Sequence-to-sequence prediction of spatiotemporal systems

Cite as: Chaos **30**, 023102 (2020); https://doi.org/10.1063/1.5133405
Submitted: 23 October 2019 . Accepted: 13 January 2020 . Published Online: 03 February 2020

Guorui Shen, Jürgen Kurths, and Ye Yuan

Sequence-to-sequence prediction of spatiotemporal systems

Cite as: Chaos **30**, 023102 (2020); doi: 10.1063/1.5133405
Submitted: 23 October 2019 · Accepted: 13 January 2020 ·
Published Online: 3 February 2020

Guorui Shen,[1] Jürgen Kurths,[2,3] and Ye Yuan[1,a)]

AFFILIATIONS

[1]School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, People's Republic of China
[2]Potsdam Institute for Climate Impact Research, Potsdam 14473, Germany
[3]Department of Physics, Humboldt University, Berlin 12489, Germany

Note: This paper is part of the Focus Issue, "When Machine Learning Meets Complex Systems: Networks, Chaos and Nonlinear Dynamics".
[a)]**Author to whom correspondence should be addressed:** yye@hust.edu.cn

ABSTRACT

We propose a novel type of neural networks known as "attention-based sequence-to-sequence architecture" for a model-free prediction of spatiotemporal systems. This architecture is composed of an encoder and a decoder in which the encoder acts upon a given input sequence and then the decoder yields another output sequence to make a multistep prediction at a time. In order to demonstrate the potential of this approach, we train the neural network using data numerically sampled from the Korteweg–de Vries equation—which describes the interaction between solitary waves—and then predict its future evolution. Furthermore, we validate the applicability of the approach on datasets sampled from the chaotic Lorenz system and three other partial differential equations. The results show that the proposed method can achieve good performance in predicting the evolutionary behavior of studied spatiotemporal dynamics. To the best of our knowledge, this work is the first attempt at applying attention-based sequence-to-sequence architecture to the prediction task of solitary waves.

Prediction of spatiotemporal systems is important for many scientific and industrial fields, which, however, is often a challenging task due to the spatial correlation and temporal dependency within data. Within the machine learning community, there is an explosive number of successful applications of neural networks, from image recognition to neural language processing. In this article, we use the attention-based sequence-to-sequence architecture—a novel type of deep neural networks originally designed for sequential tasks—to deal with high-dimensional spatiotemporal data.

I. INTRODUCTION

Spatiotemporal systems refer to systems whose state evolution involves both spatial and temporal information. It is a very important concept in areas such as reaction diffusion,[1,2] heat conduction,[3] and fluid dynamics.[4] The modeling of spatiotemporal systems allows us to better understand the essence of natural phenomena and, therefore, is of great importance. To model spatiotemporal systems, a traditional theoretical method is to derive an analytical form, a partial differential equation (PDE), for example, which governs the evolutionary behavior of the system. However, this only works fine for limited situations, since it relies on a series of restriction assumptions that are sometimes unrealistic to meet. For the remaining many complex systems, which are hard to have an analytical description, forecasting their future states is still of great interest and practical value. Hence, in this article, we consider the modeling of spatiotemporal dynamics in which a history of high-dimensional series data is accessible, while an equation-based description of how these data are generated is not accessible. Prediction approaches focusing on dealing with such data have been extensively studied. To illustrate, classical methods such as regression over local states are well established in Refs. 5 and 6. Kernel methods[7] and Gaussian process regression (GPR)[8] have been successfully applied to prediction tasks since they are capable of capturing nonlinearity from true dynamics. However, their success depends on a subjectively defined nonlinear function. Recurrent neural networks (RNNs), a class of

deep neural networks, have drawn a large amount of attention from the machine learning community due to their excellent performance in dealing with sequential modeling tasks.[9] Their success has inspired some efforts at utilizing RNNs to the modeling of spatiotemporal systems. In particular, Refs. 10 and 11 use a novel type of RNN known as reservoir computing to predict the chaotic KS system and achieve great success. In recent years, sequence-to-sequence (seq2seq) learning and attention-based seq2seq learning, a more prominent type of RNNs, have been proposed for sequential modeling. Therefore, it is natural to explore the feasibility of applying these state-of-art variants of RNNs to the prediction of spatiotemporal systems.

The rest of this article is organized as follows. Section II outlines the details about both attention-free and attention-based seq2seq architectures and how do they work for prediction tasks. Section III presents seven demonstrative examples—including three demonstrations on solitary waves, one on the chaotic Lorenz system and the remaining three on Fisher, Burger, and sine-Gordon equations, respectively. Last, the conclusion and future problems are given in Sec. IV.

II. PREDICTION METHODOLOGY

Seq2seq architecture was first proposed by Refs. 12–14 for sequential modeling, such as image captioning,[15,16] sentiment classification,[17] and language translation problems.[14] Later, it was further developed by Refs. 9, 18, and 19 in which the attention mechanism was added to help enhance its performance on large-scale sequential tasks. In this paper, we explore the feasibility of using the attention-based seq2seq method, the version outlined in the paper,[19] to predict some interesting physical phenomena. In particular, the collision of Korteweg–de Vries (KdV) solitary waves is thoroughly studied here. We begin with elaborating the basic idea inside the raw seq2seq architecture and then point out its main drawback, based on which the attention mechanism is introduced to improve its capacity. After that, we formulate the attention-based seq2seq model to suit the prediction purpose for spatiotemporal systems.

A. Seq2seq architecture

The seq2seq architecture is also termed as the encoder–decoder model due to its two important components, an encoder cascaded after by a decoder, as shown in Fig. 1. The encoder converts a given input sequence $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_S)$ into a sequence of the same length of hidden states $(\boldsymbol{h}_1, \ldots, \boldsymbol{h}_S)$. The concerned computation is completed by repeatedly unrolling a RNN cell for S times. At time step s, the current input $\boldsymbol{x}_s$ and the previous hidden state $\boldsymbol{h}_{s-1}$ are used as known information to compute the current hidden state $\boldsymbol{h}_s$, abstractly written as

$$\boldsymbol{h}_s = f_e(\boldsymbol{x}_s, \boldsymbol{h}_{s-1}), \quad s = 1, \ldots, S, \tag{1}$$

where f_e defines a RNN cell that advances the hidden state one-step forward. The hidden state $\boldsymbol{h}_s$ is referred to as a source hidden state. Popular choices of such RNN cells vary from a simple vanilla RNN unit to complicated long short-term memory (LSTM)[20] and a gated recurrent unit (GRU).[12] Take GRU, for example, Eq. (1) is expanded

