



Stacked 1D Convolutional LSTM (sConvLSTM1D) Model for Effective Prediction of Sunspot Time Series

Abhijeet Kumar¹ · Vipin Kumar¹

Received: 9 July 2023 / Accepted: 16 September 2023 / Published online: 17 October 2023
© The Author(s), under exclusive licence to Springer Nature B.V. 2023

Abstract

A multi-layer, deep-learning (DL) architecture consisting of stacked Convolutional Long Short Term Memory (sConvLSTM1D) layers is proposed to forecast the sunspot number (SSN) more effectively. The proposed model with optimized hyper-parameters performs efficiently on four kinds of sunspot data with different frequencies of time that are yearly, monthly, daily, and 13-month smoothed provided by the World Data Center-Sunspot Index and Long Term Solar Observation (WDC-SILSO), the Royal Observatory of Belgium (SILSO World Data Center). The model was contrasted with other traditional DL models on different performance metrics, namely root-mean-square error (RMSE), mean-absolute error (MAE), mean-absolute-percentage error (MAPE), and mean-absolute-scaled error (MASE). A non-parametric statistical test has also been carried out to confirm the model's effectiveness. The prediction of the highest yearly mean of total sunspot number (SSN) in Solar Cycle 25 (SC25) has also been performed. The proposed sConvLSTM1D model suggests that the solar cycle exhibits the characteristics of a weak cycle. However, it is anticipated to be stronger than the preceding Solar Cycle 24 (SC24). The year of peak sunspot number will be 2024, as per the prediction, with the peak value of yearly mean sunspot number as 140.84, which is 24.3% higher than the peak value of the yearly mean of total sunspot number, which was 113.3 in the Solar Cycle 24 in the year 2014.

Keywords Sunspot number · Solar cycle · Prediction · Time series · Deep learning · Convolutional · LSTM · Hybrid model · Stack model

1. Introduction

Sunspots are the observable dark areas that emerge on the surface of the Sun due to relatively lower temperatures in that specific region than the other regions. Since 1610, astronomers have observed the solar surface with the help of telescopes and recorded the occurrence and appearance of sunspots (Vokhmyanin, Arlt, and Zolotova, 2020). As per the study of

✉ V. Kumar
rt.vipink@gmail.com

A. Kumar
abhijeetk235@gmail.com

¹ Department of Computer Science and Information Technology, Mahatma Gandhi Central University, Motihari, Bihar, India

Marques, Leal-Júnior, and Kumar (2023), variances caused by sunspot activity in space are a significant factor in determining the architecture and design of spacecraft. The radiation hazard that is observed in space has a significant source, which is known as cosmic rays. The flux of these cosmic rays is anti-correlated with the solar cycle (Pesnell, 2008). Therefore, accurate prediction of sunspot numbers along with their peaks and resulting solar cycle is quite essential to perform various developments in astronomy. In this study, the performance of some traditional deep-learning (DL) models has been analyzed and compared with the proposed model with the primary objective of reducing the error in the prediction of sunspot numbers. As per the study of Benson et al. (2020), sunspot-activity forecasts for the upcoming solar cycle have been classified as a multi-step univariate time-series technique. Therefore, the concept of time series has been utilized along with DL technologies to design a model that can foretell the number of SSNs appearing on the surface of the Sun with less error.

Chattopadhyay, Jhajharia, and Chattopadhyay (2011) explained that the prediction based on periodicity present in time-series data of sunspot numbers is known as the numeric method, while the prediction based on geophysical parameters and methodologies related to it is known as the precursor method. In most of the studies, the data of the sunspot number has been taken from the SILSO centre of SIDC, Belgium since it has the data for an extended period, which is helpful in prediction using deep-learning approaches. Büyükşahin and Ertekin (2019) proposed a hybrid model of ANN and ARIMA and performed a comparative analysis of their suggested model's efficacy with the individual model's performances. In contrast, Pala and Atici (2019) introduced deep-learning techniques as a solution to this problem, forecasted the total monthly mean of sunspot numbers using the LSTM method, and showed that deep-learning models outperform the statistical models. The difference in the previously stated works is that the former utilized the monthly average of sunspot numbers while the latter analyzed the results on the annual data. Apart from this, Büyükşahin and Ertekin (2019) also used Empirical Mode Decomposition to fragment data into Intrinsic Mode Functions (IMFs) and then executed their proposed models. Pala and Atici (2019) utilized two stacking layers of LSTM for their study, while Elgamal (2020) utilized deep and stacked LSTM to predict SC25.

