z_t)_{LSTM}, & \text{if } p^{train}(z_t) = \prod p_i^{train}(z_t^i) > \delta \\ (\Delta z_t)_{MSM}, & \text{otherwise} \end{cases} \tag{2.14}$$

where $p^{train}(z_t)$ is an approximation of the probability density function of the training dataset and $\delta \approx 0.01$ a constant threshold tuned based on $p^{train}(z_t)$. We approximate $p^{train}(z_t)$ using a mixture of Gaussian kernels. This hybrid architecture exploits the advantages of LSTM and MSM. In case there is a high probability that the state z_i lies close to the training dataset (interpolation) the LSTM having memorized the local dynamics is used to perform inference. This ensures accurate LSTM short-term predictions. On the other hand, close to the boundaries the attractor is only sparsely sampled $p^{train}(z_i) < \delta$ and errors from LSTM predictions would lead to divergence. In this case, MSM guarantees that forecasting trajectories remain close to the attractor, and that we converge to the statistical invariant measure in the long-term.

3. Benchmark and Performance Measures

The performance of the proposed LSTM based prediction mechanism is benchmarked against the following state-of-the-art methods:

- Mean Stochastic Model (MSM)
- Gaussian Process Regression (GPR)
- Mixed Model (GPR-MSM)

In order to guarantee that the prediction performance is independent of the initial condition selected, for all applications and all performance measures considered the average value of each measure for a number of different initial conditions sampled independently and uniformly from the attractor is reported. The ground truth trajectory is obtained by integrating the discretized reference equation starting from each initial condition, and projecting the states to the reduced order space. The reference equation and the projection method are of course application dependent.

From each initial condition, we generate an empirical Gaussian ensemble of dimension N_{en} around the initial condition with a small variance σ_{en}. This noise represents the uncertainty in the knowledge of the initial system state. We forecast the evolution of the ensemble by iteratively predicting the derivatives and integrating (deterministically for each ensemble member for the LSTM, stochastically for GPR) and we keep track of the mean. The ensemble size $N_{ensemble}$ is

selected in the order of ≈ 50, which is the usual choice in environmental science, e.g. weather prediction and short term climate prediction [44].

The ground truth trajectory at each time instant z is then compared with the predicted ensemble mean $\widetilde{z}$. As a comparison measure we use the root mean square error (RMSE) defined as $RMSE(z_k) = \sqrt{1/V \sum_{i=1}^{V} \left(z_k^i - \widetilde{z}_k^i\right)^2}$,where index k denotes the k^{th} component of the reduced order state z, i is the initial condition, and V is the total number of initial conditions. The RMSE is computed at each time instant for each component k of the reduced order state, resulting in error curves that describe the evolution of error with time.

Moreover, we use the mean Anomaly Correlation (AC) [47] over V initial conditions to quantify the pattern correlation of the predicted trajectories with the ground-truth. The AC is defined as

$$AC = \frac{1}{V} \sum_{i=1}^{V} \frac{\sum_{k=1}^{r_{dim}} w_k \left(z_k^i - \overline{z}_k\right)\left(\widetilde{z}_k^i - \overline{z}_k\right)}{\sqrt{\sum_{k=1}^{r_{dim}} w_k \left(z_k^i - \overline{z}_k\right)^2 \sum_{k=1}^{r_{dim}} w_k \left(\widetilde{z}_k^i - \overline{z}_k\right)^2}}, \tag{3.1}$$

where k refers to the mode number, i refers to the initial condition, w_k are mode weights selected according to the energies of the modes after dimensionality reduction and $\overline{z}_k$ is the time average of the respective mode, considered as reference. This score ranges from -1.0 to 1.0. If the forecast is perfect, the score equals to 1.0. The AC coefficient is a widely used forecasting accuracy score in the meteorological community [46].

4. Applications

In this section, the effectiveness of the proposed method is demonstrated with respect to three chaotic dynamical systems, exhibiting different levels of chaos, from weakly chaotic to fully turbulent, i.e. the Lorenz 96 system, the Kuramoto-Sivashinsky equation and a prototypical barotropic climate model.

(a) The Lorenz 96 System

In [45] a model of the large-scale behaviour of the mid-latitude atmosphere is introduced. This model describes the time evolution of the components X_j for $j \in \{0, 1, \ldots, J-1\}$ of a spatially discretized (over a single latitude circle) atmospheric variable. In the following we refer to this model as the Lorenz 96. The Lorenz 96 is usually used ([25,46] and references therein) as a toy problem to benchmark methods for weather prediction.

The system of differential equations that governs the Lorenz 96 is defined as

$$\frac{dX_j}{dt} = (X_{j+1} - X_{j-2})X_{j-1} - X_j + F, \tag{4.1}$$

for $j \in \{0, 1, \ldots, J-1\}$, where by definition $X_{-1} = X_J$, $X_{-2} = X_{J-1}$. In our analysis $J = 40$. The right-hand side of (4.1) consists of a non-liner adjective term $(X_{j+1} - X_{j-2})X_{j-1} - X_j$, a linear advection (dissipative) term $-X_j$ and a positive external forcing term F. The discrete energy of the system remains constant throughout time and the Lorenz 96 states X_j remain bounded. By increasing the external forcing parameter F the behavior that the system exhibits changes from periodic $F < 1$ to weakly chaotic ($F = 4$) to end up in fully turbulent regimes ($F = 16$). We refer to X_j as the states of the Lorenz 96 model. These regimes can be observed in Figures 4

Following [25,44] we apply a shifting and scaling to standardize the Lorenz 96 states X_j. The discrete or Dirichlet energy is given by $E = \frac{1}{2}\sum_{j=1}^{J} X_j^2$. In order for the scaled Lorenz 96 states to have zero mean and unit energy we transform them using

$$\tilde{X}_j = \frac{X_j - \overline{X}}{\sqrt{E_p}}, \quad d\tilde{t} = \sqrt{E_p}dt, \tag{4.2}$$

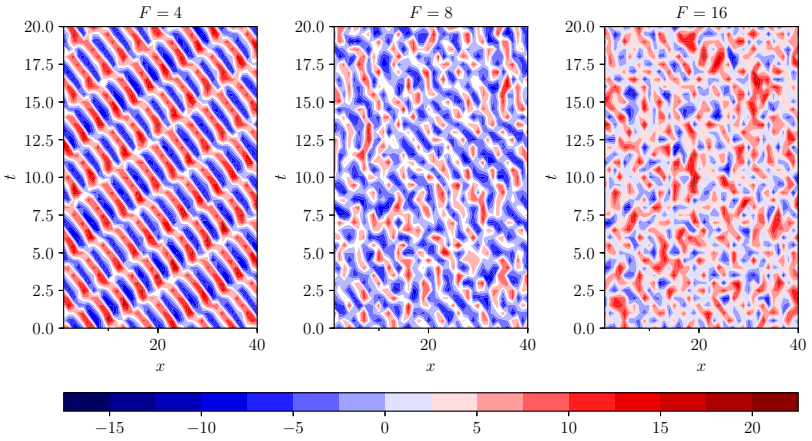

Figure 4: Lorenz 96 contour plots for different forcing regimes F. Chaoticity rises with bigger values of F.

where E_p is the average energy fluctuation, i.e.

$$E_p = \frac{1}{2T}\sum_{j=0}^{J-1}\int_{T_0}^{T_0+T}(X_j - \overline{X})^2 dt. \tag{4.3}$$

In this way the scaled energy is $\tilde{E} = \frac{1}{2}\sum_{j=0}^{J-1}\tilde{X}_j^2 = 1$ and the scaled variables have zero mean $\overline{\tilde{X}} = \frac{1}{J}\sum_{j=0}^{J-1}\tilde{X}_j = 0$, with $\overline{X}$ the mean state. The scaled Lorenz 96 states $\tilde{X}_j$ obey the following differential equation

$$\begin{aligned}\frac{d\tilde{X}_j}{d\tilde{t}} = \frac{F-\overline{X}}{E_p} + \frac{(\tilde{X}_{j+1}-\tilde{X}_{j-2})\overline{X} - \tilde{X}_j}{\sqrt{E_p}} + \\ + (\tilde{X}_{j+1}-\tilde{X}_{j-2})\tilde{X}_{j-1}\end{aligned} \tag{4.4}$$

(i) Dimensionality Reduction: Discrete Fourier Transform

Firstly, the Discrete Fourier Transform (DFT) is applied to the energy standardized Lorenz 96 states $\tilde{X}_j$. The Fourier coefficients $\hat{X}_k \in \mathbb{C}$ are given by

$$\hat{X}_k = \frac{1}{J}\sum_{j=0}^{J-1}\tilde{X}_j e^{-2\pi i k j/J} \tag{4.5}$$

while the Lorenz 96 states can be recovered from the Fourier coefficients using the inverse DFT

$$\tilde{X}_j = \sum_{k=0}^{J-1}\hat{X}_k e^{2\pi i k j/J} \tag{4.6}$$

After applying the DFT to the Lorenz 96 states we end up with a symmetric energy spectrum that can be uniquely characterized by $J/2+1$ (J is considered to be an even number) coefficients $\hat{X}_k$ for $k \in K = \{0, 1, \cdots, J/2\}$. In our case $J = 40$, thus we end up with $|K| = 21$ complex coefficients $\hat{X}_k \in \mathbb{C}$. These coefficients are referred to as the Fourier modes or simply modes. The Fourier energy of each mode is defined as

$$E_k = Var(\hat{X}_k) = \mathbb{E}\big[(\hat{X}_k(\tilde{t}) - \overline{\hat{X}}_k)(\hat{X}_k(\tilde{t}) - \overline{\hat{X}}_k)^*\big]. \tag{4.7}$$

