Research Article

A Novel Long Short-Term Memory Predicted Algorithm for BDS Short-Term Satellite Clock Offsets

Tailai Wen [ID]**, Gang Ou** [ID]**, Xiaomei Tang** [ID]**, Pengyu Zhang** [ID]**, and Pengcheng Wang** [ID]

College of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, China

Correspondence should be addressed to Gang Ou; ougangcs@139.com

The satellite clocks carried on the BeiDou navigation System (BDS) are a self-manufactured hydrogen clock and improved rubidium clock, and their on-orbit performance and stabilities are not as efficient as GPS and Galileo satellite clocks caused of the orbital diversity of the BDS and the complexity of the space operating environment. Therefore, the existing BDS clock product cannot guarantee the high accuracy demand for precise point positioning in real-time scenes while the communication link is interrupted. To deal with this problem, we proposed a deep learning-based approach for BDS short-term satellite clock offset modeling which utilizes the superiority of Long Short-Term Memory (LSTM) derived from Recurrent Neural Networks (RNN) in time series modeling, and we call it QPLSTM. The ultrarapid predicted clock products provided by IGS (IGU-P) and four widely used prediction methods (the linear polynomial, quadratic polynomial, gray system (GM (1,1)), and Autoregressive Integrated Moving Average (ARIMA) model) are selected to compare with the QPLSTM. The results show that the prediction residual is lower than clock products of IGU-P during 6-hour forecasting and the QPLSM shows a greater performance than the mentioned four models. The average prediction accuracy has improved by approximately 79.6, 69.2, 80.4, and 77.1% and 68.3, 52.7, 66.5, and 69.8% during a 30 min and 1-hour forecasting. Thus, the QPLSTM can be considered as a new approach to acquire high-precision satellite clock offset prediction.

1. Introduction

High-precision satellite clock is the cornerstone of building a GNSS time system, and establishing an accurate clock offset model is crucial to guarantee real-time global satellite precise positioning services (RT-PPP) [1]. Researches have shown that the main factors affecting the performance of satellite clocks come from the external environment in which the satellite operates and the physical characteristics of the atomic clock itself [2]. Compared with other GNSS systems, the satellites of BDS occupy three different types of orbits and the operating environment is more complicated, which make it unsuitable to establish a clock offset model by a conventional quadratic polynomial method so that we cannot meet the requirement of real-time precise point positioning while the communication is blocked [3–5]. For most of the RT-PPP users, they meet their requirement by obtaining clock product from the International GNSS Service Center (IGS) [6]. All types of products are listed in Table 1.

As can be seen from Table 1, RT-PPP users can obtain clock product in three ways.

The first is the broadcast ephemeris, which broadcasts a set of satellite clock coefficients per hour; the users receive it and estimate the real-time clock offsets through the quadratic polynomial fitting and forecasting. The precision of BDS broadcast ephemeris clock products is around five nanoseconds (ns). The second is the ultrarapid clock offset products, which provide the real-time forecasting of 24 h clock offsets based on the last 24 h observation. The precision of it is around 3 ns, and it is updated every six hours, which performs better than the first way. However, both of them cannot meet the requirement of the decimeter-level real-time navigation and positioning [1, 7–10]. The last way is to estimate the real-time satellite's clock offset by real-time observation flows of local or wide-area BDS stations, which can obtain the offset once in 5-10 seconds and cycle in 5 minutes. The precision of the last way is better than 0.5 ns, but it depends on the support of a

TABLE 1: IGS satellite clock product description (http://igs.org/products/).

Type		Accuracy	Delay	Update	Sample interval
Broadcast	Orbit	100 cm	Real time	1 h	Daily
	Clock	RMS: 5 ns SDev: 2.5 ns			
Ultrarapid (part of prediction)	Orbit	5 cm	Real time	UTC: 03, 09, 15, 21	15 min
	Clock	RMS: 3 ns SDev: 1.5 ns			
Ultrarapid (part of observation)	Orbit	3 cm	3 h-9 h	UTC: 03, 09, 15, 21	15 min
	Clock	RMS: 150 ps SDev: 50 ps			
Rapid	Orbit	3 cm	17 h-41 h	UTC: 17	15 min
	Clock	RMS: 75 ps SDev: 25 ps			5 min
Final	Orbit	2.5 cm	12-18 days	Thursday	15 min
	Clock	RMS: 75 ps SDev: 20 ps			30 s, 5 min

communication network and high-performance data processing equipment, and it is also unstable and difficult for common users to acquire [11–14]. Therefore, developing a high-precision satellite clock offset prediction method for common users to low-cost obtained high-precision clock offset becomes one of the significant research issues in recent years.

A lot of traditional time series modeling methods have been applied to satellite clock offset forecasting such as polynomial model, GM (1,1) prediction model, Kalman filter model, ARIMA time series model, and spectral analysis model and achieved good performance. But each of them has their own deficiencies [7, 15–17]. The polynomial model performs worse while the forecast time is growing. The performance of the GM (1,1) prediction model depends on its exponential coefficients, and its requirement on the train data set is strict. Only with sufficient prior knowledge about the atomic clock's own operating characteristics and its environmental characteristics can the Kalman filter model achieve great prediction results. As for the ARIMA time series model and spectral analysis model, the former's optimal fixed order is inconsistent for the clock's type and in different orbits [18]. The latter requires a large amount of prior data to analyze the environmental noise and periodic factors of the satellite clock [19]. Due to the limitation of these traditional models, researchers try to introduce machine learning methods into the modeling of satellite clock errors in recent years. M. Kim and J. Kim proposed a GA-ARIMA model to revise the RTS forecast products of IGS, which is 32% higher than the traditional ARIMA prediction method, but their research object is limited to GPS [20]. Wang et al. built a short-term satellite clock offset model by wavelet neural network which is optimized through particle swarm optimization algorithm, and it can make 1 h forecasting results better than 0.3 ns. Although the particle swarm optimization algorithm can provide an optimization method for the wavelet neural network, the particle swarm optimization algorithm also easily falls into the local optimum, and this research is based on the GPS constellation with stable spatial structure; we are not uncertain whether it is suitable for the BDS constellation [21]. He et al. established the BeiDou 2/3 short-term forecast model based on the least squares support vector machine (LS-SVM), and its 24-hour forecast accuracy is improved by 37.7% compared with the traditional method. However, for the LS-SVM model, there is no definite standard about selecting the length of the training data. It is not that the longer the input data length is, the better the forecasting effect gets [22]. Huang et al. applied the BP neural network to correct the BeiDou-2 ultrarapid clock prediction product. The forecasting accuracy for 3 hours, 6 hours, 12 hours, and 24 hours was improved by 23.1%, 21.3%, 20.2%, and 19.8% compared with IGS's original product. But in their study, they did not give a clear rule for the selection of parameters about training BP neural network such as the time interval of the input samples [23].

Most of the machine learning methods are behavior modeling algorithm using the information generated by the past behavior of organisms, and they can also accurately estimate various complex nonlinear time series [24]. Long Short-Term Memory (LSTM) stands out from most of time series prediction algorithms, which uses its memory block in hidden lay to extract and save time series information and use the forget gate in memory block to filter the stored sequence information so that it can make full use of time series information and give accurate prediction [25, 26]. Nowadays, LSTM has gradually been applied to deal with the problem of GNSS data processing. Kaselimi et al. constructed an ionospheric electron content prediction model based on LSTM to replace Global Ionospheric Map (GIM) data product which performs better than GIM product [27]. Tao et al. established a multipath CNN-LSTM network model to optimize the multi-GNSS positioning performance, which reduces its positioning root mean square error of multi-GNSS positioning in the east, north, and vertical directions by 62.3%, 70.8%, and 66.0% [28].

In this paper, a general deep learning method used for time-series modeling, LSTM, is taken to construct the clock bias prediction model. In order to avoid the complicated structure of LSTM, we only built single-layer LSTM and use the genetic algorithm (GA) to optimize the number of hidden layer units. Then, we divide satellite clock feature into system feature and environment feature, and the QP-based Long Short-Term Memory (QPLSTM) model is construct and is applied to short-term clock bias prediction for the first time. We used the IGU-O product to conduct six-hour forecasting compared to IGU-P product to evaluate the performance of QPLSTM. Furthermore, we also use precise satellite clock product to train our model and compare to four conventional methods during 30 min and 1 h forecasting. Finally, results show that we provide a more excellent method to realize high-precision short-term satellite clock offset prediction than the mentioned method above, and it can also replace the IGU-P product.

The remaining sections of the paper are organized as follows: the original RNN and its variant of LSTM used for time series modeling is introduced in detail in Section 2. Section 3 emphasizes on the construction for the clock offset prediction model—QPLSTM—based on the variant of LSTM which is introduced in Section 2. The simulation results and dissection are conducted in Section 4. Section 5 summarizes the performance of the QPLSTM and also points out the areas that need to be optimized and improved.

2. LSTM RNN

LSTM is one of the variants of RNN. This part will introduce the RNN and its LSTM variant which is used to establish the satellite clock offset prediction model in the next part [29]. It started from the basic RNN model, and then, it is derived to LSTM.

RNN can build a deep structure in the time dimension [30], and it has been widely used in time series modeling [31, 32]. Figure 1 shows an RNN model being expanded to a full network, and the following introduces the mathematical symbols in Figure 1:

(1) Set x_t as the input vector at time $t_\circ$

(2) Set H_t as the hidden unit value at time t. The state value depends on the input vector at the current moment and the value of the hidden layer unit at the previous moment. H_t can be calculated by the following formula:

$$H_t = f(W_{XH}x_t + W_{HH}H_{t-1}) \tag{1}$$

In the above formula, f is the activate function, such as sigmoid function and ReLU. Normally, H_0 is initialized to 0

(3) Set o_t as the output vector at time t, and it can be calculated by the following formula:

$$o_t = f(W_{HO}H_t) \tag{2}$$

(4) In the above formula, W_{XH} and W_{HO} represent the weight matrix of the input layer and output layer connected to the hidden layer. W_{HH} represents the transfer weight matrix between hidden layer units

Even though RNN does a good job in time series modeling, its deep gradient is calculated in the form of partial derivative and multiplication through each activate function, and when the input vector becomes a large-scale time series, the gradient calculation value easily disappears or explodes, making it difficult to calculate and update the gradient. And LSTM is one of the effective variants to solve this problem [26].

