

## Short and mid-term sea surface temperature prediction using time-series satellite data and LSTM-AdaBoost combination approach



Changjiang Xiao<sup>a,b,d,f</sup>, Nengcheng Chen<sup>a,b</sup>, Chuli Hu<sup>e</sup>, Ke Wang<sup>e</sup>, Jianya Gong<sup>a,b,c</sup>, Zeqiang Chen<sup>a,b,\*</sup>

<sup>a</sup> State Key Laboratory for Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, 129 Luoyu Road, Wuhan 430079, China

<sup>b</sup> Collaborative Innovation Center of Geospatial Technology, Wuhan 430079, China

<sup>c</sup> School of Remote Sensing and Information Engineering, Wuhan University, 129 Luoyu Road, Wuhan 430079, China

<sup>d</sup> CyberGIS Center for Advanced Digital and Spatial Studies, Department of Geography and Geographical Information Sciences, University of Illinois at Urbana-Champaign, Urbana, IL 61801, United States

<sup>e</sup> Faculty of Information Engineering, China University of Geosciences (Wuhan), Wuhan 430074, China

<sup>f</sup> College of Surveying and Geo-Informatics, Tongji University, 1239 Siping Road, Shanghai 200092, China

### ARTICLE INFO

Edited by Menghua Wang

**Keywords:**

Long short-term memory (LSTM)

AdaBoost

Sea surface temperature (SST)

Prediction

Time series satellite data

### ABSTRACT

Sea surface temperature (SST) is one of the most important parameters in the global ocean-atmospheric system, changes of which can have profound effects on the global climate and may lead to extreme weather events such as droughts and floods. Therefore, predicting the dynamics of future SSTs is of vital importance which can help identify these extreme events and alleviate the losses they cause. In this paper, a machine learning method combining the long short-term memory (LSTM) deep recurrent neural network model and the AdaBoost ensemble learning model (LSTM-AdaBoost) is proposed to predict the short and mid-term daily SST considering that LSTM is good at modelling long-term dependencies but suffers from overfitting, while AdaBoost has strong prediction capability and is not easily overfitted. By combining these two strong and heterogeneous models, the prediction errors related to variance may cancel out each other and the final results can be improved. In this method, the historical time-series satellite data of SST anomaly (SSTA) is used instead of SST itself considering that the fluctuations of SSTs are very small compared to their absolute magnitudes. The seasonality of the SSTA time series is first modelled using polynomial regression and then removed. Then, the deseasonalized time series are used to train the developed LSTM model and AdaBoost model independently. Daily SSTA predictions are made using these two models, and eventually, their predictions are combined as final predictions using the averaging strategy. A case study in the East China Sea that predicts the daily SSTA 10 days ahead shows that the proposed LSTM-AdaBoost combination model outperforms the LSTM and AdaBoost separately, as well as the optimized support vector regression (SVR) model, the optimized feedforward backpropagation neural network model (BPNN), and the stacking LSTM-AdaBoost model (S\_LSTM-AdaBoost), when judged using multiple error statistics and from different perspectives. The results suggest that the LSTM-AdaBoost combination model using the averaging strategy is highly promising for short and mid-term daily SST predictions.

### 1. Introduction and background

Sea surface temperature (SST) refers to the water temperature at the ocean's surface. It is a vitally important parameter of the world's oceans and plays a fundamental role in the exchange of energy, momentum, and moisture between the oceans and the atmosphere (Frank et al., 2000; Sumner et al., 2003). SST changes can have profound effects on

the marine ecosystem and global climate (Bouali et al., 2017; Castro et al., 2016; Chaidez et al., 2017; Herbert et al., 2010; USEPA, 2016; Yao et al., 2017); they can influence the distribution of precipitation, which may lead to extreme weather events such as droughts and floods (Salles et al., 2016), eventually affecting various environmental conditions and dynamics. Therefore, it is highly important to predict future SSTs in order to help us better understand the climatic dynamics and

\* Corresponding author at: State Key Laboratory for Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, 129 Luoyu Road, Wuhan 430079, China.

E-mail addresses: [cjxiao@whu.edu.cn](mailto:cjxiao@whu.edu.cn) (C. Xiao), [cnc@whu.edu.cn](mailto:cnc@whu.edu.cn) (N. Chen), [huchl@cug.edu.cn](mailto:huchl@cug.edu.cn) (C. Hu), [wangke@cug.edu.cn](mailto:wangke@cug.edu.cn) (K. Wang), [gongjy@whu.edu.cn](mailto:gongjy@whu.edu.cn) (J. Gong), [ZeqiangChen@whu.edu.cn](mailto>ZeqiangChen@whu.edu.cn) (Z. Chen).

identify extreme events in advance based on historical observations of earth observation systems (Chen et al., 2014; Zhang et al., 2018). However, due to the large variations in the heat flux, radiation, and diurnal wind near the sea surface, the prediction of SST is highly uncertain (Patil et al., 2013; Patil et al., 2016) and difficult.

SST prediction is an active research topic. Currently, methods for SST prediction can be divided into three categories, including physics-based numerical methods, data-driven methods and the combinations of these two. The numerical methods use mathematical models to describe the variation of SST based on physical conditions and processes, which is sophisticated and difficult to solve. They are usually used for predictions over a large area with relatively coarse resolutions instead of site-specific predictions with fine resolutions (Aparna et al., 2018; Krishnamurti et al., 2006; Stockdale et al., 2006). The data-driven methods learn patterns and relationships from historical observations and further use the learned patterns to infer future SSTs. They are much less sophisticated than numerical methods and are suitable for predicting SSTs at specific sites. The methods that fall into this category include traditional statistical methods and the latest machine learning and artificial intelligent ones. Traditional statistical methods used for SST prediction include the Markov model (Yan and Ants, 2000), and the regression model (Laepple, 2007), etc. Examples of machine learning methods applied for SST prediction include the support vector machine (SVM) (Lins et al., 2013) and neural networks (Patil and Deo, 2017; Tanggang et al., 1998; Wu et al., 2006; Zhang et al., 2017). The genetic algorithm (GA) and particle swarm optimization (PSO) are two examples of artificial intelligence for SST predictions that are either utilized alone or used together with other approaches (Lins et al., 2013; Neetu et al., 2011). In addition, the numerical and machine learning methods have also been combined for better SST prediction (Patil et al., 2016). However, the outputs of the combined numerical-machine learning models are for large scales with the same coarse resolutions as those of the numerical models, which are also difficult for site-specific high-resolution SST prediction tasks. In this paper, we focus on SST predictions at specific sites; therefore, the data-driven methods are utilized.

Among the machine learning methods, neural network models are mostly used due to their flexibility and strong ability in modelling the complex patterns hidden in data (Yue et al., 2017; Zang et al., 2019). Many kinds of neural network models are used with different variations for SST prediction, such as the ordinary feedforward neural network (Aparna et al., 2018), the wavelet neural network (Patil and Deo, 2017; Patil et al., 2016), the nonlinear autoregressive neural network (Patil et al., 2013), and long short-term memory (LSTM) (Zhang et al., 2017) which is a special kind of recurrent neural network (RNN). The RNN is good at predicting time series data, but due to the vanishing and exploding gradient problems, it cannot model the long-term dependencies in SST series well. In contrast, LSTM can remember information for much longer periods of time due to its recurrent structure and gating mechanisms, and is regarded as a state-of-the-art method for time series related problems (Chao et al., 2018). Considering that traditional machine learning methods such as SVM and ordinary neural networks are difficult to extract the long-term dependencies of SSTs while the LSTM neural network is good at it, the LSTM is chosen for the SST prediction tasks in this paper. However, though LSTM has achieved good results in prediction tasks, it still suffers from problems such as overfitting due to its high complexity, which can be inferred from the decomposition of squared prediction errors, the relationship between the model complexity and the decomposed components, and the relationship between the decomposed components and the model fitting (Fortmann-Roe, 2012; Wikipedia, 2019). For any machine learning model, the prediction error can be decomposed into three parts, including the bias error, the variance error, and the irreducible error. As the irreducible error cannot be reduced no matter what algorithm is used, the focus of a machine learning model is on reducing the bias error and the variance error. However, decreasing any one of them will increase the other one.

Therefore, a trade-off between the bias error and the variance error is needed. In the case of LSTM with high complexity, the prediction errors primarily come from the large variance of the prediction results relative to the true values. To reduce the variance error of the predictions made by the LSTM model and further improve the prediction results, combination with other heterogeneous, powerful (low bias), and meanwhile not easily overfitted predicting techniques in a certain way can be considered, during which the errors related to the variance may be smoothed while keeping the bias error low, resulting in a model with both lower variance error and low bias error, which can have better generalization performance. The advantages of the hybrid prediction mechanism have been demonstrated in similar time series prediction tasks through different models being adopted and combined (Chen and Li, 2009; Liu et al., 2015; Messias et al., 2016; Zou and Yang, 2004). In this paper, we try to combine AdaBoost, which is a strong ensemble learning method for prediction tasks with low bias error while being not easily overfitted during training (Kun, 2015; Schapire et al., 1998), with the LSTM model to achieve better results. The combination is achieved using the averaging strategy, resulting in a model with lower variance error than the LSTM (Perrone, 1993). As the LSTM model and AdaBoost model themselves are both with low bias error, the resulting LSTM-AdaBoost combination model is also with low bias error. Namely, the resulting model is with both low bias and low variance error, having better trade-off between these two error components, and thus having lower total error and being less easily overfitted. In this combining scheme, the LSTM model and the AdaBoost model are first trained independently. Predictions are then made using the trained LSTM model and AdaBoost model independently as well. Finally, their predictions are averaged as the final predictions. To the best of our knowledge, this is the first attempt to use the LSTM-AdaBoost combination model to solve the SST prediction problems.

The goals of this paper include the following: 1) develop an LSTM deep neural network model and an AdaBoost model that can predict the short and mid-term SST with high accuracy; 2) develop a strategy to combine the prediction results of the LSTM and the AdaBoost model to achieve more accurate predictions; and 3) investigate the applicability, effectiveness, and advantages of the proposed LSTM-AdaBoost combination model in predicting the short and mid-term daily SST using experiments at selected sites in the East China Sea using 36 year satellite time series data.

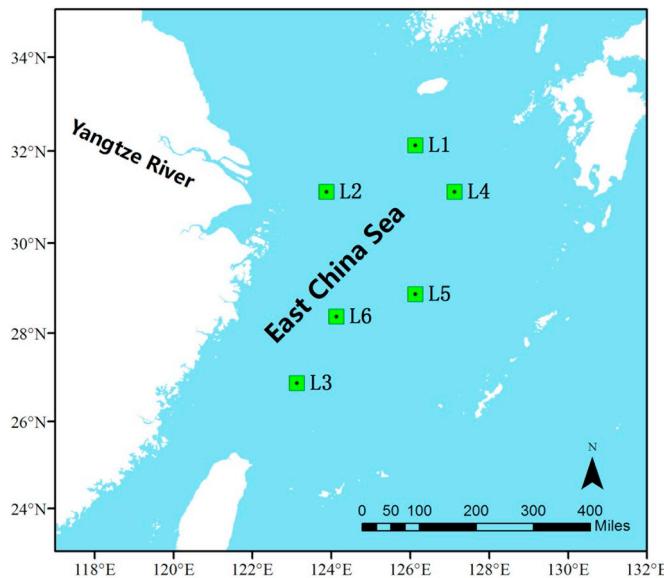
The contributions of this paper are primarily twofold: (1) A LSTM deep learning model and an AdaBoost ensemble learning model are designed for accurate short and mid-term SST prediction; (2) A simple but powerful averaging strategy is designed to combine the LSTM model and AdaBoost model, which is demonstrated in the experiments to have better performance than the single LSTM and AdaBoost model, the SVR model, the Feedforward Backpropagation Neural Network (BPNN) model, and the combination model of LSTM and AdaBoost using complex stacking generalization (Wolpert, 1992) based on k-folds cross-validation.

The remainder of the paper is organized as follows. Section 2 describes the study area and satellite-derived time-series SST data used in this study. Section 3 presents the proposed LSTM-AdaBoost method for SST prediction. The experimental results and thorough discussions are given in Section 4. Finally, Section 5 concludes the paper.

