electronics

MDPI

Article

Prediction of Sunspot Number with Hybrid Model Based on 1D-CNN, BiLSTM and Multi-Head Attention Mechanism

Huirong Chen [1], Song Liu [2], Ximing Yang [2], Xinggang Zhang [2], Jianzhong Yang [3,*] and Shaofen Fan [4,*]

[1] College of Resources and Environment, Beibu Gulf University, Qinzhou 535011, China; chenhuirong@bbgu.edu.cn
[2] College of Machinery and Shipping, Beibu Gulf University, Qinzhou 535011, China
[3] College of Electronic and Information Engineer, Beibu Gulf University, Qinzhou 535011, China
[4] Network and Educational Technology Center, Jinan University, Guangzhou 510632, China
* Correspondence: gxyjz@bbgu.edu.cn (J.Y.); fanshaofen@jnu.edu.cn (S.F.)

Abstract: Sunspots have a significant impact on human activities. In this study, we aimed to improve solar activity prediction accuracy. To predict the sunspot number based on different aspects, such as extracted features and relationships among data, we developed a hybrid model that includes a one-dimensional convolutional neural network (1D-CNN) for extracting the features of sunspots and bidirectional long short-term memory (BiLSTM) embedded with a multi-head attention mechanism (MHAM) to learn the inner relationships among data and finally predict the sunspot number. We evaluated our model and several existing models according to different evaluation indicators, such as mean absolute error (MAE) and root mean square error (RMSE). Compared with the informer, stacked LSTM, XGBoost-DL, and EMD-LSTM-AM models, the RMSE and MAE of our results were more than 42.5% and 65.1% lower, respectively. The experimental results demonstrate that our model has higher accuracy than other methods.

Keywords: sunspot prediction; 1D-CNN; BiLSTM; multi-head attention mechanism

Citation: Chen, H.; Liu, S.; Yang, X.; Zhang, X.; Yang, J.; Fan, S. Prediction of Sunspot Number with Hybrid Model Based on 1D-CNN, BiLSTM and Multi-Head Attention Mechanism. *Electronics* **2024**, *13*, 2804. https://doi.org/10.3390/electronics13142804

Academic Editor: Simeone Marino

Received: 22 June 2024
Revised: 10 July 2024
Accepted: 10 July 2024
Published: 16 July 2024

1. Introduction

Sunspots are prominent features of the Sun's surface and usually appear as darker areas [1], and their formation and activity significantly impact the magnetic field and energy output of this star. The Sun is the source of light and heat for the Earth, and its activity has a variety of effects on the planet. For instance, sunspots cause climate change on Earth. More than 100 years ago, a Swiss astronomer discovered that when there are many sunspots, our planet's climate is dry, and agriculture is impaired; on the contrary, when there are few sunspots, the climate is humid, and rain is torrential [2]. Zhu Kezhen, a famous Chinese scientist, also found that in every century in which there were many sunspots, as recorded in ancient Chinese books, there were also more particularly cold winters in China. Researchers have studied the rainfall change patterns in some areas and found that change occurs every 11 years, which is probably related to the increase and decrease in the number of sunspots [3]. Since sunspots are violent phenomena resulting from material activity on the Sun, it is natural that they have an impact on the Earth. Therefore, the accurate prediction of sunspot activity is very important for understanding solar behavior.

Many researchers have used different methods to forecast sunspot numbers. These methods can be classified into three main categories, including physical models, statistical models, and methods based on machine learning. There are two types of physical models: prediction models based on surface flux transport and prediction models based on dynamo theory. The former predicts the peak and cycle length of the solar cycle several years in advance by simulating magnetic field migration and evolution on the Sun's surface [4,5]; even though in these models, the approach to the boundary is different (some use empirical

relationships and some are data-driven), their results for Solar Cycle 25 are similar. Models based on dynamo theory are used for prediction on the basis of physics-based integrated conservation equations, which can also include a data assimilation model. The mature model-based dynamo theory for solar cycle prediction not only allows for prediction but also offers a useful means of assessing how well we comprehend the solar cycle. However, some of these models can only predict the amplitude of the cycle, not its shape, so they do not provide physical insights into how the cycle changes.

As the number of records about sunspots has increased, in the past decade, several statistical models, such as models based on probability theory and established by using mathematical–statistical methods, have been applied to predict future sunspots. The classical statistical method used is autoregression (AR). R. Werner [6] developed an AR method based on the Box–Jenkins method for predicting the annual sunspot number from 1749 to 2010; it was estimated that by 2013, the number of sunspots would have reached its maximum value of approximately 90. Exponential smoothing was an early method for predicting linear time series [7]. The logistic regression method [8] has been used to construct a model for predicting the categories of solar flares. The local linear approximation method has been adopted to predict the number of upcoming sunspots by extending a continuous dataset based on the group sunspot number and the international sunspot number [9]. The generalized autoregressive conditional heteroskedasticity (GARCH) and AR models, combined in a model denoted as AR-GARCH, have been used to predict the sunspot cycle [10]. Some improved methods based on AR have been utilized for sunspot prediction, such as the autoregressive moving average method [11,12], the autoregressive integrated moving average (ARIMA) method, and their variants. Abdel-Rahman and Marzouk [6] applied the ARIMA statistical method to predict the number of sunspots. Because the sunspot number is a classical time series that may contain seasonal components, seasonal ARIMA (SARIMA) was developed by Box and Jenkins [13]. These methods have demonstrated better performance on linearly changing time series and have long played a dominant role in the field of time series forecasting. However, their accuracy in capturing nonlinear relationships in chaotic time series is poor. Many prediction methods for nonlinear time series have been applied to forecast the number of sunspots. For example, the gray topological theory [14] has been applied to predict the average yearly sunspot number. Sabarinath and Anilkumar [15] combined the multivariate regression technique with a binary mixture of Laplace distribution functions to predict the cycle of sunspot numbers.

The application of machine learning technology has become a research hotspot and is utilized for sunspot number prediction. These methods include the Bayesian approach [16,17], support vector machine [18,19], and random forest [20]. In recent years, deep learning (DL) has become one of the mainstream methods in the time series field. DL-based models can be divided into two categories: single-neural-network-based models and hybrid-neural-network-based models. In [21], a recurrent convolutional neural network was adopted for sunspot prediction, and a predictive system was constructed with a customized graphic user interface in MATLAB 7 [22]. Amrita Prasad et al. [23] used a long short-term memory (LSTM) neural network to predict Solar Cycle 25 and predicted that the peak of sunspot activity would appear approximately in August 2023. Zhu et al. [24] optimized an LSTM neural network to predict Solar Cycle 25 by fine-tuning the different hyperparameters of LSTM. Abhijeet and Vipin [25] proposed a stacked-1D-convolution-based LSTM (sConvLSTM1D) model to predict the sunspot time series and predicted that the peak would be in 2024. Dai et al. [26] first reconstructed the sunspot numbers and then used a temporal convolutional neural network (TCN) to predict the number of sunspots in Solar Cycle 24. A gated recurrent unit (GRU) [27] was used to predict the sunspot number, and a mean absolute percentage error (MAPE) index of 9.9557% was achieved. Abduallah and Wang et al. [28] proposed a method based on a transformer neural network for sunspot number prediction. Zhou et al. [29] presented an improved model, called the informer model, based on a transformer to predict the time series of sunspot numbers and verified its prediction performance. The above models are based on a single neural network.

