

## Research Article

# A Novel Method for Sea Surface Temperature Prediction Based on Deep Learning

Xuan Yu <sup>1</sup>, Suixiang Shi <sup>1,2</sup>, Lingyu Xu <sup>1,3</sup>, Yaya Liu <sup>1</sup>, Qingsheng Miao <sup>4</sup>,  
and Miao Sun <sup>2</sup>

<sup>1</sup>School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China

<sup>2</sup>Key Laboratory of Digital Ocean, National Marine Data and Information Service, Tianjin 300171, China

<sup>3</sup>Shanghai Institute for Advanced Communication and Data Science, Shanghai University, Shanghai 200444, China

<sup>4</sup>Marine Data Center, National Marine Data and Information Service, Tianjin 300171, China

Correspondence should be addressed to Lingyu Xu; shuv60@126.com

Received 16 January 2020; Revised 27 March 2020; Accepted 18 April 2020; Published 7 May 2020

Academic Editor: Piotr Jędrzejowicz

Copyright © 2020 Xuan Yu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Sea surface temperature (SST) forecasting is the task of predicting future values of a given sequence using historical SST data, which is beneficial for observing and studying hydroclimatic variability. Most previous studies ignore the spatial information in SST prediction and the forecasting models have limitations to process the large-scale SST data. A novel model of SST prediction integrated Deep Gated Recurrent Unit and Convolutional Neural Network (DGCnetwork) is proposed in this paper. The DGCnetwork has a compact structure and focuses on learning deep long-term dependencies in SST time series. Temporal information and spatial information are all included in our procedure. Differential Evolution algorithm is applied in order to configure DGCnetwork's optimum architecture. Optimum Interpolation Sea Surface Temperature (OISST) data is selected to conduct experiments in this paper, which has good temporal homogeneity and feature resolution. The experiments demonstrate that the DGCnetwork significantly obtains excellent forecasting result, predicting SST by different lengths flexibly and accurately. On the East China Sea dataset and the Yellow Sea dataset, the accuracy of the prediction results is above 98% on the whole and all mean absolute error (MAE) values are lower than 0.33°C. Compared with the other models, root mean square error (RMSE), root mean square percentage error (RMSPE), and mean absolute percentage Error (MAPE) of the proposed approach reduce at least 0.1154, 0.2594, and 0.3938. The experiments of SST time series show that the DGCnetwork model maintains good prediction results, better performance, and stronger stability, which has reached the most advanced level internationally.

## 1. Introduction

Analyzing sea surface temperature (SST), an essential parameter for studying the marine ecosystem and global climate can efficiently help us to explore the ocean conditions and understand the climatic dynamics. For a long time, SST has been reported the role in different fields of science, such as providing significant predictive information about hydroclimatic variability [1–3], supplying basis for revealing the spatial distribution of biological environmental factors [4], and as an indicator to observe and monitor marine disasters [5, 6]. Because of large variations in heat flux, radiation, and diurnal wind near the sea surface, the prediction of SST has always been a highly uncertain issue.

Recent years, many methods have been developed for SST prediction. There are primarily two types of forecasting strategies: physical techniques and statistical techniques [7]. The former is aimed at the physical properties of the ocean, using a series of differential equations to describe the SST data. Statistical models, including linear regression [8], thogonal functions [9], support vector machines (SVM) [10, 11], and artificial neural networks (ANN) [7], are extensively used time series-based approaches for SST prediction. These models are designed to predict SST time series by establishing a relationship between historical values and a predictor. The previous studies found that the SST prediction result is often unstable. Traditional methods have some disadvantages in processing large-scale SST data, such as

slow speed, difficulty in fitting, occupying much machine memory, and computing time.

Recurrent Neural Networks (RNNs), such as Long Short-Term Memory (LSTM) [12, 13] and Gated Recurrent Unit (GRU) [14–16], have shown to achieve the state-of-the-art results in many applications with time series or sequential data. RNNs enjoy several nice properties such as strong prediction performance as well as the ability to capture long-term temporal dependencies and variable-length observations. LSTM and GRU introduce gate mechanism to overcome the problems of vanishing and explosion of gradients in traditional RNNs when learning long-term dependencies. GRU network is faster and has the simpler structure than LSTM training and performs well in sequence learning tasks [6, 17]. Recently, SST prediction progresses further with the advent of deep learning [15] and neural networks methods. Zhang et al. [13] adopted LSTM to predict SST and obtained good prediction results. Based on the existing contributions, however, there are three problems with the studies. Firstly, mining the information of time series by the model structure of a single network layer is limited. Secondly, the current examination did not consider the temporal and spatial characteristics of SST time series simultaneously. In other terms, the isolated prediction of each point ignores the interaction between the SSTs of different points. Thirdly, the previous ways did not take into account the optimization strategy of the parameters in the prediction model.

In our work, an innovative approach is constructed for SST prediction, which is the Deep Gated Recurrent Unit and Convolutional Neural Network (DGCnetwork). The DGCnetwork model is constructed combining the deep GRU and CNN. The deep GRU layers and the convolutional layer are used to extract the deep hidden temporal features and spatial characteristics of SST data, respectively. We apply one full-connected layer to combine all features into global features and map the output of the previous layer to a final prediction. Increasing the depth of a neural network is an effective way to improve the overall performance [15]. Because the proposed model has a more compact representation than the single network layer, it will be better promoted and performed when applying to prediction of SST data. Besides, temporal information and spatial information are all included in our procedure. Research shows that the SST of a specific point interacts with the SST of its surrounding points [4, 18]. Therefore, when we predict the SST of a certain location, the proposed approach combines the historical SST information of its nearby location.

The efficiency of the DGCnetwork depends on several hyperparameters, namely, the number of neurons in every layer and the number of epochs. Without choosing appropriate network parameters, it slows down the training speed and the network is vulnerable to interference in the nearest local minimum. Because the initial values of hyperparameters play a vital role in the training outputs of the neural network [19, 20], we adopt the Differential Evolution algorithm (DE) to infer optimal selection for the proposed model's hyperparameters. DE can leverage individual local information and population global information to search for the optimal solution, which has been widely applied [21, 22].