LSTM is designed to use a memory block to replace the hidden layer unit in the original RNN architecture. The memory block includes an input gate, an output gate, a forget gate, and self-circulating neurons [33], and the structure of which is shown in Figure 2. The input gate is used to determine how much current information is involved in the calculation of the self-circulating neuron state value of the memory block, the forget gate is used to determine how much historical information at the last moment is used to calculate the self-circulating neurons state value of the memory at this time, and the output gate is used to control the information that can be output in the memory block. By training the combination of these gates, not only the problem of gradient disappearance or explosion caused by the large-scale time sequence used for training is solved, but also the time series variables can be predicted more finely. And the number of parameters that LSTM needs to control is four times of RNN.

Figure 3 shows the LSTM model when multiple memory blocks are combined. Figure 3 also reflects the iterative steps of each door, and the mathematical expression of each step is shown as follows:

(1) Set x_t as the input vector to the memory block at time t

(2) Let W_i, W_f, W_c, W_o, U_i, U_f, U_c, U_o, V_i, V_f, and V_o represent the weight matrix of their respective steps

(3) Let b_i, b_f, b_c, and b_o be the bias vector

(4) Set h_t as the output value of the memory block at time t

(5) Set i_t and $\tilde{c}_t$ as the output value of the input gate and the activate value of the self-circulating neuron state, respectively, at time t. Both of them are calculated by the following formula:

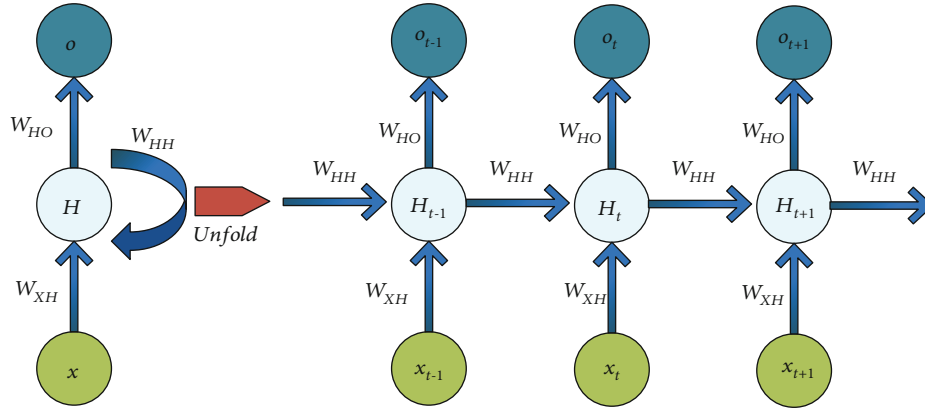$$i_t = \sigma(W_i x_t + U_i h_{t-1} + V_i c_{t-1} + b_i), \tag{3}$$

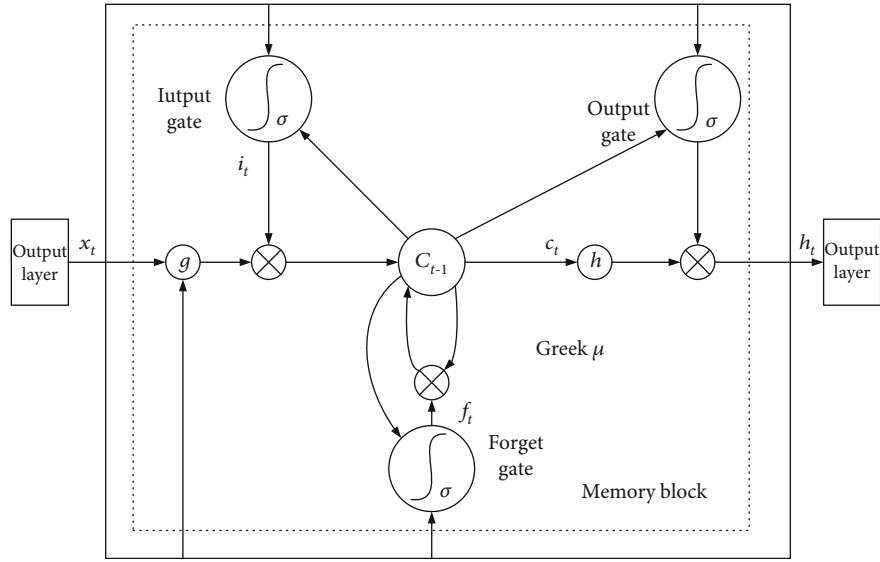FIGURE 1: A RNN with its unfolded architecture.
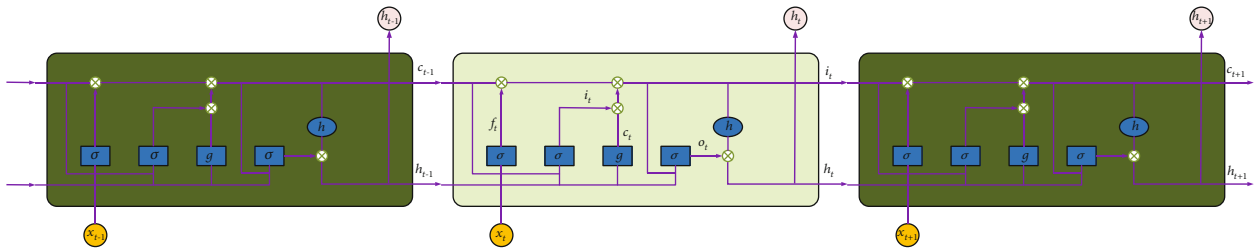

FIGURE 2: LSTM architecture of single memory block.

FIGURE 3: LSTM model with multiple memory blocks.

$$\tilde{c}_t = g(W_c x_t + U_c h_{t-1} + b_c) \tag{4}$$

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + V_f c_{t-1} + b_f\right), \tag{5}$$

$$c_t = i_t * \tilde{c}_t + f_t * c_{t-1} \tag{6}$$

(6) Set f_t and c_t as the output value of the forget gate and the value of the self-circulating neuron state, respectively, at time t. Both of them are calculated by the following formula:

(7) Set o_t and h_t as the output value of the output gate and the output value of the memory block,

respectively, at time t. Both of them are calculated by the following formula:

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + V_o c_t + b_o) \quad (7)$$

$$h_t = o_t * h(c_t) \quad (8)$$

(8) In the above formula, $\sigma(\cdot)$, $g(\cdot)$, and $h(\cdot)$, respectively, represent the standard logistic sigmoid function with range $[0, 1]$, $[-2, 2]$, and $[-1, 1]$, Both of them are defined in equations (9)–(11):

$$\sigma(x) = \frac{1}{1 + e^{-x}}, \quad (9)$$

$$g(x) = \frac{4}{1 + e^{-x}} - 2, \quad (10)$$

$$h(x) = \frac{2}{1 + e^{-x}} - 1 \quad (11)$$

A modified version of Real-Time Recurrent Learning (RTRL) that is optimized by the gradient descent method and truncated Back Propagation Through Time (BPTT) are selected as the basic theory to train LSTM RNN [34]. The objective function of training is to minimize the mean square error. The training error does not directly affect the output of the memory block but enters the linear CEC of the memory block and allows the error to pass back. In this way, the LSTM neural network has the ability to handle the large-scale time series. Before training, it is necessary to determine the number of hidden layers and the number of hidden layer units and which part of the historical data is used for training and testing. Since there is no clear theoretical basis for the selection of the number of hidden layer and the number of hidden layer units [25, 26], the single-layer LSTM is used in subsequent experiments and GA is selected to find the best number of hidden layer units. Initial learning rate, batch size, and number of interactions are 0.005, 125 and 1000, respectively. Each result of experiments converges to the optimal solution.

3. Construction of QPLSTM Used for Satellite Clock Offset Prediction

The current BDS constellation is composed of GEO, IGSO, and MEO, and the orbit type of each satellite and the atomic clock mounted on it are shown in Table 2.

The various orbit types and star clock characteristics make the traditional polynomial combined periodic term modeling method not able to achieve better forecast results for all satellites [35, 36]. The method proposed in this paper models all the satellites that can obtain observations except the satellite under testing, and the forecasting performance has been significantly improved compared with conventional methods.

In order to better understand the method proposed in this article, this chapter will first introduce the on-board characteristics of the satellite clock and then introduce the new model QPLSTM that combines the traditional method with LSTM in this article.

Assume that the satellite clock offsets is $Cx(t)$, and it can be decomposed into system variation components, periodic variation components, and random variation components [37, 38], written in the following form:

$$Cx(t) = a_0 + b_0 t + \frac{1}{2} c_0 t^2 + \sum_{i=1}^{n_p} [cs_i \cos(i\mu) + ss_i \sin(i\mu)] + \varepsilon_{cx}(t). \quad (12)$$

The first three items of the satellite clock offsets are caused by the characteristics of the satellite clock itself. a_0 is the initial phase, b_0 is the frequency deviation, and c_0 is the frequency drift. The fourth item is the periodic variation component caused by the movement of the satellite on-board, where n_p is the number of periodic variation components contained in the clock offset sequence, cs_i is the cosine coefficient of the ith period term, ss_i is the sine coefficient of the ith period term, and μ is the satellite orbit angle calculated from midnight. The fifth term is the noise caused by environmental factors, which is a random variation component.