To improve predictive accuracy, complex models have been established based on the hybridization of different neural networks and even by combining neural networks with traditional regression methods. A combined model that included a convolutional neural network and an LSTM neural network was proposed by Hu et al. [30] and was applied to predict sunspot numbers. Fully connected deep neural networks combined with LSTM have been utilized for the prediction of sunspot numbers [31]. LSTM and neural network autoregression (NNAR) [32] have also been employed in the field of sunspot number prediction. Further, Okoh et al. [33] combined regression and neural networks to forecast the sunspot number and predicted that Solar Cycle 24 would end in March 2020. A model based on empirical mode decomposition (EMD) and integrated long short-term memory (LSTM) has been used to predict sunspot numbers [34] and was further improved by Yang et al. [35] by adding an attention module. The beta+SARMA model, which is based on the beta distribution, has been combined with LSTM for predicting sunspot numbers [36]. In a previous study, researchers proposed a model composed of a convolutional neural network and bidirectional GRU and employed a novel gradient residual correction technique to improve sunspot number predictive accuracy [37]. Finally, the ensemble method, extreme gradient boosting (XGBoost), and deep learning methods have been combined into a hybrid prediction model [38].

Because LSTM can be used to process and forecast significant events in time series with comparatively lengthy intervals and delays [39], it has been widely applied to time series in different fields. In solar activity time series, there are likely relationships among sunspot numbers. A 1D-CNN model can be adopted to extract the features of sunspot numbers at different scales, and bidirectional LSTM can be utilized to learn the potential relationships among the observed data and predict the number of sunspots. Finally, unlike attention mechanisms, multi-head attention mechanisms (MHAMs) can assign different weights to each location, which can better capture long-distance dependencies in a sequence by calculating the degrees of association between each location and other locations [40]; thus, these techniques have an obvious advantage in processing long sequences and complex relationships. Based on the above analysis, we used a 1D-CNN, BiLSTM, and an MAHM to build a model to predict sunspot numbers in this study.

The remainder of this paper is organized as follows: The basic principles of the 1D-CNN, Bi-LSTM, and the multi-head attention mechanism are described in Sections 2.1–2.4. The prediction model and the evaluation indicators are described in Section 2.5 and Section 2.6, respectively. Section 3 includes the data description and experimental settings. Section 4 presents the simulation results and discussion. The conclusions are presented in Section 5.

2. Methodology

2.1. One-Dimensional Convolutional Network

A CNN, the structure of which is shown in Figure 1, is a classical neural network model in the deep learning field. To process data of different dimensions, researchers have developed three different network structures for one-dimensional sequence data, two-dimensional data (images), and three-dimensional data (video) called 1D-CNN, 2D-CNN, and 3D-CNN, respectively [41]. Among them, the 1D-CNN model has a strong feature extraction ability and can be used to analyze fixed-length signals and filter out the features of SSNs. The formulas of the convolutional layer are as follows:

$$x_j^l = f\Big(\sum_{i \in M_j} x_i^{l-1} \times k_{ij}^l + b_j^l\Big) \tag{1}$$

where x_i^l is the input of the lth layer, x_j^i is the output of the lth layer, k_{ij}^l is the convolutional kernel of the lth layer, b_j^i is the bias parameter of the lth layer, and M_j is the jth input feature vector.

$$P_I^{l+1}(j) = \underset{(j-1)W+1 \le t \le jW}{max} \{q_i^l(t)\} \quad (2)$$

where W is the width of the pool region, $q_i^l(t)$ is the value of the neural cell of the ith vector in the lth layer, and the scope of t is $[(j-1)W+1, jW]$, where $P_i^{l+1}(j)$ is the corresponding value of the neural cell.

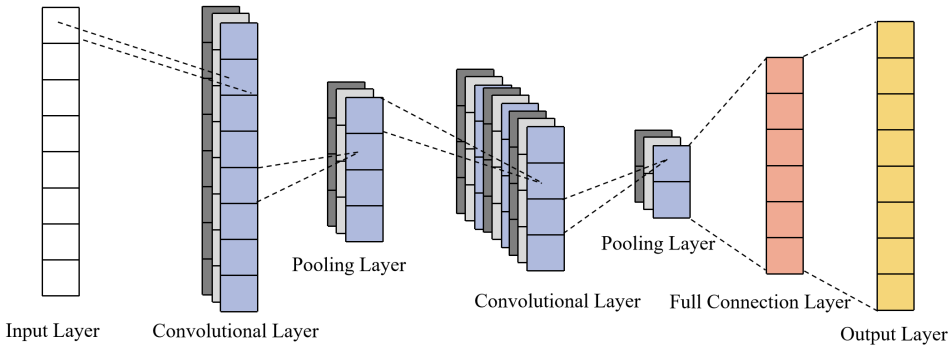

Figure 1. Structure of 1D-CNN.

2.2. Long Short-Term Memory (LSTM) Neural Network

LSTM is improved by a recurrent neural network that adds a forget gate. It can not only solve the "vanishing gradient" problem in the training process but also self-learn long-term dependencies. This method is highly suitable for predicting 1D time series and has been adopted in many applications. The basic cell is presented in Figure 2.

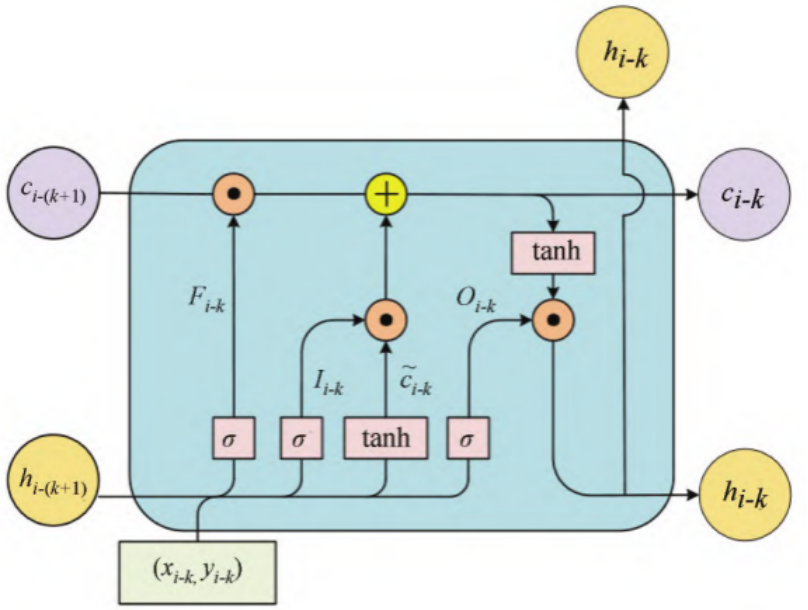

Figure 2. Basic cell of LSTM network.