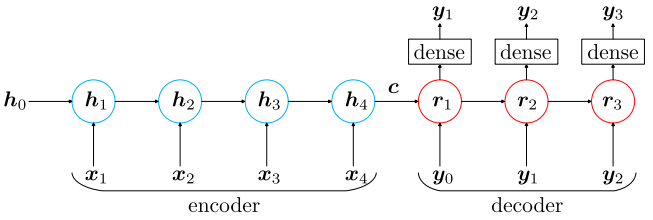


FIG. 1. The seq2seq architecture illustrates transforming a sequence $(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{x}_4)$ into another sequence $(\boldsymbol{y}_1, \boldsymbol{y}_2, \boldsymbol{y}_3)$. The blue circle represents an encoder, a RNN cell unrolling for $S = 4$ times. The red circle represents a decoder, another RNN cell unrolling for $T = 3$ times. "Dense" marks a fully connected dense layer.

as (for $s = 1, \ldots, S$)

$$\begin{aligned} \boldsymbol{z}_s &= \sigma(W_z \boldsymbol{x}_s + U_z \boldsymbol{h}_{s-1} + \boldsymbol{b}_z), \quad \boldsymbol{g}_s = \sigma(W_g \boldsymbol{x}_s + U_g \boldsymbol{h}_{s-1} + \boldsymbol{b}_g), \\ \boldsymbol{h}_s &= (1 - \boldsymbol{z}_s) \circ \boldsymbol{h}_{s-1} + \boldsymbol{z}_s \circ \tanh(W_h \boldsymbol{x}_s + U_h(\boldsymbol{g}_s \circ \boldsymbol{h}_{s-1}) + \boldsymbol{b}_h), \end{aligned} \tag{2}$$

where σ denotes the sigmoid function, $W_z, U_z, W_g, U_g, W_h, U_h$ are weight matrices, $\boldsymbol{b}_z, \boldsymbol{b}_g, \boldsymbol{b}_h$ are bias vectors, and $\boldsymbol{h}_0$ is random initialization. After S steps of unrolling, $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_S$ are produced according to Eq. (2). Thus, a context vector $\boldsymbol{c}$ can be yielded by executing a predefined function on those source hidden states, such as the mean value function $\boldsymbol{c} = (\boldsymbol{h}_1 + \cdots + \boldsymbol{h}_S)/S$, or the final hidden state function $\boldsymbol{c} = \boldsymbol{h}_S$. The resulted context vector is always seen as a summary of the entire input sequence and will be used to initialize the decoder during the next stage. In the seq2seq architecture, the decoder is another RNN characterized by its hidden state $\boldsymbol{r}_t$ and output $\boldsymbol{y}_t$. To distinguish from the encoder time step s, here, we use t to keep the time step for the decoder. At each time step t, the decoder receives as input the output produced by itself at time $t - 1$ to update its present target hidden state, which, in contrast to the source hidden state, is referred to as a target hidden state. The present hidden state will next be fed through a dense layer to generate an output that approximates the target label. Mathematically, these processes are calculated as

$$\begin{aligned} \boldsymbol{r}_t &= f_d(\boldsymbol{r}_{t-1}, \boldsymbol{y}_{t-1}, \boldsymbol{c}), \\ \boldsymbol{y}_t &= W_y \boldsymbol{r}_t, \quad t = 1, \ldots, T, \end{aligned} \tag{3}$$

where the target hidden state is initialized via $\boldsymbol{r}_0 = \boldsymbol{c}$ and the initial input $\boldsymbol{y}_0$ is treated as a trigger starting the decoding phase. During this phase, f_d, denoting the decoder cell, is also chosen as a GRU, the same type as the case used in the encoder but with a different unrolling times T. It is noteworthy that for the encoder, its input sequence $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_S$ is manually provided, while the input sequence $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{T-1}$ for the decoder is self-generated according to Eq. (3), given the initial input $\boldsymbol{y}_0$.

B. Attention-based seq2seq architecture

Up to now, we have illustrated the basic idea inside the original seq2seq architecture. Before stepping into the attention-based architecture, one should notice that in the seq2seq model above, the context vector $\boldsymbol{c}$ is used only one time to initialize the target hidden state, which sometimes is rough because the information contained within those source hidden states may not be efficiently

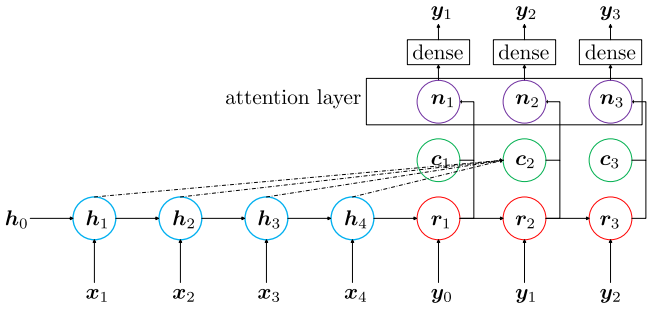


FIG. 2. Based upon the seq2seq architecture, as illustrated in Fig. 1, the attention-based seq2seq model adds an attention layer to right after the decoder. The green circle denotes the context vector, and the purple circle denotes the state of an attentional layer. The dashed lines indicate that $\boldsymbol{c}_2$ is drawing attention from all source hidden states. To be clear, dashed lines for $\boldsymbol{c}_1$ and $\boldsymbol{c}_3$ are dropped out.

exploited, and that is where the attention mechanism comes in to overcome this weakness. In an attentional architecture, the source hidden states are selectively focused on throughout the entire decoding course, making the most relevant information concentrated. The attention-based seq2seq model, as shown in Fig. 2, is much more complicated than the normal one. Based upon the normal seq2seq model, in Ref. 19, a global attention layer is added to right after the decoder in order to make full use of all source hidden states. To be clear, an attention-based seq2seq architecture calculates a so-called attentional hidden state $\boldsymbol{n}_t$ via

$$\boldsymbol{n}_t = \tanh(W_n[\boldsymbol{c}_t; \boldsymbol{r}_t]), \tag{4}$$

where $\boldsymbol{c}_t$ is now referred to as the context vector at time t. Details for how to compute $\boldsymbol{c}_t$ will be discussed later. Through a dense layer, the attentional hidden state will then be fed to produce the predicted result,

$$\boldsymbol{y}_t = W_y \boldsymbol{n}_t. \tag{5}$$

Equation (5) is implemented for T times to make the T-step forward prediction, based on the given input sequence $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_S$.