In many cases, the traditional polynomial model is proved to be a reliable method for BDS clock offset modeling [23]. Differing from using a longer time series to analyze the periodic term and the noise term in random variation component, the method proposed in paper takes the periodic term and the noise term as the feature to construct the LSTM model.

When using LSTM to predict in practice, it is very important to determine the number of input and output layers and the number of hidden layers in the network, but there is no theory that can be followed. It is usually necessary to ensure that the vector scale of the input layer and the vector scale of the output layer are the same. And then, the biggest difficulty lies in the determination of the number of hidden layer units in LSTM modeling. Before determining the number of hidden layer units, set other parameters of LSTM for satellite clock offset prediction, mainly including the selection of the loss function, training method, and the establishment of initial parameters. The training method can be expressed as follows:

(1) Selection of the loss function for LSTM training. The selection of LSTM loss function in different scenarios is different [39]. Taking into account the complexity of the clock error model, we use the mean square error loss function. This function is suitable in this scenario and is often used in nonlinear time series forecasting, which is defined as $L = \sum_i (\|\hat{o}_i - o_i\|/2)$, where $\hat{o}_i$ represents the predicted value of the output layer

(2) Selection of the training algorithm. Truncated Back Propagation Through Time (BPTT) based on

TABLE 2: The types and orbits of BDS on-board satellite clocks (the list of events as of October 2020) (http://www.igmas.org).

Generation	Orbit	PRN	Status	Equipment clock
BDS-2	MEO	C11, C12, C14	Healthy	Rb
	GEO	C01, C02, C03, C04, C05	Healthy	Rb
	IGSO	C06, C07, C08, C09, C10, C13, C16	Healthy	Rb
BDS-3	MEO	C19, C20, C21, C22, C23, C24, C32, C33, C36, C37, C45, C46	Healthy	Rb
		C25, C26, C27, C28, C29, C30, C34, C35, C41, C42, C43, C44	Healthy	H
	IGSO	C38, C39, C40	Healthy	H
	GEO	C59, C60, C61	Testing	H

gradient descent has great advantages such as rigorous derivation, solid theoretical foundation, and strong versatility and is usually used as the main learning algorithm [34, 40]. Therefore, we also use this training method to train the LSTM model for satellite clock offset prediction. Since we do not have to worry about gradient disappearance or explosion caused by gradient descent, we only need to set the number of learning iterations, the size of learning round, and dropping rate. Then, the LSTM is trained as follows:

(a) Initializing parameters of network. The method of adaptive moment estimation is used to adjust the learning rate. Set the initial learning rate η, learning round size N, dropping iteration n, and the dropping rate lrate well; after each n rounds of adaptive learning, the next round of learning rate is $\eta^{(i+1)} = \eta^{(i)} \cdot$ lrate, where $i \in [1, \lceil N/n \rceil]$

(b) Setting LSTM as a single memory block network. Setting the transfer weights of each layer to W_i, W_f, W_c, W_o, U_i, U_f, U_c, U_o, V_i, V_f, and V_o, each of them represents the weight matrix of the input of memory block, the output of memory block, and self-circulating neuron connect to the input gate, the forget gate, the previous state of self-circulating neuron, and the output gate, respectively

(c) Forward propagation self-learning and calculating the LSTM network output of memory block $\hat{o}_i$ according to formulas (3)–(11), then calculating the value of loss function L according to the actual output o_i

(d) According to the rule of BPTT. Define $\varsigma_h^t \overset{\text{def}}{=} \partial L/\partial h_t$ and $\varsigma_c^t \overset{\text{def}}{=} \partial L/\partial c_t$. Let the vector that is input into activate function in formulas (3), (4), (5), and (7) be abbreviated as a_i^t, a_c^t, a_f^t, and a_o^t respectively. The gradient descent expression and the weight update expression are calculated by the following formula though the chain rule

Output gate:

$$\frac{\partial L}{\partial a_o^t} = \sigma'\left(a_o^t\right)h(c_t)\varsigma_h^t. \tag{13}$$

And $\partial a_o^t/\partial W_o = x_t$, $\partial a_o^t/\partial U_o = h_{t-1}$, and $\partial a_o^t/\partial V_o = c_t$.

$$\Rightarrow \begin{cases} W_o = W_o - \eta\dfrac{\partial L}{\partial W_o} = W_o - \eta\sigma'\left(a_o^t\right)h(c_t)\varsigma_h^t x_t, \\[2mm] U_o = U_o - \eta\dfrac{\partial L}{\partial U_o} = U_o - \eta\sigma'\left(a_o^t\right)h(c_t)\varsigma_h^t h_{t-1}, \\[2mm] V_o = V_o - \eta\dfrac{\partial L}{\partial V_o} = V_o - \eta\sigma'\left(a_o^t\right)h(c_t)\varsigma_h^t c_t, \end{cases}$$

$$\varsigma_c^t = o_t h'(c_t)\varsigma_h^t + f_{t+1}\varsigma_c^{t+1} + V_i\frac{\partial L}{\partial a_i^{t+1}} + V_f\frac{\partial L}{\partial a_f^{t+1}} + V_o\frac{\partial L}{\partial a_o^t}. \tag{14}$$

Self-circulating neuron:

$$\frac{\partial L}{\partial a_c^t} = i_t g'\left(a_c^t\right)\varsigma_c^t. \tag{15}$$

And $\partial a_c^t/\partial W_c = x_t$ and $\partial a_c^t/\partial U_c = h_{t-1}$. Then,

$$\Rightarrow \begin{cases} W_c = W_c - \eta\dfrac{\partial L}{\partial W_c} = W_c - \eta i_t g'\left(a_c^t\right)\varsigma_c^t x_t, \\[2mm] U_c = U_c - \eta\dfrac{\partial L}{\partial U_c} = U_c - \eta i_t g'\left(a_c^t\right)\varsigma_c^t x_t. \end{cases} \tag{16}$$

Forget gate:

$$\frac{\partial L}{\partial a_f^t} = \sigma'\left(a_f^t\right)c_{t-1}\varsigma_c^t. \tag{17}$$

And $\partial a_f^t / \partial W_f = x_t$, $\partial a_f^t / \partial U_f = h_{t-1}$, and $\partial a_f^t / \partial V_f = c_{t-1}$.

$$\Rightarrow \begin{cases} W_f = W_f - \eta \dfrac{\partial L}{\partial W_f} = W_f - \eta \sigma' \left(a_f^t \right) c_{t-1} \varsigma_c^t x_t, \\[2mm] U_f = U_f - \eta \dfrac{\partial L}{\partial U_f} = U_f - \eta \sigma' \left(a_f^t \right) c_{t-1} \varsigma_c^t h_{t-1}, \\[2mm] V_f = V_f - \eta \dfrac{\partial L}{\partial V_f} = V_f - \eta \sigma' \left(a_f^t \right) c_{t-1} \varsigma_c^t c_{t-1}. \end{cases} \quad (18)$$

Output gate:

$$\frac{\partial L}{\partial a_i^t} = \sigma' \left(a_i^t \right) g \left(a_c^t \right) \varsigma_c^t. \quad (19)$$

And $\partial a_i^t / \partial W_i = x_t$, $\partial a_i^t / \partial U_i = h_{t-1}$, and $\partial a_i^t / \partial V_i = c_{t-1}$.

$$\Rightarrow \begin{cases} W_i = W_i - \eta \dfrac{\partial L}{\partial W_i} = W_i - \sigma' \left(a_i^t \right) g \left(a_c^t \right) \varsigma_c^t x_t, \\[2mm] U_i = U_i - \eta \dfrac{\partial L}{\partial U_i} = U_i - \sigma' \left(a_i^t \right) g \left(a_c^t \right) \varsigma_c^t h_{t-1}, \\[2mm] V_i = V_i - \eta \dfrac{\partial L}{\partial V_i} = V_i - \sigma' \left(a_i^t \right) g \left(a_c^t \right) \varsigma_c^t x_t c_{t-1}. \end{cases} \quad (20)$$

Before the end of the Nth iteration of training, the learning rate is constantly changing, and the gradient and weight are continuously updated according to the above formula.

(3) Optimization of the model parameters. GA is a global optimization algorithm, which follows a certain fitness function and uses selection, crossover and mutation genetic operators based on multipoint random parallel search to effectively prevent parameters from converging to local optimal solutions [41, 42]. Therefore, it is selected to optimize the number of hidden layer units in LSTM. The fitness function of the GA is set as $f_{\text{fitness}} = 1/\text{RMS}$, where RMS represent the minimum mean square error between the LSTM's forecast result and the actual value, which is defined as follows:

$$\text{RMS} = \sqrt{\frac{1}{K} \sum_{i=1}^{K} \left(Cx_i - \widehat{Cx_i} \right)} \quad (21)$$

In above, K represents the number of total targets that need to be predicted. Cx_i represent the actual observation. $\widehat{Cx_i}$ represent the prediction by LSTM.

By training the above LSTM model, the satellite clock offset prediction model we built can be written in the following form:

$$Cx(t) = a_0 + b_0 t + \frac{1}{2} c_0 t^2 + f_{\text{lstm}}. \quad (22)$$

The flow of the proposed model for forecasting is shown in Figure 4. Firstly, we use data preprocessing method to remove gross errors in the original data, and the specific methods and effects will be introduced in the next section. Obtain the system variation components of the satellite clock through the QP model, then standardize the residuals after the QP model is fitted, and divide them into training data and test data. Under a fixed number of hidden layer units, train the LSTM.

The prediction and the test data are used to calculate the RMS together to help the GA find the optimal number of hidden layer units and obtain the optimal residual fitting model. Finally, by combining the residual fitting model and the QP model, QPLSTM is constructed.