The energy spectrum of the Lorenz 96 system is plotted in Figure 5 for different values of the forcing term F. We take into account only the $r_{dim} = 6$ modes corresponding to the highest

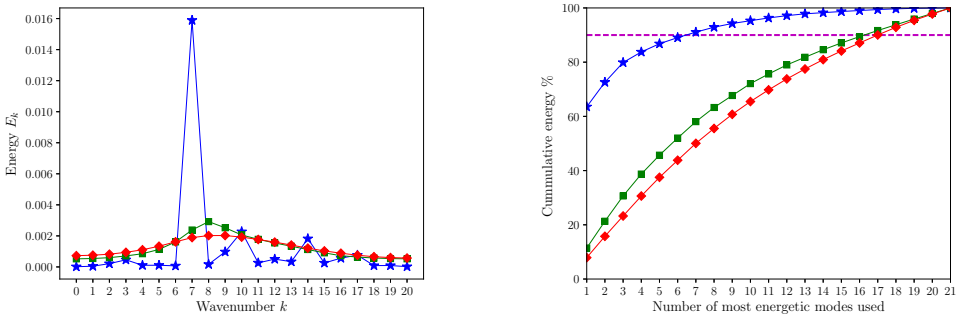

Figure 5: Energy spectrum E_k and cumulative energy with respect to the number of most energetic modes used for different forcing regimes of Lorenz 96 system. As the forcing increases, more chaoticity is introduced to the system.
$F = 4$ ——; $F = 8$——; $F = 16$ ——

Forcing	Wavenumbers k	Forcing	Wavenumbers k
$F = 4$	7,10,14,9,17,16	$F = 8$	8,9,7,10,11,6
$F = 6$	8,7,9,10,11,6	$F = 16$	8,9,10,7,11,6

Table 1: Most energetic Fourier modes used in the reduced order phase space

energies and the rest of the modes are truncated. For the different forcing regimes $F = 1, 2, 3, 4$, the six most energetic modes correspond to approximately 89%, 57.8%, 52% and 43.8% of the total energy respectively. The space where the reduced variables live in is referred to as the reduced order phase space and the most energetic modes are notated as $\hat{X}_k^r$ for $k \in \{0, 1, \ldots, r_{dim} - 1\}$. As shown in [48] the most energetic modes are not necessarily the ones that capture better the dynamics of the model. However, in this work we are not interested in an optimal reduced space representation, but rather in the effectiveness of a prediction model given this space. The respective wavenumbers of the most energetic modes as well as their energy are given in Table 1. The truncated modes are ignored for now. Nevertheless, their effect can be modelled stochastically as in [25].

Since each Fourier mode $\hat{X}_k^r$ is a complex number, it consists of a real part and an imaginary part. By stacking these real and imaginary parts of the r_{dim} truncated modes we end up with the $2\,r_{dim}$ dimensional reduced model state

$$\mathbf{X} \equiv [Re(\hat{X}_1^r), \ldots, Re(\hat{X}_{r_{dim}}^r), Im(\hat{X}_1^r), \ldots, Im(\hat{X}_{r_{dim}}^r)]^T \tag{4.8}$$

Assuming that X_j^t for $j \in \{0, 1, \ldots, J-1\}$ are the Lorenz 96 states at time instant t, the mapping X_j^t, $\forall j \rightarrow \mathbf{X}$ is unique and the reduced model state of the Lorenz 96 has a specific vector value. For high dimensions, Fourier Transform is equivalent to Principal Component Analysis.

(ii) Training and Prediction in Lorenz 96

The reduced Lorenz 96 system states $\mathbf{X}_t$ are considered as the true reference states z_t. The LSTM is trained to forecast the derivative of the reduced order state dz_t/dt as in [34]. In the following we analyze the influence of the truncation layer d and the number of hidden units h of the LSTM with respect to the chaotic Lorenz 96 system.

The influence of d in training and performance of the LSTM model is the following. On the one hand, selecting a large d makes the training more challenging, for two reasons. Firstly, the LSTM has more layers and secondly more noise might be included in the input (irrelevant information)

rendering suboptimal prediction performance. On the other hand, selecting a small d might lead to an input sequence with poor information content, leading to low prediction performance. Increasing the number of hidden nodes h rises the expressiveness of LSTM, but it is easier to overfit the training set. A *stateless LSTM* is used. The back-propagation truncation horizon is set to $d = 10$ and we use $h = 20$.

In order to obtain training data for the LSTM, we integrate the Lorenz 96 system state Eg. (4.1) starting from an initial condition X_j^0 for $j \in \{0, 1, \ldots, J-1\}$ using a Runge-Kutta 4th order method with a time step $dt = 0.01$ up to $T = 51$. In this way a time series X_j^t, $t \in \{0, 1, \cdots\}$ is constructed. Using the scaling and dimensionality reduction method explained in Section i we construct the reduced order state time series $\mathbf{X}_t$, $t \in \{0, 1, \cdots\}$, using the mapping $X_t^j \forall j \rightarrow \mathbf{X}_t$. From this time series we discard the first 10^4 initial time steps to avoid transients, ending up with a time series with $N^{train} = 50000$ samples. A similar but independent process is repeated for the validation set.

(iii) Results

The trained LSTM models are used for prediction based on the iterative procedure explained in Section 2. In this section, we demonstrate the forecasting capabilities of LSTM and compare it with the state of the art. 100 different initial conditions are simulated. For each initial condition, an ensemble with size $N_{en} = 50$ is considered by perturbing it with a normal noise with variance $\sigma_{en} = 0.0001$.

In Figures 6a, 6b, and 6c we report the mean RMSE prediction error of the most energetic mode $\hat{X}_1^r \in \mathbb{C}$, scaled with $\sqrt{E_p}$ for the forcing regimes $F \in \{6, 8, 16\}$ for the first $N = 10$ time steps ($T = 0.1$). In the RMSE the complex norm $||v||_2 = vv^*$ is taken into account. The 10% of the standard deviation of the attractor is also plotted for reference ($10\%\sigma$). As F increases, the system becomes more chaotic and difficult to predict. As a consequence, the number of prediction steps that remain under the $10\%\sigma$ threshold are decreased. The LSTM models extend this predictability horizon for all forcing regimes compared to GPR and MSM. However, when LSTM is combined with MSM the short term prediction performance is compromised. Nevertheless, hybrid LSTM-MSM models outperform GPR methods in short term prediction accuracy.

In Figures 6d, 6e, and 6f, the RMSE error for $T = 2$ is plotted. The standard deviation from the attractor σ is plotted for reference. We can observe the following

- The prediction performance of the LSTM in the quasi-periodic regime $F = 4$ is clearly superior to all other approaches. Blending LSTM with MSM guarantees accurate modeling of the steady state in the long term, but leads to a performance compromise in the short-term. LSTM-MSM outperforms GPR-MSM.
- In all forcing regimes, both GPR and LSTM eventually diverge, while MSM, and blended GPR-MSM, LSTM-MSM schemes remain close to the attractor in the long term as expected.
- For $F = 8$ although the RMSE error in the short-term is smaller for LSTM, GPR remains for a longer period close to the attractor (e.g. $T = 0.75$ for $F = 8$). However, when blended schemes are taken into account, LSTM-MSM shows superior performance in the short-term and slightly better performance in the long term compared to GPR-MSM.

In Figures 6g, 6h, and 6i, the mean AC over 1000 initial conditions is given. The predictability threshold of 0.6 is also plotted. After crossing this critical threshold, the methods do not predict better than a trivial mean predictor. For $F = 4$ GPR methods show inferior performance compared to LSTM approaches as analyzed previously in the RMSE comparison. However, for $F = 8$ LSTM models do not predict better than the mean after $T \approx 0.35$, while GPR shows better performance. In turn, when blended with MSM the compromise in the performance for GPR-MSM is much bigger compared to LSTM-MSM. The LSTM-MSM scheme shows slightly superior performance than GPR-MSM during the entire relevant time period ($AC > 0.6$). For the fully