The cell of the LSTM network is composed of three gates: the input gate, the forget gate, and the output gate. The forget gate includes input variable x_t, state memory unit S_{t-1}, and intermediate output h_{t-1}. In the input gate, vector preservation in the memory unit is co-determined by the results of the *sigma* and *tanh* functions with the same input (x_t). Output h_t is computed by using the updated S_t and output o_t. Their computational equations are as follows:

$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f) \quad (3)$$

$$i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i) \tag{4}$$

$$g_t = \phi(W_{gx}x_t + W_{gh}h_{t-1} + b_g) \tag{5}$$

$$i_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o) \tag{6}$$

$$S_t = g_t \odot i_t + S_{t-1} \odot f_t \tag{7}$$

$$h_t = \phi(S_t) \odot o_t \tag{8}$$

where f_t, i_t, g_t, o_t, h_t, and S_t are the states of the forget gate, input gate, output gate, intermediate output, and state unit, respectively. W_{fx}, W_{fh}, W_{ix}, W_{ih}, W_{gx}, W_{gh}, W_{ox}, and W_{oh} are the weight matrices of the corresponding gates multiplied by input data x_t and intermediate output h_{t-1}, b_f, b_i, b_g, and b_o are the bias terms of the corresponding gates, $\odot$ indicates that the items of the vector are bitwise multiplied, σ represents the function $sigmoid()$, and ϕ is the function $tanh()$.

2.3. Bidirectional LSTM

Because the data are transmitted in one direction, the LSTM model only captures historical information [42]. To overcome this problem, Schmidhuber and Sepp Hochreiter et al. proposed a two-directional LSTM network in 1997, which is an extended form of traditional LSTM. BiLSTM contains forward and backward LSTM networks [43], and its structure is shown in Figure 3. The forward LSTM processes the input sequence in the forward order, while the backward one processes the input sequence in the reverse order. They are combined to obtain more complete context information; thus, BiLSTM can better capture bidirectional semantic dependencies [44]. At time t, the hidden state of BiLSTM consists of two states: forward h_t^f and backward h_t^b. Thus, the hidden output of BiLSTM, which contains the forward and backward outputs, is denoted by H_t. The detailed formulas are defined as follows:

$$h_t^f = LSTM^f(h_{t-1}, x_t, c_{t-1}), t \in [1, T], \tag{9}$$

$$h_t^b = LSTM^b(h_{t-1}, x_t, c_{t-1}), t \in [1, T], \tag{10}$$

$$H_t = [h_t^f, h_t^b] \tag{11}$$

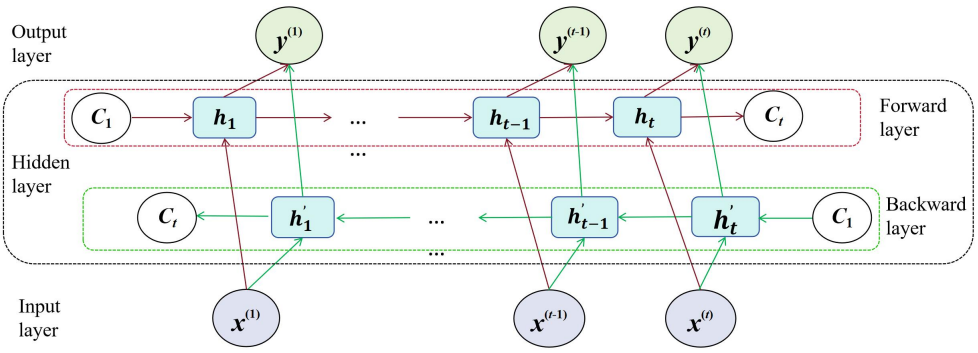

Figure 3. Structure of BiLSTM.

2.4. Multi-Head Attention Mechanism

Because the number of sunspots is very large, important information in the data has a loss probability in the model training process [45]. A multi-head attention mechanism (MHAM) was here embedded into the model, as it can effectively mine the features of the long-distance data that are relevant to the time series [46]. Its structure is shown in Figure 4. The MHAM can transform the output of BiLSTM into three same-dimensional input matrices, Q (Query), K (key), and V (Value), whose dimension size is d_k, according to three different mappings. The three output matrices are computed as follows:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{12}$$

where d_k, which is used to enlarge the weights and is normalized to [0, 1] by using softmax, is the characteristic dimension of each key.

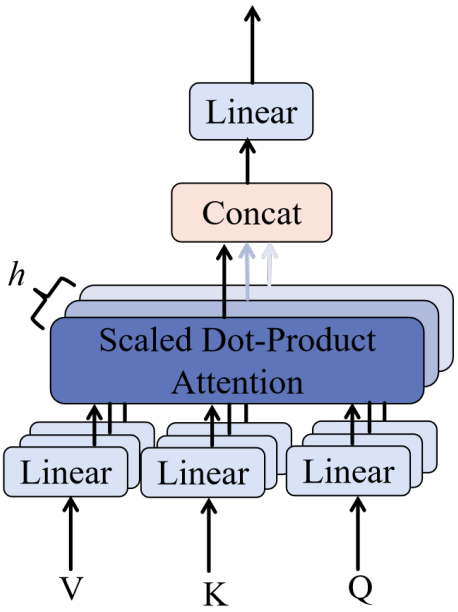

Figure 4. Structure of the multi-head attention mechanism.

To improve the ability to capture long-distance dependencies, the MHAM divides the data sequence into several sub-sequences and uses each head to compute the attention weights of all sub-sequences according to formula (13). Next, the results of h heads are spliced together for the multi-heads. Each head is concatenated, and the final result is obtained by using linear transformation.

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{13}$$

where W_i^Q, W_i^K, and W_i^V represent the weight matrices of Q, K, and V, respectively.

$$Head = MultiHead(Q, K, V) = Concat(head_1, \cdots, head_h)W^o \tag{14}$$

where W^o is the weight of the linear transformation, $head_i$ is the ith head in the multi-head module, and the function $Concat()$ is the splice operator. $MultiHead(Q, K, V)$ is the final output, and it can learn more feature information from different spaces.

2.5. Prediction Model

Because SSN data form a classical chaotic time series, they may contain complex features that are difficult to fully extract with only one technique. To improve predictive accuracy in different aspects, including feature extraction and the relationship between SNNs, we made full use of the advantages of the CNN for extracting key features, the BiLSTM model for processing time series by simultaneously considering historical information and hidden states at each timepoint, and the MHAM for using multiple sets of attention weights. Thus, as presented in this section, we developed a hybrid model called the 1D-CNN-BiLSTM-MHAM model to improve SSN predictive accuracy. Our model can be divided into two parts: deep feature extraction based on the CNN layer and time-series forecasting based on the BiLSTM and MHAH layers. Our model consists of six layers: the input layer, the CNN layer, the BiLSTM layer, the multi-head attention layer, the fully

connected (dense) layer, and the output layer. The structure of the model is presented in Figure 5. To predict longer-term values, a recursive multi-step forecasting technique was used to obtain values for the next 5 and 7 months, which correspond to the blue and red boxes, respectively.