The sequel of the paper is organized as follows. The procedures of the DGCnetwork predicting model are explained in detail in Section 2. Section 3 provides the experimental results and discussions. Finally, Section 4 summarizes the conclusions.

## 2. Methodology

*2.1. The DGCnetwork.* In order to solve the task of SST time series prediction, this paper proposes the DGCnetwork model based on deep learning with deep GRU and CNN network. The DGCnetwork architecture can adapt by learning the nonlinearity and complexity of SST time series data, which includes multiple GRU layers, one CNN layer, and one full-connected layer. After the prediction point is selected, we express the SST time series of the prediction point and its nearest points in a matrix form to input into the model. In the model, each GRU layer operates at different time scales and the CNN layer captures spatial feature. The full-connected layer combines all features into global features and maps the output of previous layer to a final prediction. They process the certain part of the prediction task. The output of the previous GRU layer is the input of the next GRU layer. The output of the last GRU layer is the input of the CNN layer and finally generates the prediction result by the full-connected layer. As such, the model is an end-to-end prediction network. Stacking more GRU layers to the recurrent connections between the units in the model and the feed-forward connections between units in a GRU layer and the GRU layer above, it is helpful to research the large-scale SST time series. This ensures an improved learning with more sophisticated conditional distributions of SST time series data. Also, it can perform hierarchical processing on difficult temporal tasks, and more naturally, capture the deep feature of data sequences. The hyperparameters in the network layers are chosen by the DE algorithm.

As shown in Figure 1, the DGCnetwork architecture has three GRU layers, one CNN layer, and one full-connected layer. We define the SST time series as  $X(x_1, x_2, \dots, x_p, \dots, x_n)$ .  $x_t$  represents the SST value at time  $t$  and  $n$  is the length of SST time series. Multiple time series constitute the input matrix  $M(X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8, X_9)$ , where  $X_1$  is the predicting point and  $X_2, X_3, X_4, X_5, X_6, X_7, X_8$ , and  $X_9$  are the surrounding points. In the DGCnetwork architecture, the input at time  $t$  and  $M_t$  is introduced to the first GRU layer along with the previous hidden state  $h_{t-1}^{(1)}$ , and the superscript (1) denotes the first GRU layer. The hidden state at time  $t$ ,  $h_t^{(1)}$  and  $h_t^{(1)}$  are computed, as shown in Section 2.2.  $h_t^{(1)}$  goes forward to the time  $t+1$  and also moves forward to the second GRU layer.  $h_{t-1}^{(2)}$  in GRU layer 2 is computed by  $h_t^{(1)}$  and  $h_{t-1}^{(2)}$ , which goes forward to the time  $t+2$  and also moves forward to the third GRU layer in the same way. The output of the third GRU layer is the input of the CNN layer.  $c_i^l$  and  $c_j^l$  are computed, as shown in Section 2.3. The output of the CNN layer is the input of the full-connected layer. Finally, the predicted value  $y_t$  is obtained by the full-connected layer.

Our proposed DGCnetwork model has three advantages. To begin with, each layer can process some part of the predicted task and GRU layer and pass it on to the CNN

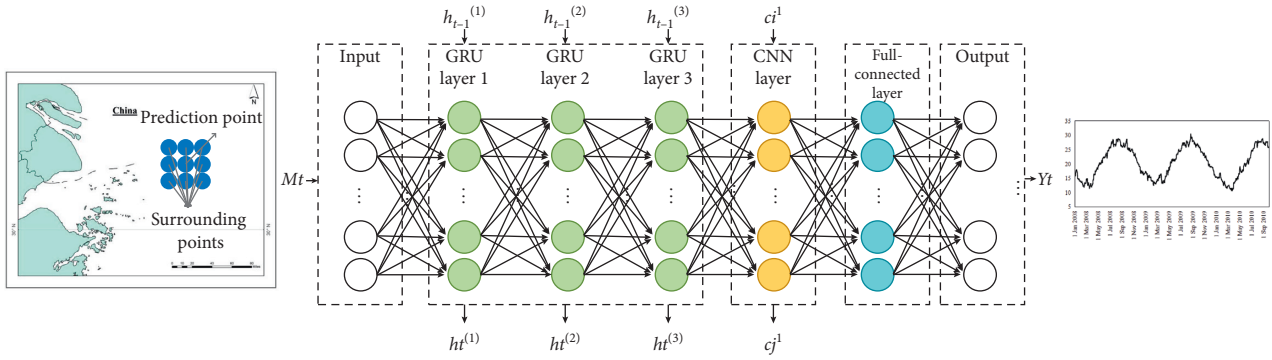


FIGURE 1: The architecture of the deep gated recurrent unit and convolutional neural network.

layer, until finally the last full-connected layer provides the predicted SST value. Secondly, the hidden state in the model at each level is allowed to deal with at a different time scale which could mine the deep spatial-temporal feature of the data. Thirdly, the optimal hyperparameters in the model are selected directly by the DE algorithm. The three advantages have great benefit in case of handling the predicting problem of large-scale SST time series data.

### 2.2. Temporal Feature Extraction by GATED Recurrent Unit.

This paper adopts GRU to capture the temporal relationship among SST time series data. GRU was first proposed by Bahdanau et al. [16], which is more accurate than conventional RNNs and more simple than LSTM. In the topological structure of GRU, the forget gate and the input gate are integrated into an update gate. GRU mixes the cell state with the hidden state, and the information flow inside it is modulated by the reset gate and the update gate. As illustrated in Figure 2,  $r_t$  and  $z_t$  are the reset gate and update gate, respectively, and  $h_t$  and  $\tilde{h}_t$  represent the activity value and the candidate activity value, respectively. The mechanism of the gates could extract the temporal relationship among time series data.

The reset gate  $r_t$  can control the influence containing information of the last implicit state  $h_{t-1}$  on the current information  $x_t$ , which determines how much information was forgotten in the past. If the value of  $r_t$  approximates 0, the information of the previous implicit state is discarded.