## 2. Study area and data

### 2.1. Study area

The study area is in the East China Sea, which is a marginal sea east of China bordering South Korea and Japan and covering an area of approximately 1,249,000 km<sup>2</sup>. The famous Yangtze River flows into the East China Sea. It is part of the Maritime Silk Road and one of the busiest seaways in the world. Therefore, understanding and predicting the dynamics of the SST in this area is of vital importance to the



**Fig. 1.** Study area in the East China Sea. The coordinates (latitude, longitude) of the centres of the 6 selected sites (green rectangles) are as follows: L1 (32.125°N, 126.125°E), L2 (31.125°N, 123.875°E), L3 (26.875°N, 123.125°E), L4 (31.125°N, 127.125°E), L5 (28.875°N, 126.125°E), and L6 (28.375°N, 124.125°E). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

production and lives of people in both China and its surrounding countries, but it has rarely been investigated in previous research.

In this study, we select 6 representative sites in the East China Sea as our studying locations, as is shown in Fig. 1. The 6 sites are denoted as L1 to L6, with L1 and L2 nearer to coastal areas and L3 to L6 in the open sea.

## 2.2. Data

The data used in this study is the National Oceanic and Atmospheric Administration (NOAA) 1/4° daily Optimum Interpolation Sea Surface Temperature (daily OISST, version 2). It's an analysis data constructed by combining observations from different platforms, including satellites, ships and buoys, on a regular global grid. Interpolation is applied to fill in the gaps and make it a spatially complete SST data set. There are two kinds of daily SST, named after the satellite SST sensors that are used, including the Advanced Very High Resolution Radiometer (AVHRR) and the Advanced Microwave Scanning Radiometer on the Earth Observing System (AMSR-E). The OISST using satellite SSTs from the AVHRR are called AVHRR-Only, and OISST using data from both AVHRR and AMSR-E are called AVHRR + AMSR. However, since the AMSR-E stopped working in October 2011, AVHRR + AMSR production ended (NOAA). To obtain a continuous long time-series of SSTs, we choose the AVHRR-Only SSTs, which cover the global ocean from 89.975°S to 89.875°N, 0.125°E to 359.875°E and provide the daily SST values from 1981/09/01 to present. Here, in this study, the daily OISST-V2-AVHRR data from 1982/01/01 to 2017/12/31 (13,149 days in total) are used. As the fluctuations of SSTs are very small compared to their absolute magnitudes, the SST anomalies (SSTAs) are preferred instead of SSTs themselves (Aparna et al., 2018; Neetu et al., 2011; Patil and Deo, 2017; Wu et al., 2006). Thus, in this research, the SSTAs are used to establish the prediction models, and to compare the prediction performance of different models. The SSTA is a variable of the OISST-V2-AVHRR data, which is calculated relative to 1971–2000 climatology (PSD).

Basic statistics of the SSTAs at the 6 selected sites are presented in Table 1. We can see that SSTAs at these sites have similar and meanwhile varying characteristics. L1, L2 and L3, L4, L5 and L6 are the

groups with similar mean SSTAs and standard deviations of SSTAs. L1, L2, and L3 generally have bigger mean, standard deviation, minimum, 1st quantile, median, 3rd quantile and maximum SST values than L4, L5 and L6 except for that L1 has the lowest minimum SSTA among all the 6 locations. In addition, L1 and L2 have higher standard deviations than the other sites, which may be in part due to that they are near the coasts and have shallower depths than the other locations that are in the open sea. The rich characteristics existing in the SSTA time series at the selected locations make them appropriate for comprehensively investigating the applicability and effectiveness of the proposed prediction method.

## 3. Method

In this study, a rolling prediction scheme is adopted. In this scheme, an LSTM and AdaBoost model are each trained to find a pattern between the SSTA at time  $t_i$  and its  $n$  preceding values at times  $t_{i-1}$ ,  $t_{i-2}$ , ..., and  $t_{i-n}$  (also called a time window) using historical observations. The future prediction is made based on its preceding  $n$  values using the identified patterns. By sliding the time window one step ahead and using the same pattern, one more step-ahead prediction is made. Through repeating this process,  $k$ -days-ahead predictions can be achieved. The final prediction for each day ahead is obtained by combining the predictions of the LSTM model and the AdaBoost model for that day using the averaging strategy.

### 3.1. LSTM deep neural network model

LSTM is a variation of the RNN, which was first proposed by Sepp Hochreiter and Jürgen Schmidhuber in 1997 as a solution to the vanishing and exploding gradient problem of the RNN (Sepp and Jürgen, 1997). It is capable of learning long-term dependencies and works well on a large variety of problems.

As shown in Fig. 2, LSTM has a chain-like structure. The basic building block of LSTM is a cell whose state is the key to LSTM. LSTM uses three kinds of gates to regulate a cell state, including a forget gate, an input gate, and an output gate. A gate is a way to control how much information can pass through. As shown in the second cell, a gate is comprised of a fully-connected neural network layer activated by a sigmoid function, and a pointwise multiplication operation (Olah, 2015). The working mechanism of the gates and information flow can be expressed using the following equations:

$$\begin{aligned} f_t &= \sigma(W_f \cdot [h_{t-1}x_t] + b_f) \\ i_t &= \sigma(W_i \cdot [h_{t-1}x_t] + b_i) \\ C'_t &= \tanh(W_C \cdot [h_{t-1}x_t] + b_C) \\ C_t &= f_t * C_{t-1} + i_t * C'_t \\ o_t &= \sigma(W_o \cdot [h_{t-1}x_t] + b_o) \\ h_t &= o_t * \tanh(C_t) \end{aligned} \quad (1)$$

where  $f_t$ ,  $i_t$ , and  $o_t$  are the outputs of three sigmoid functions  $\sigma$ . Their values are between 0 and 1 and control the information that is forgotten in the old cell state  $C_{t-1}$ , the information that is stored ( $C'_t$ ) in the new cell state  $C_t$ , and the information that is output ( $h_t$ ) from the cell, respectively.  $W_f$ ,  $W_i$ ,  $W_C$ , and  $W_o$  are the weights applied to the concatenation (represented by  $[]$ ) of the new input  $x_t$  and output  $h_{t-1}$  from previous cell.  $b_f$ ,  $b_i$ ,  $b_C$ , and  $b_o$  are the corresponding biases.

Based on LSTM, we build a 5-layer deep neural network model for SSTA prediction, as shown in Fig. 3. It consists of an input layer, two LSTM layers and two Dense layers (also called the fully connected layers). We have conducted several experiments and found that this neural network design can achieve the best prediction performance (Text S1 in the Supplementary file). The input of the whole network is in 3D tensor form and expressed as  $(n\_samples, n\_time\_steps, n\_features)$ .  $n\_samples$  is the batch size for training, the best of which is set by trial

**Table 1**

Statistics of the SSTA time series at L1 to L6 from 1982/01/01 to 2017/12/31.

Site	Mean (°C)	Std. dev. (°C)	Minimum (°C)	25% (°C)	50% (°C)	75% (°C)	Maximum (°C)
L1	0.094	1.173	-5.770	-0.660	0.170	0.900	4.530
L2	0.348	1.152	-4.570	-0.400	0.360	1.130	4.620
L3	0.275	1.060	-3.620	-0.430	0.250	0.930	4.570
L4	-0.109	1.035	-4.310	-0.760	-0.080	0.560	3.910
L5	-0.250	1.014	-4.950	-0.880	-0.190	0.420	2.910
L6	-0.322	1.011	-4.840	-0.950	-0.250	0.390	3.300

and error and may vary for SSTAs at different locations.  $n\_time\_steps$  is the size of the time window used to predict the SSTAs. In our method, we set it to 40 (Text S2 in the Supplementary file), which means that we use the previous 40 days' SSTA sequence to predict the 41st day's SSTA. Since we only use SSTAs as predictors,  $n\_features$  is set to 1. By trial and error, we set the number of cells in the first LSTM layer, the second LSTM layer, and the first Dense layer to 30, 25, and 35 respectively (Text S1 in the Supplementary file). A dropout mechanism is applied to the inputs of the second LSTM layer and the inputs of the first Dense layer to help prevent overfitting during training. Both dropout rates are empirically set to 0.5.

### 3.2. AdaBoost ensemble learning model

AdaBoost, short for Adaptive Boosting, is a kind of boosting algorithm that was introduced by Freund and Schapire in 1995 (Freund and Schapire, 1997) to improve the performance of machine learning learners. The AdaBoost regressor is the application of the AdaBoost for regression problems, to which the SSTA prediction problem in this study belongs. The core principle of AdaBoost regressor is to learn a sequence of weak regressors which have high bias error but with low variance error, on repeatedly reweighted training instances according to the error of prediction at each boosting iteration. Through this, the subsequent regressor puts more emphasis on the incorrectly predicted

instances at the previous step. The final prediction is obtained by combining the results of all the weak regressors, resulting in a model with lower bias error and low variance error (Buitinck et al., 2013; Pedregosa et al., 2011). It's not easily overfitted during training, which has been theoretically explained and proved in previous literatures (Kun, 2015; Schapire et al., 1998).

In this research, we choose the decision tree regressor as the basic regressor boosted by AdaBoost, which has been demonstrated to be powerful for predictions (Che et al., 2011; Kim and Upneja, 2014). The decision tree regressor is a kind of non-parametric supervised learning method for regression problems. It uses a white box model, which is easy to understand and interpret. The *if-then-else* decision rules are learned by a decision tree regressor from the data features to approximate the target variable. The classification and regression trees (CART) algorithm (Leo et al., 1984; Pedregosa et al., 2011) is adopted to build the decision tree regressor for SSTA prediction in this study.

The architecture of the adopted decision tree regressor boosted using AdaBoost is shown in Fig. 4. The dotted-line part demonstrates the generation of the boosted decision tree regressors based on repeatedly reweighted training data. The weights to combine the predicted SSTAs using each trained decision tree regressor are calculated during boosting according to the prediction errors it produces. The whole process can be expressed in Algorithm 1 as follows:

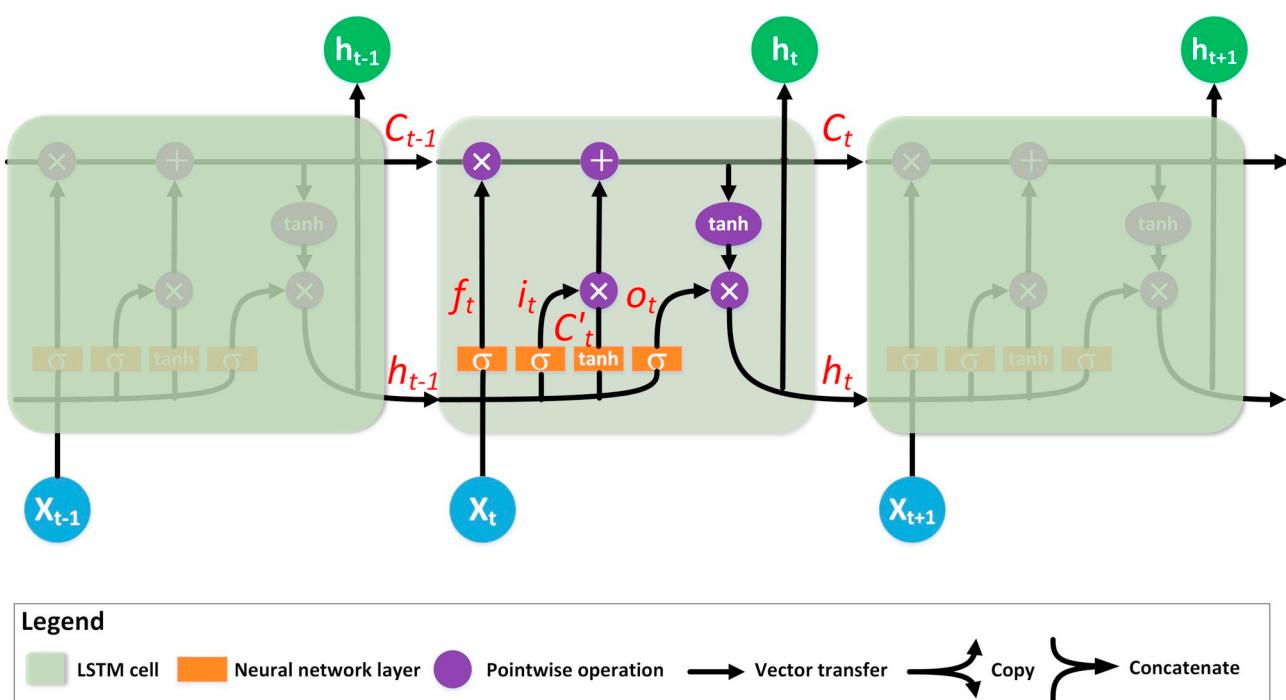


Fig. 2. Structure of LSTM (Olah, 2015).