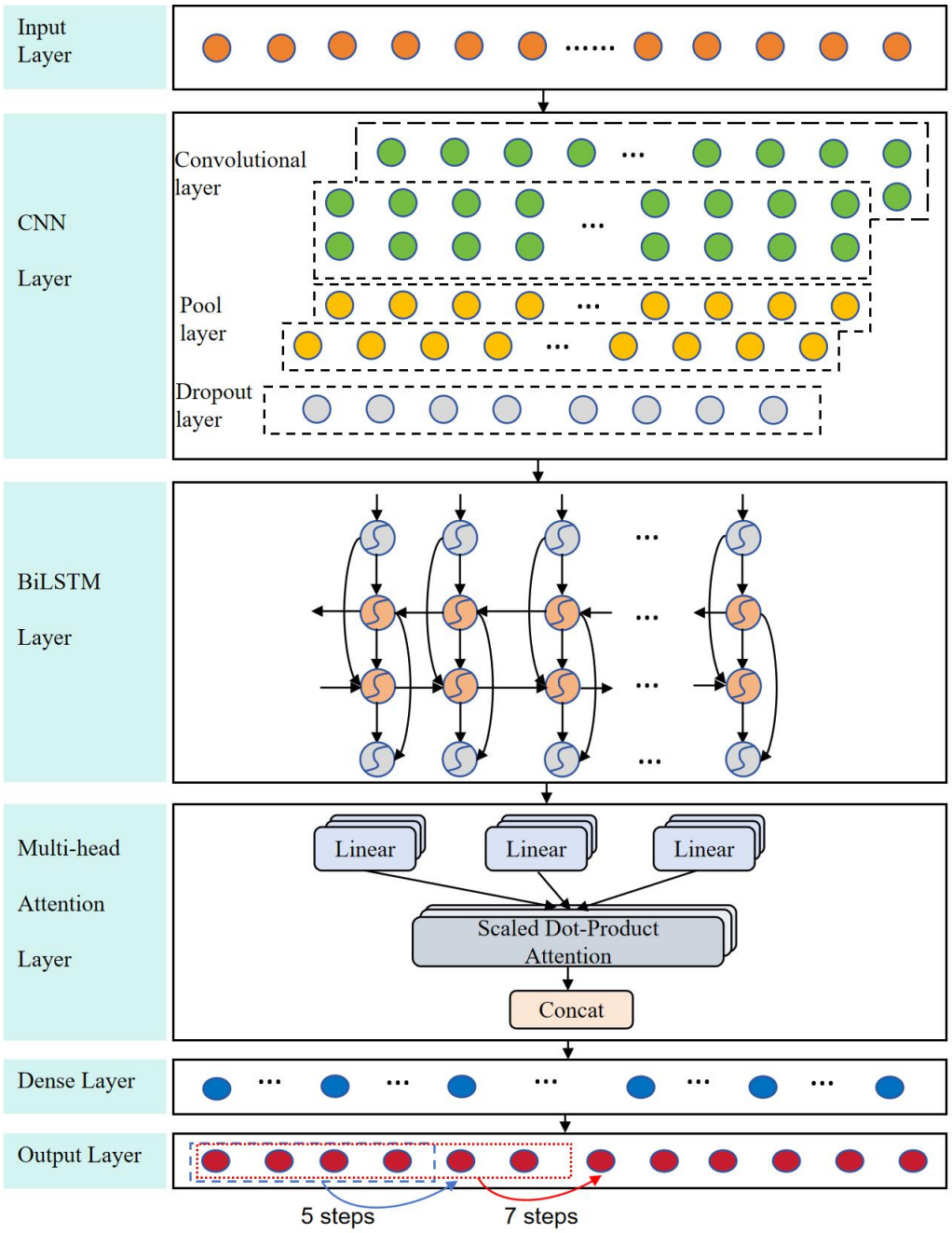

Figure 5. The structure of our algorithm.

2.6. Evaluation Indicators

To measure the predictive accuracy of our model, we calculated six evaluation indicators, namely, mean square error (*MSE*), root mean square error (*RMSE*), mean absolute error (*MAE*), mean absolute percentage error (*MAPE*), symmetric mean absolute percentage error (*SMAPE*), and correlation coefficient (*Corr*), based on the prediction results and the true values. The formulas of these indicators are listed in Equations (15)–(19). When these

indicators, except *Corr*, are small, the model has better predictive ability, and if *Corr* is close to 1, the model has better performance.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\widehat{y}_i - y_i)^2} \tag{15}$$

$$MAE = \frac{1}{n}|(\widehat{y}_i - y_i)| \tag{16}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}|\frac{\widehat{y}_i - y_i}{y_i}| \times 100\% \tag{17}$$

$$SMAPE = \frac{100\%}{n}\sum_{i=1}^{n}\frac{|\widehat{y}_i - y_i|}{(|\widehat{y}_i| + |y_i|)/2} \tag{18}$$

$$Corr = \frac{\sum_{i=1}^{n}(y_i - \overline{y})(\widehat{y}_i - \overline{\widehat{y}}_i)}{\sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2}\sqrt{\sum_{i}^{n}(\widehat{y}_i - \overline{\widehat{y}}_i))^2}}, \tag{19}$$

where y and $\widehat{y}$ are the true data and the prediction results, respectively. The mean values of y and $\widehat{y}$ are $\overline{y}$ and $\overline{\widehat{y}}$, respectively; n is the length of the data, and i is the index.

3. Data and Experimental Settings

3.1. Data Description and Preprocessing

We downloaded SSN data from the SIDC website (URL: "https://www.sidc.be/silso/home", accessed on 1 May 2022) in CSV format. The file contained 3279 samples, with many zero values before the year 1811. Therefore, we selected the time scope of the monthly mean total sunspot number starting from January 1811 and ending in March 2022. The statistical features, including the maximum and minimum values and the standard deviation (Std), are listed in Table 1. The table shows that the minimum was 0.2 and the maximum was 285. Thus, we normalized the data with Formula (20) before inputting them into our model.

$$x' = \frac{x - Minimum}{Maximum - Minimum} \tag{20}$$

Table 1. Statistical description of dataset.

Dataset	Time Period	Type	Size	Std	Minimum	Maximum
Sunspots	1811.1–2022.3	Monthly	2535	48.37	0.2	285

To facilitate network training, the data were split in chronological order into a training set (1632 records from January 1811 to December 1946), a validation set (408 records from January 1947 to December 1980), and a testing set (495 records from January 1981 to March 2022). The ratio of the training set to the testing set was 8:2. The training set was used to preprocess the model to determine the optimal hyperparameters, the validation set was used to monitor the hyperparameter tuning process, and the testing set was used to assess the performance of each trained prediction model.