The update gate  $z_t$  is used to control the importance of the past implicit state  $h_{t-1}$  at the present moment  $h_t$ . If the value of  $z_t$  is always approximately 1, the information of  $h_{t-1}$  is always saved through time and passed to  $h_t$ . This makes the gradient reversely propagate, effectively solving the gradient vanishing problem of RNN. The whole computation can be defined by a series of equations as follows:

$$\begin{aligned} r_t &= \sigma(W_r \cdot [h_{t-1}, x_t]), \\ z_t &= \sigma(W_z \cdot [h_{t-1}, x_t]), \\ \tilde{h}_t &= \tanh(W_{\tilde{h}} \cdot [r_t * h_{t-1}, x_t]), \\ h_t &= (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t, \\ y_t &= \sigma(W_o \cdot h_t), \end{aligned} \quad (1)$$

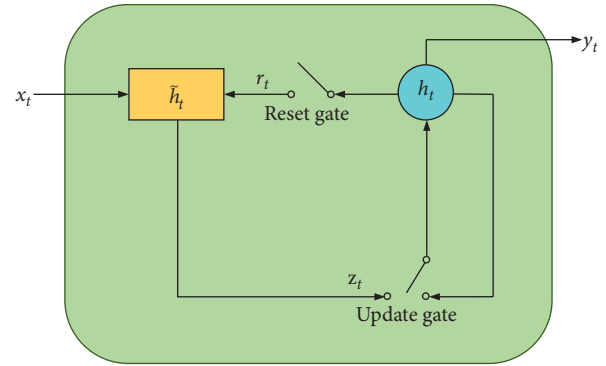


FIGURE 2: The architecture of gated recurrent unit.

where  $\sigma$  denotes the sigmoid function,  $W_r$ ,  $W_z$ , and  $W_{\tilde{h}}$  are the recurrent weight matrices.  $[\ ]$  represents the two vectors are connected and  $*$  is the multiplication of matrix elements.

The eigenvalues are required to enter in the chronological order when GRU networks are dealing with the SST time series. Both the sigmoid function  $\sigma$  and the hyperbolic cosine function  $\tanh$  are adopted as activation functions in the structure. During the training process, the loss of the objective function from the training sets is minimized.

### 2.3. Spatial Feature Extraction by Convolutional Neural Network.

CNN is a special structure of ANN, which has the ability to deal with high-dimensional data. It is general utilized in image recognition, recommender systems, and natural language processing [23]. Since there is interaction between the SST of the adjacent positions, this paper combines the historical SST information of the prediction point and its surrounding points to forecast the target point. In the proposed model, we apply the CNN layer as a module to mine the spatial information of SST time series (Figure 3). After processing the matrix  $M$  in the GRU layers, the matrix  $M'$  is input into the CNN layer. To begin with, multiple two-dimensional matrices at different time periods are stacked into three-dimensional matrix blocks. Then, spatial feature extraction can be achieved by a roll over convolution layer. Afterwards, the outputs of convolution operation are adopted in pooling process. The role of the pooling layer is lowering the computational burden and improving

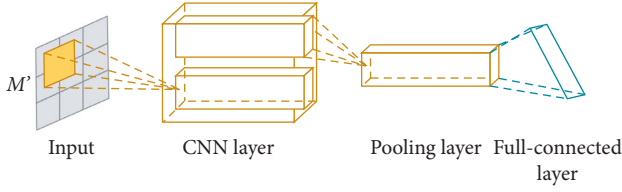


FIGURE 3: Spatial feature extraction by convolutional neural network.

operation efficiency by compressing the feature map. Finally, the abstract feature set is flattened to a one-dimensional vector and connected with the full-connected layer. CNN has the advantages of local perception, sparse interactions, and parameter sharing. Its weight-sharing network structure makes it more similar to the biological neural network and has achieved good results in time series research [24]. The output of the CNN layer can be written as follows:

$$c_j^l = f\left(\sum_{i \in Z_j} c_i^{l-1} \times k_{ij}^l + b_j^l\right), \quad (2)$$

where  $Z_j$  is the collection of input maps. Each output map is given an additive bias  $b$ ; however, for a particular output map, the input maps will be convolved with distinct kernels. The kernels applied to map  $i$  are different for output maps  $j$  and  $k$  when output map  $j$  and map  $k$  both sum over input map  $i$ .

**2.4. Optimization of Network Parameters by Differential Evolution Algorithm.** There are some decision parameters to be optimized for the DGCnetwork's training. This paper applies DE algorithm to optimally select the values of each hyperparameter in the predicting model, including the number of neurons in the GRU layers and the number of epochs. The optimization strategy is convenient for us to seek out the best model's structure in order to minimize the difference between the predicting and actual values. The DE algorithm is a simple, population-based, and direct-search algorithm for optimizing the multimodal functions [25]. DE is reliable due to its ability to reach global optimum values and rapid convergence with fewer control parameters. Previous research states that the DE outperforms several other well-known optimization algorithms in terms of convergence speed and stability [26]. The standard DE consists of four main operations, which are initialization, mutation, crossover, and selection. The four operations make the model evolve to a higher fitness to achieve the goal of optimal solution.

### 3. Experiments

**3.1. Data and Software.** The data used in our research is the Optimum Interpolation Sea Surface Temperature (OISST), an optimally interpolated SST, from the National Oceanic and Atmospheric Administration (NOAA). Because OISST has good temporal homogeneity and feature resolution [27], it is applied to the analysis and prediction of time series in

our work, studying the OISST data is helpful to research the oceanic features. The data we used in the paper is global grid data, the spatial resolution is  $1^\circ \times 1^\circ$ , and the time resolution is days. We choose the East China Sea and the Yellow Sea as the experimental objects (see Figure 4). This paper creates two SST datasets which are the East China Sea dataset and the Yellow Sea dataset, respectively. Six points are randomly selected on the two datasets. The time length is from January 1, 2001, to July 15, 2017 (6,040 days).

SST data preprocessing and handling are conducted in Python 3.6, relying on the packages numpy and pandas. Deep learning GRU and CNN networks are implemented with keras, a high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano.

**3.2. Evaluation Standard.** In the study, five different indexes are measured in order to estimate the forecasting precision, error, and performance evaluation of the prediction task [28].