Compared to the above works, Lee (2020) proposed a novel model with a hybrid of LSTM and EMD suggesting it as a better alternative for modeling the sunspot time series because of its cyclic nature. In this study, ensemble EMD has been utilized to obtain the IMFs. Unlike the above-mentioned strategies, Panigrahi et al. (2021) introduced a combination of machine-learning algorithms and statistical methods. In this research, SVM has been utilized alongside ARIMA and Exponential Smoothing with error, trend, and seasonality combined, known as ETS, where seasonality means repeating patterns over a fixed time interval. In contrast to this work regarding methodology, Arfianti et al. (2021) utilized deep-learning techniques namely GRU and LSTM individually for forecasting and summarized that GRU outperformed LSTM in their experiments. Similarly, Prasad et al. (2022) experimented with stacked LSTM to forecast the Solar Cycle 25 based on 13-month smoothed observations of sunspot numbers and compared their work with the proposed model of Wang 2021. They concluded that the performance improved significantly.

Apart from the studies mentioned above, Hasoon and Al-Hashimi (2022) proposed three models, namely RNN, DNN, and a hybrid of DNN and LSTM, and concluded that the hybrid model has higher detection performance than the individual models. From a comparison point of view, an elaborate study was carried out by Dang et al. (2022) where methodologies not associated with deep learning, namely Prophet, Exponential Smoothing, and SARIMA, were compared with approaches associated with deep learning, namely Transformer, GRU,

Informer, and LSTM where the Informer model outperformed all other models. Then, an ensemble of all models was also built based on mean, median, error, regression, and XGBoost, and it was deduced that an ensemble of deep-learning algorithms based on XGBoost is better than all other experimented combinations.

Another study by Ramadevi and Bingi (2022) suggested that a Nonlinear Autoregressive Network (NAR), a neural network, performs better on 13-month smoothed observations of sunspot numbers. The NAR neural network is a neural network of feed-forward orientation with input and output layers along hidden layers. Specific activation functions “purelin” and “tansig” have been utilized to achieve optimal results. Nghiem et al. (2022) proposed a model that is a hybrid of CNN and LSTM along with Bayesian optimization and compared its performance with six state-of-the-art models, including four approaches of the DL methodology that are Informer, GRU, Transformer, and LSTM along with two models not associated with deep learning namely Prophet and SARIMA, and summarized that the CNN-Bayes LSTM model performed more effectively than these six models.

Kumar, Sunil, and Yadav (2023) also proposed a hybrid model, combining a statistical model with a deep-learning model. In their research, the β SARMA and LSTM models were combined to predict yearly sunspot numbers, and the performance was compared with other traditional models such as ARIMA, LSTM, and MLP along with an ensemble of ARIMA and ANN, and the proposed model was found to be good in most of the cases. In most studies, the monthly mean of total SSN has been utilized. Some of the researchers such as Chattopadhyay, Jhajharia, and Chattopadhyay (2011), Büyüksahin and Ertekin (2019), Elgama (2020), and Panigrahi et al. (2021) utilized the annual average of total sunspot numbers as well. A few researchers also performed forecasts over 13-month smoothed observations of sunspot numbers such as Prasad et al. (2022) and Ramadevi and Bingi (2022).

While comparing the performance measures, it was found that Elgama (2020) used RMSE and MAPE along with RMSE, while Panigrahi et al. (2021) used MASE and MAE along with RMSE. Hasoon and Al-Hashimi (2022) used MSE and MAE to measure the effectiveness of their proposed deep-learning models. Dang et al. (2022) used RMSE and MAE measures to compare the effectiveness of deep-learning techniques, methodologies other than deep-learning techniques and their ensemble models. From the above, most researchers have utilized RMSE as a standard metric to measure the effectiveness of the proposed models.