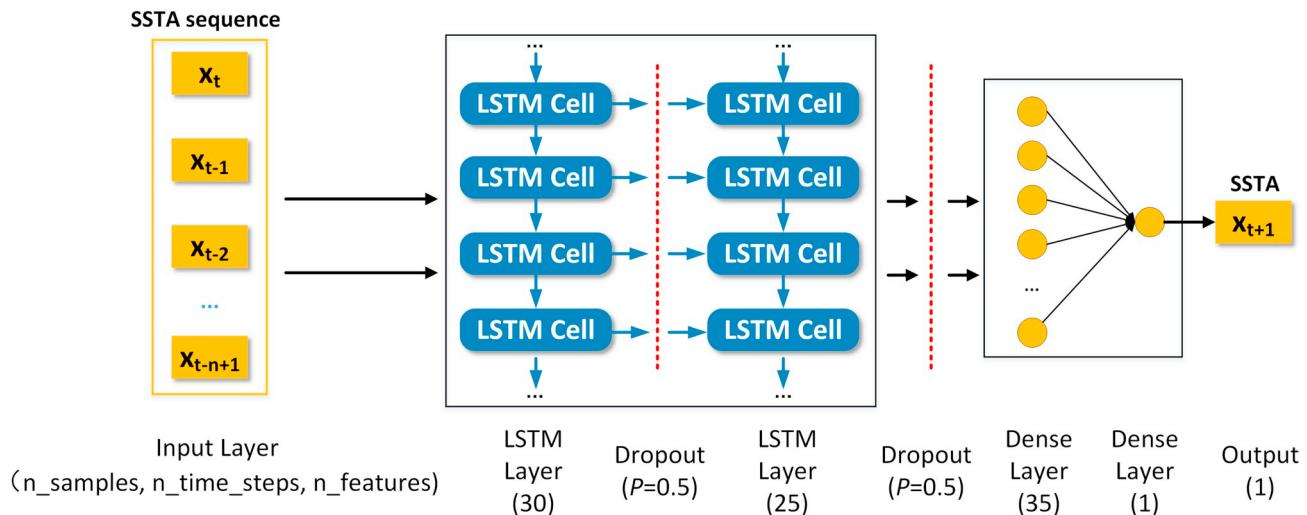


Fig. 3. The architecture of the proposed LSTM deep neural network model for SSTA prediction.

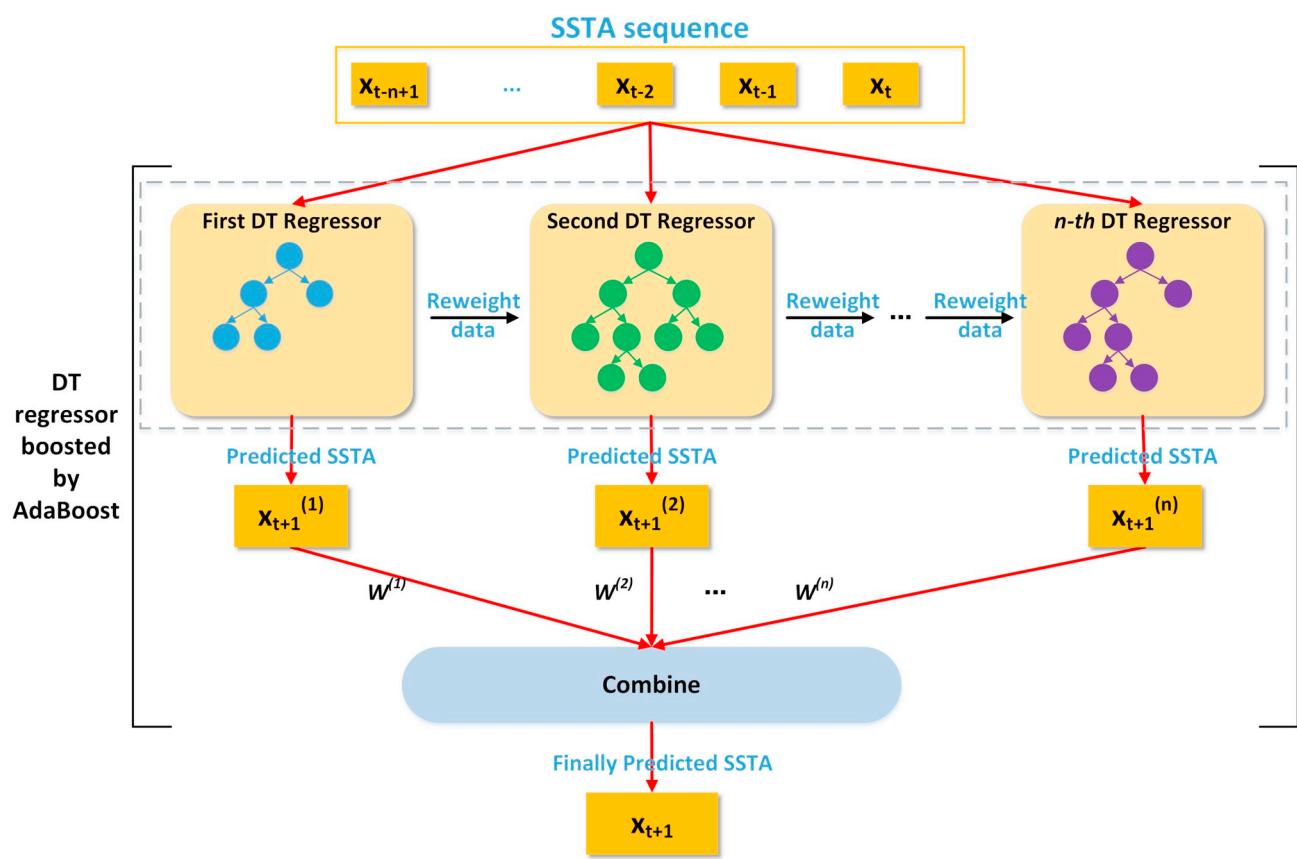
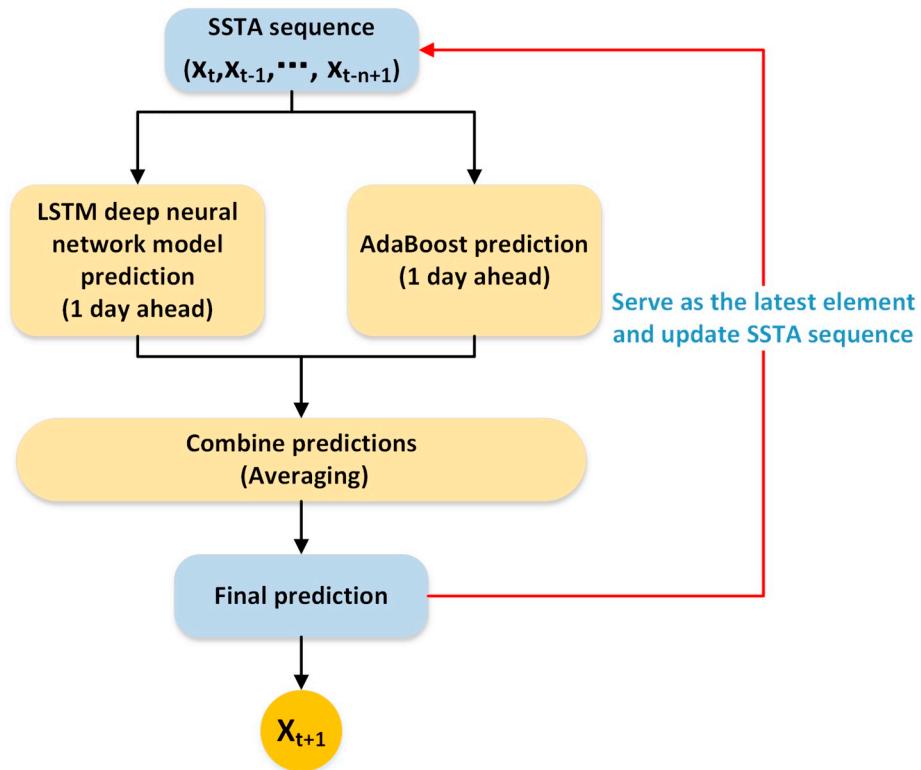
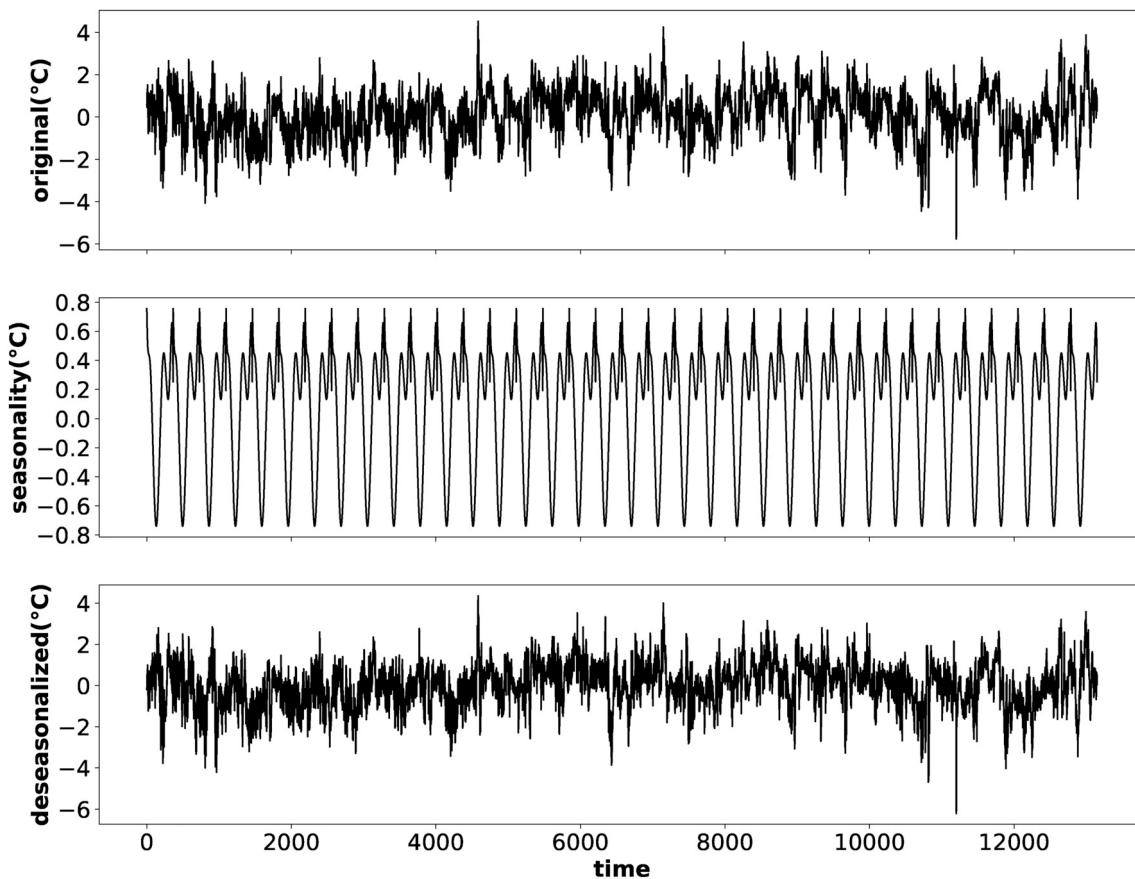


Fig. 4. The architecture of the adopted decision tree (DT) regression model boosted by AdaBoost for SSTA prediction.



**Fig. 5.** The architecture of the proposed LSTM-AdaBoost combination method for SSTA prediction.



**Fig. 6.** Deseasonalization of the SSTA time series of location L1.

**Algorithm 1.** AdaBoost using the decision tree regressor as the base regressor for SSTA prediction (Drucker, 1997; Pardoe and Stone, 2010).