3.2. The Parameter Settings of the Model

Because our method is a hybrid model, the parameters of different modules may influence its predictive accuracy. Therefore, parameter setting is also an important task. We set the different parameters, such as the number of epochs, the number of convolutional kernels, the number of neurons in BiLSTM, and the convolutional step size, as follows: first, we set different numbers of neurons, including 16, 32, 64, and 128, in the BiLSTM model and used it on its own to predict the data; then, we chose similar settings for the number of convolutional kernels to be combined with BiLSTM. The corresponding RMSE

and MAE values are listed in Table 2. Next, the convolution steps were assigned values of 1, 2, and 3; then, the RMSE and MAE values were computed. We found that the accuracy was higher with one convolution step. The detailed parameter settings of the different models are listed in Table 3.

Table 2. Predictive accuracy in terms of RMSE and MAE corresponding to numbers of BiLSTM neurons and convolutional kernels in CNN-BiLSTM.

Number	Neural Cell in BiLSTM		Convolutional Kernel in CNN-BiLSTM	
	RMSE	MAE	RMSE	MAE
16	3.12	2.72	2.1	1.65
32	3.75	1.54	3.75	2.83
64	3.38	1.35	1.64	1.19
128	2.57	1.21	4.01	3.21

Table 3. Detailed parameter settings of different models.

Parameter	CNN	BiLSTM	CNN-BiLSTM	CNN-BiLSTM-MHAM
LR	0.001	0.001	0.001	0.001
Epochs	350	350	350	350
Batch size	64	64	64	64
Drop ratio	-	-	0.2	0.2
Convolutional kernels	64	-	64	64
Step size	1	-	1	1
Number of neurons in BiLSTM	128	128	128	128
Optimizer	Adam	Adam	Adam	Adam

4. Simulation Results and Discussion

4.1. Analysis of Predictive Error

Figure 6 shows our prediction results (red line) and the original data (black line). We can see from the figure that the two curves are relatively close, but there are some obvious differences in some places, especially around large fluctuations. To observe the discrepancy between the prediction results ($\widehat{y}$) and the original data (y), the formula $y - \widehat{y}$ was used to calculate the predictive error (PE), as shown in Figure 7. Many error values were found to be near the centerline ($y = 0$), and the number of predictive errors greater than 0 was relatively large. If the PE is larger than 0, it means that the prediction result may be ahead of schedule. In contrast, there was a lag in the prediction results. We also found that the lag error was larger than the predictive error because the minimum and maximum of the amplitudes (y-axis) were close to -8 and 6, respectively. Figure 8 shows the statistical results of the PE. The PEs relative to the different scopes of $[-8, -1)$, $[-1, 0)$, $[0, 1]$, and $(1, 6]$ were 118, 76, 123, and 186, respectively. The number of non-negative PEs (309 samples) was greater than that of negative PEs (194 samples).

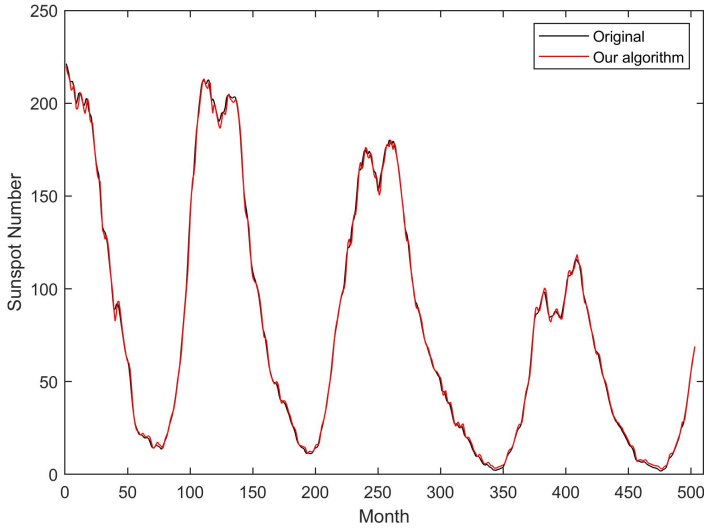

Figure 6. Comparison of original data (black line) and prediction results (red line).

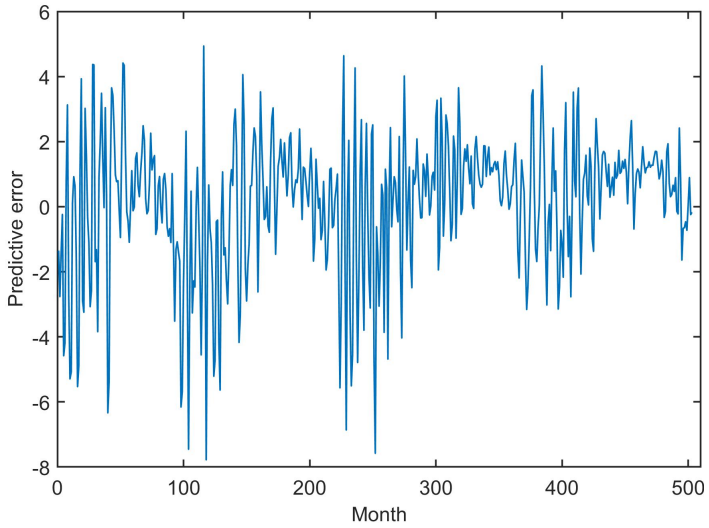

Figure 7. Errors between original data and prediction results.

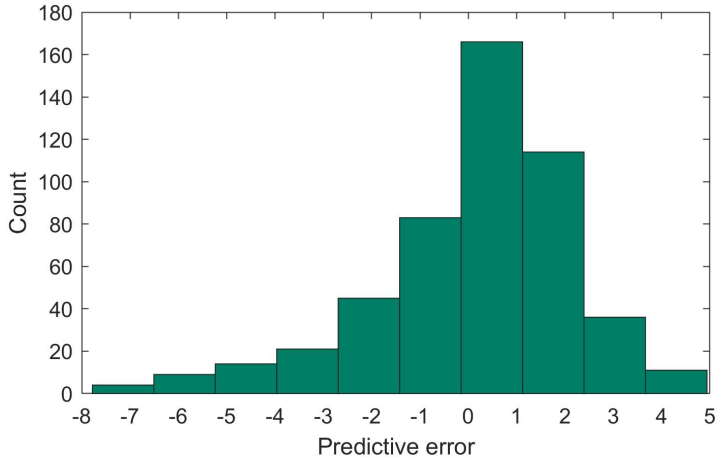

Figure 8. Histogram of errors between original data and prediction results.

4.2. Comparison and Analysis of Different Methods

To demonstrate that our algorithm has significant accuracy in predicting SSNs, ARIMA, CNN, BiLSTM, and CNN-BiLSTM were selected for comparison with our algorithm,

and their prediction results and the original data are shown in Figure 9. From the figure, we can see that the results of the single-method CNN (blue line) are quite distant from the original data (black line). The results of CNN-BiLSTM (green line) are close to the original data in the comparison methods. In general, in terms of distance from the true value, the models are ordered as CNN, BiLSTM, ARIMA, CNN-BiLSTM, and our algorithm from worst to best results.