We now begin to work out the details of computing the context vector. In most literature studies, a context vector is defined as a weighted summation over all source hidden states, implying selectively taking advantage of all information from the input sequence at each time step t. The key point now remains how to design reasonable weights for each $\boldsymbol{c}_t$. One simple way of determining which source hidden state should receive more or less attention is to measure the similarity between the current target hidden state $\boldsymbol{r}_t$ and every single source hidden state $\boldsymbol{h}_s$ by their inner product,

$$a_{ts} = \text{softmax}(\boldsymbol{r}_t, \boldsymbol{h}_s) = \frac{\exp(\boldsymbol{r}_t^T \boldsymbol{h}_s)}{\sum_{q=1}^{S} \exp(\boldsymbol{r}_t^T \boldsymbol{h}_q)}, \tag{6}$$

where softmax is carried out to ensure that two requirements are satisfied: (i) each component of the resulted weight vector lies in the interval $(0, 1)$ and (ii) all elements will add up to one. The vector $\boldsymbol{a}_t = \begin{pmatrix} a_{t1} & \cdots & a_{tS} \end{pmatrix}^T$ is named the attentional alignment weight vector. By each alignment weight vector, the context vector can then be expressed as a linear combination of all source hidden states,

$$\boldsymbol{c}_t = \sum_{s=1}^{S} a_{ts} \boldsymbol{h}_s. \tag{7}$$

C. Attention-based seq2seq for spatiotemporal prediction

Next, we apply the attention-based seq2seq architecture to the prediction task. The data collected for training the model are a history of time series, $\boldsymbol{u}(t), t = 1, 2, \ldots, L$, where $\boldsymbol{u}(t) = \left[u_1(t), \ldots, u_Q(t)\right]^T \in R^Q$ denotes the system's states captured at time t. For spatiotemporal data, Q represents the number of points sampled along the spatial axis. Both the input and output dimension of the decoder are set to q, such that Q is an integer multiple of it, i.e., $d = Q/q$, which allows us to partition the entire states $\boldsymbol{u}(t)$ into d groups with each group of the form $\boldsymbol{w}_i(t) = \left[u_{(i-1)\cdot q+1}(t), \ldots, u_{i\cdot q}(t)\right]^T, i = 1, \ldots, d$. To construct the input sequence for the encoder, we also define another vector $\boldsymbol{v}_i(t)$ as $\boldsymbol{v}_i(t) = \left[u_{(i-1)\cdot q+1-l}(t), \ldots, u_{i\cdot q+l}(t)\right]^T, i = 2, \ldots, d-1$. Here, in general, l is a positive number less than q, ensuring that the ith group of states has some overlapping areas with that of other groups on its both sides. As shown in Fig. 3, $\boldsymbol{v}_i(-3), \boldsymbol{v}_i(-2), \boldsymbol{v}_i(-1), \boldsymbol{v}_i(0)$ are used as a source input sequence to make a three-step prediction for the target sequence $\boldsymbol{w}_i(1), \boldsymbol{w}_i(2), \boldsymbol{w}_i(3)$. In this case, the model parameters are configured as $S = 4, T = 3, q = 3, l = 2$. The recipe for constructing $\boldsymbol{v}_i(t), i = 1, d$, requires more consideration, since some states are missing. To alleviate this dilemma, we put an extra restriction on the boundary near-by states. More concretely, for systems with periodic boundary conditions, $\boldsymbol{v}_1(t)$ and $\boldsymbol{v}_d(t)$ use periodic indexes, namely, $\boldsymbol{v}_1(t) = \left[u_{Q+1-l}(t), \ldots, u_Q(t), u_1(t), \ldots, u_{q+l}(t)\right]^T$

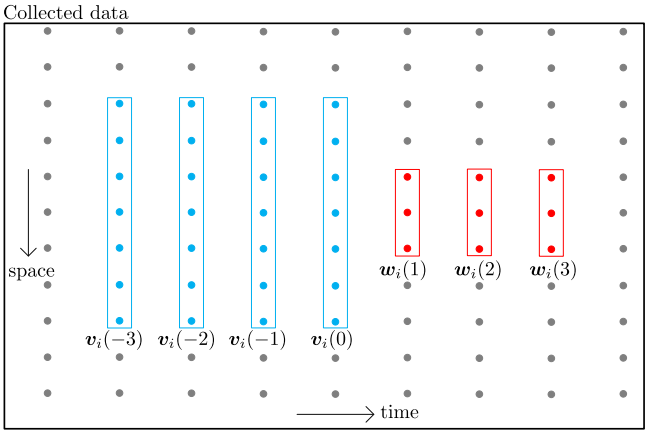


FIG. 3. An illustration of constructing input sequence and output sequence from collected spatiotemporal data. The length of input and output sequences are $S = 4$ and $T = 3$, and the dimension of each single input and output are 7 and 3, respectively. In our attention-based architecture, the encoder receives input sequence $\boldsymbol{v}_i(-3), \boldsymbol{v}_i(-2), \boldsymbol{v}_i(-1), \boldsymbol{v}_i(0)$, and next, the decoder generates the output sequence $\boldsymbol{w}_i(1), \boldsymbol{w}_i(2), \boldsymbol{w}_i(3)$, with $\boldsymbol{w}_i(0)$ triggering the decoding phase.

and $\boldsymbol{v}_d(t) = \left[u_{Q-q+1-l}(t), \ldots, u_Q(t), u_1(t), \ldots, u_l(t)\right]^T$. For nonperiodic systems, $\boldsymbol{v}_1(t)$ and $\boldsymbol{v}_d(t)$ are designated as $\boldsymbol{v}_1(t) = [u_1(t), \ldots, u_{q+2l}(t)]^T$ and $\boldsymbol{v}_d(t) = \left[u_{Q-q+1-2l}(t), \ldots, u_Q(t)\right]^T$. After doing so, each single input maintains their dimension of $q+2l$ to match the encoder accurately. With all of the above training sequences collected: $\{\boldsymbol{v}_i(t+1), \ldots, \boldsymbol{v}_i(t+S), \boldsymbol{w}_i(t+S+1), \ldots, \boldsymbol{w}_i(t+S+T)\}_{i=1,t=0}^{d,L-S-T}$, the unknown parameters can be determined by minimizing the mean squared error (MSE).

$$\frac{1}{N}\sum_{i=1}^{d}\sum_{t=0}^{L-S-T}\sum_{j=1}^{T}\|\boldsymbol{w}_i^p(t+S+j) - \boldsymbol{w}_i(t+S+j)\|_2^2, \tag{8}$$

where $N = d \cdot (L-S-T+1)$ represents the number of training examples and the superscript "p" denotes predicted values. Besides, we also define another MSE,

$$\text{Error}(t) = \frac{1}{Q}\|\boldsymbol{u}^p(t) - \boldsymbol{u}(t)\|_2^2, t = L+1, \ldots, \tag{9}$$

to measure the difference between the predicted values and the actual ones at time step t.