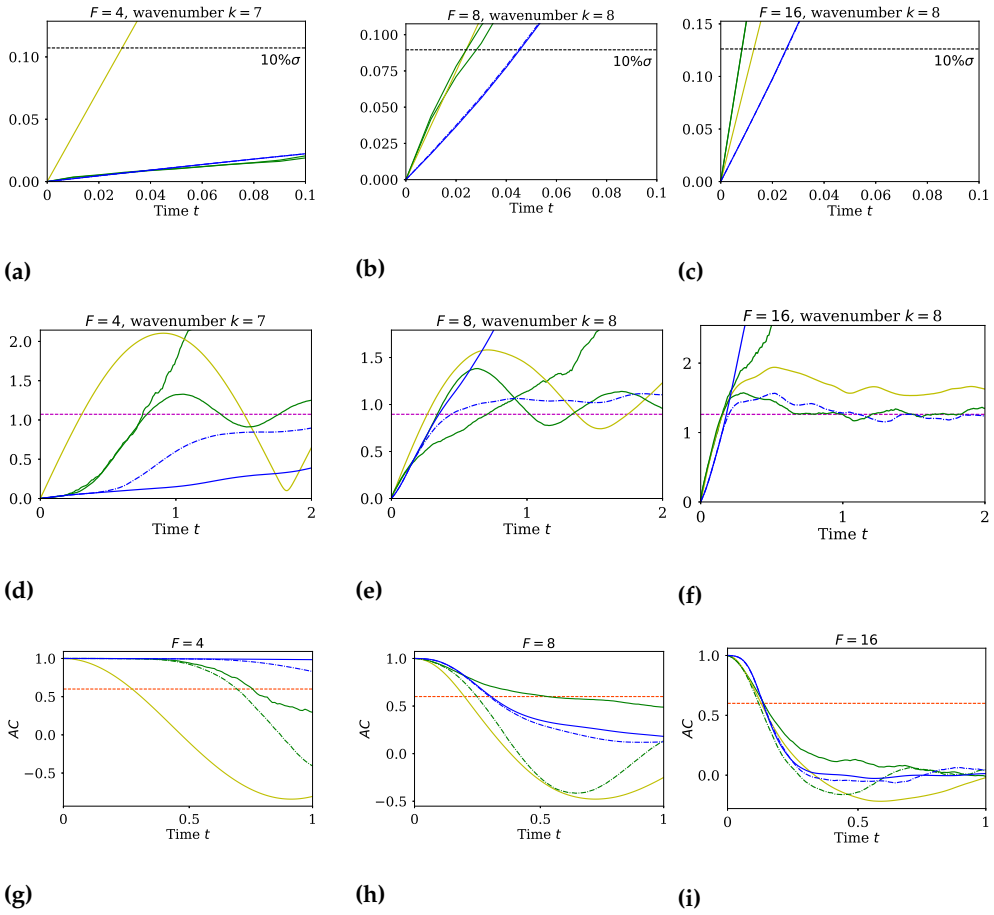

Figure 6: Mean RMSE of the most energetic mode and mean AC over 1000 initial conditions for the Lorenz 96 system. 10% of the standart deviation from the atractor - - -; Standart deviation from the atractor- - -; AC predictability threshold- - -; MSM——; GPR——; GPR-MSM- - -; LSTM——; LSTM-MSM- - -

turbulent regime $F = 16$, LSTM shows comparable performance with both GPR and MSM and all methods converge as chaoticity rises, since the intrinsic dimensionality of the system attractor increases and the system become inherently unpredictable.

In Figure 7, the evolution of the mean RMSE over 1000 initial conditions of the wavenumbers $k = 8, 9, 10, 11$ of the Lorenz 96 with forcing $F = 8$ is plotted. In contrast to GPR, the RMSE error of LSTM is much lower in the moderate and low energy wavenumbers $k = 9, 10, 11$ compared to the most energetic mode $k = 8$. This difference among modes is not observed in GPR. This can be attributed to the highly non-linear energy transfer mechanisms between these lower energy modes as opposed to the Gaussian and locally linear energy transfers of the most energetic mode.

As illustrated before, the hybrid LSTM-MSM architecture effectively combines the accurate short-term prediction performance of LSTM with the long-term stability of MSM. The percentage of ensemble members in the hybrid scheme explained by LSTM is plotted with respect to time in Figure 8. In parallel with the GPR results presented in [25], the slope of the percentage drop increases with F up to time $t \approx 1.5$. However, in contrast to the results from GPR reported in [25], LSTM shows a more stable behavior as a bigger percentage of the ensembles is explained by it

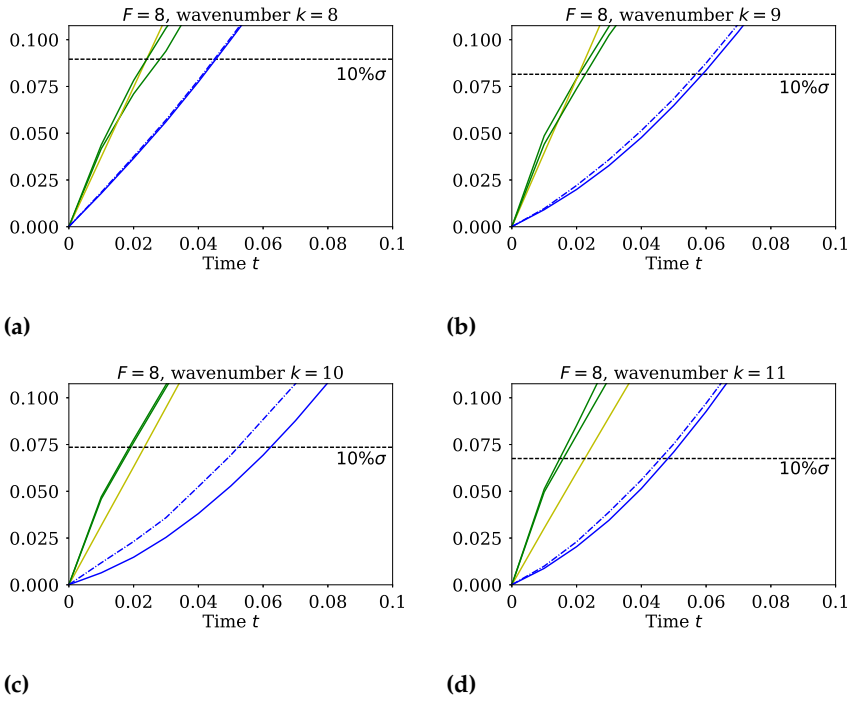

Figure 7: Mean RMSE of the most energetic mode ($k = 8$) and medium and low energy modes ($k = 9, 10, 11$) over 1000 initial conditions for the Lorenz 96 system with forcing $F = 8$. 10% of the standart deviation from the atractor - - -; MSM——; GPR——; GPR-MSM- - -; LSTM——; LSTM-MSM- - -

compared to GPR in general. This is because LSTM is a local nonlinear attractor approximator and can better capture the mean local dynamics, while GPR is locally linear.

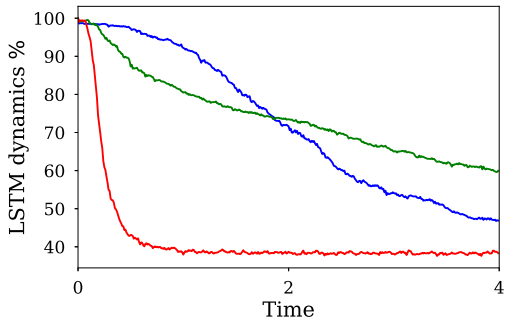

Figure 8: Average percentage over 500 initial conditions of the ensemble members evaluated using LSTM dynamics over time for different Lorenz 96 forcing regimes in the hybrid LSTM-MSM method. $F = 4$ ——; $F = 8$——; $F = 16$ ——

(b) Kuramoto-Sivashinsky Equation

The Kuramoto-sivashinsky (K-S) system is extensively used in many scientific fields to model a multitude of chaotic physical phenomena. It was first derived by Kuramoto [49,50] as a turbulence model of the phase gradient of a slowly varying amplitude in a reaction-diffusion type medium with negative viscosity coefficient. Later, Sivashinsky [51] studied the spontaneous instabilities of the plane front of a laminar flame ending up with the K-S equation, while in [52] the K-S equation is found to describe the surface behavior of viscous liquid in a vertical flow.

For our study, we restrict ourselves to the one dimensional K-S equation with boundary and initial conditions given by

$$\begin{aligned} &\frac{\partial u}{\partial t} = -\nu \frac{\partial^4 u}{\partial x^4} - \frac{\partial^2 u}{\partial x^2} - u\frac{\partial u}{\partial x}, \\ &u(0,t) = u(L,t) = \left.\frac{\partial u}{\partial x}\right|_{x=0} = \left.\frac{\partial u}{\partial x}\right|_{x=L} = 0, \\ &u(x,0) = u_0(x), \end{aligned} \tag{4.9}$$

where $u(x,t)$ is the modeled quantity of interest depending on a spatial variable $x \in [0, L]$ and time $t \in [0, \infty]$. The negative viscosity is modeled by the parameter $\nu > 0$. We impose Dirichlet and second-type boundary conditions to guarantee ergodicity [53]. In order to spatially discretize (4.9) we use a grid size Δx with $D = L/\Delta x$ the number of nodes. Further, we denote with $u_i = u(i\Delta x)$ the value of u at node $i \in \{0, \dots, D\}$. Discretization using a second order finite differences scheme yields

$$\begin{aligned} \frac{du_i}{dt} = &-\nu \frac{u_{i-2} - 4u_{i-1} + 6u_i - 4u_{i+1} + u_{i+2}}{\Delta x^4} \\ &- \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2} - \frac{u_{i+1}^2 - u_{i-1}^2}{4\Delta x}. \end{aligned} \tag{4.10}$$

Further, we impose $u_0 = u_{D+1} = 0$ and add ghost nodes $u_{-1} = u_1$, $u_{D+2} = u_D$ to account for the Dirichlet and second-order boundary conditions. In our analysis, the number of nodes is $D = 512$. The Kuramoto-Sivashinsky equation exhibits different levels of chaos depending on the bifurcation parameter $\tilde{L} = L/2\pi\sqrt{\nu}$ [54]. Higher values of $\tilde{L}$ lead to more chaotic systems [25].