The machine learning method is to use prior data to learn the complex nonlinear relationship between the past and the future. In practical applications, if we need to predict Tl period of data in the future, there are Nl periods of prior data. In order to make better use of the selectivity of LSTM for time series features, we use all $Nl - Tl$ data for training and use GA to find the best modified model and save it. Then, run the trained model online to provide real-time QP model correction items, and during online operation, there is no need to self-train each data set. When the model is inaccurate, change back to the traditional prediction method, and at the same time, collect new data and train a new LSTM model to construct QPLSTM.

In this section, QPLSTM is constructed and the entire model algorithm flow can be as shown in Figure 4. The data preprocessing method and the performance of it would be shown at the beginning of Section 4, and then, the experiments and evaluation for QPLSTM are taken through the processed data.

4. Results and Discussion

4.1. Data Preprocessing. During the operation of the atomic clock on-board, changes in the external environment and abnormal hardware signal output of the atomic clock can cause abnormal conditions such as gross errors and data discontinuities in the satellite time-frequency sequence.

Therefore, before establishing the satellite clock offset prediction model, it is necessary to ensure that the observed time-frequency characteristic sequence of the on-board atomic clock is clean and stable [42, 43]. The combined gross error detection method is taken to simultaneously deal with the gross errors contained in the phase and frequency [44].

The method is as follows.

Firstly, process the phase sequence of the satellite clock offset observations through 3-sigma rules.

Define the step bias of residual as

$$\delta = \sqrt{\frac{v^T v}{(n-1)}}, \quad (23)$$

$$v = Cx - \text{polyfit}(Cx),$$

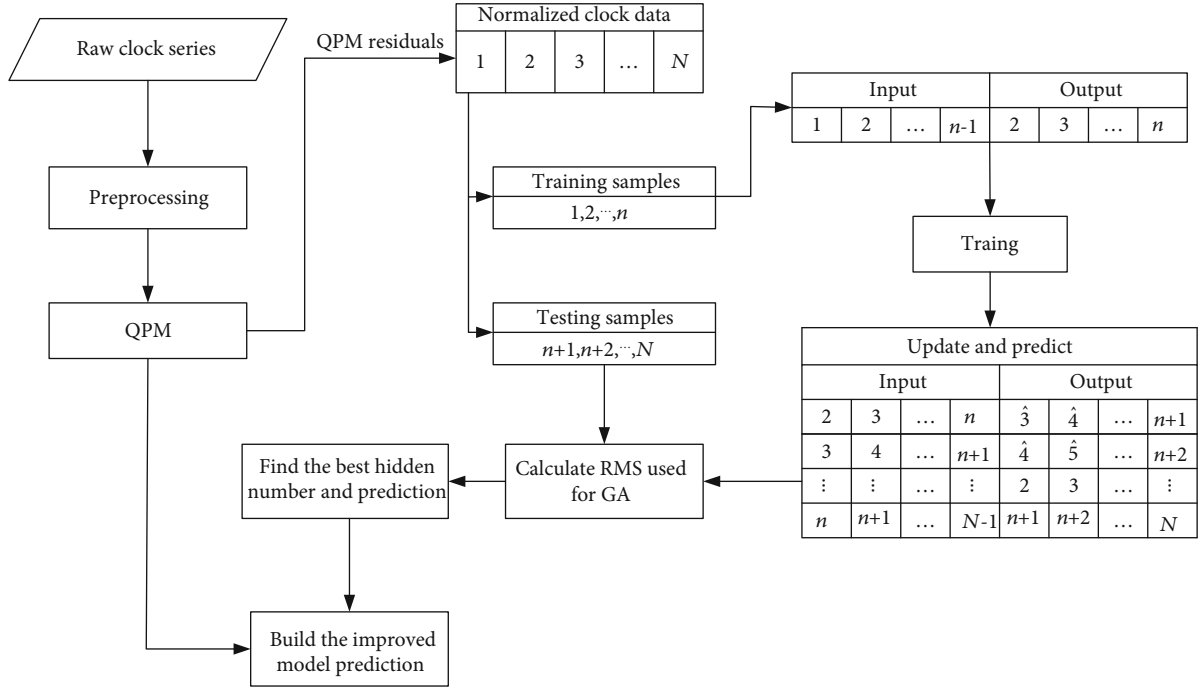where Cx represents the satellite clock offset phase sequence

Figure 4: Algorithm flow chat of QPLSTM.

obtained by observation; v represent the quadratic polynomial fitting residual vector of the clock offset phase sequence; and n represent the number of epochs of phase data. If the j th phase residual v_j satisfies $|v_j - \text{mean}(v)| > 3\delta$, we regard it as a gross error, where $\text{mean}(v)$ means the mean value of v.

After the phase data is processed, a well-known gross detection method, Median Absolute Deviation (MAD) method, is applied to process the frequency data of the satellite clock offset observations.

The MAD method can be expressed as follows:

$$b_u = \frac{(Cx_j - Cx_{j-1})}{\Delta t},$$
$$M = \text{median}\left\{ \frac{|b_u - m|}{0.6745} \right\}, \qquad (24)$$
$$m = \text{median}\{b_u\},$$

where M is the threshold of the gross error detection model; u is the number of epochs of the frequency data; and Δt is the sampling interval of the phase data. m is the median of the frequency data. When the kth epoch value in the frequency data satisfies $|b_u^k| > (m + r * M)$, it is the gross error ($r = 3$ in this model).

In order to verify the effectiveness of the above method, we select the precise clock offset product of the C36 satellite on October 15, 2020, as a representative to show the effect after data preprocessing. Both the raw data and the processed data are shown in Figure 5.

In Figure 5, the gross error is obviously eliminated through the combined gross error detection method.

The following data used in experiments are all preprocessed, and the selected data is continuous with nonjumping in the period of the day. Unless otherwise specified, we regard the 30-second sampling interval precision clock offset product provided by IGS as the most accurate value of the clock offset.

LP, QP, GM (1,1), and ARIMA models are also used as comparisons to QPLSTM in following experiments. The root mean squared error (RMS) and mean absolute error (MAE) are used to evaluate the performance of prediction model calculated through the prediction result of each model and the real data which is selected as training data. The mathematical expressions are

$$\text{RMS} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\text{Ypre}_i\text{-Yreal}_i)^2},$$
$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|(\text{Ypre}_i\text{-Yreal}_i)|, \qquad (25)$$

where Ypre_i represents the ith output data of the prediction method. Yreal_i represents the target testing data.

4.2. Experiment One. We selected the ultrarapid clock offset product with a sampling interval of 15 min provided by the Wuhan University GNSS monitoring station updated at 03:00 on October 15, 2020, for the experiment and used 12 hours of ultrarapid observation data as a priori data to train the LSTM model, then predict the subsequent 6 hours of clock offset data, and compare it with IGU-P product.

As the current ultrarapid clock offset products only have BDS-2 satellites, we give the 6 h prediction residual curve of
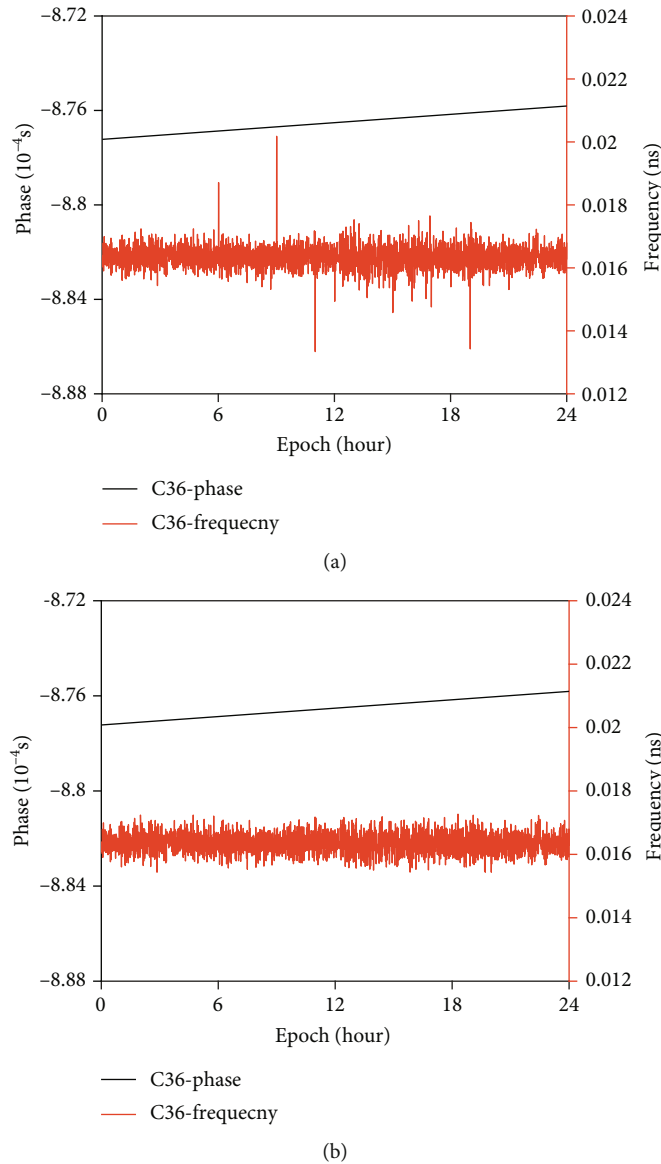
(a)

(b)

Figure 5: The combined gross error detection method. (a) The black line represents the phase data of the original clock offset, and the red line represents the frequency data of the original clock offset. (b) The black and red lines represent the phase and frequency data after correction by the gross error detection method, respectively.

QPLSTM and IGU-P as the following for each type of condition, which is calculated by prediction result and observation value. Experiments were conducted on all satellites that can obtain in the ultrarapid clock offset products during the day, and they are divided into three categories, namely, BDS2-MEO-Rb, BDS2-GEO-Rb, and BDS2-IGSO-Rb.