III. DEMONSTRATION ON EXAMPLES

In this section, we demonstrate the effectiveness of the attention-based seq2seq architecture to predict the evolutionary behavior of spatiotemporal dynamic systems. Specifically, in the first part of this section, the simulated propagation of a single solitary wave and the collision of two solitary waves under two different speed ratios are studied by means of our approach. In the second part, we compare the performance of our method with another type of neural networks known as "reservoir computing" on the Lorenz

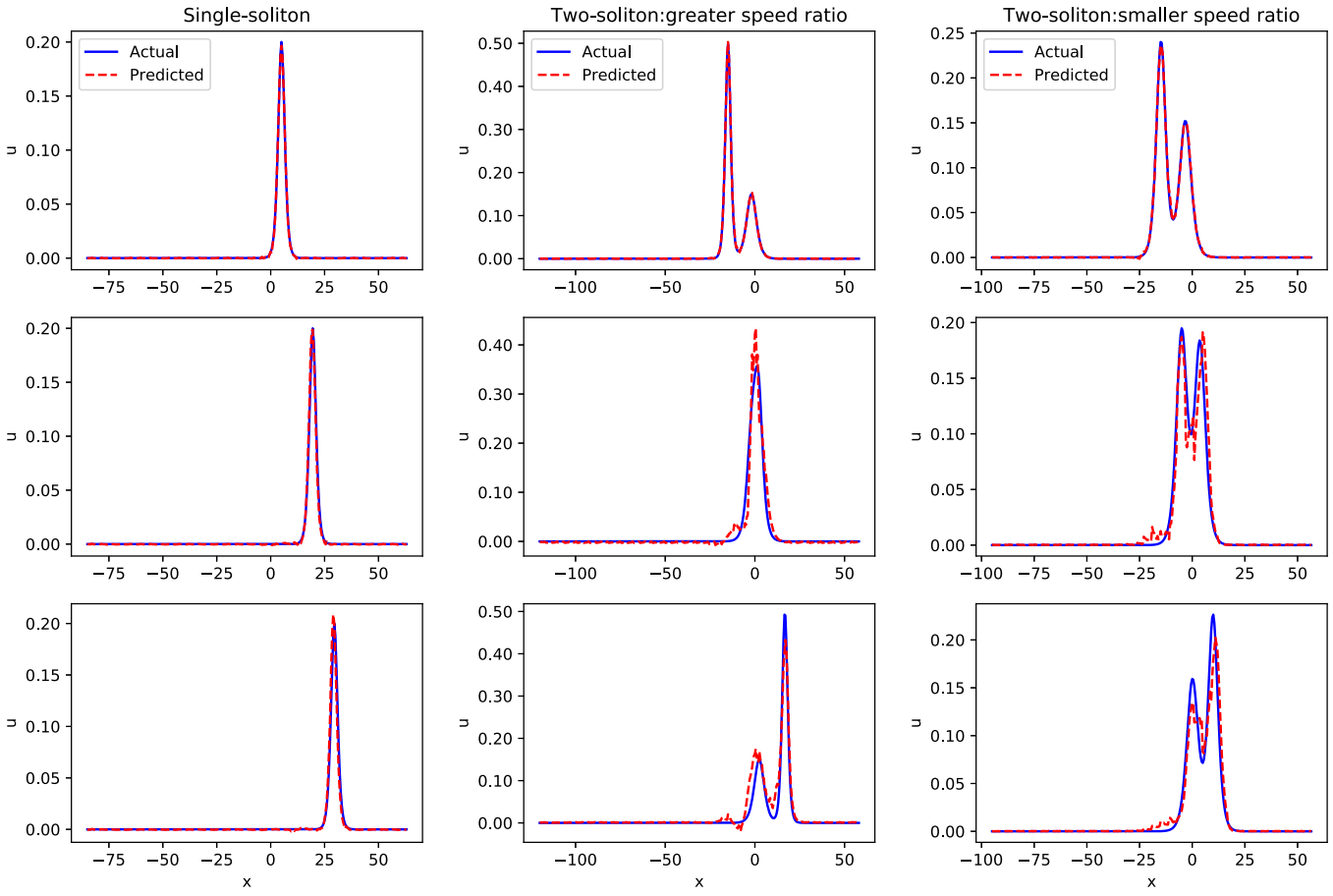


FIG. 4. In the leftmost column, the prediction for a single solitary wave is exhibited in which the upper, middle, and lower parts represent the 1st, 30th, and 50th-step forward prediction, respectively. The same applies to the middle column where the 1st, 20th, and 40th-step forward predictions are made for two solitary waves when the speed ratio is greater than three. On the rightmost column, the 1st, 30th, and 50th-step forward predictions are exhibited for two solitary waves when the speed ratio is set to less than three.

system, which is hard to predict due to its chaotic behavior. What is more in the third part, we further testify the capability of our approach by applying it to the prediction task of three more PDEs.

A. Solitary waves

It has been found that solitary waves exist in a wide range of scientific fields such as geomorphology,[21] optics,[22,23] telecommunication,[24] and geographic magnetics.[25] Current literature studies about the soliton theory mainly focus on deriving and solving an equation-based description of the original system either from a pure mathematical perspective[23] or from a numerical simulation angle.[22] This work, by contrast, avoids the requirement of a mathematical equation and resorts to from a data-driven angle to predict the future evolution of interested solitary phenomena. It only needs historical time series data to train an encoder–decoder model, offering a new perspective to the study of solitary waves.

A solitary wave, also termed as a soliton, is a wave that maintains its shape when it moves at a constant speed. The first solitary wave was observed and recorded by the engineer John Scott Russell[26] in 1834. Later, in 1895, it was further studied in a rigorous way by the two physicists Korteweg and de Vries.[27] They established the famous Korteweg–de Vries (KdV) equation,

$$u_t + u_{xxx} - 6u_x u = 0, \tag{10}$$

and argued that it could describe the phenomenon observed by Russell. Besides continuous systems, solitary waves also appear in