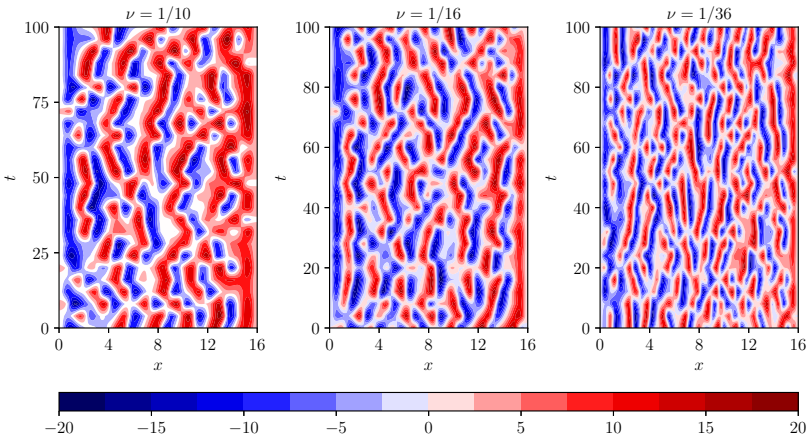

Figure 9: Contour plots of $u(x,t)$ for different values of ν in steady state. Chaoticity rises with smaller values of ν.

In our analysis the spatial variable bound is held constant to $L = 16$ and chaoticity level is controlled through the negative viscosity ν, where a smaller value leads to a system with a

higher level of chaos (see Figure 9). The temporal average of the state and the cumulative energy are plotted in Figure 10. As ν declines, chaoticity in the system rises and higher oscillations of the mean towards the Dirichlet boundary conditions are observed, while the number of modes needed to capture most of the energy is higher. In our study, we consider two values, namely $\nu = 1/10$ and $\nu = 1/16$ to benchmark the prediction skills of the proposed method. The discretized equation (4.10) is integrated with a time interval $dt = 0.02$ up to $T = 11000$. The data points up to $T = 1000$ are discarded as initial transients. Half of the remaining data ($N = 250000$ samples) are used for training and the other half for validation.

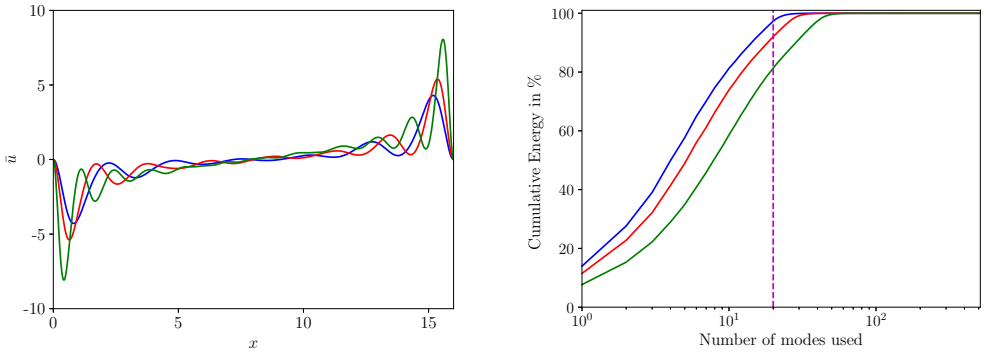

Figure 10: Temporal average $\overline{\mathbf{u}}$ and cumulative mode (PCA) energy for different values of ν. $1/\nu = 10$ —; $1/\nu = 16$ —; $1/\nu = 36$ —

(i) Dimensionality Reduction: Singular Value Decomposition

The dimensionality of the problem is reduced using Singular Value Decomposition (SVD). By subtracting the temporal mean $\overline{\mathbf{u}}$ and stacking the data, we end up with the data matrix $\mathbf{U} \in \mathbb{R}^{N\times 513}$, where N is the number of data samples ($N = 500000$ in our case). Performing SVD on $\mathbf{U}$ leads to

$$\mathbf{U} = \mathbf{M}\Sigma\mathbf{V}^T, \quad \mathbf{M} \in \mathbb{R}^{N\times N}, \quad \Sigma \in \mathbb{R}^{N\times 513}, \quad \mathbf{V} \in \mathbb{R}^{513\times 513}, \tag{4.11}$$

with Σ diagonal, with descending diagonal elements. The right singular vectors corresponding to the r_{dim} largest singular values are the first columns of $\mathbf{V} = [\mathbf{V_r}, \mathbf{V_{-r}}]$. Stacking these singular vectors yields $\mathbf{V_r} \in \mathbb{R}^{513\times r_{dim}}$. Assuming that $\mathbf{u}_t \in \mathbb{R}^{513}$ is a vector of the discretized values of $u(x,t)$ in time t, in order to get a reduced order representation corresponding to the components with the highest energies (singular values) we multiply

$$\mathbf{c} = \mathbf{V_r}^T\mathbf{u}, \quad \mathbf{c} \in \mathbb{R}^{r_{dim}}. \tag{4.12}$$

Applying SVD on the data matrix $\mathbf{U}$ is equivalent with Principal Component Analysis on the covariance matrix as in [25]. The percentage of cumulative energy w.r.t. to the number of components (modes) considered is plotted in Figure 10. Further, the 90% threshold is plotted. In our study, we pick $r_{dim} = 20$ (out of 512) most energetic modes, as they explain approximately 90% of the total energy. The reduced model state is then given by:

$$\mathbf{c} \equiv [c_1, \ldots, c_{r_{dim}}]^T. \tag{4.13}$$

(ii) Results

We train *stateless* LSTM models with $h = 100$ and $d = 50$. For testing, starting from 1000 initial conditions uniformly sampled from the attractor, we generate a Gaussian ensemble of dimension $N = 50$ centered around the initial condition in the original space with standard deviation of

$\sigma = 0.1$. This ensemble is propagated using the LSTM prediction models, and GPR, MSM and GPR-MSM models trained as in [25]. The root mean square error between the predicted ensemble mean and the ground-truth is plotted in Figures 11a, 11b for different values of the parameter ν. All methods reach the invariant measure much faster for $1/\nu = 16$ compared to the less chaotic regime $1/\nu = 10$ (note the different integration times $T = 4$ for $1/\nu = 10$, while $T = 1.5$ for $1/\nu = 16$).

In both chaotic regimes $1/\nu = 10$ and $1/\nu = 16$, the reduced order LSTM outperforms all other methods in the short term before escaping the attractor. However, in the long term, LSTM does not stabilize and will eventually diverge faster than GPR (see Figure 11b). Blending LSTM with MSM alleviates the problem and both accurate short term predictions and long term stability is attained. Moreover, the hybrid LSTM-MSM has better forecasting capabilities compared to GPR.

The need for blending LSTM with MSM in the KS equation is less imperative as the system is less chaotic than the Lorenz 96 and LSTM methods diverge much slower, while they sufficiently capture the complex nonlinear dynamics. As the intrinsic dimensionality of the attractor rises LSTM diverges faster.

The mean Anomaly Correlation (3.1) is plotted with respect to time in Figures 11c and 11d for $\nu = 10$ and 16 respectively. The evolution of the AC justifies the aforementioned analysis. The mean AC of the trajectory predicted with LSTM remains above the predictability threshold of 0.6 for a highest time duration compared to other methods. This predictability horizon is approximately 2.5 for $\nu = 1/10$ and 0.6 for $\nu = 1/16$, since the chaoticity of the system rises and accurate predictions become more challenging.

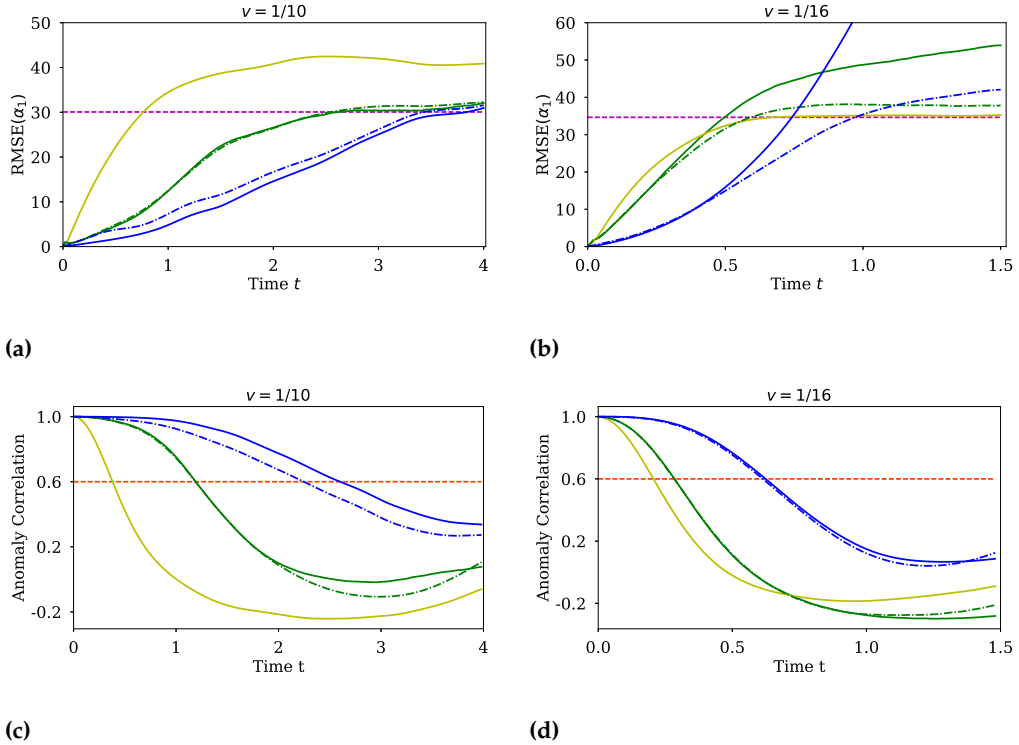

Figure 11: Mean RMSE of the most energetic mode and mean AC over 1000 initial conditions for the K-S equation with $1/\nu = 10$ (11a,11c) and $1/\nu = 16$ (11c,11d). Standard deviation from the attractor- - -; AC predictability threshold- - -; MSM——; GPR——; GPR-MSM- - -; LSTM——; LSTM-MSM- - -