---

**Algorithm 1** AdaBoost using the decision tree regressor as the base regressor for SSTA prediction (Drucker 1997; Pardoe and Stone 2010)

**Input:**  $T = \{(X_i, y_i)\}$  ( $i = 1, 2, 3, \dots, m$ ) is the training dataset built from SSTA time series.  
 $m$  is the size of the training dataset.  
 $K$  is the number of decision tree regressors (predictors) to be learned

1: Initialize weights of samples in training dataset  $T$ :

$$D(1) = (w_{11}, w_{12}, \dots, w_{1m}) \\ w_{1i} = \frac{1}{m}, i = 1, 2, \dots, m$$

2: **for**  $k = 1: K$  **do**

3:     Sample from training dataset  $T$  based on weights  $D(k)$  and train a decision tree predictor  $H_k(x)$

4:     Calculate the maximum prediction error  $E_k$  on  $T$ , the linear error  $\xi_{ki}$  on each sample of  $T$  and the error rate  $\xi_k$  using predictor  $H_k(x)$ :

$$E_k = \max(|y_i - H_k(X_i)|) \\ \xi_{ki} = \frac{|y_i - H_k(X_i)|}{E_k}, i = 1, 2, \dots, m \\ \xi_k = \sum_{i=1}^m w_{ki} \xi_{ki}$$

4: Calculate  $a_k$ , a measure of confidence in predictor  $H_k(x)$ :

$$a_k = \frac{\xi_k}{1 - \xi_k}$$

5: Update the weights of samples in  $T$ :

$$D(k+1) = (w_{k+1,1}, w_{k+1,2}, \dots, w_{k+1,m}) \\ w_{k+1,i} = \frac{w_{ki}}{Z_k} a_k^{1-\xi_{ki}}, i = 1, 2, \dots, m \\ Z_k = \sum_{i=1}^m w_{ki} a_k^{1-\xi_{ki}}$$

6: **end for**

---

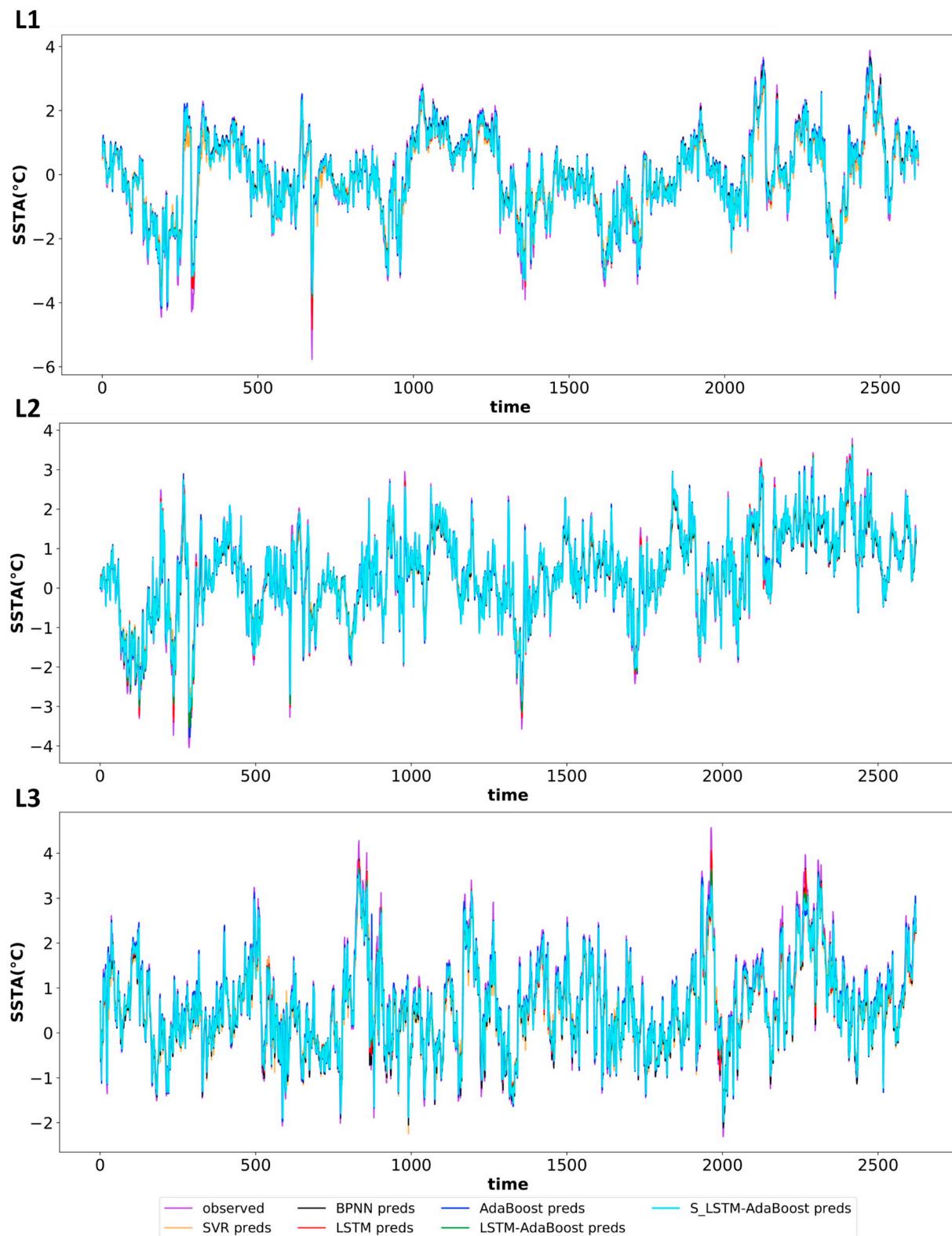
**Output:** A strong SSTA predictor  $H(x)$  that is the weighted median of  $H_k(x)$  using

---


$$W^{(k)} = \ln \frac{1}{a_k} \text{ as the weight for } H_k(x). k = 1, 2, \dots, K$$


---

For the AdaBoost algorithm, there are some hyper-parameters to tune, including the number of regression tree predictors, the learning rate, and the loss function, etc. We use the grid search (Pedregosa et al., 2011) to exhaustively evaluate the possible combinations of a set of parameter values for SSTA prediction and the best combination is retained for each location.



**Fig. 7.** One-day-ahead prediction results on the held-out testing samples using the LSTM-AdaBoost, LSTM, AdaBoost, SVR, BPNN and S\_LSTM-AdaBoost methods at the 6 locations (L1 – L6).

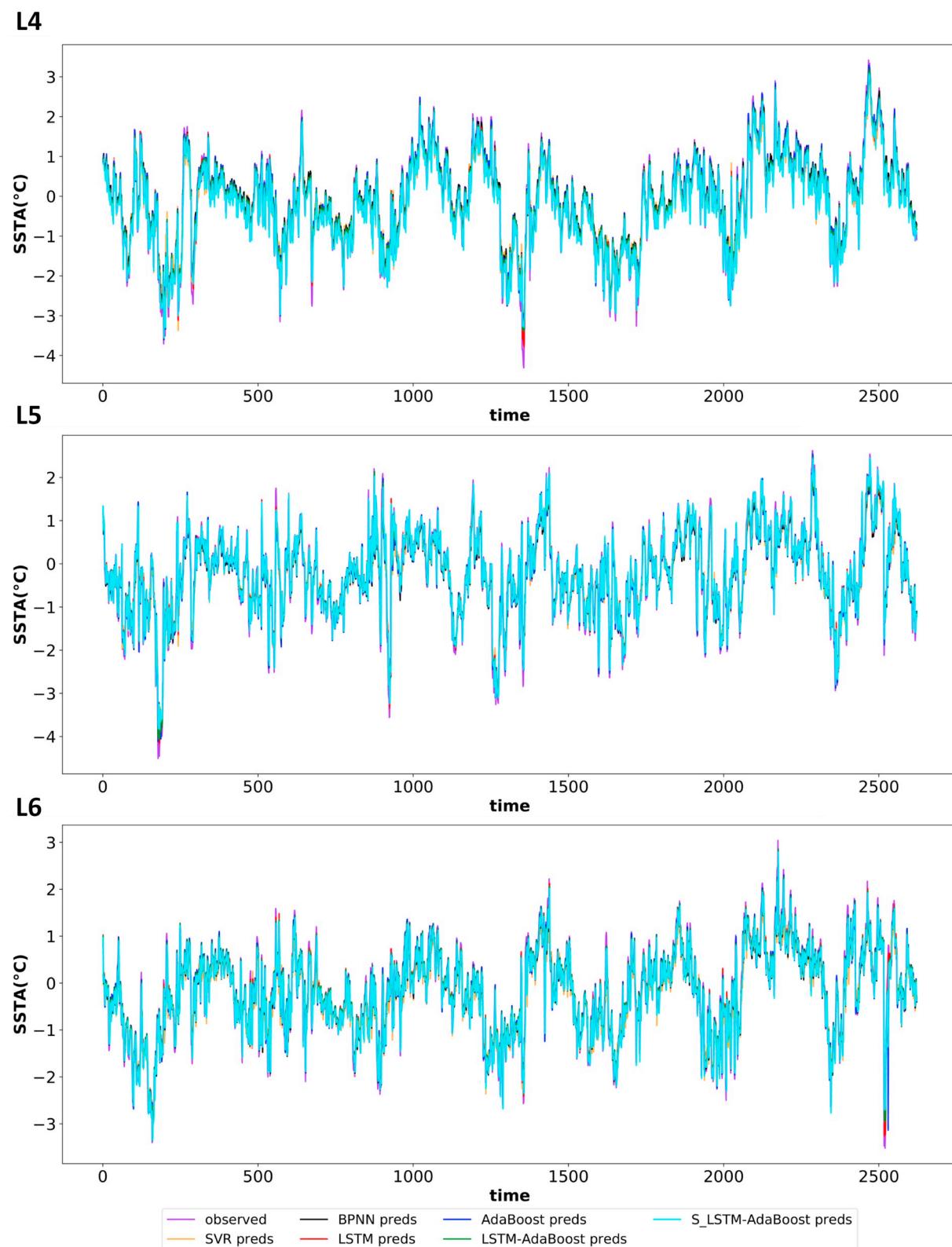
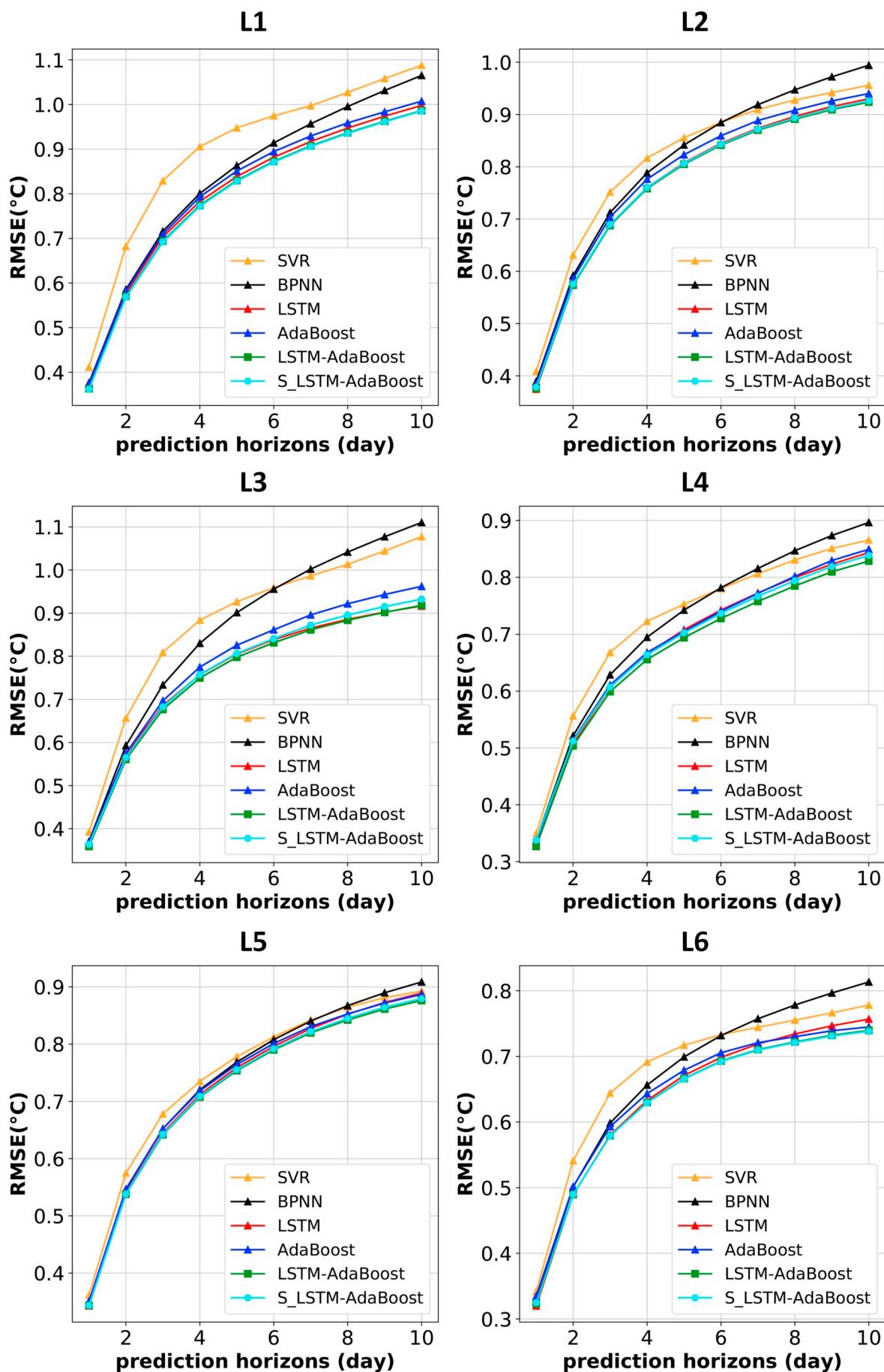


Fig. 7. (continued)



(caption on next page)

**Fig. 8.** RMSE of the predictions using the LSTM-AdaBoost, LSTM, AdaBoost, SVR, BPNN and S\_LSTM-AdaBoost methods at the 6 locations (L1 – L6) for different prediction horizons (1–10 days).