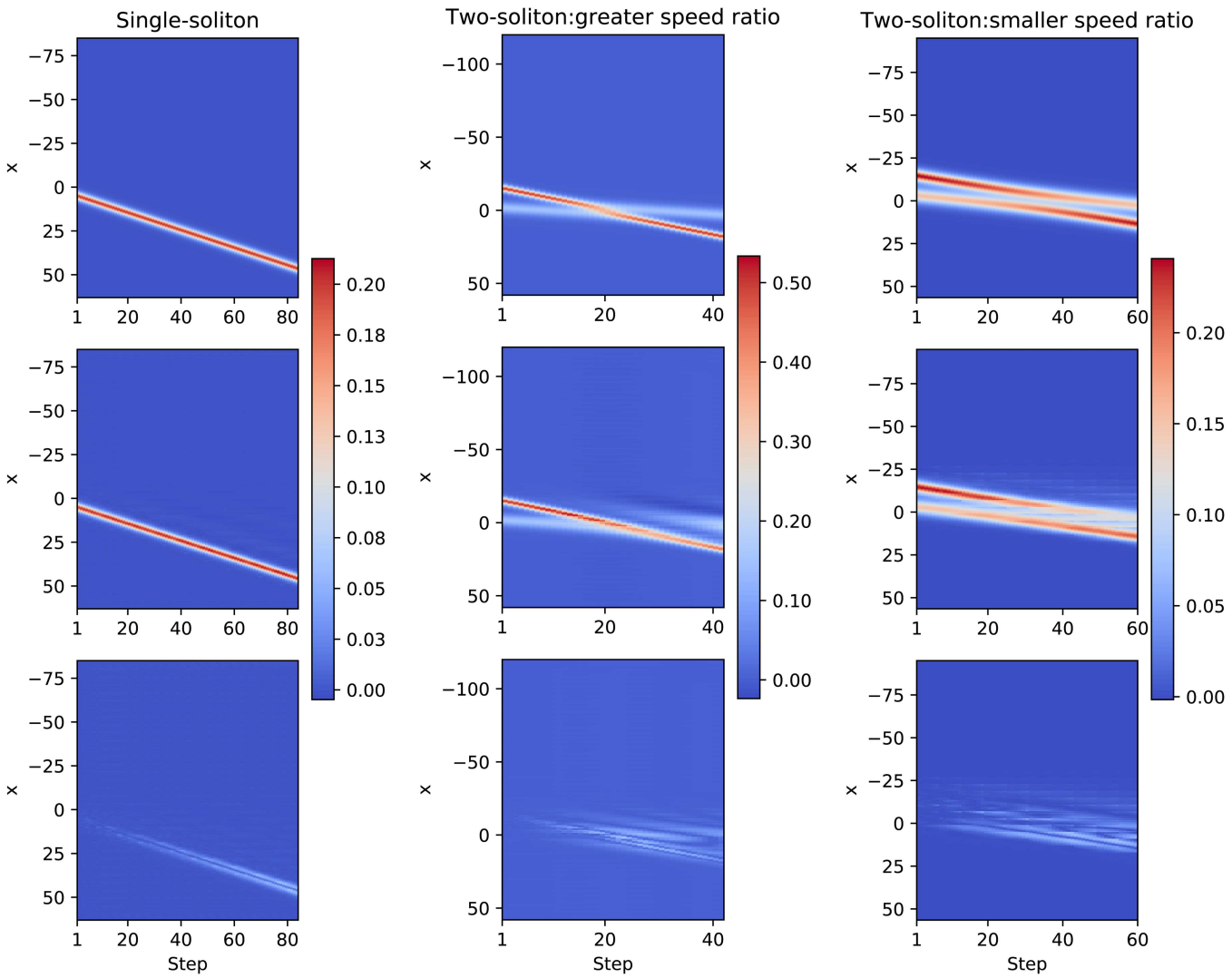


FIG. 5. In the leftmost column, the actual and predicted states and the absolute values of their difference are exhibited, respectively. The same applies to the middle and rightmost column, where the future evolution of two-soliton waves under two distinct speed ratios are exhibited.

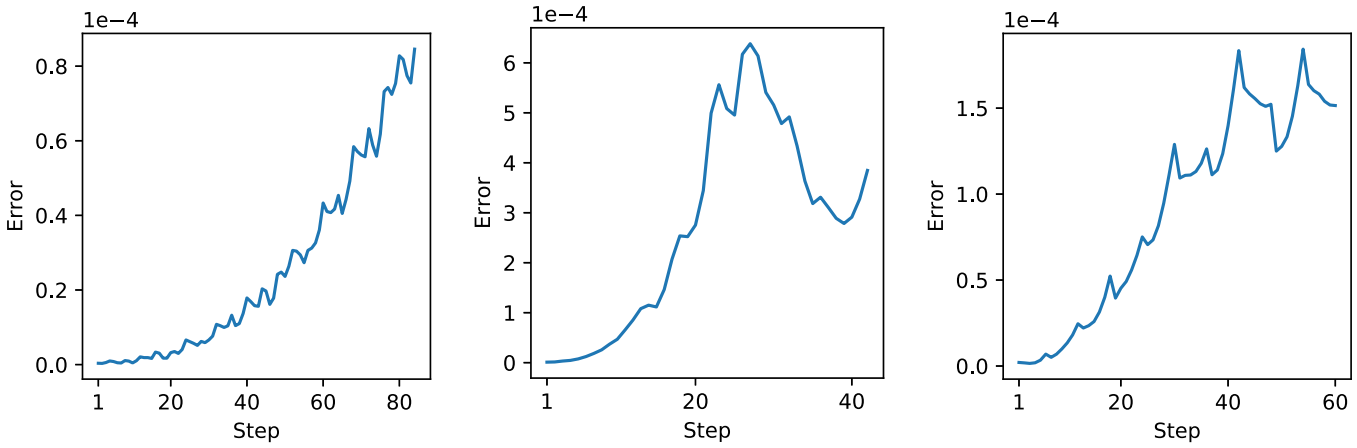


FIG. 6. Prediction error vs time step. From left to right are the plots for a single solitary wave, the collision of two solitary waves under a larger and smaller speed ratio, respectively. In general, the prediction error shows an increasing tendency along with the increase of the prediction time step.

discrete systems, such as the Toda lattice.[28] One beautiful feature in the interaction process of two or more solitary waves is that they can cross each other without change of any property. That means, their magnitude, shape, and velocity are well conserved after mutual interaction. As the main demonstrative example, we would consider in this section to replicate this interesting feature by using the attention-based seq2seq approach.

1. Single solitary wave

The KdV equation (10) is particularly notable because it is one of the earliest models known to have soliton solutions. By means of the inverse scattering transform, one can explicitly write down the single soliton solution to (10),

$$u(x,t) = \frac{1}{2}c \cdot \operatorname{sech}^2\left[\frac{\sqrt{c}}{2}(x - ct)\right], \tag{11}$$

where c, assigned with the value of 1, represents the phase speed. One can also verify this solution by simply taking it back to Eq. (10). For the single solitary wave, the position of its peak at time t is $x = c \cdot t$.

From Eq. (11), the synthetic data for a solitary wave can be sampled equally on the time domain $[-60, 50]$ with an interval of $\Delta t = 0.5$. The spatial domain $[-85, 65]$ is also equally discretized with a grid step of $\Delta x = 0.5$. With all of training sequences designed and collected from the synthetic data, our attention-based model, as described in Sec. II, is trained to simulate the evolution of the single solitary wave. As shown in the leftmost column of both Figs. 4 and 5, the model achieves fairly accurate even a 40-step ahead prediction. Furthermore, we also calculated the prediction error, according to Eq. (9), and found that it grows exponentially with the increase of the prediction time step, as shown in the first picture of Fig. 6. More details about the model parameters are summarized in Table I.