The simulation result is shown in Figures 6–8. Different colored curves represent satellites of different PRN numbers. The curve marked with stars represents the IGU-P product prediction errors. The curve marked with a plus sign represents the QPLSTM prediction errors. It can be seen that the clock offset value predicted by QPLSTM is closer to the actual clock offset observation value than the IGU-P product. When the external complex environment has a small impact on the clock offset, the QPLSTM method will only

slightly improve compared to the IGU-P method, such as PRN16 and PRN14.

4.3. Experiment Two. In order to further highlight the superiority of the QPLSTM method, the precise clock offset products provided by IGS on October 15, 2020, is used to conduct experiments and uses the data from the first 5 hours (600 epochs, 0:00~5:00) of the day to predict the clock offset in the next 30 minutes and 60 minutes. And compare QPLSTM with the other four traditional methods. The real precise data is selected as the testing data to verify the accuracy of the output in each prediction model. We select (C12), (C01, C03), (C08, C09, and C16), (C19, C20, C37, and C45), (C27, C28, C34, and C42), and (C40) satellites as representations of BDS clock in different scenarios.
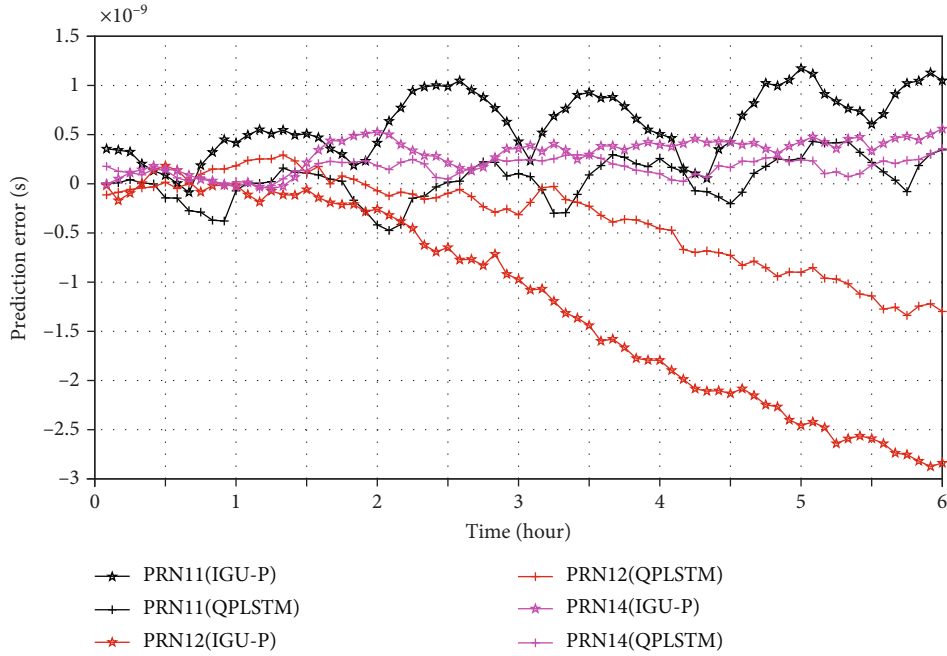
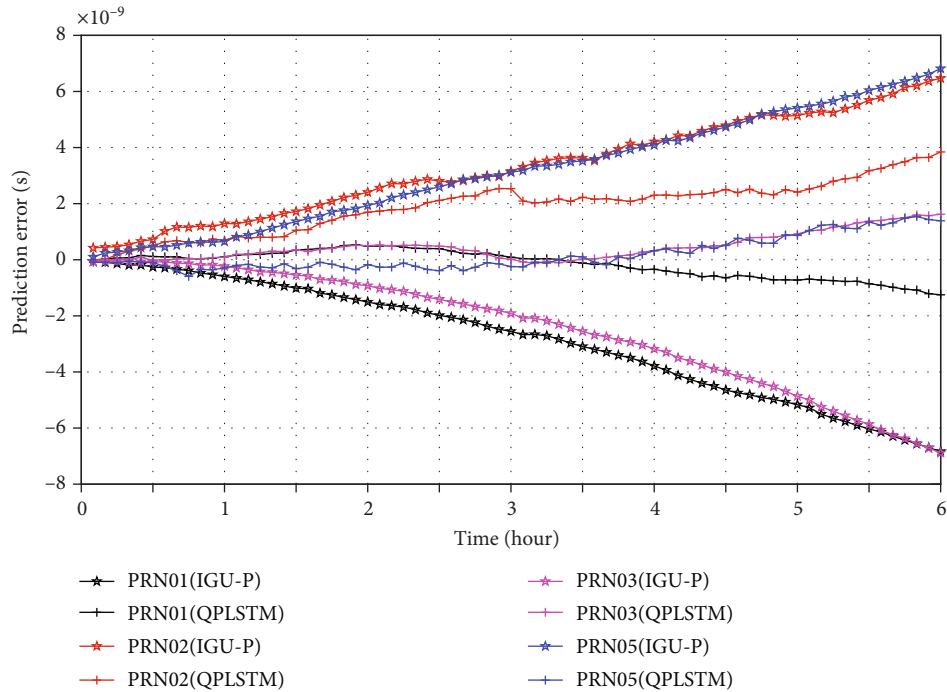Figure 6: Prediction residuals for IGU-P and QPLSTM for BDS2-MEO-Rb.

Figure 7: Prediction residuals for IGU-P and QPLSTM for BDS2-GEO-Rb.

Figure 9 represents the prediction accuracy of the clock offset for the selected 15 BDS satellites of the five models for two durations. It can be seen that each of LP, QP, GM (1,1), and ARIMA model cannot make the RMS lower than 0.3 ns in 1 h forecasting for every type of satellites.

As Figure 9 shows, the prediction accuracy for the QPLSTM model is better than that of the other four models in the two prediction durations for the 15 satellites. In par-

ticular, for C01, QPLSTM exhibits high prediction accuracy in both two durations.

Tables 3 and 4 calculate the average prediction RMS and MAE of each category of satellite on October 15 using the first 5-hour (600 epochs, 0:00~5:00) data to forecast the next 30 min and 1 h. In the table, "All" represents the average prediction accuracy of all satellites observed.
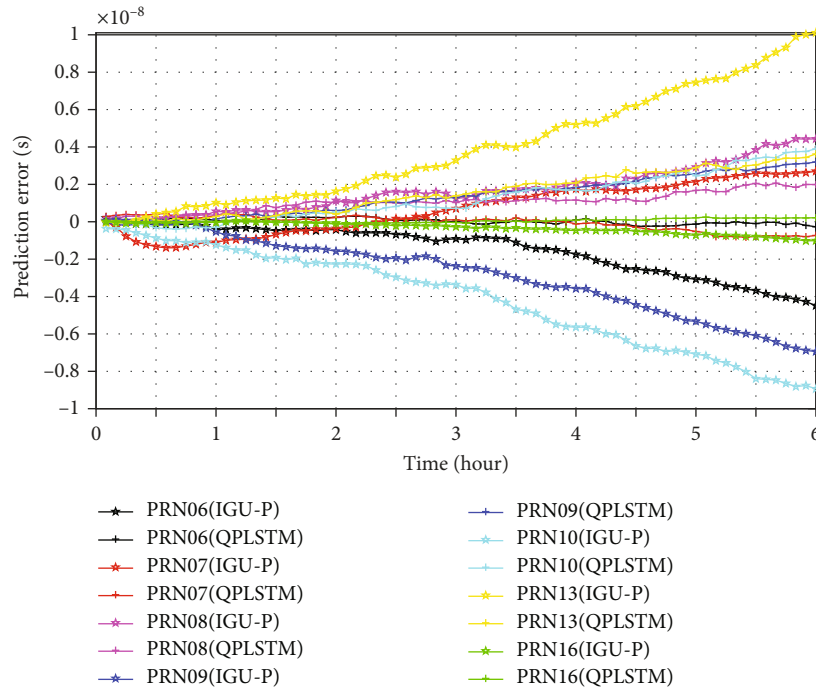
Figure 8: Prediction residuals for IGU-P and QPLSTM for BDS2-IGSO-Rb.

In Tables 5 and 6, the averaged prediction RMS and MAE are calculated during every hour on October 15, and five-hour previous data are used for training and testing the next 30 min and 60 min prediction result (e.g., data in 19:00~24:00 on October 14 is used for training. Data in 0:00~1:00 on October 15 is used for testing). Every single hour is selected to test and take the average in 24 h for different category clocks. "All" represents the average prediction accuracy for 24 single hour of all satellites observed.

As can be seen from Tables 3 and 4, the QPLSTM model is ahead of the other four models in terms of prediction accuracy for each category of satellite. Compared with the other 4 methods, QPLSTM has approximately achieved 82.7, 70.9, 83.5, and 81.9% and 75.1, 61.3, 76.4, and 77.5% improvement in the prediction RMS of 30 min and 60 min and 84.3, 72.8, 83.7, and 77.9% and 77.9, 69.5, 79.5, and 78.4% improvement in the prediction MAE of 30 min and 60 min. Tables 3 and 4 also show that the prediction accuracy of the five methods decreases as the forecast time increases, but for QPLSTM, the prediction accuracy of each category of satellite is stabilized within 0.1 ns. There are two reasons for this kind of phenomenon. First, LSTM makes full use of the prior data at the same period of time and performs well in correcting the QP model. Especially for BDS3-MEO-Rb, QPLSTM has a more significant improvement compared to the other four models, which is improved (84.3, 70.3, 83.6, and 82.5% and 77.0, 63.9, 76.3, and 85.2% in RMS; 89.2, 77.9, 88.5, and 84.1% and 80.9, 80.4, 80.3, and 86.4% in MAE), indicating that QPLSTM can better model the BDS3-MEO-Rb clock offset compared to the other categories of satellites. Second, the GA is used to optimize the number of hidden layer units of LSTM, which increases the applicability of LSTM. Compared with the

other four models, the forecasting capabilities of various satellites have been significantly improved.