### 3.3. LSTM-AdaBoost combination model

In a similar wind speed prediction task, it is concluded that a hybrid method always has better prediction performance than the single ones (Liu et al., 2015). Considering the heterogeneity and strong prediction capabilities of the LSTM and the AdaBoost model that have independent error distributions, a prediction method that combines these two, which is named LSTM-AdaBoost, is developed to further improve the prediction accuracy. Fig. 5 shows the architecture of the proposed LSTM-AdaBoost combination method. The SSTA sequence in a time window form is first input into both LSTM and AdaBoost for independent predictions. Then, their predictions on each horizon are combined through averaging to produce the final prediction. The final prediction is then used as the latest element of the input SSTA sequence to update the sequence, which is further used to predict one more day ahead SSTA. By repeating this process,  $n$ -days-ahead predictions can be achieved. As we predict the short and mid-term SSTA,  $n$  is set to 10.

## 4. Experiments and discussion

### 4.1. Data pre-processing

Before feeding the historical SSTA observations into LSTM and the AdaBoost model for training, the SSTA time series of each selected location is deseasonalized and normalized. The deseasonalization, which is also called seasonal adjustment, is useful for exploring the trends, cyclical, and any remaining irregular components of a time series and has been proved to contribute to more accurate predictions than using the raw data (Michael et al., 1999). Predictions are first performed on the deseasonalized SSTA data. Then, the seasonality is added back to obtain the final predictions. In addition, normalization is applied to the deseasonalized SSTA time series in each location to make the data more centred, which is good for the models' functioning.

#### 4.1.1. Deseasonalization of the satellite-derived-SSTA time series

To deseasonalize the OISST-V2-AVHRR time series, we model the seasonality directly and then subtract it from the raw observations. By investigating the raw series, it is found that the seasonal component looks like a polynomial curve over a generally fixed period. Therefore, we use a curve-fitting method to approximate the seasonal component.

A new time series is constructed based on the original time series by transforming the observation date to a time index (day of the year). Polynomial regression is used to fit the new time series using the x-axis values (time index), the y-axis values (SSTA observations) and the order of the polynomial. The resulting model takes the following form:

$$y = \sum_{i=0}^n a_i x^i \quad (2)$$

where  $n$  is the order of the polynomial,  $y$  is the fitted seasonality value at time index  $x$ , and  $a_i$  is the coefficient of the term  $x^i$ .  $n$  is set to 7 for location L1 and 6 for all the other 5 locations through trial and error. We can use this model to estimate the seasonality component for both historical observations and any new future observations.

The fitted seasonality curve is subtracted from the original observations to create a seasonality-adjusted series, which is further used to train, validate and test the LSTM and AdaBoost model used in this research. Fig. 6 shows the deseasonalizing result of the SSTA time series at location L1 as an example.

#### 4.1.2. Normalization

Before feeding the deseasonalized SSTA time series into the adopted machine learning models, it is normalized to the range of [0 1] for each

of the 6 locations using the following algorithm:

$$x_{\text{norm}} = \frac{(y_{\max} - y_{\min})(x - x_{\min})}{(x_{\max} - x_{\min})} + y_{\min} \quad (3)$$

where  $y_{\max} = 1$ ;  $y_{\min} = 0$ ;  $x_{\max}$  and  $x_{\min}$  are the maximum and minimum value of each time series, respectively;  $x$  is the value to be normalized, and  $x_{\text{norm}}$  is the normalized value of  $x$ .

### 4.2. Experiment setup

For the SSTA prediction experiment in each location, we set the time window of the input SSTA sequence to 40 to prepare the datasets for the LSTM, AdaBoost, SVR and BPNN model (SVR and BPNN are used for comparison purposes). The window size is set by trial and error (Text S2 in the Supplementary file). We then select the first 80% of the prepared dataset as training samples and the remaining 20% as held-out testing samples. 5% of the training samples are further split out for validation purposes during training. To guarantee the fairness of the comparison, grid search is used to tune the hyperparameters of the SVR and BPNN to achieve their best performances.

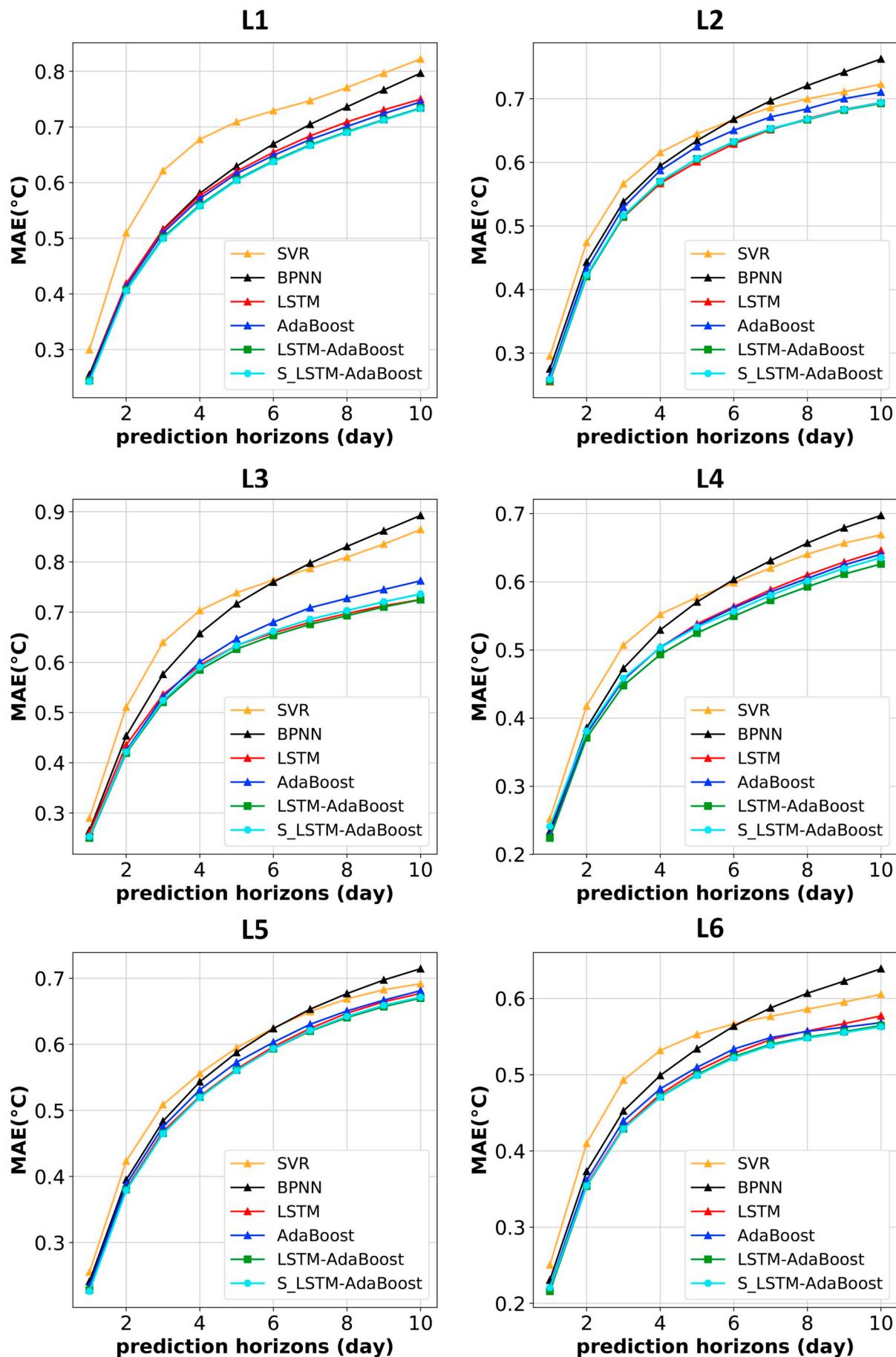
Besides, to demonstrate that the averaging strategy for combining the predictions of LSTM and AdaBoost is a preferring strategy, the comparisons with LSTM and AdaBoost combination using stacking generalization strategy (S\_LSTM-AdaBoost) are conducted. To build an S\_LSTM-AdaBoost model, the following steps are adopted: (1) Split the training dataset into three disjoint sets with the same number of samples in each set; (2) Train LSTM and AdaBoost model (called base learners) on two of the three disjoint sets. The LSTM and AdaBoost model have the same structure as those described in Section 3.1 and Section 3.2; (3) Make predictions on the remaining set with the learned two base learners; (4) Repeat step (2)–(3) until each two of the three disjoint sets have been used for training the base learners once and the remaining set has been used for predictions based on the trained base learners; (5) Use the predictions made in step (2)–(4) and the target SSTA to train a higher level meta-learner. In this research, a BPNN model is chosen as the meta-learner as it performs best among a variety of learners including linear model, BPNN model, Random Forest model and Gradient Boosting Regression Tree model in experiments. The BPNN meta-learner is trained to automatically learn how to combine the predictions made by the LSTM model and the AdaBoost model to produce the final predictions. The best parameters of BPNN meta-learner including the number of hidden layers, the number of units in each hidden layer and batch size for training etc. are chosen by grid search.

The LSTM deep neural network is implemented with Keras (Francois, 2015) using TensorFlow 1.5.0 (GPU version) (Abadi et al., 2016) as the backend. The AdaBoost, SVR, and BPNN are implemented using Scikit-learn 0.19.1 (Pedregosa et al., 2011).