For the hybrid LSTM-MSM, the percentage of the ensemble members that are explained by LSTM dynamics is plotted in Figure 12. The quotient drops slower for $1/\nu = 10$ in the long run as the intrinsic dimensionality of the attractor is smaller and trajectories diverge slower. However, in the beginning the LSTM percentage is higher for $1/\nu = 16$ as the MSM drives initial conditions close to the boundary faster towards the attractor due to the higher damping coefficients compared to the case $1/\nu = 10$. This explains the initial knick in the graph for $1/\nu = 16$. The slow damping coefficients for $1/\nu = 10$ do not allow the MSM to drive the trajectories back to the attractor in a faster pace than the diffusion caused by the LSTM forecasting.

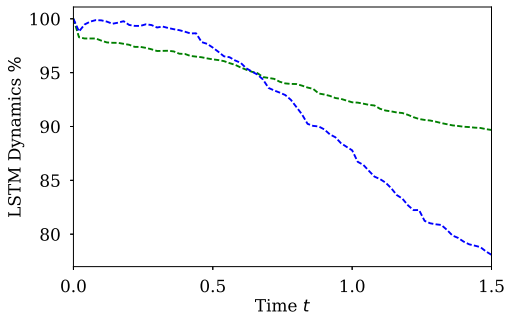

Figure 12: Mean over 1000 initial conditions of the percentage of ensemble members explained by the LSTM dynamics for the Kuramoto-Sivashinsky ($T = 1.5$)
$1/\nu = 10$ ——; $1/\nu = 16$ ——

(c) A Barotropic Climate Model

In this section, we examine a standard barotropic climate model [55] originating from a realistic winter circulation. The model equations are given by

$$\frac{\partial \zeta}{\partial t} = -\mathcal{J}(\psi, \zeta + f + h) + k_1\zeta + k_2\delta^3\zeta + \zeta^*, \tag{4.14}$$

where ψ is the streamfunction, $\zeta = \delta\psi$ the relative vorticity, f the Coriolis parameter, ζ^* a constant vorticity forcing, while k_1 and k_2 are the Ekman damping and the scale-selective damping coefficient. $\mathcal{J}$ is the Jacobi operator given by

$$\mathcal{J}(a, b) = \Big(\frac{\partial a}{\partial \lambda}\frac{\partial B}{\partial \mu} - \frac{\partial a}{\partial \mu}\frac{\partial B}{\partial \lambda}\Big), \tag{4.15}$$

where μ and λ denote the sine of the geographical latitude and longitude respectively. The equation of the barotropic model (4.14) is non-dimensionalized using the radius of the earth as unit length and the inverse of the earth angular velocity as time unit. The non-dimensional orography h is related to the real Northern Hemisphere orography $h^{'}$ by $h = 2sin(\phi_0)A_0h^{'}/H$, where phi_0 is a fixed amplitude of $45^{\circ}N$, A_0 is a factor expressing the surface wind strength blowing across the orography, and H a scale height [55]. The stream-function ψ is expanded into a spherical harmonics series and truncated at wavenumber 21, while modes with an even total wavenumber are excluded, avoiding currents across the equator and ending up with a hemispheric model with 231 degrees of freedom.

The training data are obtained by integrating the Eq. (4.14) for 10^5 days after an initial spin-up period of 1000 days, using a fourth-order Adams-Bashforth integration scheme with a 45-min time step in accordance with [25], with $k_1 = 15$ days, while k_2 is selected such that wavenumber 21 is damped at a time scale of 3 days. In this way we end up with a time series ζ_t with 10^4 samples. The spherical surface is discretized into a $D = 64 \times 32$ mesh with equally spaces latitude and longitude. From the gathered data, 90% is used for training and 10% for validation. The mean and variance of the statistical steady state are shown in Figure 13a.

(i) Dimensionality Reduction: Classical Multidimensional Scaling

The original problem dimension of 231 is reduced using a generalized version of the classical multidimensional scaling method [56]. The procedure tries to identify an embedding with a lower dimensionality such that the pairwise inner products of the dataset are preserved. Assuming that the dataset consists of points ζ_i, $i \in \{1, \ldots, N\}$, whose reduced order representation is denoted with $\mathbf{y}_i$, the procedure is equivalent with the solution of the following optimization problem

$$\underset{\mathbf{y}_1,\ldots,\mathbf{y}_N}{\text{minimize}} \sum_{i<j} \big(\langle \zeta_i, \zeta_j\rangle_\zeta - \langle \mathbf{y}_i, \mathbf{y}_j\rangle_\mathbf{y}\big)^2, \tag{4.16}$$

where $\langle\cdot,\cdot\rangle_\zeta$, and $\langle\cdot,\cdot\rangle_\mathbf{y}$ denote some well defined inner product of the original space ζ and the embedding space $\mathbf{y}$ respectively. Problem (4.16) minimizes the total squared error between pairwise products. In case both products are the scalar products, the solution of (4.16) is equivalent with PCA. Assuming only $\langle\cdot,\cdot\rangle_\mathbf{y}$ is the scalar product, problem (4.16) also accepts an analytic solution. Let $W_{ij} = \langle \zeta_i, \zeta_j\rangle_\zeta$ be the coefficients of the Gram matrix, $|k_1| \geq |k_2| \geq \cdots \geq |k_N|$ its eigenvalues sorted in descending absolute value and $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_N$ the respective eigenvectors. The optimal d-dimensional embedding for a point ζ_n is given by

$$\mathbf{y}_n = \begin{pmatrix} k_1^{1/2}\mathbf{u}_1^n \\ k_2^{1/2}\mathbf{u}_2^n \\ \vdots \\ k_d^{1/2}\mathbf{u}_d^n \end{pmatrix}, \tag{4.17}$$

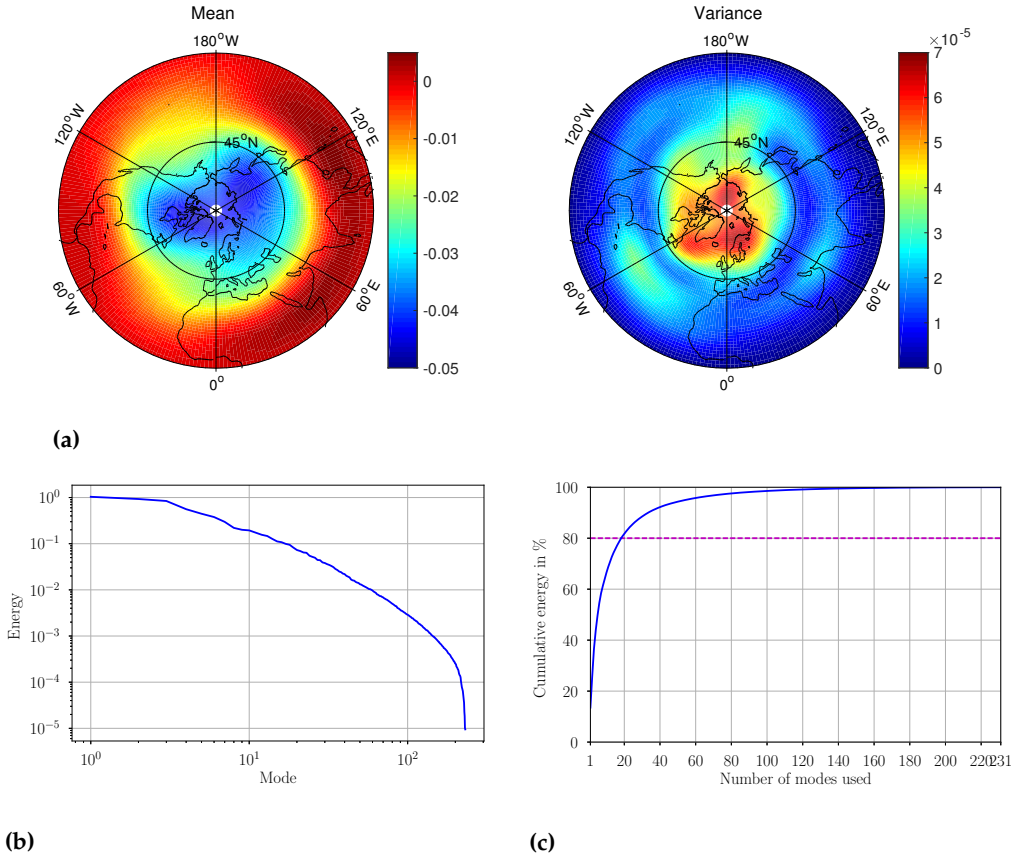

Figure 13: Mean, variance and energy distribution of the Barotropic model at statistical steady state.