Root of mean squared error (RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{N-L+1-i} \sum_{i=1}^n (y_i^{\text{obs}} - y_i^{\text{pre}})^2}. \quad (3)$$

Root mean square percentage error (RMSPE):

$$\text{RMSPE} = \sqrt{\frac{1}{n} \sum_{i=1}^n \left(\frac{y_i^{\text{pre}} - y_i^{\text{obs}}}{y_i^{\text{obs}}}\right)^2} \times 100. \quad (4)$$

Mean absolute percentage Error (MAPE):

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^n \left(\frac{|y_i^{\text{pre}} - y_i^{\text{obs}}|}{y_i^{\text{obs}}}\right) \times 100. \quad (5)$$

Mean absolute error (MAE):

$$\text{MAE} = \frac{1}{n} |y_i^{\text{pre}} - y_i^{\text{obs}}|. \quad (6)$$

Accuracy (ACC):

$$\text{ACC} = \frac{1}{n} \sum_{i=1}^n \frac{|y_i^{\text{pre}} - y_i^{\text{obs}}|}{y_i^{\text{obs}}}. \quad (7)$$

Among them,  $y_i^{\text{obs}}$  and  $y_i^{\text{pre}}$  represent the true value and its predicted value, respectively. The degree of freedom in RMSE is  $N-L+1-i$ , where  $N$  is the number of samples,  $L$  is the length of observations, and  $i$  represents the number of independent variables. In this paper,  $i=2$ . RMSE is smaller and its degree of freedom is larger indicating that the model is more effective and universal [29, 30]. The important property of the RMSPE, MAPE, and MAE is their values closer to 0 imply higher accuracy of the predicting model. The range of ACC is  $[0, 1]$ , and the value closer to 1 corresponds to better performance of the forecasting model. It is widely demonstrated in the previous literature that the five measures are the appropriate tools to assess the performance of the forecasting model [31].

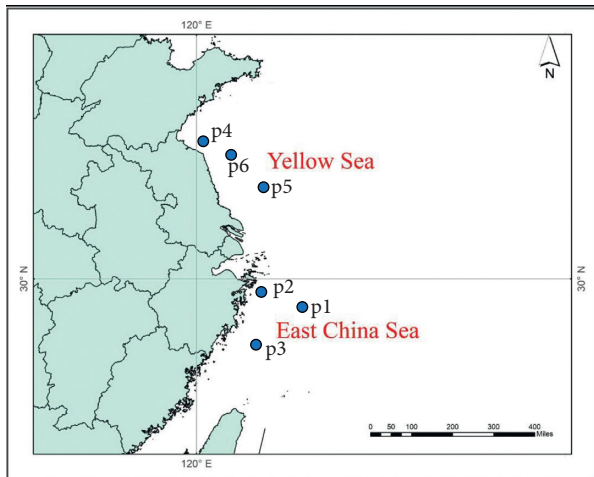


FIGURE 4: Location maps of the study area.

**3.3. Results and Analysis.** There are some important settings in the DGCnetwork model to be determined beforehand. Firstly, we utilize early stopping to prevent overfitting as a further mechanism. This paper sets the maximum early stopping duration to 15. Secondly, the data is split into training, testing, and validation set following the ratio of 3 : 1 : 1. The training set is used for training and the test results are obtained on the verification set. Furthermore, we set the batch size as 40 in the experiments.

We proceed now to show the quantitative and visual results of the proposed DGCnetwork. The results shown in all tables and figures indicate the performance of the model in the validation set. This has been done in concurrence with the widely demonstrated fact, which states, the genuine evaluation for forecasting performance should be based on unseen data not the historical (training and testing) data, which is already seen by the model [31].

In the experiment, we use the different lengths of historical observations to predict the future SST value. The length of historical observations is denoted as  $H$ . In general, if  $H$  is too small, there may not be sufficient sequence information to predict future SST values. Otherwise, with the increase of  $H$ , there may be more noise in the training samples [10, 32]. When the length of historical observations is from 1 day to 60 days, we apply the DGCnetwork to predict the SST for one day. Figures 5(a) and 6(a) show the forecasting accuracy on the East China Sea dataset and the Yellow Sea dataset, respectively. From the results, the accuracy of the six points on the two datasets are all more than 98% with the different  $H$ . Experiments display that the length of historical observations has little effect on the prediction accuracy when the predicting length is one day. Then, this paper adopts the DGCnetwork to predict the SST for one week with the length of historical observations from 7 days to 60 days (as shown in Figures 5(b) and 6(b)). That is to say, SST data from the past  $H$  days are applied to forecast the value for the seventh day in the future. Considering the problem of the insufficient information, our experiment does not perform the case, where  $H$  is less than the predicting length. It is worth mentioning in view of the results

that, as  $H$  increases, the forecasting accuracy has a raise in tendency. This could be attributed to more sequence information which is needed when we predict the longer length. Overall, whether it is forecasting the SST value of the first day or the seventh day in the future with different  $H$ , the prediction effect on the two datasets could achieve satisfying accuracy (98%~99%). Moreover, it is interesting that the accuracy of p1 is better than p2 and p3 on the East China Sea dataset. As we all know, the temperature changes in the distant sea are relatively stable, while the fluctuations in the coastal water's temperature are greater. By observing the location of the three points on the map, we can observe that p1 is farther away from the coast than p2 and p3. This is demonstrated that the temperature changes at p1 are relatively stable; therefore, the forecast performance of p1 is better than p2 and p3. On the Yellow Sea dataset, we could obtain the same finding. The forecast accuracy of p5 is better than p4 and p6 which are near the land.

Since the DGCnetwork contains DE algorithm module, the values of each hyperparameter have been optimally selected. This paper analyzes the best model's structure and the prediction results with the different predicting lengths. On the two SST datasets, the optimal model is used to forecast SST value with the historical observations of 30 days used as an example, the predicting length is set as 3 days, 5 days, 1 week, 2 weeks, and 1 month, respectively. DE algorithm in our predicting model makes it convenient to adjust the deep network to the optimal state when the prediction range changes, avoiding the trouble of parameter adjustment. Table 1 lists the predicting results of p1 and p4 on the two SST datasets, and it is easy to notice the number of neurons in hidden layers accumulate between 10 and 20 and is larger as the predicting length increases. The number of neurons in the neural network determines the number of input features. Very few neurons can cause part data to be lost. The numbers of epochs in the optimal models are clustered around 100. The forecast result gets better obviously when the predicting length reduces; among them, ACC is near 0.99 when the third day's data is forecasted in the future on the two SST datasets. The error of the model remains small when we forecast the SST data after a month (RMSE is 0.6729 on the East China Sea dataset and 0.5681 on the Yellow Sea dataset). The experimental process also indicates the DGCnetwork optimized by DE may be a good choice for SST time series forecasting.