2. The collision of two solitary waves

Besides the single-soliton solution, the KdV equation also admits multiple-soliton solutions; see Ref. 29 for the explicit form of the two-soliton solution and Ref. 30 for the n-soliton solution. In this section, we consider the interaction between two right-moving solitary waves. The entire interaction process includes three phases. In the first phase, a fast solitary wave with speed c_1 chases after another slow solitary wave with speed c_2. After a finite duration of time, they enter the second phase during which the two waves are expected to collide with each other and merge as a parent entity. Finally, in the third phase, the parent entity divides into two solitary waves. After the interaction, the two solitary waves keep their original speeds and shapes unchanged, but their orders are exchanged in a way that the slow wave now chases after the fast wave with an increasing distance between them. Even more interesting is, when the speed ratio c_1/c_2 is greater than the critical value of three, the parent entity has only one single peak. While for $c_1/c_2 < 3$, there will be two peaks, which are equal in height for the parent entity.

TABLE I. Summary of model parameters of all systems: q represents the dimension of input and output of the decoder, l helps control the input dimension of the encoder (namely, the input dimension of the encoder is $q + 2l$), and S and T are the unrolling times of the encoder and decoder, respectively.

System	q	l	S	T
Single-soliton	9	4	10	4
Two-soliton: greater speed ratio	7	4	7	2
Two-soliton: smaller speed ratio	8	5	14	6
Lorenz	3	0	10	4
Fisher	9	4	10	4
Burger	9	4	12	4
Sine-Gordon	9	4	10	4

As the main demonstrative example, the aim here is to make prediction for the interesting phenomenon occurring in the second phase but based on the data sampled from the first phase. In the first case, we set $c_1 = 1$ and $c_2 = 0.3$ so that $c_1/c_2 > 3$. The numerical solution to Eq. (10) is constructed by discretizing its exact two-soliton solution on the domain $(x, t) \in [-120, 60] \times [-100, 50]$ at a grid resolution of 360×200. While in the second case, the wave speeds are set to $c_1 = 0.5$ and $c_2 = 0.3$ so that $c_1/c_2 < 3$. The data are sampled on the domain $(x, t) \in [-95, 60] \times [-100, 50]$ at a grid resolution of 310×200. Compared with the single solitary wave, the evolutionary behavior of two solitary waves is much more complicated due to their mutual interaction and, therefore, pose a special challenge to the forecasting task. However, as vividly shown in the middle column of both Figs. 4 and 5, our approach succeeds in capturing the occurrence of the single-peak parent entity for the case $c_1/c_2 > 3$. Likewise, under the speed ratio $c_1/c_2 < 3$, the happening of the merged double-peak entity is predicted and exhibited in the rightmost panel of Figs. 4 and 5. To evaluate the forecasting performance of our approach, we use the MSE, defined in Eq. (9), as the prediction error measure. As plotted in Fig. 6, the prediction error increases as the prediction horizon increases, generally. The successful application of prediction for such intriguing phenomena further testifies the higher potential of our method in boosting interdisciplinary research between engineering and physics. The configuration of the model parameters can be found in Table I.

B. Chaotic Lorenz system

To examine the effectiveness of the attention-based seq2seq model, we compare it with another kind of technique known as "reservoir computing" on a canonical model for dynamics, the Lorenz system. We compare with the previous best result of using the reservoir computing approach to this prediction task, which was published in Ref. 31. To ensure these two methods are comparable, we train the attention-based seq2seq network on the same training set and evaluated on the same test set as theirs.

The evolution of the Lorenz system is given by three differential equations,

$$\begin{aligned} \dot{x} &= \sigma(y - x), \\ \dot{y} &= x(\rho - z) - y, \\ \dot{z} &= xy - \beta z, \end{aligned} \tag{12}$$

where the parameters are chosen as certain values $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$ to produce chaotic solutions that evolve around the so-called Lorenz attractor. Starting from the initial value $[x(0), y(0), z(0)] = [2, 1, 1]$, the numerical solution to Eq. (12) is collected from the interval $t \in [0, 225]$ with a sampling time of $\Delta t = 0.25$. However, the data sampled on $t \in [0, 100]$ are discarded to avoid a transient evolution process but take only data sampled from $[100, 200]$ to train the model, and the remaining data are used for validation. This procedure for picking up data is exactly the same as used in Ref. 31. For the Lorenz system, $\boldsymbol{u}(t) = [x(t), y(t), z(t)]^T$, and thus, $Q = 3$. Since Q is quite small, we simply choose $q = 3$ and $l = 0$, which results in $\boldsymbol{v}(t) = \boldsymbol{u}(t)$ and $\boldsymbol{w}(t) = \boldsymbol{v}(t)$. In Fig. 7, the upper three panels display the comparison between the predicted and true trajectories, while in the lower panel, the prediction error vs time span is shown. From the upper three panels of Fig. 7, we can find that, after $t = 9.5$, the predicted trajectories begin to deviate from the original trajectories significantly. In the meantime, there is an obvious error peak occurred around this moment. Therefore, we use an error threshold, plotted with a horizontal dashed line that goes right across this peak, to help distinguish the divergent moment. Using the same training data, our method is capable of closely tracking the actual trajectories for about a time span of 9.5, better than that reported in Ref. 31 where they use reservoir computing to successfully predict the Lorenz system over a time span of 7.