where $\mathbf{u}_m^n$ denotes the n^{th} component of the m^{th} eigenvector. The optimality of (4.17) can be proven by the Eckart-Young-Mirsky theorem, as problem (4.16) is equivalent with finding the best d rank approximation in the Frobenius norm. In our problem, the standard kinetic energy product is used to preserve the nonlinear symmetries of the system dynamics [25]:

$$\langle \zeta_i, \zeta_j \rangle_\zeta = \int_{\mathcal{S}} \nabla\psi_i \cdot \nabla\psi_j d\mathcal{S} = -\int_{\mathcal{S}} \zeta_i \psi_j d\mathcal{S} = -\int_{\mathcal{S}} \zeta_j \psi_i d\mathcal{S}, \tag{4.18}$$

where the last identities are derived using partial integration and the fact that $\zeta = \Delta\mathbf{y}$. The energy spectrum of the modes of the reduced order space $\mathbf{y}$ is plotted in Figure 13a.

Solution (4.17) is only optimal w.r.t. the N training data points used to construct the Gram matrix. In order to calculate the embedding for a new point, it is convenient to compute the empirical orthogonal functions (EOFs) which form an orthonormal basis of the reduced order space $\mathbf{y}$ [25]. The EOFs are given by

$$\phi = \sum_{n=1}^{N} k_m^{-1/2} \mathbf{u}_m^n \zeta_n, \tag{4.19}$$

where m runs from 1 to d. The EOFs are sorted in descending order according to their energy level. The first four EOFs are plotted in Figure 14. EOF analysis has been used to identify individual realistic climatic modes such as the Arctic Oscillation (AO) [57,58] known as `teleconnections`. The first EOF is characterized by a center of action over the Arctic that

is surrounded by a zonal symmetric structure in mid-latitudes. This pattern resembles the Arctic Oscillation/Northern Hemisphere Annular Mode (AO/NAM) [57] and explains approximately 13.5% of the total energy. The second, third and fourth EOFs are quantitatively very similar to the East Atlantic/West Russia [59], the Pacific/North America (PNA) [60] and the Tropical/Northern Hemisphere (TNH) [61] patterns end account for 11.4%, 10.4% and 7.1% of the total energy respectively. Since these EOFs feature realistic climate teleconnections, performing accurate predictions of them is of high practical importance.

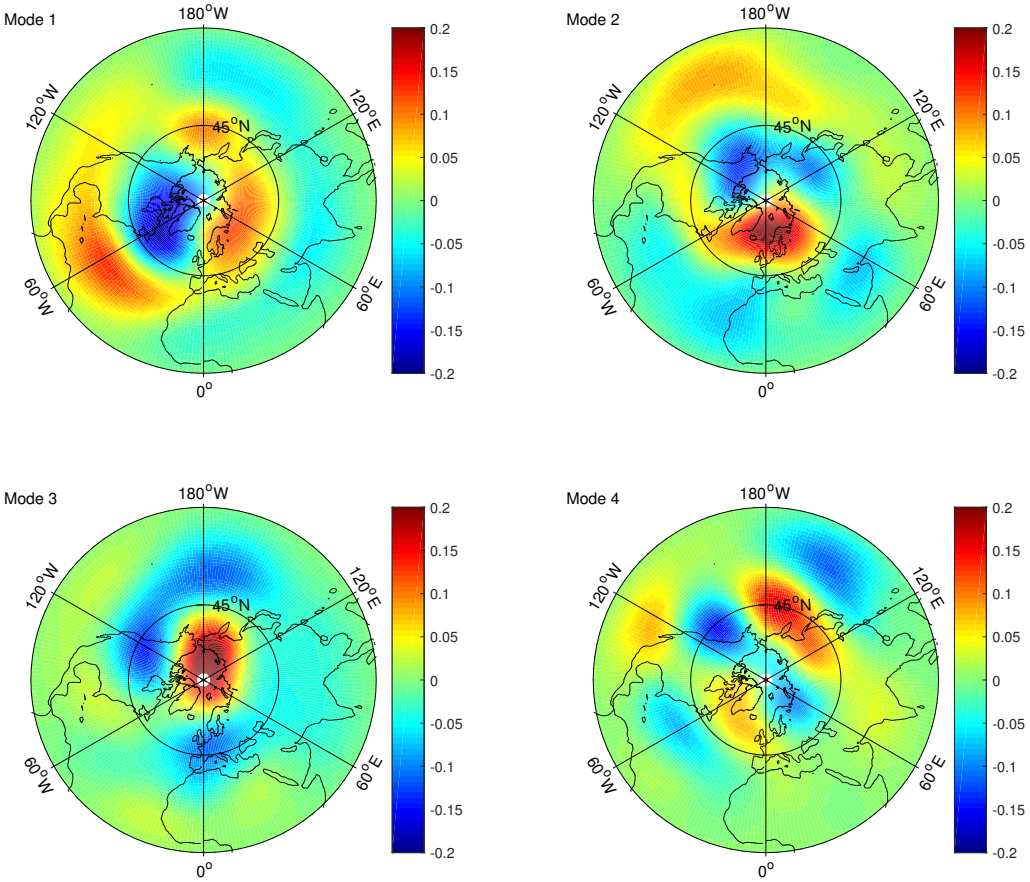

Figure 14: The four most energetic empirical orthogonal functions of the barotropic model

As a consequence of the orthogonality of the EOFs w.r.t. the kinetic energy product, the reduced representation $\mathbf{y}^*$ of a new state ζ^* can be recovered from

$$\mathbf{y}^* = \begin{pmatrix} \langle \zeta^*, \phi_1 \rangle_\zeta \\ \langle \zeta^*, \phi_2 \rangle_\zeta \\ \vdots \\ \langle \zeta^*, \phi_d \rangle_\zeta \end{pmatrix}. \tag{4.20}$$

In essence, the EOFs act as an orthogonal basis of the reduced order space and the new state ζ^* is projected to this basis. Only the d coefficients corresponding to the most energetic EOFs form the reduced order state $\mathbf{y}^*$. In our study, the dimensionality of the reduced space is $r_{dim} = 30$, as $\phi_3 0$ contains only 3.65% of the energy of ϕ_1, while the 30 most energetic modes contain approximately 82% of the total energy, as depicted in Figure 13c.

(ii) Training and Prediction

The reduced order state that we want to predict using the LSTM are the 30 components of $\mathbf{y}$. A *stateless* LSTM with $h = 140$ hidden units is considered, while the truncated back-propagation horizon is set to $d = 10$. The prototypical system is less chaotic than the KS equation and the Lorenz 96, which enables us to use more hidden units. The reason is that as chaoticity is decreased trajectories sampled from the attractor as training and validation dataset become more interconnected and the task is inherently easier and less prone to overfitting. In the extreme case of a periodic system, the information would be identical. 500 points are randomly picked from the attractor as initial conditions for testing. A Gaussian ensemble with a small variance ($\sigma_{en} = 0.001$) along each dimension is formed and marched using the reduced-order GPR, MSM, Mixed GPR-MSM and LSTM methods.

(iii) Results

The RMSE error of the four most energetic reduced order space variables $\mathbf{y}_i$ for $i \in \{1, \ldots, 4\}$ is plotted in Figure 15. The LSTM takes $400 - 500$ h to reach the attractor, while GPR based methods generally take $300 - 400$ h. In contrast, the MSM reaches the attractor already after 1 hour. This implies that the LSTM can better capture the non-linear dynamics compared to GPR. Note that the barotropic model is much less chaotic than the Lorenz 96 system with $F = 16$, where all methods show comparable prediction performance. Blended LSTM models with MSM are omitted here, as LSTM models only reach the attractor standard deviation towards the end of the simulated time and MSM-LSTM shows identical performance.