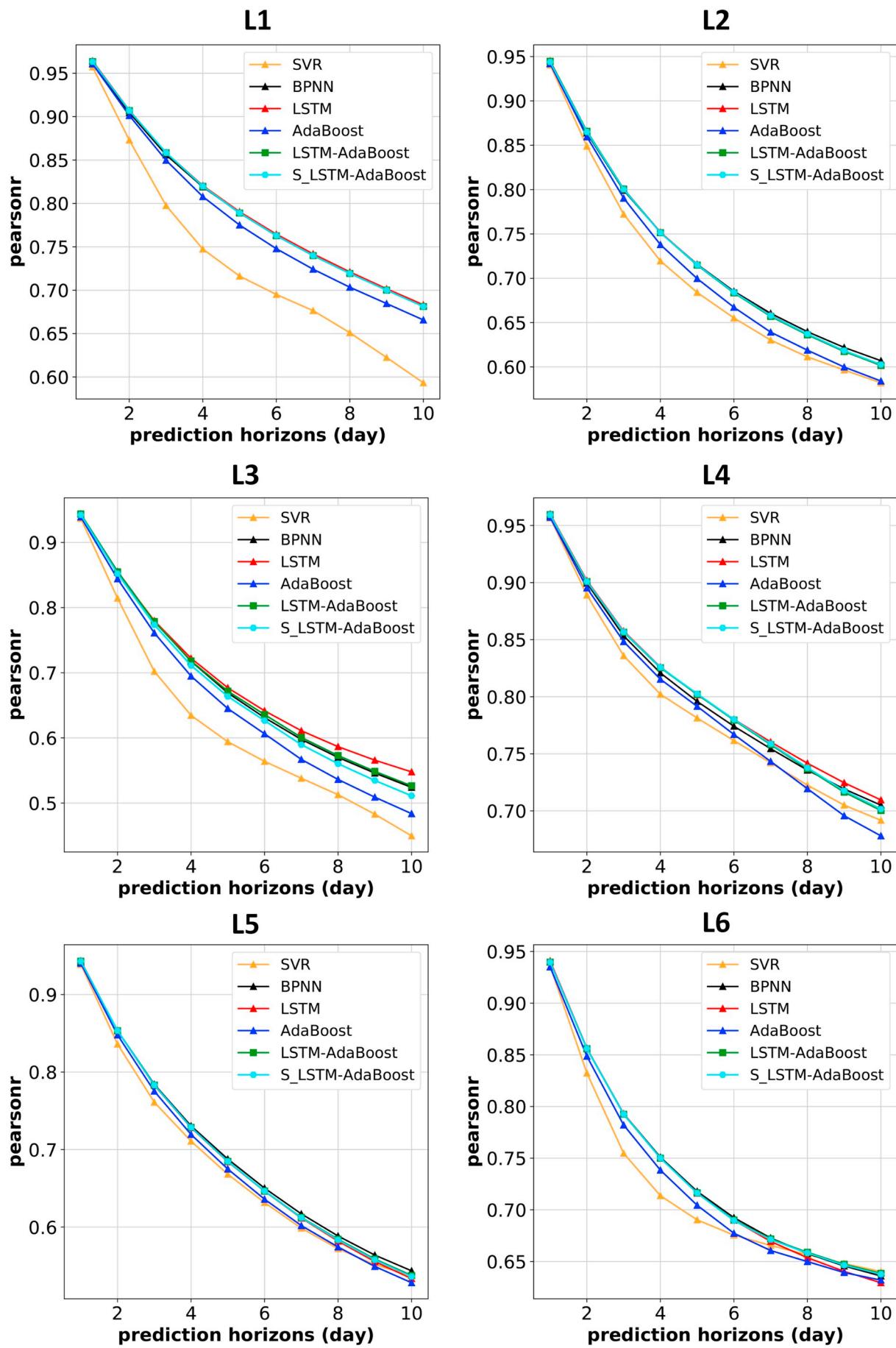
Three indexes are used to measure the performances of the different methods for SSTA prediction, including the root mean square error (RMSE), the mean absolute error (MAE), and the Pearson correlation coefficient ( $r$ ), which are defined as follows:

$$\begin{aligned} \text{RMSE} &= \sqrt{\frac{\sum_1^n d_i^2}{n}} \\ \text{MAE} &= \frac{1}{n} \sum_1^n |y_i - y'_i| \\ r &= \frac{\sum_1^n (y_i - \bar{y})(y'_i - \bar{y}')}{\sqrt{\sum_1^n (y_i - \bar{y})^2} \sqrt{\sum_1^n (y'_i - \bar{y}')^2}} \end{aligned} \quad (4)$$

where  $d_i$  is the error vector calculated by the difference between the desired SSTA value  $y_i$  and the predicted SSTA value  $y'_i$ .  $\bar{y}$  and  $\bar{y}'$  are the

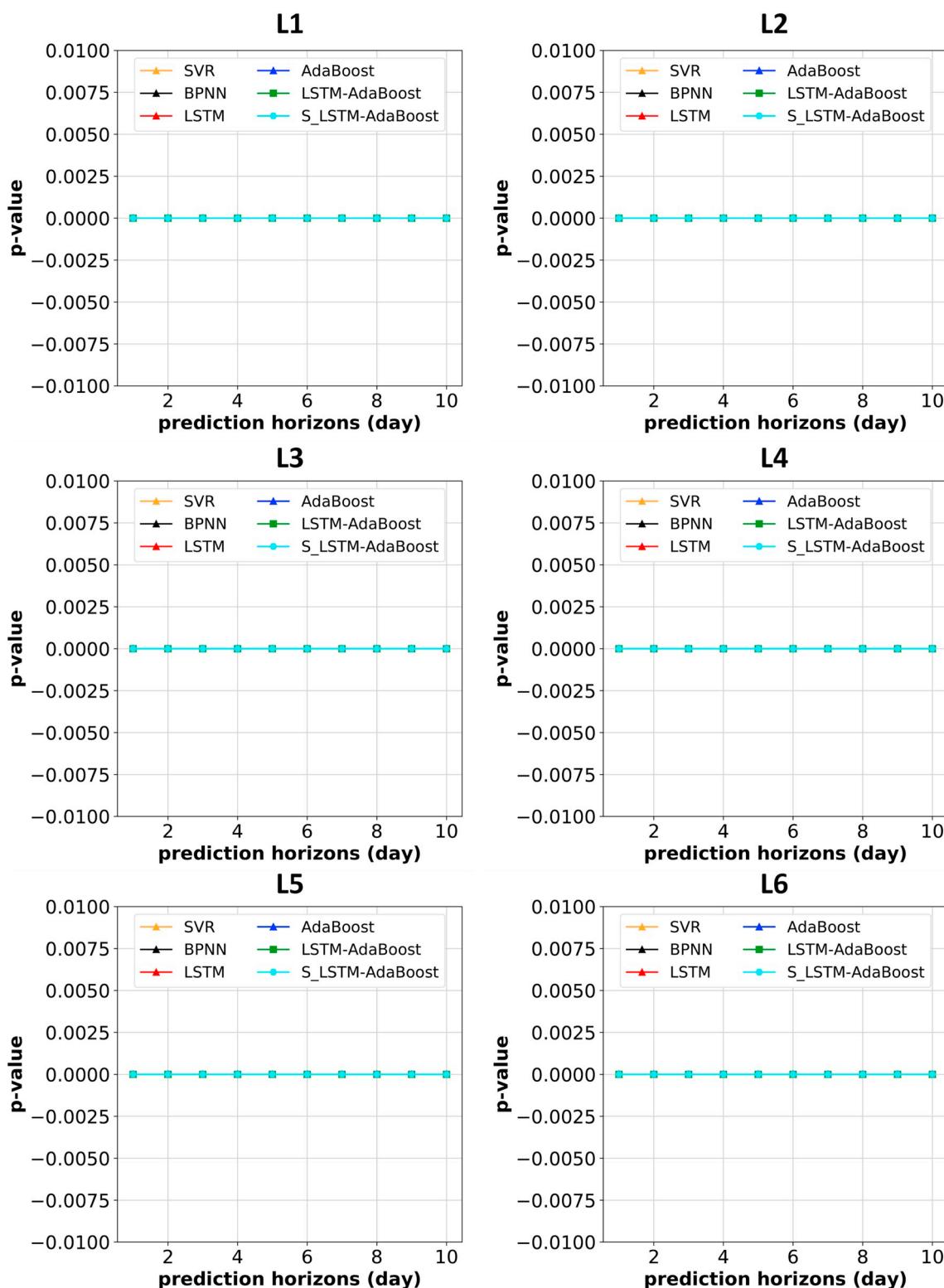


**Fig. 9.** MAE of the predictions using the LSTM-AdaBoost, LSTM, AdaBoost, SVR, BPNN and S\_LSTM-AdaBoost methods at the 6 locations (L1 – L6) for different prediction horizons (1–10 days).

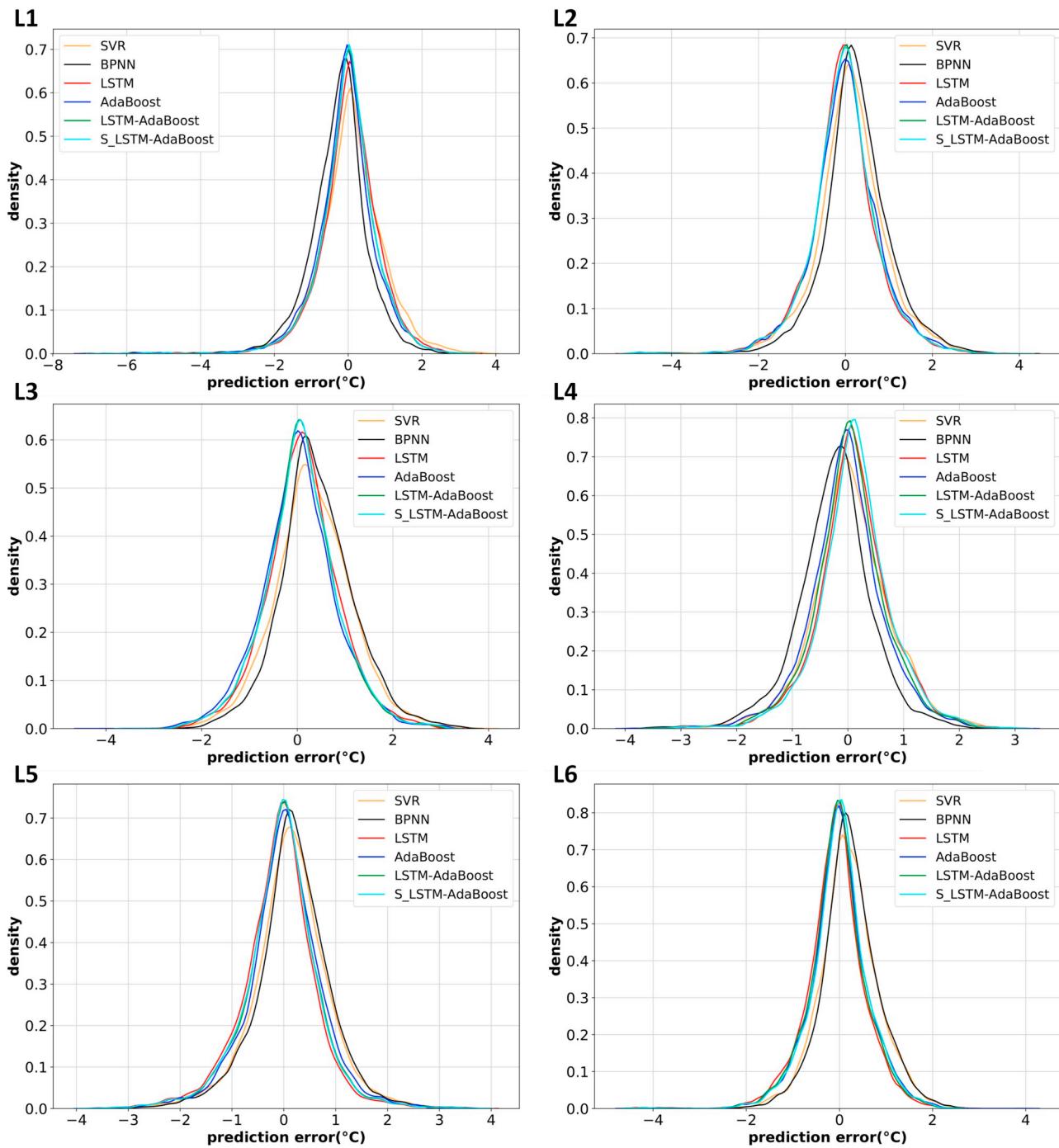


(caption on next page)

**Fig. 10.** Pearson's  $r$  of the predictions using the LSTM-AdaBoost, LSTM, AdaBoost, SVR, BPNN and S\_LSTM-AdaBoost methods at the 6 locations (L1 – L6) for different prediction horizons (1–10 days).  
←



**Fig. 11.**  $p$  values for the Pearson's  $r$  of the predictions using the LSTM-AdaBoost, LSTM, AdaBoost, SVR, BPNN and S\_LSTM-AdaBoost methods at the 6 locations (L1 – L6) for different prediction horizons (1–10 days).



**Fig. 12.** Gaussian kernel density estimation of the prediction errors using the LSTM-AdaBoost, LSTM, AdaBoost, SVR, BPNN and S\_LSTM-AdaBoost methods at the 6 locations (L1 – L6) for all prediction horizons (1–10 days).

mean value of the desired SSTA values and the predicted SSTA values, respectively.  $n$  is the number of test samples. The respective indexes are calculated for the predictions on each horizon (1 to 10 days) separately.

#### 4.3. Experiment results and discussion

SSTA predictions are made on the 6 selected sites described in Section 2.1 independently for 1 to 10 days ahead. Figs. 7 to 12, Fig. S1 to Fig. S6 (in the Supplementary file), Tables 2 and 3 show the comparisons of the prediction performance of the proposed LSTM-AdaBoost combination method with LSTM, AdaBoost, SVR, BPNN and S\_LSTM-AdaBoost using different statistics and from different perspectives.