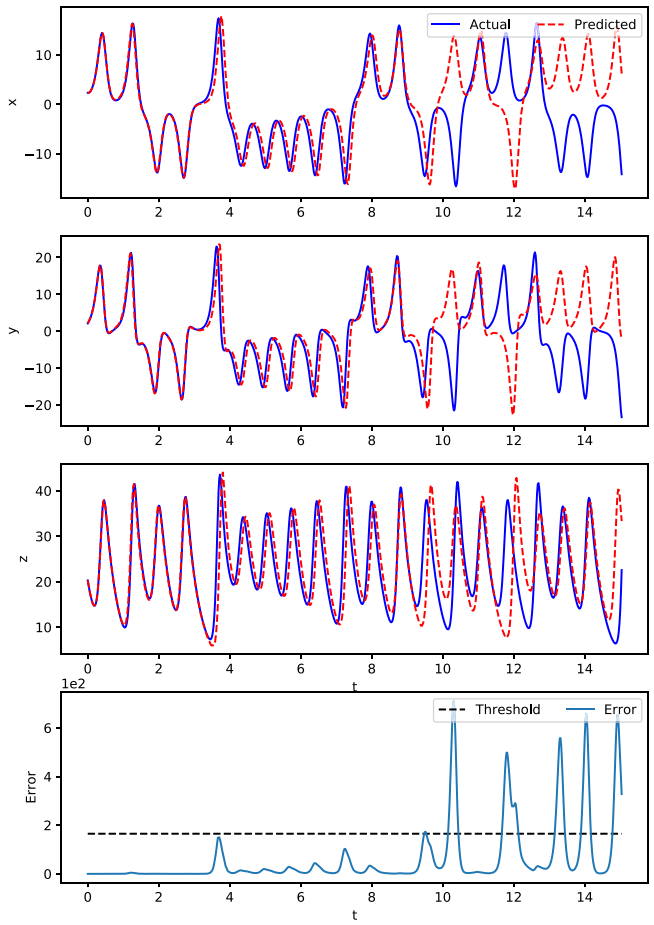
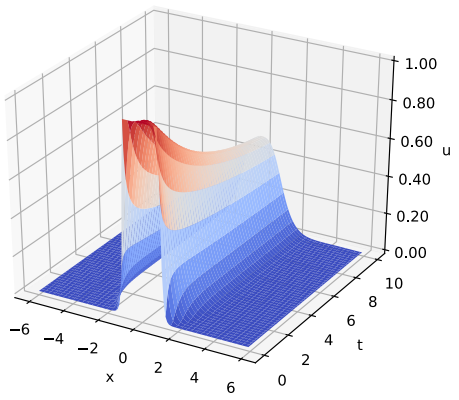
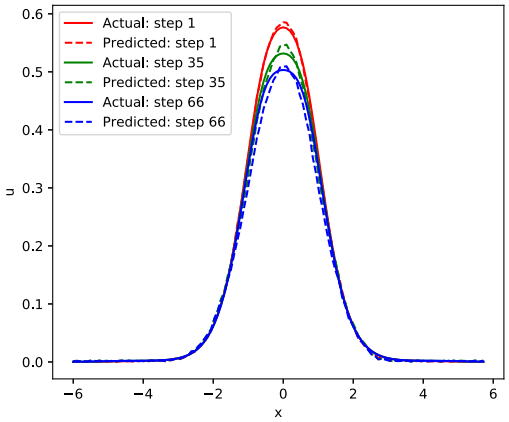
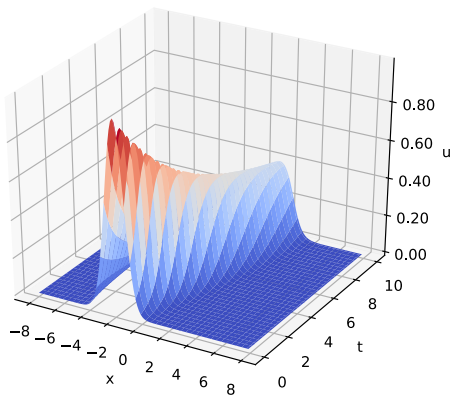
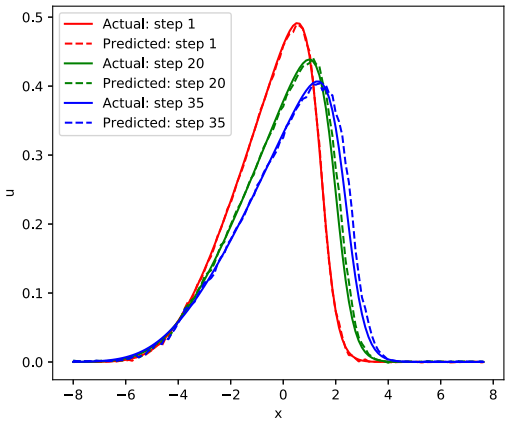
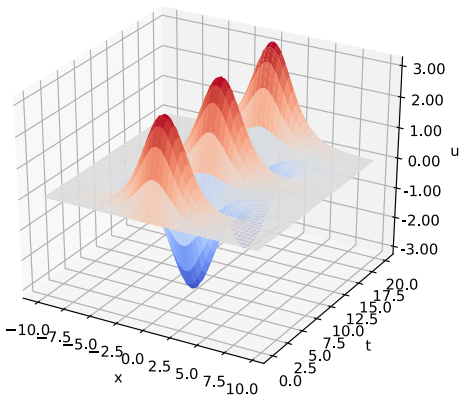
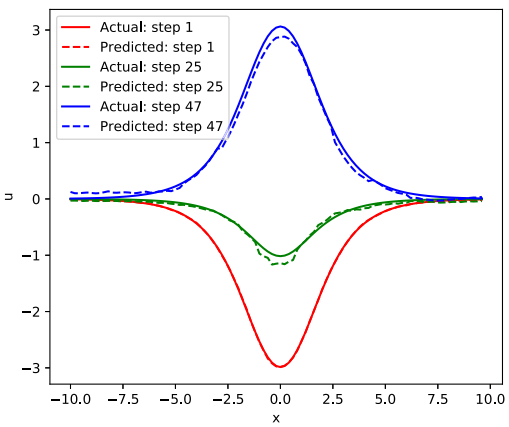


FIG. 7. In the upper three panels, the prediction of three trajectories of the Lorenz system is exhibited. The blue solid line is plotted with actual values, and the red dashed line is plotted with predicted values. In the lower panel, prediction error, calculated according to Eq. (9), vs time span is displayed. Besides, an error threshold is plotted with a black dashed line to help distinguish the moment, which is around $t = 9.5$, indicating significant deviation between predicted trajectories and original trajectories.

TABLE II. Summary of predicted results for three canonical models of mathematical physics. In each example, their equation, collected data, and multistep prediction are exhibited.

PDE	Collected data	Predicted vs actual values
Fisher: $u_t = u_{xx} + u - u^2$, $x \in [-6, 6]$, $\Delta x = 0.937$, $t \in [0, 10]$, $\Delta t = 0.05$.		
Burger: $u_t + uu_x = \epsilon u_{xx}$, $x \in [-8, 8]$, $\Delta x = 0.125$, $t \in [0, 10]$, $\Delta t = 0.1$.		
Sine-Gordon: $u_{tt} = u_{xx} - \sin(u)$, $x \in [-10, 10]$, $\Delta x = 0.2$, $t \in [0, 14]$, $\Delta t = 0.1$.		