From the study of the previous works, it is clear that the effective prediction of SSNs is essential, and researchers have made efforts to improve the efficacy of the forecasting model. DL models perform well in the prediction of SSN, but errors still exist in the prediction of SSN, so there is still scope for improvement for the DL models. In this study, a novel stacked model based on ConvLSTM1D is proposed to improve the prediction of SSN, which is validated on different performance measures and datasets. In comparison to earlier studies where researchers utilized a maximum of two sunspot datasets with two different frequencies of time, we have utilized four kinds of sunspot data with four different frequencies of time that are daily, yearly, monthly, and 13-month smoothed for better evaluation of the proposed prediction model. This study evaluates the proposed model based on RMSE, MASE, MAE, and MAPE for better comparison. The proposed model is compared with traditional deep-learning models based on non-parametric statistical tests to validate its efficacy. In contrast to the earlier studies where researchers utilized a statistical approach, machine-learning models, vanilla LSTM, GRU, hybrid models, ensemble DL models, or stacked LSTM model, we have utilized a novel stacked ConvLSTM1D DL model for the predictions, which resulted in effective predictions over all four variants of the data. Finally, the proposed model is utilized for a more precise prediction of SC25, along with its peak

and trough. The rest of the article contains the Methodology in Section 2, Data Analysis and Experimental Setup in Section 3, Results and Analysis in Section 4, and Conclusions in Section 5.

2. Methodology

This study includes basic DL approaches, namely LSTM, CNN, GRU, RNN, and BiLSTM, for the comparison purpose, while the proposed model is a two-layer stacked architecture of ConvLSTM1D with a repeat-vector layer embedded in between the ConvLSTM1D layers followed by dropout and fully connected layers.

2.1. Mathematical Background

This section consists of fundamental mathematical aspects of the dataset and different basic models.

2.1.1. Mathematical Principles Related to the Dataset

The time-series univariate dataset is denoted as “D” consisting of finite data points d_1, d_2, d_3 , and so on, with time as the variable. X_{Data} and Y_{Data} are the lags as training features and target features, respectively. X_{Data} is trifurcated into their corresponding set of X_{train} , $X_{\text{validation}}$, and X_{test} . Y_{Data} is trifurcated into the set of Y_{train} , $Y_{\text{validation}}$, and Y_{test} . Furthermore, the *Input Sequence* is a combination of both X_{train} and Y_{train} for the training of the models, which is utilized to feed the DL models.

2.1.2. Mathematical Principles Related to LSTM

LSTM was initially proposed by Hochreiter and Schmidhuber (1997) in the field of neuro-computing. LSTM is a DL model specifically linked with RNN, which can address the vanishing or exploding gradient issue. LSTM can divulge the temporal dynamic behavior associated with the time series (Bai et al., 2019). The memory cell has three gates: forget, input, and output gate. Of these three gates, the forget gate is crucial since it is the decision-making gate that decides whether information from the preceding time step needs to be carried forward or forgotten. The other two gates are the input and output gate, which regulate the input’s activation flow in the memory unit’s direction and the information stream from the memory unit to the output. The LSTM architecture is depicted in Figure 1. LSTM models are designed to work with sequential data consisting of one-dimensional vectors over time and can forecast the future sequence (Zhang et al., 2019):

$$i_t = \sigma(\mathbf{W}_i \cdot [h_{t-1}, x_t] + b_i), \quad (1)$$

$$f_t = \sigma(\mathbf{W}_f \cdot [h_{t-1}, x_t] + b_f), \quad (2)$$

$$o_t = \sigma(\mathbf{W}_o \cdot [h_{t-1}, x_t] + b_o). \quad (3)$$

Equation 1, Equation 2, and Equation 3 represent the equations for input, forget, and output gate, respectively, where x_t denotes the input at the current time step t and \mathbf{W}_f , \mathbf{W}_o , and \mathbf{W}_i are the weight matrices for forget, output, and input gates. Apart from these, b_f ,