Besides, comparisons with an existing work (Zhang et al., 2017) is also conducted, which are shown and analysed in the Supplementary file (Text S4).

Fig. 7 demonstrates 1-day-ahead prediction results on the held-out testing samples using different methods. The observed SST is plotted using a purple line. From this comparison, we can see that all the six methods perform well for 1-day-ahead prediction task except for some small underestimations at crests and overestimations at troughs. Generally, the predictions of the LSTM-AdaBoost, LSTM, AdaBoost, BPNN and S\_LSTM-AdaBoost agree more with the observed values than SVR does, which is also demonstrated in Fig. S1 to Fig. S6.

Fig. 8 shows the RMSE of the predictions for different prediction

**Table 2**

Percentage improvements of the RMSE by LSTM-AdaBoost relative to LSTM and AdaBoost alone for each prediction horizon.

Relative to	Location	1 day	2 day	3 day	4 day	5 day	6 day	7 day	8 day	9 day	10 day
LSTM	L1	0.01	1.18	<b>1.35</b>	1.10	1.02	1.02	1.04	1.11	1.17	1.16
	L2	-0.39	0.11	0.20	0.24	0.35	0.42	0.41	0.58	0.64	<b>0.68</b>
	L3	1.15	<b>1.90</b>	1.41	0.97	0.88	0.85	0.47	0.26	0.06	-0.17
	L4	0.06	0.86	1.24	1.54	1.98	<b>1.99</b>	1.95	1.82	1.62	1.76
	L5	-0.24	0.38	0.51	0.60	0.71	0.78	0.92	1.16	1.33	<b>1.46</b>
	L6	-0.96	-0.01	0.18	0.47	0.71	0.78	1.19	1.65	1.93	<b>2.26</b>
AdaBoost	L1	<b>2.96</b>	2.11	2.14	2.40	2.46	2.42	2.36	2.30	2.15	2.09
	L2	2.27	2.08	2.24	<b>2.30</b>	2.21	2.10	2.07	1.88	1.80	1.74
	L3	2.72	2.42	2.89	3.25	3.39	3.49	3.86	4.14	4.37	<b>4.61</b>
	L4	2.31	1.98	1.91	1.84	1.58	1.70	1.85	2.10	2.39	<b>2.43</b>
	L5	<b>2.01</b>	1.63	1.61	1.48	1.40	1.39	1.31	1.24	1.19	1.20
	L6	<b>3.58</b>	2.43	2.34	2.09	1.86	1.80	1.46	1.10	0.92	0.70

The bold value indicates the biggest percentage improvements of the RMSE by LSTM-AdaBoost relative to LSTM and AdaBoost alone at each location.

**Table 3**

Percentage improvements of the MAE by LSTM-AdaBoost relative to LSTM and AdaBoost alone for each prediction horizon.

Relative to	Location	1 day	2 day	3 day	4 day	5 day	6 day	7 day	8 day	9 day	10 day
LSTM	L1	1.87	2.58	2.48	<b>2.86</b>	2.37	2.35	2.35	2.48	2.38	2.09
	L2	-0.38	0.2	-0.15	-0.47	-0.73	-0.46	-0.16	<b>0.28</b>	0.19	0.21
	L3	<b>3.73</b>	3.7	2.83	1.58	1.13	0.75	0.62	0.62	0.35	0.06
	L4	1.73	1.48	1.83	2.03	2.51	2.54	2.62	2.82	2.8	<b>3.06</b>
	L5	0.15	0.4	0.58	0.26	0.32	0.51	0.67	1.03	<b>1.11</b>	1.07
	L6	-0.16	0.65	0.47	0.64	0.94	0.83	1.11	1.47	1.83	<b>2.19</b>
AdaBoost	L1	1.62	1.45	1.63	<b>1.98</b>	1.73	1.52	1.41	1.36	1.47	1.44
	L2	2.75	2.59	2.73	3.06	<b>3.11</b>	2.84	2.83	2.51	2.57	2.49
	L3	1.05	1.13	2.00	2.59	3.15	3.88	<b>4.66</b>	4.73	4.65	<b>4.93</b>
	L4	1.96	1.65	1.82	2.13	2.06	2.19	2.08	2.02	2.15	<b>2.22</b>
	L5	<b>2.71</b>	1.99	2.09	2.06	2.09	1.61	1.65	1.54	1.48	1.66
	L6	<b>2.65</b>	2.12	2.35	2.15	1.84	1.87	1.58	1.36	1.00	0.66

The bold value indicates the biggest percentage improvements of the MAE by LSTM-AdaBoost relative to LSTM and AdaBoost alone at each location.

horizons using the six methods. It can be found that LSTM-AdaBoost, S\_LSTM-AdaBoost, LSTM, and AdaBoost perform much better than SVR and BPNN for all prediction horizons at all six locations. However, as with LSTM and AdaBoost, they do not outperform each other for all prediction horizons at all the six locations, namely, one of them predicts better at some locations while the other one predicts better at other locations, and one of them predicts better at some prediction horizons while another one predicts better at other horizons. Besides, the prediction horizons they are good at are not fixed for all the six locations. This is the reason why we attempt to combine these two methods. Fig. 8 also shows that the RMSEs of the predictions of LSTM-AdaBoost are smaller than or equal to those of S\_LSTM-AdaBoost for all the prediction horizons at all six locations, demonstrating that the averaging strategy is better than the complex stacking strategy for combining the predictions of LSTM and AdaBoost for the short and mid-term SSTAs prediction task. Table 2 calculates the percentage improvement of the RMSE by LSTM-AdaBoost relative to LSTM and AdaBoost alone. The biggest improvement at each location is set in bold. From this table, we can see that the RMSE of the predictions have been improved at almost all locations and for almost all prediction horizons due to the combination of LSTM and AdaBoost using averaging strategy. The biggest improvement relative to LSTM is 2.26% at location L6 for the 10-days-ahead prediction. The biggest improvement relative to AdaBoost is 4.61% at location L3 for the 10-days-ahead prediction. The only exceptions are for LSTM at location L2 for 1-day-ahead prediction, location L3 for 10-days ahead prediction, location L5 for 1-day-ahead prediction, and location L6 for 1- and 2-days-ahead predictions with RMSE increased by 0.39%, 0.17%, 0.24%, 0.96%, and 0.01% respectively, which however are negligible. Another finding is that the RMSEs at location L1 are bigger than those at the other 5 locations for almost all six methods. This may be in part due to that L1 is near the coast where mixing pixel of sea and

land affects the data quality. Though the variation of SSTA at L2 is bigger than that at L3 (see Table 1), the RMSEs of the predictions at L3 are not guaranteed bigger than those at L2. The same phenomenon happens between L4 and L5, from which it may be concluded that the variations of the SSTA time series are not positively related to the prediction performance of the six methods. It can also be found that the RMSE increases with the prediction horizons due in part to the decreasing predictability of SSTAs along with the increasing prediction horizons, and due in part to the rolling prediction scheme adopted in this study. In the rolling prediction scheme, the current predicted SSTA is used as the latest element of the input SSTA sequence to infer the SSTA on next day, during which the prediction errors are propagated and accumulated as the prediction horizon increases.

Fig. 9 is the comparison of the MAE of the predictions for different prediction horizons using the six methods. It shows almost the same behaviour as in Fig. 8. The biggest MAE improvements by LSTM-AdaBoost relative to LSTM and AdaBoost are 3.73% at location L3 for the 1-day-ahead prediction and 4.93% at location L3 for the 10-days-ahead prediction, as shown in Table 3.

To further measure and compare the quality of predicted SSTAs, Pearson's  $r$  is calculated and shown in Fig. 10 with the corresponding  $p$  values shown in Fig. 11. All the  $p$  values are equal to 0, showing that all the  $r$  values are statistically significant. It can be found that  $r$  decreases quickly as the prediction horizon increases from 1 to 10 days, showing that the linear correlation between the observed SSTAs and the predicted SSTAs becomes weaker and weaker as the prediction horizon becomes longer and longer. The  $r$  values of SVR and AdaBoost are always smaller than those of AdaBoost-LSTM for all prediction horizons at all the six locations. The  $r$  values of LSTM are smaller than or equal to those of AdaBoost-LSTM at location L1, L2, L5, and L6 for all prediction horizons. At location L3 and L4, the  $r$  values of LSTM are slightly bigger

than those of LSTM-AdaBoost. However, LSTM-AdaBoost can achieve smaller RMSEs and MAEs than LSTM do at location L3 and L4 as analysed above. Besides, it can be found that the  $r$  values of LSTM-AdaBoost are bigger than or equal to those of S\_LSTM-AdaBoost for all the prediction horizons at all 6 locations. Therefore, from the perspective of  $r$ , LSTM-AdaBoost generally performs better than SVR, LSTM, AdaBoost and S\_LSTM-AdaBoost. Exceptions are that the  $r$  values of BPNN are equal to or slightly bigger than those of LSTM-AdaBoost at location L1, L2, L3 and L5 for almost all prediction horizons, at location L4 for 9 to 10 days ahead predictions, and at L6 for 5 to 7 days ahead predictions. However, the LSTM-AdaBoost has much smaller RMSEs and MAEs than BPNN for all prediction horizons at all the six locations.

Fig. 12 shows the investigation of the prediction performance of the different methods from the perspective of error distributions. It considers the prediction errors on all prediction horizons for each method and estimates the kernel density of the prediction error distributions using Gaussian kernels (Silverman, 1986). It can be seen from Fig. 12 that the curves of the kernel density estimation (KDE) of the prediction errors of LSTM-AdaBoost are denser with means closer (almost equivalent) to 0 than those of other methods at locations L2 and L3. At locations L4 and L6, the KDE curves of LSTM-AdaBoost are slightly less dense than those of S\_LSTM-AdaBoost, but with mean closer to 0. At location L1, the KDE curve of LSTM-AdaBoost is a little bit less dense than that of AdaBoost and S\_LSTM-AdaBoost, but with mean closer to 0 than that of AdaBoost and S\_LSTM-AdaBoost. At location L5, the KDE curve of LSTM-AdaBoost is slightly less dense than that of S\_LSTM-AdaBoost, but almost the same as that of LSTM and denser than that of SVR, BPNN, and AdaBoost with mean much closer to 0. Generally, LSTM-AdaBoost produces errors that are spreading more densely around 0 than LSTM, AdaBoost, SVR, BPNN, and S\_LSTM-AdaBoost do.

## 5. Conclusions

SST is a key physical parameter of the world ocean. It can influence the ocean marine system, climate change, the distribution of precipitation and may even lead to extreme weather events such as droughts and floods. To achieve accurate short and mid-term daily SST predictions, a machine learning method combining two powerful prediction models, namely the LSTM model and the AdaBoost model is proposed. It is an ensemble of heterogeneous predictors that can benefit from their state-of-the-art time series prediction capabilities while overcoming their shortages using the averaging strategy. The 36-year daily SSTA time series data derived from AVHRR satellite sensors at 6 selected sites in the East China Sea are used to train and test the proposed method. Comparisons with LSTM, AdaBoost, SVR, BPNN and S\_LSTM-AdaBoost using different statistics, including the RMSE, MAE, Pearson's  $r$  and KDE, show that the proposed method can achieve more accurate predictions on almost all prediction horizons from 1 to 10 days. This site-specific SST prediction can benefit the planning and safety of marine activities. In the future, the long-term SST predictions and the spatiotemporal SST field predictions over large areas can be investigated with the help of cyberGIS and high-performance computing (Wang and Goodchild, 2019; Wright and Wang, 2011). Besides, application of the proposed method to the prediction of other critical oceanic, atmospheric, and environmental parameters, such as sea surface salinity (SSS), precipitation and air pollutants can be investigated as well (Alvera-Azcárate et al., 2016; Boutin et al., 2016; Mao et al., 2018; Wang et al., 2017).

## Acknowledgment

This research was supported by the National Key Research and Development Program, China (No. 2018YFB2100500), the National Natural Science Foundation of China (NSFC), China Program (No. 41890822; 41771422), Creative Research Groups of Natural Science Foundation of Hubei Province, China (No. 2016CFA003), and the

Fundamental Research Funds for the Central Universities, China (No. 2042017kf0211). Changjiang Xiao was also supported by the China Scholarship Council (CSC), China under the State Scholarship Fund (No. 201706270127) to pursue his study at University of Illinois at Urbana-Champaign. Changjiang Xiao also thanks Professor Shaowen Wang for providing guiding suggestions and the high-performance computing infrastructure to conduct this research, and Prof. Fangling Pu, Mr. Zewei Xu and Mr. Yaping Cai for providing kind help during experiments.