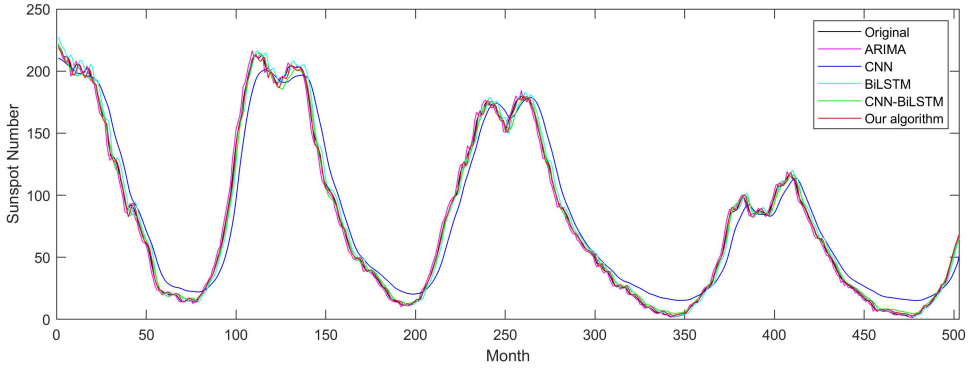

Figure 9. Our algorithm compared with ARIMA, CNN, BiLSTM, and CNN-BiLSTM.

Five indicators were selected to objectively evaluate these algorithms, and their results are listed in Table 4. For single-method algorithms, the *RMSE*, *MAE*, *MAPE*, and *SMAPE* were 3.93, 2.86, 12.13%, and 9.81 for ARIMA; 13.43, 11.05, 51.08%, and 28.84 for the CNN; and 5.35, 4.04, 8.93%, and 8.51 for BiLSTM, respectively. Compared with those of the CNN, the results of the ARIMA and BiLSTM models showed a large improvement. Furthermore, the *RMSE* and *MAE* of the combined CNN and BiLSTM model decreased by more than 0.89 and 0.44, respectively. However, the *SMAPE* values of BiLSTM and CNN-BiLSTM were very close, and the correlation coefficient increased by approximately 0.0021. For the first four indexes, our algorithm showed values that were lower by 1.07, 0.85, 4.85%, and 3.4, respectively. For the *Corr* index, our algorithm achieved a value of 0.9996, which was better than that of the others. Therefore, we can see that all the modules play a role in our algorithm.

Table 4. Performance comparison of ARIMA, CNN, BiLSTM, CNN-BiLSTM, and CNN-BiLSTM-MHAM.

Method	RMSE	MAE	MAPE	SMAPE	Corr
ARIMA	3.93	2.86	12.13%	9.81	0.9858
CNN	13.43	11.05	51.08%	28.84	0.9802
BiLSTM	5.35	4.04	8.93%	8.51	0.9970
CNN-BiLSTM	3.14	2.42	10.54%	8.59	0.9991
CNN-BiLSTM-MHAM (ours)	2.07	1.57	5.69%	5.11	0.9996

4.3. Comparison with Previous Studies

We employed methods from several previous studies for comparison with our algorithm to demonstrate the advantages of our method. Two types of algorithms were compared, including single-method algorithms and hybrid algorithms. The former included informer ([29]) and stacked LSTM with two stacked 50 LSTM units [47]. We further considered the hybrid algorithms XGBoost-DL [38] and EMD-LSTM-AM [35]. We downloaded the codes of informer and XGBoost-DL from GitHub (https://github.com/yd1008/ts_ensemble_sunspot, accessed on 5 December 2023) and used the original models' parameters, except when updating the dataset. Table 5 lists the results of the previous studies and our study. Informer produced a larger error, and LSTM was better than informer because of the stacking of multilayer LSTM cells. The hybrid XGBoost-DL algorithm was better

than the informer method, and EMD-LSTM-AM outperformed the other three algorithms, which was attributed to the fact that it adopts the EMD method to decompose the chaotic sunspot series and reduce their complexity. Because we used a CNN to extract the deep features of sunspots and combined the CNN and BiLSTM with the MHAM module, our method had better performance. The *RMSE* and *MAE* values of EMD-LSTM-AM were 3.6 and 4.5, respectively, and those of our algorithm were 2.07 and 1.57, respectively. Compared with these algorithms, our method achieved reductions in *RMSE* and *MAE* of more than 1.53 and 2.93, respectively.

Table 5. Comparison with other studies based on performance indicators.

Method	RMSE	MAE
ARIMA	3.93	2.86
Informer	29.1	22.56
Stacked LSTM	13.38	9.57
XGBoost-DL	24.5	18.34
EMD-LSTM-AM	3.6	4.5
CNN-BiLSTM-MHAM	2.07	1.57

4.4. Short-Term Forecasting

We predicted SSNs for 1, 5, and 7 months into the future to validate the network's generalization ability. Figure 10 shows the results of the 1-month (black line), 5-month (blue line), and 7-month (green line) predictions and the original data (black line). As the prediction time increased, the prediction results became increasingly different from the original values. We can note that these lines exhibit differences in some places, especially around fluctuations.

Table 6 lists the results of the three evaluation indicators. The RMSE and MAE values for the 1-month, 5-month, and 7-month predictions were 2.07 and 1.57, 2.89 and 2.15, and 3.39 and 2.63, respectively. We found that the two indices showed increases of 0.82 and 0.58 between the 1- and 5-month predictions and of 0.5 and 0.48 between the 5- and 7-month predictions, respectively. The Corr value decreased by 0.0015 between the 1- and 5-month predictions and by 0.0003 between the 5- and 7-month predictions. The table shows that the 1-month predictive accuracy is better than that for the other time windows.

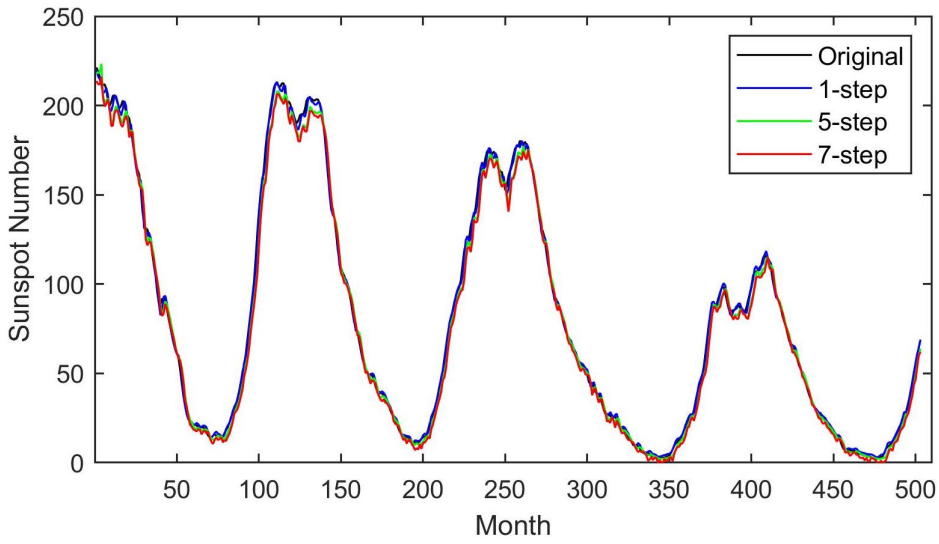

Figure 10. The comparison of the ground truth (original) and the multi-step prediction results of our algorithm.