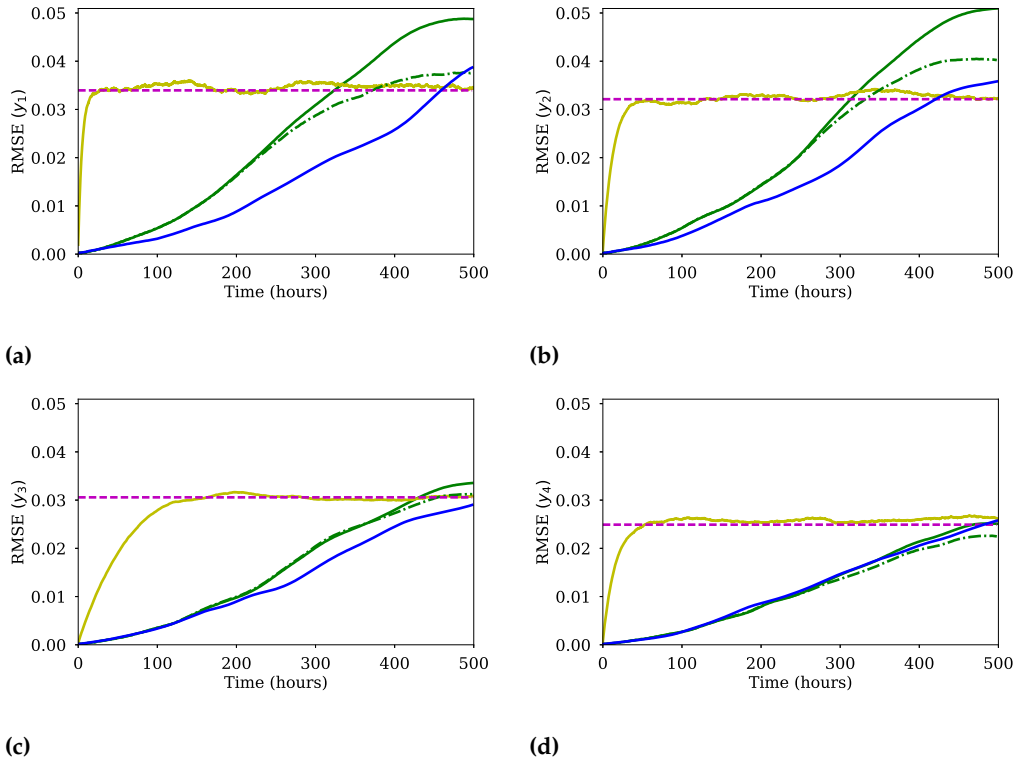

Figure 15: Mean RMSE of the most energetic EOFs over 500 initial conditions for the Barotropic climate model. Standard deviation from the attractor - - -; MSM ——; GPR ——; GPR-MSM - - -; LSTM ——

5. A Comment on Computational Cost of Prediction

The computational cost of making a single prediction can be quantified by the number of operations (multiplications and additions) needed. In GPR based approaches the computational cost in the Landau notation is $O(N^2)$, where N is the number of samples used in training. For GPR methods illustrated in the previous section $N \approx 2500$. The GPR models the global dynamics by uniformly sampling the attractor and "carries" this training dataset at each time instant to identify the geometric relation between the input and the training dataset and make (exact) probabilistic inference on the output.

In contrast, LSTM learns the behavior by adjusting its parameters, which leads to a prediction computational complexity that does not depend on the number of samples used for training. The inference complexity is roughly $O(d_i \cdot d \cdot h + d \cdot h^2)$, where d_i is the dimension of each input, d is the number of inputs and h is the number of hidden units. This complexity is significantly smaller than GPR, which can be translated to faster prediction.

Especially in real-time applications that require fast short-term predictions of a complex system, the LSTM has an advantage. However, it is logical that the LSTM is more prone to diverge from the attractor, as there is no guarantee that the infrequent training samples near the attractor limits where memorized. This remark explains the faster divergence of LSTM in the more turbulent regimes considered in Section 4.

6. Conclusions

We propose a data-driven method, based on long-short term memory networks, for modeling and prediction in the reduced space of chaotic dynamical systems. The LSTM uses the short term history of the reduced order variable to predict the state derivative and uses it for one-step prediction. The network is trained on time-series data and it requires no prior knowledge of the underlying governing equations. Using the trained network, long-term predictions are made by iteratively predicting one step forward.

The features of the proposed technique are showcased through comparisons with GPR and MSM on bench-marked cases. Three applications are considered, the Lorenz 96 system, the Kuramoto-Sivashinsky equation and a barotropic climate model. The chaoticity of these systems ranges from weakly chaotic to fully turbulent, ensuring a complete simulation study. Comparison measures include the RMSE and AC between the predicted trajectories and trajectories of the real dynamics.

In all cases, the proposed approach performs better, in short term predictions, as the LSTM is more efficient in capturing the local dynamics and complex interactions between the modes. However, the prediction error propagates fast and the prediction similar to GPR does not converge to the invariant measure. Furthermore in the cases of increased chaoticity the LSTM diverges faster than GPR. This may be attributed to the non-presence of certain attractor regions in the training data, insufficient training, and propagation of the exponentially increasing prediction error. To mitigate this effect, LSTM is also combined with MSM, following ideas presented in [25], in order to guarantee convergence to the invariant measure. Blending LSTM or GPR with MSM leads to a deterioration in the short term prediction performance but the steady-state statistical behavior is captured. The hybrid LSTM-MSM exhibits a slightly superior performance than GPR-MSM in all systems considered in this study.

In the Kuramoto-Sivashinsky equation LSTM can capture better the local dynamics compared to Lorenz 96 due to the lower intrinsic dimensionality of the attractor. The LSTM shows comparable forecasting accuracy with GPR in the barotropic model. The intrinsic dimensionality is significantly smaller than Kuramoto-Sivashinsky and Lorenz 96 and both methods can effectively capture the dynamics. Moreover, the prediction error does not propagate as rapidly as in Lorenz 96 and the blended LSTM-MSM scheme is omitted.

Future directions include modeling the lower energy modes and interpolation errors using a stochastic component in the LSTM to improve the forecasting accuracy. Another possible

research direction is to model the attractor in the reduced space using a mixture of LSTM models, one model for each region. The LSTM proposed in this work models the attractor globally. However, different attractor regions may exhibit very different dynamic behaviors, which cannot be simultaneously modeled using only one network. Moreover, these local models can be combined with a closure scheme compensating for truncation and modeling errors. This local modeling approach may further improve prediction performance.

7. Data Accessibility Statement

The machine learning library `TensorFlow` in `python 3.0` was used for the implementation of the LSTM architectures while MATLAB was used for GPR. The code and data used in this work are available in https://polybox.ethz.ch/index.php/s/eAwDx32qlnTnT7e.

8. Competing Interests Statement

We have no competing interests.

9. Authors' Contributions

PRV conceived the idea of the blended LSTM-MSM scheme, implemented the neural network architectures and the simulations, interpreted the computational results, and wrote the manuscript. WB supervised the work and contributed to the implementation of the LSTM. ZYW implemented the GPR and made contributions to the manuscript. PK had the original idea of the LSTM scheme and contributed to the manuscript. PK and TPS contributed to the interpretation of the results and offered consultation. All authors gave final approval for publication.

10. Funding Statement

TPS and ZYW have been supported by an Air Force Office of Scientific Research grant FA9550-16-1-0231, an Office of Naval Research grant N00014-15-1-2381, and an Army Research Office grant 66710-EG-YIP. PK and PRV gratefully acknowledge support from the European Research Council (ERC) Advanced Investigator Award (No. 341117).

11. Acknowledgments

Do not apply in this work.

References

1. Kingma DP, Ba J. 2017 Adam: A Method for Stochastic Optimization, *ArXiv preprint*. (arXiv:1412.6980)
2. Williams MO, Kevrekidis IG, Rowley CW. 2015 A data-driven approximation of the Koopman operator: extending dynamic mode decomposition, *Journal of Nonlinear Science*. **25** (6), pp 1307–1346. (doi:10.1007/s00332-015-9258-5)
3. Tu JH, Rowley CW, Luchtenburg DM, Brunton SL, Kutz JN. 2014 On dynamic mode decomposition: theory and applications, *Journal of Computational Dynamics*. **1** (2), pp 391–421. (doi:10.3934/jcd.2014.1.391)
4. Kutz JN, Fu X, Brunton SL. 2016 Multiresolution dynamic mode decomposition, *SIAM Journal on Applied Dynamical Systems*. **15** (2), 713–735. doi:10.1073/pnas.1517384113
5. Arbabi H, Mezic I. 2017 Study of dynamics in unsteady flows using Koopman mode decomposition, *ArXiv preprint*. (arXiv:1704.00813)
6. Arbabi H, Mezic I. 2017 Ergodic theory, Dynamic Mode Decomposition and computation of spectral properties of the Koopman operator, *ArXiv preprint*. (arXiv:1611.06664)