The authors would like to thank the following data and tool providers: NOAA for providing OISST-V2-AVHRR data (<https://www.ncdc.noaa.gov/oisst>), Google for providing the open source machine learning framework TensorFlow (<https://www.tensorflow.org/>), Keras (<https://keras.io/>) for making it easier to use TensorFlow to conduct LSTM deep learning experiment, and Scikit-learn (<http://scikit-learn.org/stable/>) for providing the machine learning library based on which the experiments are conducted.

## Declaration of competing interest

None.

## Appendix A. Supplementary data

Supplementary data to this article can be found online at <https://doi.org/10.1016/j.rse.2019.111358>.

## References

- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D.G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., Zheng, X., 2016. TensorFlow: a system for large-scale machine learning. In: Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation. USENIX Association, Savannah, GA, USA, pp. 265–283.
- Alvera-Azcárate, A., Barth, A., Parard, G., Beckers, J.M., 2016. Analysis of SMOS sea surface salinity data using DINEOF. *Remote Sens. Environ.* 180, 137–145.
- Aparna, S.G., D'Souza, S., Arjun, N.B., 2018. Prediction of daily sea surface temperature using artificial neural networks. *Int. J. Remote Sens.* 39, 4214–4231.
- Bouali, M., Sato, O.T., Polito, P.S., 2017. Temporal trends in sea surface temperature gradients in the South Atlantic Ocean. *Remote Sens. Environ.* 194, 100–114.
- Boutin, J., Martin, N., Kolodziejczyk, N., Reverdin, G., 2016. Interannual anomalies of SMOS sea surface salinity. *Remote Sens. Environ.* 180, 128–136.
- Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., Vanderplas, J., Joly, A., Holt, B., Varoquaux, G., 2013. API design for machine learning software: experiences from the scikit-learn project. In: European Conference on Machine Learning and Principles and Practices of Knowledge Discovery in Databases. Czech Republic, Prague.
- Castro, S.L., Wick, G.A., Steele, M., 2016. Validation of satellite sea surface temperature analyses in the Beaufort Sea using UpTempO buoys. *Remote Sens. Environ.* 187, 458–475.
- Chaigneau, V., Dreano, D., Agusti, S., Duarte, C.M., Hoteit, I., 2017. Decadal trends in Red Sea maximum surface temperature. *Sci. Rep.* 7, 8144.
- Chao, Z., Pu, F., Yin, Y., Han, B., Chen, X., 2018. Research on real-time local rainfall prediction based on MEMS sensors. *J. Sens.* 2018, 9.
- Che, D., Liu, Q., Rasheed, K., Tao, X., 2011. Decision tree and ensemble learning algorithms with their applications in bioinformatics. In: Arabnia, H.R., Tran, Q.-N. (Eds.), Software Tools and Algorithms for Biological Systems. Springer New York, New York, NY, pp. 191–199.
- Chen, Y., Li, Y., 2009. Entropy-based combining prediction of grey time series and its application. In: 2009 Second International Conference on Intelligent Computation Technology and Automation, pp. 37–40.
- Chen, N., Wang, K., Xiao, C., Gong, J., 2014. A heterogeneous sensor web node meta-model for the management of a flood monitoring system. *Environ. Model. Softw.* 54, 222–237.
- Drucker, H., 1997. Improving regressors using boosting techniques. In: Proceedings of the Fourteenth International Conference on Machine Learning. Morgan Kaufmann Publishers Inc, pp. 107–115.
- Fortmann-Roe, S., 2012. Understanding the Bias-Variance Tradeoff. <http://scott-fortmann-roes.com/docs/BiasVariance.html>, Accessed date: 5 May 2019.
- Francois, C., 2015. Keras. Github. <https://github.com/keras-team/keras>.
- Frank, J., Wentz, C.G., Smith, D., Chelton, D., 2000. Satellite measurements of sea surface temperature through clouds. *Science* 288, 847–850.
- Freund, Y., Schapire, R.E., 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* 55, 119–139.
- Herbert, T.D., Peterson, L.C., Lawrence, K.T., Liu, Z., 2010. Tropical ocean temperatures over the past 3.5 million years. *Science* 328, 1530–1534.
- Kim, S.Y., Upneja, A., 2014. Predicting restaurant financial distress using decision tree and AdaBoosted decision tree models. *Econ. Model.* 36, 354–362.

- Krishnamurti, T.N., Chakraborty, A., Krishnamurti, R., Dewar, W.K., Clayson, C.A., 2006. Seasonal prediction of sea surface temperature anomalies using a suite of 13 coupled atmosphere–ocean models. *J. Clim.* 19, 6069–6088.
- Kun, J., 2015. The boosting margin, or why boosting doesn't overfit. <https://jeremykun.com/2015/09/21/the-boosting-margin-or-why-boosting-doesnt-overfit/>, Accessed date: 6 July 2019.
- Laepple, T.J., Stephen (2007). Five Year Ahead Prediction of Sea Surface Temperature in the Tropical Atlantic: A Comparison Between IPCC Climate Models and Simple Statistical Methods.
- Leo, B., Jerome, F., Charles, J.S., & R.A., O. (1984). Classification and Regression Trees. Chapman and Hall/CRC.
- Lins, I.D., Araujo, M., Moura, M., Silva, M.A., Drogue, E.L., 2013. Prediction of sea surface temperature in the tropical Atlantic by support vector machines. *Comput. Stat. Data Anal.* 61, 187–198.
- Liu, H., Tian, H., Li, Y., Zhang, L., 2015. Comparison of four Adaboost algorithm based artificial neural networks in wind speed predictions. *Energy Convers. Manag.* 92, 67–81.
- Mao, F., Pan, Z., Henderson, D.S., Wang, W., Gong, W., 2018. Vertically resolved physical and radiative response of ice clouds to aerosols during the Indian summer monsoon season. *Remote Sens. Environ.* 216, 171–182.
- Messias, V.R., Estrella, J.C., Ehlers, R., Santana, M.J., Santana, R.C., Reiff-Marganiec, S., 2016. Combining time series prediction models using genetic algorithm to auto-scaling Web applications hosted in the cloud infrastructure. *Neural Comput. & Appl.* 27, 2383–2406.
- Michael, N., Tim, H., William, R., Marcus, O.C., 1999. Time series forecasting using neural networks: should the data be deseasonalized first? *J. Forecast.* 18, 359–367.
- Neetu, Sharma, R., Basu, S., Sarkar, A., & Pal, P.K. (2011). Data-adaptive prediction of sea-surface temperature in the Arabian Sea. *IEEE Geosci. Remote Sens. Lett.*, 8, 9–13.
- NOAA Optimum Interpolation Sea Surface Temperature (OISST). <https://www.ncdc.noaa.gov/oisst>, Accessed date: 17 July 2018.
- Olah, C., 2015. Understanding LSTM Networks. <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>, Accessed date: 19 July 2018.
- Pardoe, D., Stone, P., 2010. Boosting for regression transfer. In: Proceedings of the 27th International Conference on International Conference on Machine Learning. Omnipress, Haifa, Israel, pp. 863–870.
- Patil, K., Deo, M.C., 2017. Prediction of daily sea surface temperature using efficient neural networks. *Ocean Dyn.* 67, 357–368.
- Patil, K., Deo, M.C., Ghosh, S., Ravichandran, M., 2013. Predicting sea surface temperatures in the North Indian Ocean with nonlinear autoregressive neural networks. *Int. J. Oceanogr.* 2013, 11.
- Patil, K., Deo, M.C., Ravichandran, M., 2016. Prediction of sea surface temperature by combining numerical and neural techniques. *J. Atmos. Ocean. Technol.* 33, 1715–1726.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, É., 2011. Scikit-learn: machine learning in python. *J. Mach. Learn. Res.* 12, 2825–2830.
- Perrone, M.P., 1993. Improving Regression Estimation: Averaging Methods for Variance Reduction With Extensions to General Convex Measure Optimization. Doctoral dissertation. Brown University.
- PSD NOAA OI SST V2 High Resolution Dataset. <https://www.esrl.noaa.gov/psd/data/gridded/data.noaa.oisst.v2.highres.html>, Accessed date: 23 April 2019.
- Salles, R., Mattos, P., Iorgulescu, A.-M.D., Bezerra, E., Lima, L., Ogasawara, E., 2016. Evaluating temporal aggregation for predicting the sea surface temperature of the Atlantic Ocean. *Eco. Inform.* 36, 94–105.
- Schapire, R.E., Freund, Y., Bartlett, P., Lee, W.S., 1998. Boosting the margin: a new explanation for the effectiveness of voting methods. *Ann. Stat.* 26, 1651–1686.
- Sepp, H., Jürgen, S., 1997. Long short-term memory. *Neural Comput.* 9, 1735–1780.
- Silverman, B.W., 1986. Density Estimation for Statistics and Data Analysis. Chapman and Hall/CRC, United Kingdom.
- Stockdale, T.N., Balmaseda, M.A., Vidard, A., 2006. Tropical Atlantic SST prediction with coupled ocean-atmosphere GCMs. *J. Clim.* 19, 6047–6061.
- Sumner, M.D., Michael, K.J., Bradshaw, C.J.A., Hindell, M.A., 2003. Remote sensing of Southern Ocean sea surface temperature: implications for marine biophysical models. *Remote Sens. Environ.* 84, 161–173.
- Tanggang, F.T., Hsieh, W.W., Tang, B., 1998. Forecasting regional sea surface temperatures in the tropical Pacific by neural network models, with wind stress and sea level pressure as predictors. *J. Geophys. Res. Oceans* 103, 7511–7522.
- USEPA, 2016. Climate change indicators: sea surface temperature. <https://www.epa.gov/climate-indicators/climate-change-indicators-sea-surface-temperature>, Accessed date: 28 April 2018.
- Wang, S., Goodchild, M.F. (Eds.), 2019. CyberGIS for Geospatial Discovery and Innovation, 1st. 118 Springer, Dordrecht, Netherlands.
- Wang, W., Mao, F., Du, L., Pan, Z., Gong, W., Fang, S., 2017. Deriving hourly PM2.5 concentrations from Himawari-8 AODs over Beijing–Tianjin–Hebei in China. *Remote Sens.* 9 (8), 858.
- Wikipedia, 2019. Bias–Variance Tradeoff. <https://en.wikipedia.org/wiki/Bias%E2%80%93variance:tradeoff>, Accessed date: 5 May 2019.
- Wolpert, D.H., 1992. Stacked generalization. *Neural Netw.* 5, 241–259.
- Wright, D.J., Wang, S., 2011. The emergence of spatial cyberinfrastructure. *Proc. Natl. Acad. Sci. USA* 108 (14), 5488–5491.
- Wu, A., Hsieh, W.W., Tang, B., 2006. Neural network forecasts of the tropical Pacific sea surface temperatures. *Neural Netw.* 19, 145–154.
- Yan, X., Ants, L., 2000. Forecasts of tropical Pacific SST and sea level using a Markov model. *Geophys. Res. Lett.* 27, 2701–2704.
- Yao, S.-L., Luo, J.-J., Huang, G., Wang, P., 2017. Distinct global warming rates tied to multiple ocean surface temperature changes. *Nat. Clim. Chang.* 7, 486.
- Yue, L., Shen, H., Zhang, L., Zheng, X., Zhang, F., Yuan, Q., 2017. High-quality seamless DEM generation blending SRTM-1, ASTER GDEM v2 and ICESat/GLAS observations. *ISPRS J. Photogramm. Remote Sens.* 123, 20–34.
- Zang, L., Mao, F., Guo, J., Wang, W., Pan, Z., Shen, H., Zhu, B., Wang, Z., 2019. Estimation of spatiotemporal PM1.0 distributions in China by combining PM2.5 observations with satellite aerosol optical depth. *Sci. Total Environ.* 658, 1256–1264.
- Zhang, Q., Wang, H., Dong, J., Zhong, G., Sun, X., 2017. Prediction of sea surface temperature using long short-term memory. *IEEE Geosci. Remote Sens. Lett.* 14, 1745–1749.
- Zhang, X., Chen, N., Chen, Z., Wu, L., Li, X., Zhang, L., Di, L., Gong, J., Li, D., 2018. Geospatial sensor web: a cyber-physical infrastructure for geoscience research and application. *Earth Sci. Rev.* 185, 684–703.
- Zou, H., Yang, Y., 2004. Combining time series models for forecasting. *Int. J. Forecast.* 20, 69–84.