This paper adopts the GRU network to make the comparative analysis of the prediction errors with the proposed method. Figures 7 and 8 depict the prediction results by the two methods when the length of historical observation is 7 days and the predicting length is 1 day. According to the results on the two datasets, it should be pointed out that the prediction results of the six points reflect the same problem. The prediction errors obtained by GRU are more larger near the maximum SST value. However, the DGCnetwork model always maintains small prediction errors and the prediction results are very close to the true SST value. After searching the previous SST prediction studies [13, 32, 33], we find that, in the literature [13], the SST predicting results also have the larger errors near the

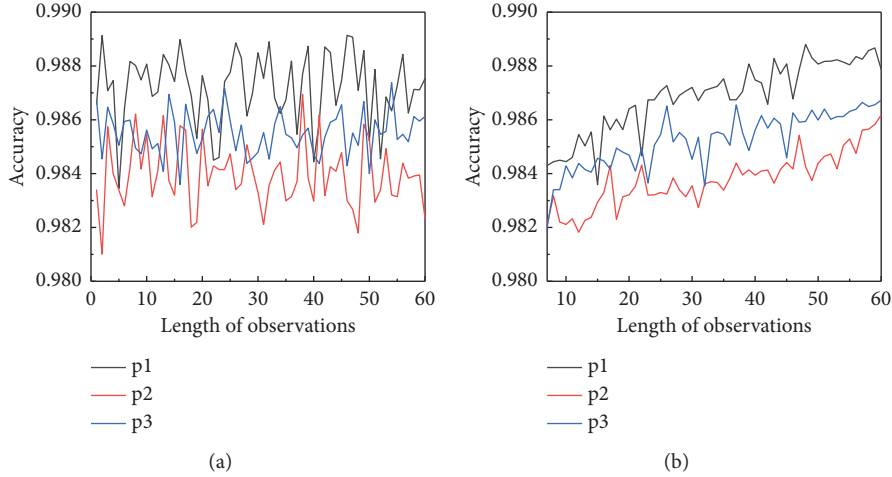


FIGURE 5: The prediction accuracy on the East China Sea dataset with different  $H$ . p1, p2, and p3 represent different points. (a) The results when the predicting length is one day. (b) The results when the predicting length is one week.

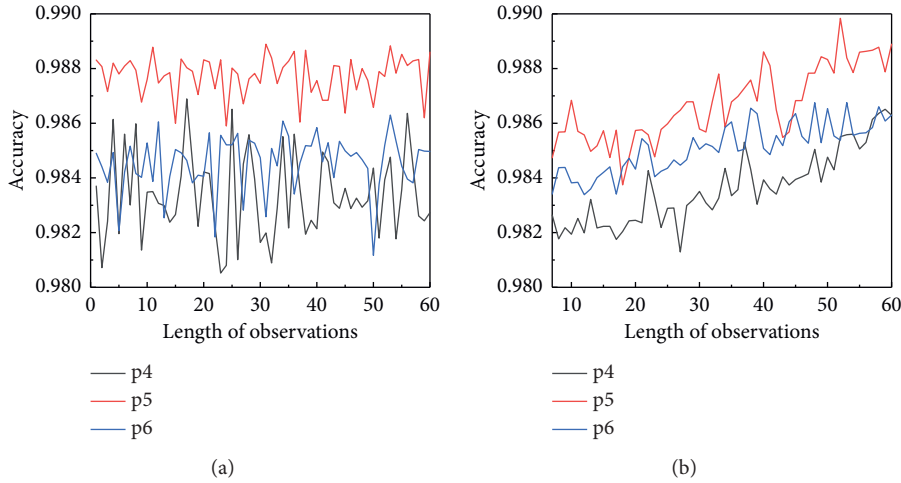


FIGURE 6: The prediction accuracy on the Yellow Sea dataset with different  $H$ . p4, p5, and p6 represent different points. (a) The results when the predicting length is one day. (b) The results when the predicting length is one week.

TABLE 1: The best performance results of the proposed model on the two dataset with different predicting lengths ( $H = 30$  days).

Predicting length	East China Sea dataset				Yellow Sea dataset			
	No. of neurons	No. of epochs	RMSE	ACC	No. of neurons	No. of epochs	RMSE	ACC
3 days	[14, 12, 13]	89	0.4503	0.9895	[15, 14, 13]	90	0.4425	0.9888
5 days	[15, 16, 16]	92	0.4573	0.9842	[14, 15, 16]	95	0.4558	0.9875
1 week	[16, 17, 18]	90	0.5501	0.9828	[16, 16, 17]	100	0.4593	0.9865
2 weeks	[18, 17, 17]	101	0.6631	0.9815	[17, 17, 18]	99	0.4612	0.9836
1 month	[19, 19, 19]	99	0.6729	0.9803	[19, 20, 19]	105	0.5681	0.9817

maximum SST value. So far, however, there has been little discussion about the reason for this phenomenon. This paper analyzes the issue from two aspects: data and method. First of all, SST time series presents obvious periodicity tendency. That is to say, SST data generally reaches its maximum in summer each year. This was demonstrated in some studies that showed in the last two decades; SST has been warming up in the coastal areas of China, and the intensity of extreme high temperature has been significantly

enhanced, especially in spring and summer [18, 34]. Secondly, the shallow architecture, i.e., the single-layer neural network cannot represent efficiently the complex features of time series data, particularly when attempting to process highly nonlinear and long interval time series datasets [35, 36]. On the whole, the single-layer GRU network is difficult to capture the trend of SST data in summer. The proposed method in our research has the higher prediction accuracy because it uses a deep network structure, which can