In Tables 5 and 6, we can see that QPLSTM is stable for any hour of the day. Compared with the other four models, the average prediction accuracy of 30 min and 60 min forecasts is improved by 79.6, 69.2, 80.4, and 77.1% and 68.3, 52.7, 66.5, and 69.8% in RMS and 83.1, 73.0, 83.8, and 75.6% and 73.8, 58.8, 74.5, and 74.6% in MAE. It shows that the forecasting ability of the QPLSTM method is not accidental. The performance of QPLSTM can actually achieve a high precise short-term prediction than the other four models in each category of BDS satellite clock.

4.4. Experiment Three. In order to further explore the versatility and flexibility of the proposed method, the precise clock offset data provided by IGS on November 9, 2020, is used for this part of experiments, taking the data of the first 5 hours and the first 10 hours of the day to predict the clock offset in the same next 1 hour (120 epochs, 10:00~11:00). And compare QPLSTM method with the other four traditional methods. Figure 10 represents the prediction accuracy of the satellite clock offset for the selected 15 BDS satellites of the five models. It can be seen that the data of different days has a significant impact on the accuracy of the clock offset prediction for the LP, QP, GM (1,1), and ARIMA models and the accuracy of most satellites is less than 0.3 ns. But for the QPLSTM model, it performs stalely and better than the other four models.

It can be seen from Figure 10 that the modeling effect of the QPLSTM model is better than the other four models even the scale of prior data used for training differs. For the C34 satellite, it can be clearly seen that with the increase of prior data, the prediction accuracy of the
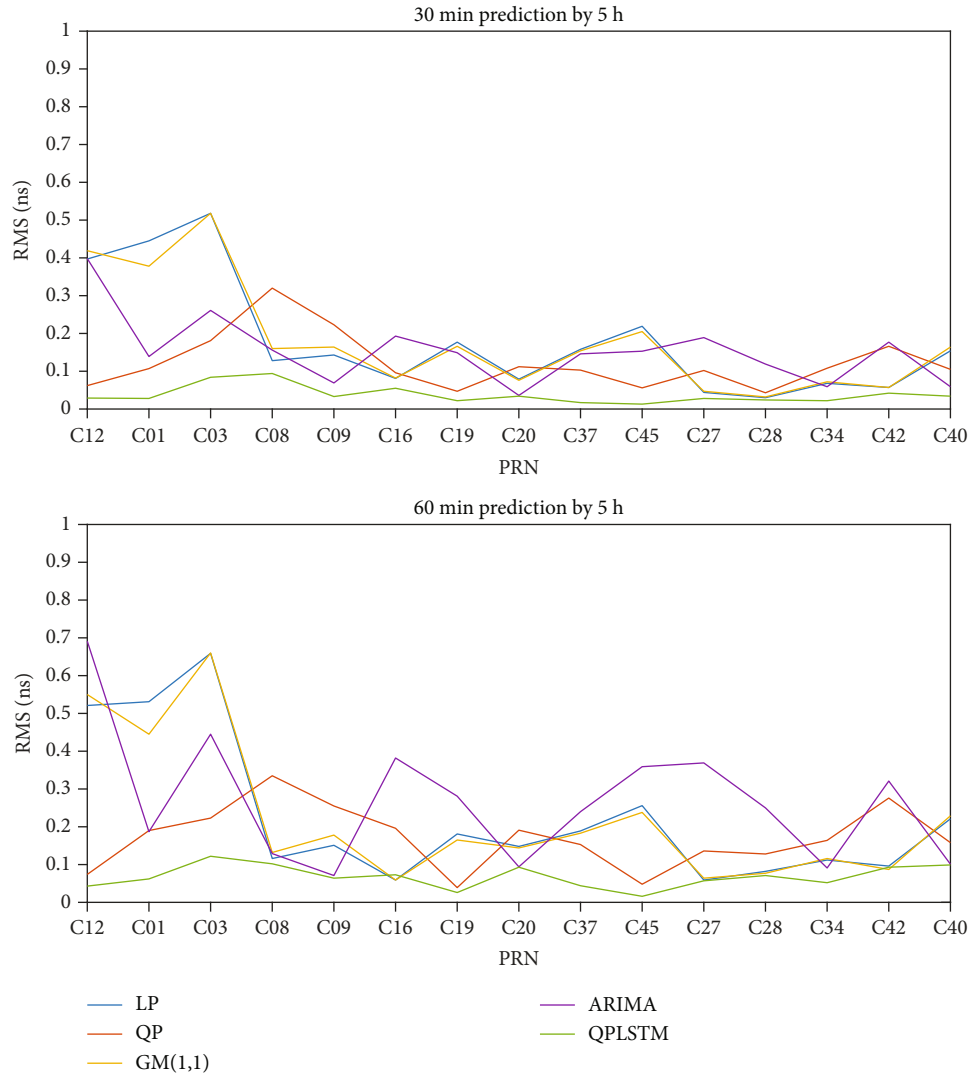
FIGURE 9: Prediction precisions of the five models for 30 min and 60 min forecasting for each BDS satellite.

TABLE 3: 30 min and 60 min average prediction RMS for six different clock categories based on the five models for 1 h.

Clock type	5 h training predict 30 min (ns)					5 h training predict 60 min (ns)				
	LP	QP	GM (1,1)	ARIMA	QPLSTM	LP	QP	GM (1,1)	ARIMA	QPLSTM
BDS2-MEO-Rb	0.242	0.075	0.243	0.251	0.025	0.360	0.075	0.415	0.386	0.041
BDS2-GEO-Rb	0.433	0.109	0.367	0.179	0.042	0.443	0.194	0.382	0.276	0.065
BDS2-IGSO-Rb	0.159	0.178	0.262	0.115	0.053	0.159	0.214	0.286	0.160	0.068
BD3-MEO-Rb	0.158	0.079	0.150	0.121	0.021	0.174	0.111	0.169	0.271	0.040
BD3-MEO-H	0.049	0.105	0.052	0.136	0.029	0.102	0.161	0.101	0.234	0.057
BD3-IGSO-H	0.190	0.071	0.194	0.055	0.023	0.243	0.092	0.246	0.087	0.054
All	0.186	0.110	0.194	0.178	0.032	0.225	0.145	0.238	0.249	0.056

four models is significantly reduced, but the prediction accuracy of the QPLSTM model is improved. For the C27 and C19 satellite, the prediction accuracy of LP and GM (1,1) models is improved with the increase of prior data. For the C16 satellite, the prediction performance of the five models has decreased even with the increase of prior data.

In order to show prediction accuracy of the five models and the influence of the increased prior data more accurately, Tables 7 and 8 list the average prediction RMS and MAE for each category satellites using 5 h or 10 h prior data to forecast the same one hour. Every one hour of the day on November 9, 2020, is included to test and then calculate the averaged RMS (e.g., data in 19:00~24:00 and data in

TABLE 4: 30 min and 60 min average prediction MAE for six different clock categories based on the five models for 1 h.

Clock type	5 h training predict 30 min (ns)					5 h training predict 60 min (ns)				
	LP	QP	GM (1,1)	ARIMA	QPLSTM	LP	QP	GM (1,1)	ARIMA	QPLSTM
BDS2-MEO-Rb	0.225	0.065	0.235	0.211	0.021	0.332	0.062	0.391	0.340	0.034
BDS2-GEO-Rb	0.427	0.105	0.363	0.162	0.036	0.427	0.185	0.384	0.252	0.057
BDS2-IGSO-Rb	0.105	0.195	0.122	0.121	0.053	0.147	0.198	0.270	0.136	0.055
BD3-MEO-Rb	0.157	0.077	0.148	0.107	0.017	0.168	0.164	0.163	0.236	0.032
BD3-MEO-H	0.045	0.096	0.048	0.119	0.023	0.082	0.119	0.082	0.197	0.047
BD3-IGSO-H	0.189	0.066	0.191	0.051	0.019	0.235	0.083	0.238	0.077	0.044
All	0.179	0.103	0.172	0.127	0.028	0.204	0.148	0.220	0.209	0.045

TABLE 5: 30 min and 60 min average prediction RMSE for six different clock categories based on the five models for 24 h.

Clock type	5 h training predict 30 min (ns)					5 h training predict 60 min (ns)				
	LP	QP	GM (1,1)	ARIMA	QPLSTM	LP	QP	GM (1,1)	ARIMA	QPLSTM
BDS2-MEO-Rb	0.278	0.097	0.269	0.229	0.030	0.346	0.116	0.371	0.363	0.055
BDS2-GEO-Rb	0.366	0.166	0.324	0.179	0.047	0.549	0.152	0.458	0.302	0.069
BDS2-IGSO-Rb	0.208	0.218	0.294	0.151	0.068	0.195	0.233	0.300	0.315	0.085
BD3-MEO-Rb	0.161	0.094	0.158	0.156	0.026	0.212	0.117	0.193	0.311	0.045
BD3-MEO-H	0.075	0.101	0.077	0.118	0.032	0.108	0.153	0.109	0.221	0.058
BD3-IGSO-H	0.201	0.119	0.203	0.048	0.040	0.261	0.125	0.264	0.077	0.067
All	0.192	0.127	0.199	0.171	0.039	0.249	0.167	0.236	0.262	0.079

TABLE 6: 30 min and 60 min average prediction MAE for six different clock categories based on the five models for 24 h.