7. Rowley CW. 2005Model reduction for fluids, using balanced proper orthogonal decomposition, *Int. J. Bifurcat. Chaos*. **15** (3), 997–1013.
8. Sapsis TP. Majda AJ. 2013 Statistically accurate low-order models for uncertainty quantification in turbulent dynamical systems, *Proc. Natl Acad. Sci.*. **110**, 13705–13710.
9. Brunton SL, Proctor JL, Kutz JN. 2016 Discovering governing equations from data by sparse identification of nonlinear dynamical systems, *Proceedings of the National Academy of Sciences*. **113** (15), 3932–3937. (doi: 10.1073/pnas.1517384113)
10. Duriez T. Brunton SL, Noack BR. 2016 Machine Learning Control: Taming Nonlinear Dynamics and Turbulence, *Springer*.
11. Majda AJ, Lee Y. 2014 Conceptual dynamical models for turbulence, *Proc. Natl Acad. Sci*. **111**, 6548–6553.
12. Schaeffer H. 2017 Learning partial differential equations via data discovery and sparse optimization, *Proc. R. Soc. A*. **473**, 20160446.
13. Farazmand M, Sapsis TP. 2016 Dynamical indicators for the prediction of bursting phenomena in high-dimensional systems, *Physical Review E*. **94**, 032212.
14. Lee Y, Majda AJ. 2016 State estimation and prediction using clustered particle filters, *PNAS*. **113** (51), 14609–14614. (doi:10.1073/pnas.1617398113)
15. Comeau D, Zhao Z, Giannakis D, Majda AJ. 2017 "Data-driven prediction strategies for low-frequency patterns of North Pacific climate variability, *Climate Dynamics*. **48** (5-6), 1855–1872. (doi:10.1007/s00382-016-3177-5)
16. Tatsis K, Dertimanis V, Abdallah I, Chatzi E. 2017 A substructure approach for fatigue assessment on wind turbine support structures using output-only measurements, *X International Conference on Structural Dynamics, EURODYN 2017*. **199**, 1044–1049.
17. Quade M, Abel M, Shafi K, Niven RK, Noack BR. 2016 Prediction of dynamical systems by symbolic regression, *Phys. Rev. E.*. **94**, 012214.
18. Cousins W, Sapsis TP. 2014 Quantification and prediction of extreme events in a one-dimensional nonlinear dispersive wave model, *Physica D*. **280-281** (48–58).
19. Cousins W, Sapsis TP. 2016 Reduced order precursors of rare events in unidirectional nonlinear water waves, *Journal of Fluid Mechanics*. **790** (368–388).
20. Lorenz EN. 1969 Atmospheric predictability as revealed by naturally occurring analogues, *J. Atmos. Sci.* **26** 636–646.
21. Xavier PK, Goswami BN. 2007 An analog method for real-time forecasting of summer monsoon subseasonal variability, *Monthly Weather Review* **135** (12) 4149–4160.doi:10.1175/2007MWR1854.1.
22. Zhao Z, Giannakis D. 2016 Analog forecasting with dynamics-adapted kernels, *Nonlinearity* **29** 2888–2939.
23. Sonday BE, Singer A, Gear CW, Kevrekidis IG. 2010 Manifold learning techniques and model reduction applied to dissipative pdes, 1–20.
24. Chiavazzo E, Gear CW, Dsilva CJ, Rabin N, Kevrekidis IG. 2014 Reduced models in chemical kinetics via nonlinear data-mining, *Processes* **2** (1) 112–140.
25. Wan ZY, Sapsis TP. 2017 Reduced-space Gaussian Process Regression for data-driven probabilistic forecast of chaotic dynamical systems, *Physica D: Nonlinear Phenomena* **345** 40–55.
26. Rasmussen CE, Williams CKI. 2006 Gaussian Processes for Machine Learning, The MIT Press.
27. Hochreiter S, Schmidhuber J. 1997 Long short-term memory, *Neural Computation* **9** 1735–1780.
28. Graves A, Mohamed AR, Hinton G. 2013 Speech Recognition with Deep Recurrent Neural Networks, *Acoustics, Speech and Signal Processing (ICASSP) IEEE International Conference*. 6645–6649.
29. Graves A, Schmidhuber, J. 2005 Framewise phoneme classification with bidirectional LSTM and other neural network architectures, *Neural Networks*.**18**(5-6): 602–610 (doi:10.1016/j.neunet.2005.06.042)
30. Fernández S, Graves A, Schmidhuber, J. 2007 An Application of Recurrent Neural Networks to Discriminative Keyword Spotting, *Proceedings of the 17th International Conference on Artificial Neural Networks. ICANN, Berlin, Heidelberg: Springer-Verlag*: 220–229 (ISBN 3540746935)
31. Graves A, Fernández S, Liwicki M, Bunke H, Schmidhuber, J. 2007 Unconstrained Online Handwriting Recognition with Recurrent Neural Networks, *Proceedings of the 20th International Conference on Neural Information Processing Systems*. 577–584 USA: Curran Associates Inc.: (ISBN 9781605603520)
32. Wierstra D, Schmidhuber J, Gomez FJ. 2005 Evolino: Hybrid Neuroevolution/Optimal Linear

Search for Sequence Learning, *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI)*, Edinburgh, 853–858 p[l;,.
33. Wu Y, Schuster M, Chen Z, Le QV, Norouzi M, Macherey W, Krikun M, Cao Y, Gao Q. 2016 Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation, *arXiv:1609.08144*.
34. Gers FA, Eck D, Schmidhuber J. 2012 Applying LSTM to Time Series Predictable Through Time-Window Approaches, *Neural Nets WIRN Vietri-01: Proceedings of the 12th Italian Workshop on Neural Nets*. Springer, London, 193–200.
35. Martens J, Sutskever I. 2011 Training Recurrent Neural Networks with Hessian Free optimization, *28th Annual International Conference on Machine Learning (ICML)*.
36. Jaeger M, Haas H. 2004 Harnessing Nonlinearity: Predicting Chaotic Systems and Saving Energy in Wireless Communication, *Science* **304** (5667) 78–80. (doi:10.1126/science.1091277).
37. Chatzis SP, Demiris Y. 2011 Echo State Gaussian Process, *IEEE Transactions on Neural Networks*, **22** (9) 1435–1445.
38. Broomhead DS, Lowe D. 1988 Multivariable functional interpolation and adaptive networks, *Complex Systems* **2** 321–355.
39. Kim KB, Park JB, Choi YH, Chen G. 2000 Control of chaotic dynamical systems using radial basis function network approximators, *Inf. Sci.* **130**, 165-183.
40. Takens F. 1981 *Detecting strange attractors in fluid turbulence. In Symposium on dynamical systems and turbulence*, Ed. DA Rand and LS Young. Springer, Berlin, 366–381.
41. Hochreiter J, Untersuchungen zu dynamischen neuronalen Netzen, Master thesis Institut fur Informatik, Technische Universitat, Munchen.
42. Bengio Y, Simard P, Frasconi P. 1994 Long short-term memory, *IEEE Transactions on Neural Networks*, **5** (2), 157–166.
43. Majda A, Harlim J. 2012 *Filtering Complex Turbulent Systems*, Cambridge University Press.
44. Majda A, Grote MJ, Abramov RV. 2005 *Information Theory and Stochastics for Multiscale Nonlinear Systems* Vol. 25, AMS and Centre de Recherches Mathematiques.
45. Lorenz NE. 1996 Predictability - A problem partly solved, *Proc. Seminar on Predictability*. Reading, Berkshire, 1–18.
46. Basnarkov L, Kocarev L. 2012 Forecast improvement in Lorenz96 system, *Nonlin. Processes Geophy.* **19**, 569–575.
47. Allgaier NA, Harris KD, Danforth CM.: 2012 Empirical correction of a toy climate model, *Phys. Rev. E* **85** (2), 026201
48. Crommelin DT, Majda AJ. 2004 Strategies for model reduction: Comparing different optimal bases. *J. Atmos. Sci.*, **61** (17) 2206–2217.
49. Kuramoto Y, Tsuzuki T. 1976 Persistent Propagation of Concentration Waves in Dissipative Media Far from Thermal Equilibrium, *Progress of Theoretical Physics* **55** (2), 356–369.
50. Kuramoto Y. 1978 Diffusion-induced chaos in reaction systems, *Progress of Theoretical Physics Supplement* **64**, 346–367.
51. Sivashinsky G, 1977. Nonlinear analysis of hydrodynamic instability in laminar flamesâĂŤi. derivation of basic equations, *Acta Astronautica* **4**, 1177–1206.
52. Sivashinsky G, Michelson DM. 1980 On irregular wavy flow of a liquid film down a vertical plane, *Progress of Theoretical Physics* **63** (6), 2112–2114.
53. Blonigan PJ, Wang Q. 2014 Least squares shadowing sensitivity analysis of a modified Kuramoto-Sivashinsky equation, *Chaos, Solitons and Fractals*. **64**, 16–25.
54. Kevrekidis IG, Nicolaenko B, Scovel JC. 1990 Back in the saddle again: A computer assisted study of the kuramoto-sivashinsky equation. *SIAM J. Appl. Math.* **50** (3) 760–790.
55. Selten FM. 1995 An efficient description of the dynamics of barotropic flow, *Journal of the Atmospheric Sciences* **52** (7) 915–936.
56. Cox MAA, Cox TF. 2001 *Multidimensional Scaling*. Second edition., Chapman and Hall.
57. Thompson DWJ, Wallace JM. 2000 Annular modes in the extratropical circulation: Part i: Month-to-month variability. *J. Climate* **13**, 1000–1016.
58. Thompson DWJ, Wallace JM. 1998 The arctic oscillation signature in wintertime signature in wintertime geopotential height and temperature fields. *Geophys. Res. Lett.* **25**, 1297–1300.
59. Barnston AG, Livezey RE. 1987 Classification, seasonality and persistence of low-frequency atmospheric circulation patterns, *Mon. Weather Rev.* **115**, 1083–1126.
60. Wallace JM, Gutzler DS. 1981 Teleconnections in the geopotential height field during the northern hemisphere winter, *Mon. Weather Rev.* **109**, 784–812.

61. Mo KC, Livezey RE. 1986 Tropical-extratropical geopotential height teleconnections during the northern hemisphere winter. *Mon. Weather Rev.* **114**, 2488–2512.