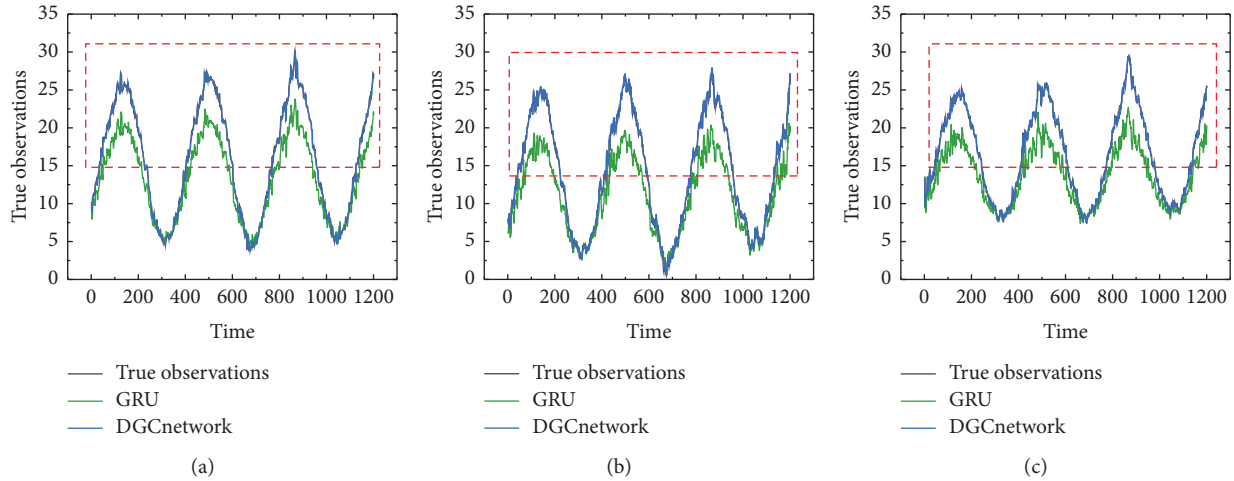


FIGURE 7: Prediction results by GRU and the DGCnetwork on the East China Sea dataset. (a–c) represent the results of p1, p2, and p3, respectively.

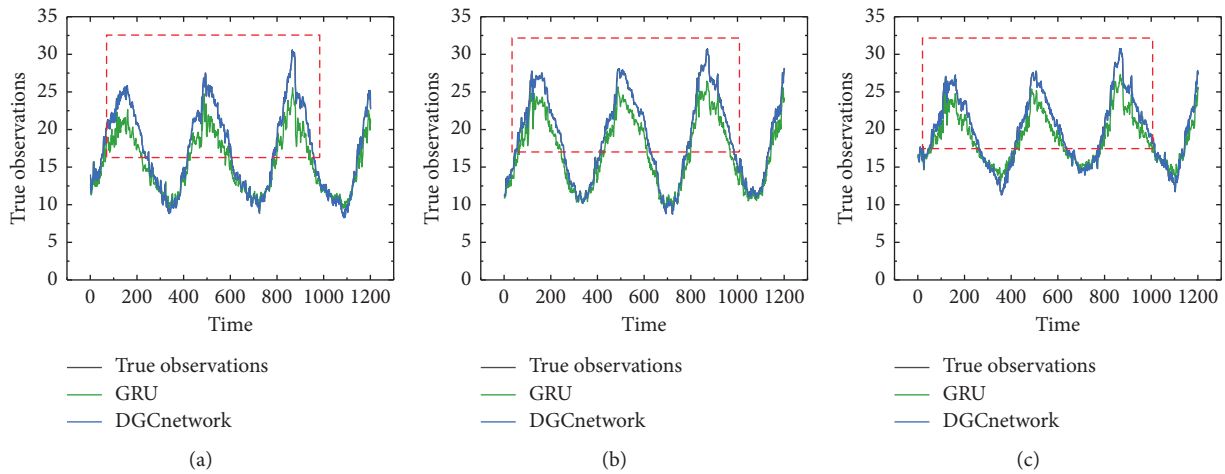


FIGURE 8: Prediction results by GRU and the DGCnetwork on the Yellow Sea dataset. (a–c) represent the results of p4, p5, and p6, respectively.

dig deeper into the spatiotemporal characteristics of SST data. Besides, we set the length of historical observation is 3 days, 15 days, 30 days, and 45 days, respectively, for more experiments. On the East China Sea dataset and the Yellow Sea dataset, the conclusion obtained by the two methods is consistent with 7 days.

Furthermore, in the comparative evaluation experiment, 12 predicting methods are deployed using the two datasets and experimental conditions via different error measures, which covered the classical time series predicting methods and newly published methods in recent years. The 12 predicting methods are Support Vector Regression (SVR), Support Vector Machine (SVM), Autoregressive Integrated Moving Average (ARIMA), Back Propagation Neural Network (BPNN), Radical Basis Function Neural Network (RBFNN), RNN, GRU, LSTM, updated-LSTM [13], GRU-SVM [14], WNN [37], and CEEMDAN-LSTM [12].

The results of the experiment on the two datasets which predict 1 day’s SST value with the length of historical

observation is one week (7 days) are shown in Tables 2 and 3. The smaller RMSE, RMSPE, MAPE, and MAE, the better the prediction results, while ACC is the opposite. The ACC results on the two datasets are 98.81% and 98.30%, which improved 0.86% to 13.90% in contrast with other 12 methods. Our datasets demonstrated that the DGCnetwork plays a role in large-scale SST time series prediction. On the East China Sea dataset, RMSE, RMSPE, MAPE, and MAE of the proposed approach reach 0.4471, 2.0932, 1.5018, and 0.3218°C, respectively, which decreased by 0.1154, 0.2594, 0.3938, and 0.1082°C than the best of the other models. RMSE, RMSPE, MAPE, and MAE is 0.3637, 1.7382, 1.2915, and 0.2673°C, respectively, on the Yellow Sea dataset. The results of evaluation indicators indicate that the method in this paper is more effective than traditional methods or other existing predicting models. The DGCnetwork model has the advantages of higher forecasting precision, better performance, and stronger stability.