Clock type	5 h training predict 30 min (ns)					5 h training predict 60 min (ns)				
	LP	QP	GM (1,1)	ARIMA	QPLSTM	LP	QP	GM (1,1)	ARIMA	QPLSTM
BDS2-MEO-Rb	0.251	0.082	0.248	0.185	0.026	0.317	0.094	0.343	0.314	0.046
BDS2-GEO-Rb	0.361	0.162	0.319	0.164	0.042	0.534	0.142	0.468	0.271	0.061
BDS2-IGSO-Rb	0.198	0.203	0.283	0.123	0.057	0.205	0.204	0.270	0.268	0.070
BD3-MEO-Rb	0.138	0.075	0.133	0.125	0.018	0.170	0.148	0.163	0.238	0.031
BD3-MEO-H	0.073	0.088	0.076	0.119	0.028	0.132	0.114	0.128	0.205	0.045
BD3-IGSO-H	0.199	0.096	0.200	0.045	0.019	0.253	0.118	0.256	0.071	0.049
All	0.184	0.115	0.192	0.127	0.031	0.222	0.141	0.228	0.229	0.058

14:00~24:00 on November 8 are used for training. Data in 0:00~1:00 on November 9 is used for testing).

As can be seen from Tables 7 and 8, with the increase of prior data, the average prediction accuracy of QPLSTM on different clock categories of satellites improves, while the prediction accuracy of the other four models does not show the same trend.

For the category of BDS2-MEO-Rb, the prediction accuracy of the LP, GM (1,1) and ARIMA models performs stably and is not changed much. The prediction accuracy of the QP model decreases significantly, while the QPLSTM is improved significantly. For the categories of BDS2-GEO-Rb and BDS2-IGSO-Rb, the prediction accuracy of LP and GM (1,1) decreases significantly, while the performance of the other three models is stable. For the category of BD3-MEO-Rb, the prediction accuracy of LP and GM (1,1) decreases significantly, while that of QP, ARIMA, and QPLSTM improves. For the category of BD3-MEO-H, the prediction accuracy of QP and ARIMA models decreases, while the performance of LP and GM (1,1) is relatively stable, and the prediction accuracy of the QPLSTM model improves. For the category of BD3-IGSO-H, the prediction accuracy of LP, GM (1,1) and ARIMA models decreases, while the accuracy of QP and QPLSTM improves.

Finally, the summarization about short-term clock offset prediction performance of the five different models is made as follows:

(1) For LP and QP models, noise is the main factor that causes the prediction accuracy of the model to vary with the prior data. In addition, the performance of the LP model and the GM (1,1) model is basically similar, and in some cases, they are better than the QP model. At the same time, with the increase of
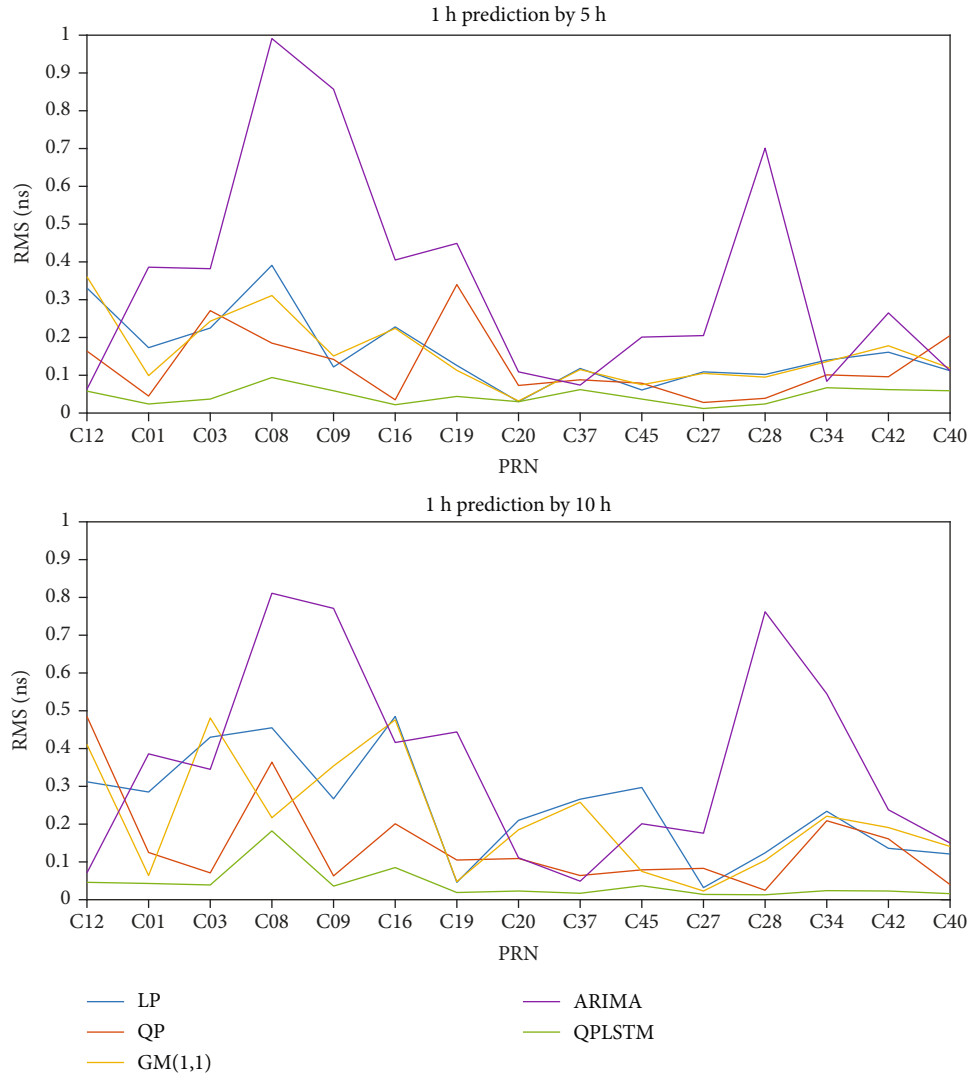
Figure 10: Prediction precisions for the five models by 5 h and 10 h modeling for each BDS satellite.

Table 7: 60 min average prediction RMS for six different clock categories based on the five models using 5 h or 10 h prior data.

Clock type	5 h training predict 60 min (ns)					10 h training predict 60 min (ns)				
	LP	QP	GM (1,1)	ARIMA	QPLSTM	LP	QP	GM (1,1)	ARIMA	QPLSTM
BDS2-MEO-Rb	0.259	0.197	0.261	0.465	0.094	0.234	0.328	0.285	0.540	0.053
BDS2-GEO-Rb	0.171	0.143	0.165	0.393	0.041	0.337	0.095	0.325	0.437	0.045
BDS2-IGSO-Rb	0.255	0.279	0.237	0.657	0.074	0.504	0.236	0.449	0.654	0.093
BD3-MEO-Rb	0.100	0.142	0.102	0.192	0.039	0.191	0.082	0.194	0.185	0.019
BD3-MEO-H	0.160	0.057	0.129	0.278	0.035	0.211	0.146	0.117	0.371	0.021
BD3-IGSO-H	0.152	0.177	0.156	0.119	0.058	0.601	0.099	0.609	0.291	0.045
All	0.174	0.156	0.163	0.349	0.052	0.319	0.154	0.291	0.398	0.042

prior data, the performance of both decreases in different degrees. And the prediction performance of these two models is also greatly affected by the satellite's environment, which limits the wide application of these two clock offset prediction models in BDS. The prediction accuracy of the ARIMA model is greatly affected by its AR and MA orders. If it is put into practice, it is necessary to constantly search for its optimal order while satellite changes or using different input data and frequently changing the model. Thus, in a range of certain order, it preforms unevenly

TABLE 8: 60 min average prediction MAE for six different clock categories based on the five models using 5 h or 10 h prior data.

Clock type	5 h training predict 60 min (ns)					10 h training predict 60 min (ns)				
	LP	QP	GM (1,1)	ARIMA	QPLSTM	LP	QP	GM (1,1)	ARIMA	QPLSTM
BDS2-MEO-Rb	0.237	0.161	0.244	0.387	0.071	0.221	0.310	0.244	0.448	0.042
BDS2-GEO-Rb	0.157	0.137	0.151	0.349	0.030	0.327	0.085	0.151	0.382	0.036
BDS2-IGSO-Rb	0.234	0.307	0.215	0.558	0.058	0.492	0.208	0.215	0.573	0.079
BD3-MEO-Rb	0.096	0.132	0.098	0.167	0.033	0.187	0.078	0.069	0.161	0.015
BD3-MEO-H	0.158	0.047	0.128	0.242	0.027	0.208	0.144	0.128	0.329	0.017
BD3-IGSO-H	0.145	0.163	0.148	0.095	0.046	0.592	0.087	0.148	0.241	0.032
All	0.164	0.151	0.154	0.299	0.041	0.312	0.142	0.147	0.346	0.035

(2) Compared with the other four models, the QPLSTM model is suitable for every category clock, and the change of clock on-board environment will not have a significant impact on its ability of prediction. QPLSTM can further optimize its modeling accuracy with the improvement of prior data, and its modeling accuracy can reach below 0.2 ns. Compared with the clock offset prediction products provided by IGU-P, QPLSTM is more effective. And LSTM is introduced into the model as a correction item, making its application scenarios more flexible

5. Conclusions

Aimed at improving the performance of the clock offset prediction method for BDS, this paper proposes a high-precision short-term clock offset prediction method, which is based on LSTM considering the characteristics of equipped atomic clock, named QPLSTM. In order to eliminate the outliers in the clock offset data and improve the LSTM modeling performance, the combined gross error detection method is introduced to preprocess the original data and the GA optimization method is used to optimize the number of LSTM hidden layer units. Experiments have shown that the QPLSTM method is better than the IGU-P solution, whose prediction accuracy of 30 min and 60 min is improved by approximately 79.6, 69.2, 80.4, and 77.1% and 68.3, 52.7, 66.5, and 69.8% in RMS, compared with the four conventional models. And QPLSTM is more versatile and stable. In addition, this modeling method that combines physical properties and neural networks can also provide references for other fields.