Table 6. The prediction results of the multi-step prediction for 1, 5, and 7 months.

Multi-Step (Months)	RMSE	MAE	Corr
1	2.07	1.57	0.9996
5	2.89	2.15	0.9981
7	3.39	2.63	0.9978

5. Conclusions

Given its ultimate aim of predicting the sunspot number, our prediction model was considered from different perspectives:

(1) We experimented with different numbers of convolutional kernels for the 1D-CNN to extract features at different scales and with different numbers of neurons for BiLSTM. We found that the optimal parameters for 1D-CNN and BiLSTM are 64 kernels and 128 neurons, respectively.
(2) GRU is structurally simple compared with LSTM, and while they perform similarly on many tasks, when the dataset is large, LSTM has better performance. Compared with LSTM, BiLSTM can effectively utilize the forward and backward feature information of the dataset, thus improving the predictive accuracy in general.
(3) With a standard attention mechanism, we only compute a weighted context vector to represent the information of the sunspot number. With an MHAM, we use multiple sets of attention weights, where each set of weights can learn different types of semantic information and produce a context vector. Finally, these context vectors are concatenated, and a linear transformation is applied to produce the final output. Thus, an MHAM can learn the relationships among data and be used to extract more efficient information for improving the predictive accuracy.

Based on the above analysis, a hybrid model was developed. Common evaluation indicators were used to measure the performances of different models. In this study, our algorithm was compared with classical and recently published models. The experiments showed that the developed model is superior in terms of accuracy and precision for predicting sunspot numbers. Sunspot activity has a great impact on human activities, especially in the field of aerospace communication and daily information communication. It is crucial to construct a method that is accurate, fast, and lightweight for sunspot prediction. In further work, the structure of the model will continue to be optimized to make the model more lightweight and reduce computational resources. New mechanisms will be introduced to improve accuracy in sunspot prediction.

Author Contributions: Conceptualization, H.C.; methodology, H.C. and S.L.; software, X.Y. and X.Z.; validation, H.C., S.L. and X.Y.; formal analysis, H.C. and J.Y.; resources, H.C.; data curation, S.F.; writing—original draft preparation, H.C. and J.Y.; writing—review and editing, J.Y. and S.F.; visualization, X.Y.; supervision, J.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Arlt, R.; Vaquero, J.M. Historical sunspot records. *Living Rev. Sol. Phys.* **2020**, *17*, 1. [CrossRef]
2. Li, G.; Ma, X.; Yang, H. A Hybrid Model for Forecasting Sunspots Time Series Based on Variational Mode Decomposition and Backpropagation Neural Network Improved by Firefly Algorithm. *Comput. Intell. Neurosci.* **2018**, 3713410. [CrossRef]
3. Maitra, A.; Saha, U.; Adhikari, A. Solar control on the cloud liquid water content and integrated water vapor associated with monsoon rainfall over India. *J. Atmos. Sol.-Terr. Phys.* **2014**, *121*, 157–167. [CrossRef]
4. Bhowmik, P.; Nandy, D. Prediction of the strength and timing of sunspot cycle 25 reveal decadal-scale space environmental conditions. *Nat. Commun.* **2018**, *9*, 5209. [CrossRef] [PubMed]

5. Upton, L.A.; Hathaway, D.H. An updated solar cycle 25 prediction with AFT: The modern minimum. *Geophys. Res. Lett.* **2018**, *45*, 8091–8095. [CrossRef]
6. Abdel-Rahman, H.; Marzouk, B. Statistical method to predict the sunspots number. *NRIAG J. Astron. Geophys.* **2018**, *7*, 175–179. [CrossRef]
7. Tabassum, A.; Rabbani, M.; Omar, S.B. An Approach to Study on MA, ES, AR for Sunspot Number (SN) prediction and to forecast SN with seasonal variations along with trend component of time series analysis using Moving Average (MA) and Exponential Smoothing (ES). In *Advances in Electrical and Computer Technologies: Select Proceedings of ICAECT 2019*; Springer: Singapore, 2020; pp. 373–386. [CrossRef]
8. Song, H.; Tan, C.; Jing, J.; Wang, H.; Yurchyshyn, V.; Abramenko, V. Statistical assessment of photospheric magnetic features in imminent solar flare predictions. *Sol. Phys.* **2009**, *254*, 101–125. [CrossRef]
9. Volobuev, D.; Makarenko, N. Forecast of the decadal average sunspot number. *Sol. Phys.* **2008**, *249*, 121–133. [CrossRef]
10. Zaffar, A.; Abbas, S.; Ansari, M.R.K. Model estimation and prediction of sunspots cycles through AR-GARCH models. *Indian J. Phys.* **2022**, *96*, 1895–1903. [CrossRef]
11. Brajša, R.; Wöhl, H.; Hanslmeier, A.; Verbanac, G.; Ruždjak, D.; Cliver, E.; Svalgaard, L.; Roth, M. On solar cycle predictions and reconstructions. *Astron. Astrophys.* **2009**, *496*, 855–861. [CrossRef]
12. Borisova, D.; Kostadinova, G.; Petkov, G.; Dospatliev, L.; Ivanova, M.; Dermendzhieva, D.; Beev, G. Assessment of CH_4 and CO_2 Emissions from a Gas Collection System of a Regional Non-Hazardous Waste Landfill, Harmanli, Bulgaria, Using the Interrupted Time Series ARMA Model. *Atmosphere* **2023**, *14*, 1089. [CrossRef]
13. Box, G. Box and Jenkins: Time series analysis, forecasting and control. In *A Very British Affair: Six Britons and the Development of Time Series Analysis during the 20th Century*; Springer: London, UK, 2013; pp. 161–215. [CrossRef]
14. Jie, T. Application of the Grey Topological Theory in the Prediction of Yearly Mean Sunspot Numbers. *Chin. Astron. Astrophys.* **2015**, *39*, 45–53. [CrossRef]
15. Sabarinath, A.; Anilkumar, A. Sunspot cycle prediction using multivariate regression and binary mixture of Laplace distribution model. *J. Earth Syst. Sci.* **2018**, *127*, 84. [CrossRef]
16. Wheatland, M. A Bayesian approach to solar flare prediction. *Astrophys. J.* **2004**, *609*, 1134. [CrossRef]
17. Yu, Y.; van Dyk, D.A.; Kashyap, V.L.; Young, C.A. A Bayesian analysis of the correlations among sunspot cycles. *Sol. Phys.* **2012**, *281*, 847–862. [CrossRef]
18. Peng, L.; Yan, H.; Yang, Z. Prediction on sunspot activity based on fuzzy information granulation and support vector machine. *Aip Conf. Proc.* **2018**, *1955*, 040152. [CrossRef]
19. Abd, M.A.; Majed, S.F.; Zharkova, V. Automated classification of sunspot groups with support vector machines. In *Technological Developments in Networking, Education and Automation*; Springer: Dordrecht, The Netherlands, 2010; pp. 321–325. [CrossRef]
20. Siagian, R.C.; Alfaris, L.; Ahmad, G.N.; Laeiq, N.; Muhammad, A.C.; Nyuswantoro, U.I.; Nasution, B. Relationship between Solar Flux and Sunspot Activity Using Several Regression Models. *J. Ilmu Fis.* **2023**, *15*, 146–165. [CrossRef]
21. Chattopadhyay, G.; Chattopadhyay, S. Monthly sunspot number time series analysis and its modeling through autoregressive artificial neural network. *Eur. Phys. J. Plus* **2012**, *127*, 43. [CrossRef]
22. Samin, R.E.; Kasmani, R.M.; Khamis, A.; Isa, S. Forecasting sunspot numbers with recurrent neural networks (rnn) using'sunspot neural forecaster'system. In Proceedings of the 2010 Second International Conference on Advances in Computing, Control, and Telecommunication Technologies, Jakarta, Indonesia, 2–3 December 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 10–14. [CrossRef]
23. Prasad, A.; Roy, S.; Sarkar, A.; Chandra Panja, S.; Narayan Patra, S. Prediction of solar cycle 25 using deep learning based long short-term memory forecasting technique. *Adv. Space Res.* **2022**, *69*, 798–813. [CrossRef]
24. Zhu, H.; Chen, H.; Zhu, W.; He, M. Predicting Solar cycle 25 using an optimized long short-term memory model based on sunspot area data. *Adv. Space Res.* **2023**, *71*, 3521–3531. [CrossRef]
25. Kumar, A.; Kumar, V. Stacked 1D Convolutional LSTM (sConvLSTM1D) Model for Effective Prediction of Sunspot Time Series. *Sol. Phys.* **2023**, *298*, 121. [CrossRef]
26. Dai, S.; Liu, Y.; Meng, J. Sunspot forecast using Temporal Convolutional Neural (TCN) network based on phase space reconstruction. In Proceedings of the 2021 33rd Chinese Control and Decision Conference (CCDC), Kunming, China, 22–24 May 2021; pp. 2895–2900. [CrossRef]
27. Arfianti, U.I.; Novitasari, D.C.R.; Widodo, N.; Hafiyusholeh, M.; Utami, W.D. Sunspot Number Prediction Using Gated Recurrent Unit (GRU) Algorithm. *Indones. J. Comput. Cybern. Syst.* **2021**, *15*, 141–152. [CrossRef]
28. Abduallah, Y.; Wang, J.T.; Wang, H.; Xu, Y. Operational prediction of solar flares using a transformer-based framework. *Sci. Rep.* **2023**, *13*, 13665. [CrossRef] [PubMed]
29. Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; Zhang, W. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 2–9 February 2021; pp. 11106–11115. [CrossRef]
30. Hu, A.; Rodriguez, A. A Hybrid CNN-LSTM Model for Predicting Solar Cycle 25. *J. Stud. Res.* **2023**, *12*, 121. [CrossRef]
31. Yang, J.; Liu, S.; Xuan, S.; Chen, H. A Hybrid Model Based on CEEMDAN-GRU and Error Compensation for Predicting Sunspot Numbers. *Electronics* **2024**, *13*, 1904. [CrossRef]
32. Pontoh, R.S.; Toharudin, T.; Ruchjana, B.N.; Gumelar, F.; Putri, F.A.; Agisya, M.N.; Caraka, R.E. Jakarta pandemic to endemic transition: Forecasting COVID-19 using NNAR and LSTM. *Appl. Sci.* **2022**, *12*, 5771. [CrossRef]