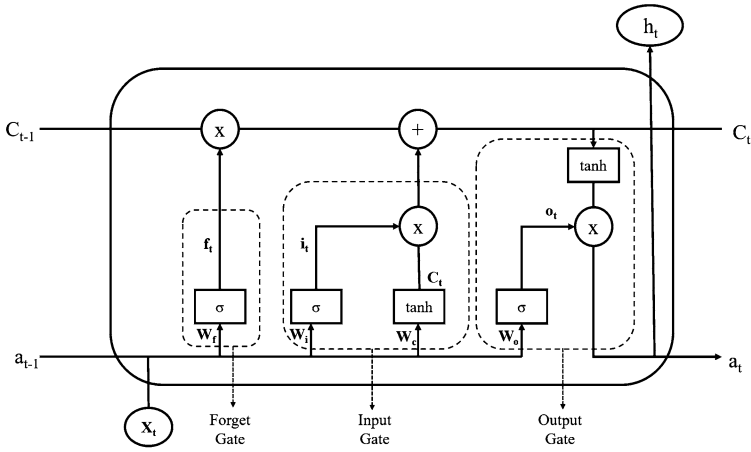
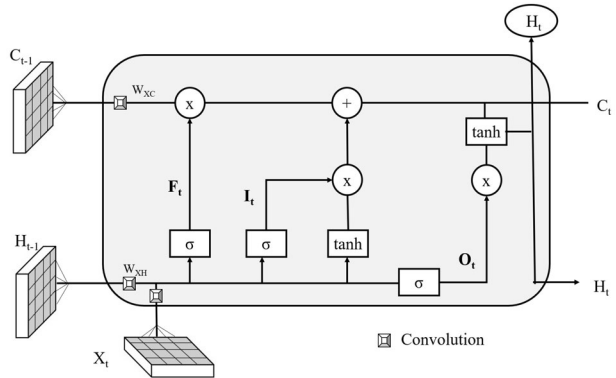


Figure 1 Schematic architecture of LSTM cell.

Figure 2 Schematic architecture of Convolutional LSTM cell.



b_o , and b_i are the bias terms for the respective gates. σ and \tanh represent sigmoid and hyperbolic tangent activation functions, respectively:

$$c_t = f_t c_{t-1} + i_t g_t, \tag{4}$$

$$g_t = \tanh(W_g [h_{t-1}, x_t] + b_g), \tag{5}$$

$$h_t = o_t \tanh(c_t). \tag{6}$$

Equation 4, Equation 5, and Equation 6 represent the equations for memory cell, candidate memory, and hidden state, respectively, where h_{t-1} and c_{t-1} denote hidden state and memory state at the previous time step ($t - 1$).

2.1.3. Mathematical Principles Related to ConvLSTM1D

The ConvLSTM1D arrangement is an adaptation of LSTM. Changes are made to the architecture of LSTM to produce the ConvLSTM1D, which is presented in Figure 2. Based on the

equations of LSTM, the main equations for ConvLSTM1D are expressed in the following manner:

$$i_t = \sigma(\text{ConvID}(\mathbf{W}_i, \mathcal{X}) + \text{ConvID}(\mathbf{U}_i, \mathbf{H}) + b_i). \quad (7)$$

Equation 7 represents the input gate and \mathbf{W}_i and b_i , the learnable parameters indicating the weight and bias associated with the input gate applied to input tensor \mathcal{X} at the current time step, whereas \mathbf{U}_i represents the weight matrix associated with the input gate applied to the hidden state \mathbf{H} from the preceding time step.

$$f_t = \sigma(\text{ConvID}(\mathbf{W}_f, \mathcal{X}) + \text{ConvID}(\mathbf{U}_f, \mathbf{H}) + b_f). \quad (8)$$

Equation 8 represents the forget gate where σ represents the sigmoid activation function operating over the learnable parameters \mathbf{W}_f , b_f , and \mathbf{U}_f associated with the forget gate.

$$o_t = \sigma(\text{ConvID}(\mathbf{W}_o, \mathcal{X}) + \text{ConvID}(\mathbf{U}_o, \mathbf{H}) + b_o). \quad (9)$$

Similarly, Equation 9 represents the output state o_t as a sigmoid function with learnable parameters \mathbf{W}_o , b_o , and \mathbf{U}_o associated with the output state.