TABLE 2: The results of p1 on the East China Sea dataset which predict 1 day's SST value ( $H=7$ ).

Methods	RMSE	RMSPE	MAPE	MAE (°C)	ACC
SVR	1.0802	6.2985	3.2420	0.4300	0.9568
SVM	1.7932	13.7467	5.6707	0.6530	0.9380
ARIMA	1.6182	3.2015	5.0987	0.6035	0.9395
BPNN	3.7294	15.2178	14.0391	3.8745	0.8491
RBFNN	2.8274	15.3271	13.8721	2.6102	0.8691
RNN	2.8104	14.4212	12.2157	2.1739	0.8718
GRU	1.5281	7.7845	7.3024	1.5981	0.9325
LSTM	1.7102	8.2459	7.7929	1.7291	0.9283
Updated-LSTM	0.5625	2.3526	1.8956	0.5895	0.9795
GRU-SVM	1.6711	8.3571	7.5042	1.7328	0.9359
WNN	0.6812	2.5032	1.9817	0.6647	0.9632
CEEMDAN-LSTM	1.5397	7.2351	6.9083	1.5402	0.9368
DGCnetwork	<b>0.4471</b>	<b>2.0932</b>	<b>1.5018</b>	<b>0.3218</b>	<b>0.9881</b>

TABLE 3: The results of p4 on the Yellow Sea dataset which predict 1 day's SST value ( $H=7$ ).

Methods	RMSE	RMSPE	MAPE	MAE (°C)	ACC
SVR	1.1780	6.4541	3.4226	0.5137	0.9482
SVM	1.4618	10.0451	6.2955	0.8457	0.9360
ARIMA	1.4307	9.1634	5.8763	0.8147	0.9398
BPNN	3.6241	14.1903	13.0661	3.7138	0.8503
RBFNN	2.8173	14.1829	12.1397	2.5015	0.8711
RNN	3.0071	15.2845	13.7467	2.4630	0.8548
GRU	1.0814	9.9060	7.8468	1.8768	0.9287
LSTM	1.8271	10.0376	8.8478	2.3915	0.9194
Updated-LSTM	0.7930	2.1523	2.4121	0.8415	0.9613
GRU-SVM	1.5029	7.3219	7.0121	1.5015	0.9259
WNN	0.9180	2.6591	2.9758	1.2371	0.9528
CEEMDAN-LSTM	1.8274	13.879	11.2769	2.0746	0.9017
DGCnetwork	<b>0.3637</b>	<b>1.7382</b>	<b>1.2915</b>	<b>0.2673</b>	<b>0.9830</b>

## 4. Conclusions

In this study, we propose a deep GRU and CNN based on the DGCnetwork network to model the spatiotemporal relationship of SST to predict the future value. DE algorithm is adopted to infer optimal selection for the hyperparameters of the model. The contributions of this paper are four folds. (1) The DGCnetwork has a compact structure and focuses on learning deep long-term dependencies in SST time series. Each layer in the DGCnetwork model processes the part of the predicted task. (2) Apart from temporal information, spatial information is combined in our work to forecast the SST data. (3) We randomly select the points on the East China Sea and the Yellow Sea datasets to experiment. The results show that the DGCnetwork overcomes the disadvantage of GRU network which has larger prediction errors near the maximum SST value. We have conducted the comprehensive experiments and compared with the leading time series predicting models. The experiments have demonstrated that the DGCnetwork model achieves a state-of-the-art performance and outperforms many existing predicting models. (4) The model can be applied to more time

series data. Finally, our future studies would also work on analyzing other types of SST, such as Group for High Resolution Sea Surface Temperature (GHRSSST) data.

## Data Availability

The data used in our research is an open dataset, the Optimum Interpolation Sea Surface Temperature (OISST), from the National Oceanic and Atmospheric Administration (NOAA).

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This work was supported by the National Program on Key Research Project of China (2016YFC1401900) and Open Fund of the Key Laboratory of Digital Ocean, State Oceanic Administration, China (B201801030).

## References

- [1] A. D. Mehr, V. Nourani, B. Hrnjica, and A. Molajou, "A binary genetic programming model for teleconnection identification between global sea surface temperature and local maximum monthly rainfall events," *Journal of Hydrology*, vol. 555, pp. 397–406, 2017.
- [2] V. Nourani and A. Molajou, "Application of a hybrid association rules/decision tree model for drought monitoring," *Global and Planetary Change*, vol. 159, pp. 37–45, 2017.
- [3] V. Nourani, M. T. Sattari, and A. Molajou, "Threshold-based hybrid data mining method for long-term maximum precipitation forecasting," *Water Resources Management*, vol. 31, no. 9, pp. 2645–2658, 2017.
- [4] C. Ji, Y. Zhang, Q. Cheng, J. Tsou, T. Jiang, and X. S. Liang, "Evaluating the impact of sea surface temperature (SST) on spatial distribution of chlorophyll-a concentration in the East China Sea," *International Journal of Applied Earth Observation and Geoinformation*, vol. 68, pp. 252–261, 2018.
- [5] A. O. Tarakanov and A. V. Borisova, "Galapagos indicator of El Niño using monthly SST from NASA Giovanni system," *Environmental Modelling & Software*, vol. 50, pp. 12–15, 2013.
- [6] A. Chanda, S. Das, A. Mukhopadhyay et al., "Sea surface temperature and rainfall anomaly over the Bay of Bengal during the el Niño-Southern oscillation and the extreme Indian ocean dipole events between 2002 and 2016," *Remote Sensing Applications: Society and Environment*, vol. 12, pp. 10–22, 2018.
- [7] K. Patil, M. C. Deo, and M. Ravichandran, "Prediction of sea surface temperature by combining numerical and neural techniques," *Journal of Atmospheric and Oceanic Technology*, vol. 33, no. 8, pp. 1715–1726, 2016.
- [8] J. S. Kug, I. S. Kang, J. Y. Lee, and J. G. Jhun, "A statistical approach to Indian Ocean sea surface temperature prediction using a dynamical ENSO prediction," *Geophysical Research Letters*, vol. 31, no. 9, pp. 399–420, 2004.
- [9] R. Neetu, R. Sharma, S. Basu, A. Sarkar, and P. K. Pal, "Data-adaptive prediction of sea-surface temperature in the Arabian Sea," *IEEE Geoscience and Remote Sensing Letters*, vol. 8, no. 1, pp. 9–13, 2011.