Due to the limitation of the LSTM theory, although the LSTM network itself has many variants and the ways of applying, the amount of calculation will increase significantly when increasing the prior data scale to pursue high prediction performance. The gradient problem of RNN has been solved to a certain extent by LSTM, which does not mean that the gradient will not disappear or explode when a much larger-scale time series is used as the input for network training. What needs to be done next for using LSTM construct a satellite clock offset prediction model is to find the way that not only maintains a high-precision prediction but also avoids the increased computational burden and the gradient problem caused by deep learning structure.

Data Availability

The experimental data in the manuscript are all public data and can be downloaded from the analysis center of Wuhan University website (http://www.igs.gnsswhu.cn/).

Conflicts of Interest

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

References

[1] G. W. Huang, Q. Zhang, and G. C. Xu, "Real-time clock offset prediction with an improved model," *GPS Solutions*, vol. 18, no. 1, pp. 95–104, 2014.

[2] D. W. Allan, "Time and frequency (time-domain) characterization, estimation, and prediction of precision clocks and oscillators," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 34, no. 6, pp. 647–654, 1987.

[3] O. Montenbruck, A. Hauschild, P. Steigenberger, U. Hugentobler, P. Teunissen, and S. Nakamura, "Initial assessment of the COMPASS/BeiDou-2 regional navigation satellite system," *GPS Solutions*, vol. 17, pp. 211–222, 2012.

[4] H. Liu and X. Cheng, "GNSS performance research for MEO, GEO, and HEO," *China Satellite Navigation Conference (CSNC) 2017 Proceedings: Volume III*, J. Sun, J. Liu, and Y. Yang, Eds., , pp. 37–45, Springer, Singapore, 8th edition, 2017.

[5] P. Steigenberger, U. Hugentobler, A. Hauschild, and O. Montenbruck, "Orbit and clock analysis of compass GEO and IGSO satellites," *Journal of Geodesy*, vol. 87, no. 6, pp. 515–525, 2013.

[6] O. Montenbruck, P. Steigenberger, L. Prange et al., "The multi-GNSS experiment (MGEX) of the International GNSS Service (IGS) - achievements, prospects and challenges," *Advances in Space Research*, vol. 59, no. 7, pp. 1671–1697, 2017.

[7] G. Busca and Q. Wang, "Time prediction accuracy for a space clock," *Metrologia*, vol. 40, pp. 265–269, 2003.

[8] K. L. Senior, J. R. Ray, and R. L. Beard, "Characterization of periodic variations in the GPS satellite clocks," *GPS Solutions*, vol. 12, no. 3, pp. 211–225, 2008.

[9] Y. J. Heo, J. Cho, and M. B. Heo, "Improving prediction accuracy of GPS satellite clocks with periodic variation behaviour," *Measurement Science and Technology*, vol. 21, no. 7, article 073001, 2010.

[10] G. W. Huang and Q. Zhang, "Real-time estimation of satellite clock offset using adaptively robust Kalman filter with classified adaptive factors," *GPS Solutions*, vol. 16, no. 4, pp. 531–539, 2012.

[11] Y. Lv, Z. Dai, Q. Zhao, S. Yang, J. Zhou, and J. Liu, "Improved short-term clock prediction method for real-time positioning," *Sensors*, vol. 17, no. 6, p. 1308, 2017.

[12] M. R. Ge, G. Gendt, G. Dick, F. P. Zhang, and M. Rothacher, "A new data processing strategy for huge GNSS global networks," *Journal of Geodesy*, vol. 80, no. 4, pp. 199–203, 2006.

[13] M. R. Ge, J. P. Chen, and J. Dousa, "A computationally efficient approach for estimating high-rate satellite clock corrections in real-time," *GPS Solutions*, vol. 15, pp. 9–17, 2011.

[14] A. Hauschild, O. Montenbruck, J. M. Sleewaegen, L. Huisman, and P. J. Teunissen, "Characterization of compass M-1 signals," *GPS Solutions*, vol. 16, no. 1, pp. 117–126, 2012.

[15] F. Vernotte, J. Delporte, M. Brunet, and T. Tournier, "Uncertainties of drift coefficients and extrapolation errors: application to clock error prediction," *Metrologia*, vol. 38, no. 4, pp. 325–342, 2001.

[16] G. Panfilo and P. Tavella, "Atomic clock prediction based on stochastic differential equations," *Metrologia*, vol. 45, pp. 108–116, 2008.

[17] J. Davis, S. Bhattarai, and M. Ziebart, "Development of a Kalman filter based GPS satellite clock time-offset prediction algorithm," in *2012 European Frequency and Time Forum*, pp. 152–156, Gothenburg, Sweden, 2012.

[18] S. Stein and J. Evans, "The application of Kalman filters and ARIMA models to the study of time prediction errors of clocks for use in the Defense Communication System (DCS)," in *44th Annual Symposium on Frequency Control*, pp. 630–635, Baltimore, MD, USA, 2002.

[19] J. C. Lu, C. Zhang, Y. Zheng, and R. P. Wang, "Fusion-based satellite clock bias prediction considering characteristics and fitted residue," *Journal of Navigation*, vol. 71, pp. 1–16, 2018.

[20] M. Kim and J. Kim, "GA-ARMA model for predicting IGS RTS corrections," *International Journal of Aerospace Engineering*, vol. 2017, Article ID 6316590, 7 pages, 2017.

[21] X. Wang, H. Chai, and C. Wang, "A high-precision short-term prediction method with stable performance for satellite clock bias," *GPS Solutions*, vol. 24, no. 4, pp. 1–11, 2020.

[22] L. He, H. Zhou, Z. Liu, Y. Wen, and X. He, "Improving clock prediction algorithm for BDS-2/3 satellites based on LS-SVM method," *Remote Sensing*, vol. 11, no. 21, p. 2554, 2019.

[23] G. W. Huang, B. B. Cui, Q. Zhang, W. J. Fu, and P. L. Li, "An improved predicted model for BDS ultra-rapid satellite clock offsets," *Remote Sensing*, vol. 10, no. 2, p. 60, 2018.

[24] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359–366, 1989.

[25] H. Palangi, L. Deng, Y. L. Shen et al., "Deep sentence embedding using long short-term memory networks: analysis and application to information retrieval," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 4, pp. 694–707, 2016.

[26] H. Palangi, R. Ward, and L. Deng, "Distributed compressive sensing: a deep learning approach," *IEEE Transactions on Signal Processing*, vol. 64, no. 17, pp. 4504–4518, 2016.

[27] M. Kaselimi, A. Voulodimos, N. Doulamis, A. Doulamis, and D. Delikaraoglou, "A causal long short-term memory sequence to sequence model for TEC prediction using GNSS observations," *Remote Sensing*, vol. 12, no. 9, p. 1354, 2020.

[28] Y. Tao, C. Liu, T. Chen et al., "Real-time multipath mitigation in multi-GNSS short baseline positioning via CNN-LSTM method," *Mathematical Problems in Engineering*, vol. 2021, 12 pages, 2021.

[29] K. L. Chu and S. KSM, "Behavior recognition for humanoid robots using long short-term memory," *International Journal of Advanced Robotic Systems*, vol. 13, no. 6, 2016.

[30] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio Speech & Language Processing*, vol. 20, no. 1, pp. 30–42, 2012.

[31] A. J. Robinson, "An application of recurrent nets to phone probability estimation," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 298–305, 1994.

[32] A. Graves, "Sequence transduction with recurrent neural networks," *Computer Science*, vol. 58, no. 3, pp. 235–242, 2012.

[33] F. A. Gers, J. Schmidhuber, and F. Cummins, *Learning to Forget: Continual Prediction with LSTM*, 1999.

[34] H. Jaeger, "A tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the "echo state network" approach," Technical Report GMD Report 148, German National Research Center for Information Technology, 2002.

[35] B. Wang, Y. D. Lou, J. Liu, Q. Zhao, and X. Su, "Analysis of BDS satellite clocks in orbit," *GPS Solutions*, vol. 20, no. 4, pp. 783–794, 2016.

[36] G. W. Huang, Q. Zhang, H. Li, and W. Fu, "Quality variation of GPS satellite clocks on-orbit using IGS clock products," *Advances in Space Research*, vol. 51, no. 6, pp. 978–987, 2013.

[37] O. Montenbruck, P. Steigenberger, E. Schoenemann et al., "Flight characterization of new generation GNSS satellite clocks," *Navigation, Journal of the Institute of Navigation*, vol. 59, no. 4, pp. 291–302, 2012.

[38] K. Senior, P. Koppang, and J. Ray, "Developing an IGS time scale," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 50, no. 6, pp. 585–593, 2003.

[39] L. Rosasco, E. D. Vito, A. Caponnetto, M. Piana, and A. Verri, "Are loss functions all the same?," *Neural Computation*, vol. 16, no. 5, pp. 1063–1076, 2004.

[40] H. Jaeger, "Long short-term memory in echo state networks: details of a simulation study," Jacobs University Technical Report no. 27, 2012.

[41] X. Z. Gao and S. Ovaska, "Genetic algorithm training of Elman neural network in motor fault detection," *Neural Computing & Applications*, vol. 11, no. 1, pp. 37–44, 2002.

[42] T. Kerh, D. Gunaratnam, and Y. Chan, "Neural computing with genetic algorithm in evaluating potentially hazardous metropolitan areas result from earthquake," *Neural Computing and Applications*, vol. 19, no. 4, pp. 521–529, 2010.

[43] H. R. Guo and Y. X. Yang, "Analyses of main error sources on time-domain frequency stability for atomic clocks of navigation satellites," *Geomatics and Information Science of Wuhan University: Wuhan, China*, vol. 34, pp. 218–221, 2009.

[44] L. Galleani and L. Sacerdote, "A mathematical model for the atomic clock error in case of jumps," *Metrologia*, vol. 40, pp. 257–264, 2003.