$$\tilde{\mathbf{C}} = \tanh(\text{ConvID}(\mathbf{W}_c, \mathcal{X}) + \text{ConvID}(\mathbf{U}_c, \mathbf{H}) + b_c). \quad (10)$$

Equation 10 represents the candidate cell state, which is the information that can be potentially added to the cell state in the current time step where the learnable parameters \mathbf{W}_c , b_c , and \mathbf{U}_c are associated with the candidate state. In this equation, \tanh , represents the hyperbolic tangent activation function, which squashes the value between -1 and $+1$.

$$\mathbf{C} = f_t \mathbf{C} + i_t \tilde{\mathbf{C}}, \quad (11)$$

$$\mathbf{H} = o_t \tanh(\mathbf{C}). \quad (12)$$

Equations 11 and 12 represent the revised state of the cell stated denoted as \mathbf{C} and the hidden state \mathbf{H} , respectively.

The proposed model in this study utilizes the ConvLSTM1D layer similar to the models mentioned in the study of Shi et al. (2015), Cantillo-Luna et al. (2023), and Shi et al. (2022) but it is different from their models as it is using two layers of ConvLSTM1D stacked over each other with a layer of RepeatVector embedded between them. Apart from these, the ConvLSTM1D layer utilized in our study uses a “swish” activation function with 32 units in the first layer and 16 units in the second layer. A dropout layer has been attached at the end, followed by a dense layer.

2.2. Proposed Model Framework

The proposed model takes the sequential input at the first layer. Then, there is a ConvLSTM1D layer with 32 units, a “swish” activation function and a kernel size equal to 12 followed by a RepeatVector layer with repetition factor two, which is placed after flattening the output tensor obtained from the previous ConvLSTM1D layer. After the RepeatVector layer, another ConvLSTM1D layer is placed with 16 units, a “swish” activation function, and a kernel size equal to 12. The “swish” activation function has been utilized for the design of the model since it has been observed that for DL models a “swish” activation function outperforms “relu” and other similar activation functions (Ramachandran, Zoph,

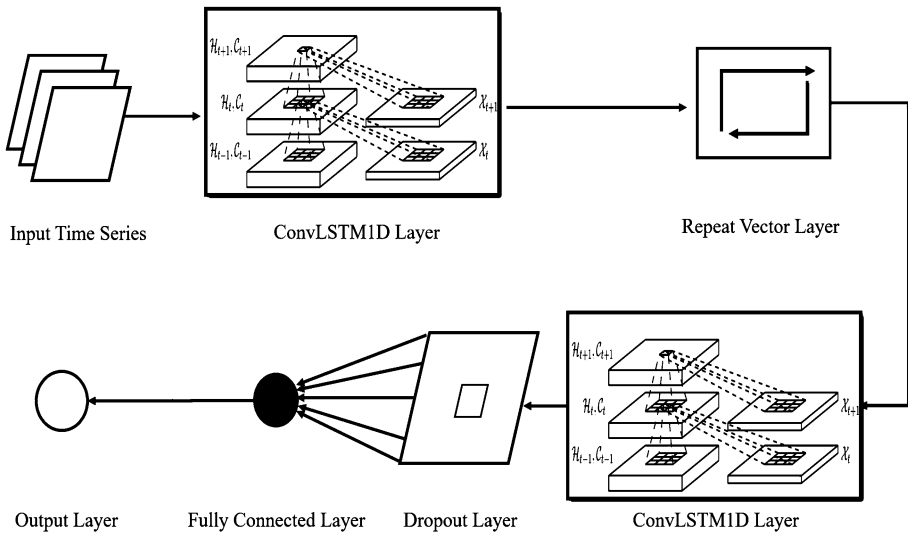


Figure 3 Schematic Diagram of the Proposed Model.