- [10] I. D. Lins, M. Araujo, M. C. Moura, M. A. Silva, and E. L. Drogue, "Prediction of sea surface temperature in the tropical Atlantic by support vector machines," *Computational Statistics & Data Analysis*, vol. 61, pp. 87–198, 2013.
- [11] I. D. Lins, M. Moura, M. A. Silva et al., "Sea surface temperature prediction via support vector machines combined with particle swarm optimization," in *Proceedings of the International Probabilistic Safety Assessment & Management Conference*, Seattle, WA, USA, June 2010.
- [12] J. Cao, Z. Li, and J. Li, "Financial time series forecasting model based on CEEMDAN and LSTM," *Physica A: Statistical Mechanics and Its Applications*, vol. 519, pp. 127–139, 2019.
- [13] Q. Zhang, H. Wang, J. Dong, G. Zhong, and X. Sun, "Prediction of sea surface temperature using long short-term memory," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 10, pp. 1745–1749, 2017.
- [14] G. Shena, Q. Tana, H. Zhang, P. Zeng, and J. Xu, "Deep learning with gated recurrent unit networks for financial sequence predictions," *Procedia Computer Science*, vol. 131, 2018.
- [15] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [16] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proceedings of the 3rd International Conference on Learning Representations*, San Diego, CA, USA, May 2015.
- [17] M. Ding, H. Zhou, H. Xie, M. Wu, Y. Nakanishi, and R. Yokoyama, "A gated recurrent unit neural networks based wind speed error correction model for short-term wind power forecasting," *Neurocomputing*, vol. 365, pp. 54–61, 2019.
- [18] Q. H. Qi and R. S. Cai, "Analysis on climate characteristics of sea surface temperature extremes in coastal China seas," *Haiyang Xuebao*, vol. 41, no. 7, pp. 36–51, 2019.
- [19] W. H. Joerding and J. L. Meador, "Encoding a priori information in feedforward networks," *Neural Networks*, vol. 4, no. 6, pp. 847–856, 1991.
- [20] H.-G. Han and J.-F. Qiao, "A structure optimisation algorithm for feedforward neural network construction," *Neurocomputing*, vol. 99, pp. 347–357, 2013.
- [21] L. Wang, H. Qu, T. Chen, and F. P. Yan, "An effective hybrid self-adapting differential evolution algorithm for the joint replenishment and location-inventory problem in a three-level supply chain," *The Scientific World Journal*, vol. 2013, Article ID 270249, 11 pages, 2013.
- [22] J. X. Du, D. S. Huang, X. F. Wang, and X. Gu, "Shape recognition based on neural networks trained by differential evolution algorithm," *Neurocomputing*, vol. 70, no. 4–6, pp. 896–903, 2007.
- [23] A. Severyn and A. Moschitti, "Learning to rank short text pairs with convolutional deep neural networks," in *Proceedings of the Thirty-Eighth International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 373–382, Santiago, Chile, August, 2015.
- [24] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *Proceedings of the 2017 International Conference on Engineering and Technology (ICET)*, IEEE, Antalya, Turkey, August 2017.
- [25] K. V. Price, "Differential evolution: a fast and simple numerical optimizer," in *Proceedings of North American Fuzzy Information Processing*, Berkeley, CA, USA, June 1996.
- [26] Y. Wang, Z. Cai, and Q. Zhang, "Differential evolution with composite trial vector generation strategies and control parameters," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 55–66, 2011.
- [27] E. K. Fiedler, A. McLaren, V. Banzon et al., "Intercomparison of long-term sea surface temperature analyses using the GHRSSST multi-product ensemble (GMPE) system," *Remote Sensing of Environment*, vol. 222, pp. 18–33, 2019.
- [28] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International Journal of Forecasting*, vol. 22, no. 4, pp. 679–688, 2006.
- [29] N. M. Faber, "Estimating the uncertainty in estimates of root mean square error of prediction: application to determining the size of an adequate test set in multivariate calibration," *Chemometrics and Intelligent Laboratory Systems*, vol. 49, no. 1, pp. 79–89, 1999.
- [30] A.-L. Schubert, D. Hagemann, A. Voss, and K. Bergmann, "Evaluating the model fit of diffusion models with the root mean square error of approximation," *Journal of Mathematical Psychology*, vol. 77, pp. 29–45, 2017.
- [31] R. J. Hyndman, "Measuring forecast accuracy," in *Business Forecasting: Practical Problems and Solutions*, M. Gilliland, L. Tashman, and U. Sglavo, Eds., pp. 177–183, John Wiley & Sons, Hoboken, NJ, USA, 2016.
- [32] C. J. Xiao, N. C. Chen, C. L. Hu et al., "Short and mid-term sea surface temperature prediction using time-series satellite data and LSTM-AdaBoost combination approach," *Remote Sensing of Environment*, vol. 233, 2019.
- [33] L. J. kang, Z. Ying, L. H. Lin et al., "SST forecast based on BP neural network and improved EMD algorithm," *Climatic and Environmental Research*, vol. 22, no. 5, pp. 587–600, 2017.
- [34] Q. Zhang, *The Interdecadal Variation of SST in the Coastal China Seas and Its Response to Global Warming*, Ocean University of China, Qingdao, China, 2014.
- [35] E. Guresen, G. Kayakutlu, and T. U. Daim, "Using artificial neural network models in stock market index prediction," *Expert Systems with Applications*, vol. 38, no. 8, pp. 10389–10397, 2011.
- [36] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proceedings of the 30th International Conference on Machine Learning*, pp. 1310–1318, Atlanta, GA, USA, June 2013.
- [37] R. Pascanu, C. Gulcehre, K. Cho, and Y. Bengio, "How to construct deep recurrent neural networks," in *Proceedings of the Second International Conference on Learning Representations*, Banff, Canada, April 2014.