C. Other PDEs

Apart from the examples presented above, we further test the applicability of our approach on three more prototypical PDEs appeared in various scientific subjects, including Burger's equation, Fisher's equation, and sine-Gordon equation. Among them, Fisher's equation has a long-standing history in the context of population dynamics to describe the spatial spread of an advantageous allele.[32] Burger's equation is a model for a nonlinear wave propagation, especially in fluid mechanics. The sine-Gordon equation governs the propagation of a slip in an infinite chain of elastically bound atoms lying over a fixed lower chain of similar atoms.[33] To be short, their equations, domains of definition as well as sampling intervals are listed in Table II. All of these dynamics generate high-dimensional data evolving across space as well as time, making it difficult to predict the future evolution. For both Fisher's and Burger's equations, we use the original datasets, which are available in Ref. 34, to train the models and then make a prediction. As for the sine-Gordon equation, it admits a breather solution of the form

$$u(x,t) = 4 \cdot \arctan\left(\frac{\sin(t/\sqrt{2})}{\cosh(x/\sqrt{2})}\right), \tag{13}$$

from which the data needed are directly created. In the case of Fisher's equation, we predict up to 66 steps ahead given $S = 10$ steps of states. For Burger's equation, we predict up to 35 steps ahead with $S = 12$ steps of states as inputs. For the sine-Gordon equation, we predict up to 47 steps ahead using $S = 10$ steps of states. As reported in Table II, the attention-based seq2seq model can predict the future evolution of all these dynamics with great precision.

IV. CONCLUSIONS

In summary, we have presented a novel attention-based seq2seq architecture for spatiotemporal system prediction. By encoding the given input sequence into a series of source hidden states, this neural network forms a sort of memory, where accumulated information is stored. Later, during the decoding phase, the output sequence is generated to make a multistep forward prediction. In particular, we have demonstrated the potential application value of this approach by applying it to forecast the evolution of a single solitary wave and the interesting interaction behavior between two solitary waves. In addition, we have also tested its effectiveness on a chaotic system and three further spatiotemporal systems. In the future work, we will focus on combining a convolutional neural network, which is believed to be an excellent feature extractor, with the attention-based seq2seq architecture to achieve better performance in spatiotemporal data prediction. In addition, more experiments shall be conducted, especially using real-world datasets.

ACKNOWLEDGMENTS

We thank the authors of Ref. 31, especially Zhixin Lu and Edward Ott, for sharing their code.

REFERENCES

[1]R. S. Cantrell and C. Cosner, *Spatial Ecology via Reaction-Diffusion Equations* (John Wiley & Sons, 2004).
[2]J. Smoller, *Shock Waves and Reaction-Diffusion Equations* (Springer Science & Business Media, 2012), Vol. 258.
[3]S. Patankar, *Numerical Heat Transfer and Fluid Flow* (CRC Press, 2018).
[4]B. R. Munson, T. H. Okiishi, W. W. Huebsch, and A. P. Rothmayer, *Fluid Mechanics* (Wiley, Singapore, 2013).
[5]U. Parlitz and C. Merkwirth, "Prediction of spatiotemporal time series based on reconstructed local states," Phys. Rev. Lett. **84**, 1890 (2000).
[6]L. Guo and S. A. Billings, "State-space reconstruction and spatio-temporal prediction of lattice dynamical systems," IEEE Trans. Automat. Control **52**, 622–632 (2007).
[7]J. Shawe-Taylor, N. Cristianini *et al.*, *Kernel Methods for Pattern Analysis* (Cambridge University Press, 2004).
[8]C. K. Williams and C. E. Rasmussen, "Gaussian processes for regression," in *Advances in Neural Information Processing Systems* (MIT Press, 1996), pp. 514–520.
[9]K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," arXiv:1502.03044 (2015).
[10]J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott, "Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach," Phys. Rev. Lett. **120**, 024102 (2018).
[11]H. Jaeger and H. Haas, "Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication," Science **304**, 78–80 (2004).
[12]K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," arXiv:1406.1078 (2014).
[13]K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," arXiv:1409.1259 (2014).
[14]I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in Neural Information Processing Systems* (Curran Associates, 2014), pp. 3104–3112.
[15]A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2015), pp. 3128–3137.
[16]O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2015), pp. 3156–3164.
[17]D. Tang, B. Qin, and T. Liu, "Document modeling with gated recurrent neural network for sentiment classification," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (Association for Computational Linguistics, 2015), pp. 1422–1432.
[18]D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," arXiv:1409.0473 (2014).
[19]M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," arXiv:1508.04025 (2015).
[20]S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Comput. **9**, 1735–1780 (1997).
[21]V. Schwämmle and H. J. Herrmann, "Geomorphology: Solitary wave behaviour of sand dunes," Nature **426**, 619 (2003).
[22]S. Amiranashvili, U. Bandelow, and N. Akhmediev, "Few-cycle optical solitary waves in nonlinear dispersive media," Phys. Rev. A **87**, 013805 (2013).
[23]A. Choudhuri and K. Porsezian, "Dark-in-the-bright solitary wave solution of higher-order nonlinear Schrödinger equation with non-Kerr terms," Opt. Commun. **285**, 364–367 (2012).
[24]A. Hasegawa and Y. Kodama, *Solitons in Optical Communications* (Oxford University Press, New York, 1995), Vol. 7.
[25]C. A. Cattell, J. Crumley, J. Dombeck, J. R. Wygant, and F. Mozer, "Polar observations of solitary waves at the earth's magnetopause," Geophys. Res. Lett. **29**, 9-1, https://doi.org/10.1029/2001GL014046 (2002).
[26]S. Russell, in *Fourteenth Meeting of the British Association of the Advancement of Science* (John Murray, 1844).
[27]D. J. Korteweg and G. De Vries, "XLI. On the change of form of long waves advancing in a rectangular canal, and on a new type of long stationary waves," Lond. Edinb. Dubl. Philos. Mag. J. Sci. **39**, 422–443 (1895).
[28]M. Toda, "Vibration of a chain with nonlinear interaction," in *Selected Papers of Morikazu Toda* (World Scientific, 1993), pp. 97–102.

[29]S. Anco and M. R. Willoughby, see http://lie.math.brocku.ca/sanco/solitons/kdv_solitons.php for "KdV solitons" (accessed 26 June 2019).
[30]G. B. Whitham, *Linear and Nonlinear Waves* (John Wiley & Sons, 2011), Vol. 42.
[31]J. Pathak, Z. Lu, B. R. Hunt, M. Girvan, and E. Ott, "Using machine learning to replicate chaotic attractors and calculate Lyapunov exponents from data," Chaos **27**, 121102 (2017).
[32]R. A. Fisher, "The wave of advance of advantageous genes," Ann. Eugen. **7**, 355–369 (1937).
[33]A. Barone, F. Esposito, C. Magee, and A. Scott, "Theory and applications of the sine-Gordon equation," La Rivista del Nuovo Cimento (1971–1977) **1**, 227–267 (1971).
[34]S. H. Rudy, S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Data-driven discovery of partial differential equations," Sci. Adv. **3**, e1602614 (2017).