33. Li, Q.; Wan, M.; Zeng, S.G.; Zheng, S.; Deng, L.H. Predicting the 25th solar cycle using deep learning methods based on sunspot area data. *Res. Astron. Astrophys.* **2021**, *21*, 184. [CrossRef]
34. Lee, T. EMD and LSTM hybrid deep learning model for predicting sunspot number time series with a cyclic pattern. *Sol. Phys.* **2020**, *295*, 82. [CrossRef]
35. Yang, J.; Fu, N.; Chen, H. The Sunspot Number Forecasting Using a Hybridization Model of EMD, LSTM and Attention Mechanism. *IEEJ Trans. Electr. Electron. Eng.* **2023**, *18*, 1791–1798. [CrossRef]
36. Kumar, B.; Sunil; Yadav, N. A novel hybrid model combining beta+SARMA and LSTM for time series forecasting. *Appl. Soft Comput.* **2023**, *134*, 110019. [CrossRef]
37. Kumar, A.; Kumar, V. Forecast of solar cycle 25 based on Hybrid CNN-Bidirectional-GRU (CNN-BiGRU) model and Novel Gradient Residual Correction (GRC) technique. *Adv. Space Res.* **2024**, 73, 4342–4362. [CrossRef]
38. Dang, Y.; Chen, Z.; Li, H.; Shu, H. A Comparative Study of non-deep Learning, Deep Learning, and Ensemble Learning Methods for Sunspot Number Prediction. *Appl. Artif. Intell.* **2022**, *36*, 2074129. [CrossRef]
39. Sirisha, U.M.; Belavagi, M.C.; Attigeri, G. Profit Prediction Using ARIMA, SARIMA and LSTM Models in Time Series Forecasting: A Comparison. *IEEE Access* **2022**, *10*, 124715–124727. [CrossRef]
40. Li, J.; Wang, X.; Tu, Z.; Lyu, M.R. On the diversity of multi-head attention. *Neurocomputing* **2021**, *454*, 14–24. [CrossRef]
41. Zhao, J.; Mao, X.; Chen, L. Speech emotion recognition using deep 1D & 2D CNN LSTM networks. *Biomed. Signal Process. Control* **2019**, *47*, 312–323. [CrossRef]
42. Smagulova, K.; James, A.P. A survey on LSTM memristive neural network architectures and applications. *Eur. Phys. J. Spec. Top.* **2019**, *228*, 2313–2324. [CrossRef]
43. Mouakher, A.; Inoubli, W.; Ounoughi, C.; Ko, A. Expect: EXplainable prediction model for energy ConsumpTion. *Mathematics* **2022**, *10*, 248. [CrossRef]
44. Schuster, M.; Paliwal, K.K. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* **1997**, *45*, 2673–2681. [CrossRef]
45. Ma, C.; Dai, G.; Zhou, J. Short-term traffic flow prediction for urban road sections based on time series analysis and LSTM_BILSTM method. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 5615–5624. [CrossRef]
46. Reza, S.; Ferreira, M.C.; Machado, J.J.M.; Tavares, J.M.R. A multi-head attention-based transformer model for traffic flow forecasting with a comparative analysis to recurrent neural networks. *Expert Syst. Appl.* **2022**, *202*, 117275. [CrossRef]
47. Moustafa, S.S.R.; Khodairy, S.S. Comparison of different predictive models and their effectiveness in sunspot number prediction. *Phys. Scr.* **2023**, *98*, 045022. [CrossRef]