and Le, 2017; Szandała, 2021). The output tensor received from this layer is again flattened and passed through a dropout layer with a 10% dropout rate, and ultimately, a fully connected layer is positioned at the final stage. The model is used with a RMSProp optimizer because, based on empirical evidence, RMSProp has demonstrated effectiveness and practicality as an optimization technique for DNN (Goodfellow, Bengio, and Courville, 2016). The illustration of the proposed model is presented in Figure 3. The proposed model framework depicts the flow of the experiment. The steps of the model framework are explained in Algorithm 1, and the corresponding steps are illustrated in Figure 4. Input data are taken as univariate data concerning time. As a part of data pre-processing, missing-value imputation is performed. Then, data standardization is performed with the help of Equation 13 where μ represents the average and σ the standard deviation of the dataset

$$d_{\text{new}} = (d_i - \mu) / \sigma, \forall d_i \in \mathbf{D}. \tag{13}$$

After this, time-series data are generated using the given dataset and a fixed value of lookback is obtained after trial and error for an optimized value of the lag. Then, the dataset is bifurcated into training and test sets, along with further partitioning of the training dataset into training and validation sets.

The proposed framework has been utilized for analyzing the effectiveness of the proposed sConvLSTM1D model with the optimized hyper-parameters. Hyper-parameter optimization for the proposed sConvLSTM1D model has been carried out using trial and error. The detailed layout of the proposed framework is presented in Figure 4.

2.3. Model-Performance Measures

This study utilizes five performance metrics: RMSE, MAE, MAPE, R^2 , and MASE. The selection of performance measures is based on a literature survey. The most frequently

Algorithm 1 The Proposed sConvLSTM1D Model Framework.

- 1: Input the SSN data: $[d1, d2, d3...] = \mathbf{D}$
- 2: Perform missing-value imputation: $\forall d_i$ in \mathbf{D} , where d_i is -1 Perform data standardization on the dataset \mathbf{D} Perform time-series data generation with a lag value of 11 to obtain X_{Data} and Y_{Data}
- 3: Divide the dataset X_{Data} into training dataset X_{train} , validation dataset $X_{\text{validation}}$, and test dataset X_{test} in the ratio of 81:9:10
- 4: Divide the dataset Y_{Data} into training dataset Y_{train} , validation dataset $Y_{\text{validation}}$, and test dataset Y_{test} in the ratio of 81:9:10
- 5: $Patience = 20$
- 6: $Epochs = 250$
- 7: For training the model, $InputSequence \leftarrow [X_{\text{train}}, Y_{\text{train}}]$, and follow sub-algorithm at Step 10
- 8: Training Algorithm
 - 1: **while** EarlyStopping **do**
 - 2: $InputLayer \leftarrow InputSequence$
 - 3: $ConvLSTM1DLayer \leftarrow Output(InputLayer)$
 - 4: $RepeatVectorLayer \leftarrow Output(ConvLSTM1DLayer)$
 - 5: $ConvLSTM1DLayer \leftarrow Output(RepeatVectorLayer)$
 - 6: $DropoutLayer \leftarrow Output(ConvLSTM1DLayer)$
 - 7: $DenseLayer \leftarrow Output(DropoutLayer)$
 - 8: Calculate the $Validation_{\text{Loss}}$ and $Training_{\text{Loss}}$
 - 9: Check for Early Stopping Criteria
 - 10: **end while**
 - 9: **if** $Validation_{\text{Loss}} \sim Training_{\text{Loss}} \sim \text{Low \& Constant}$ **then**
 - 10: Predict \hat{y} for the X_{test}
 - 11: **else**
 - 12: Optimize Hyper-parameter for the proposed model
 - 13: Goto Step 7
 - 14: **end if**
 - 15: Calculate performance of the proposed sConvLSTM1D model based on different performance metrics

used performance metric has been taken for better comparison and evaluation of the performance and effectiveness of the proposed model. These performance metrics have been compared for the traditional models, state-of-the-art models, and the proposed sConvLSTM1D model.

These measures depict the deviation of the predicted result from the actual values from different aspects. Before utilizing these performance metrics, the concept of residual-error is of utmost importance, which is the difference between y and \hat{y} , where y is the actual value of SSN and \hat{y} is the predicted value of SSN in the context of these experiments. The equations related to the RMSE are represented in Equation 14, where m is the number of samples along with y_j and \hat{y}_j are the j th actual and predicted values, respectively,

$$RMSE = \sqrt{\frac{1}{m} \sum_{j=1}^m (y_j - \hat{y}_j)^2}. \quad (14)$$

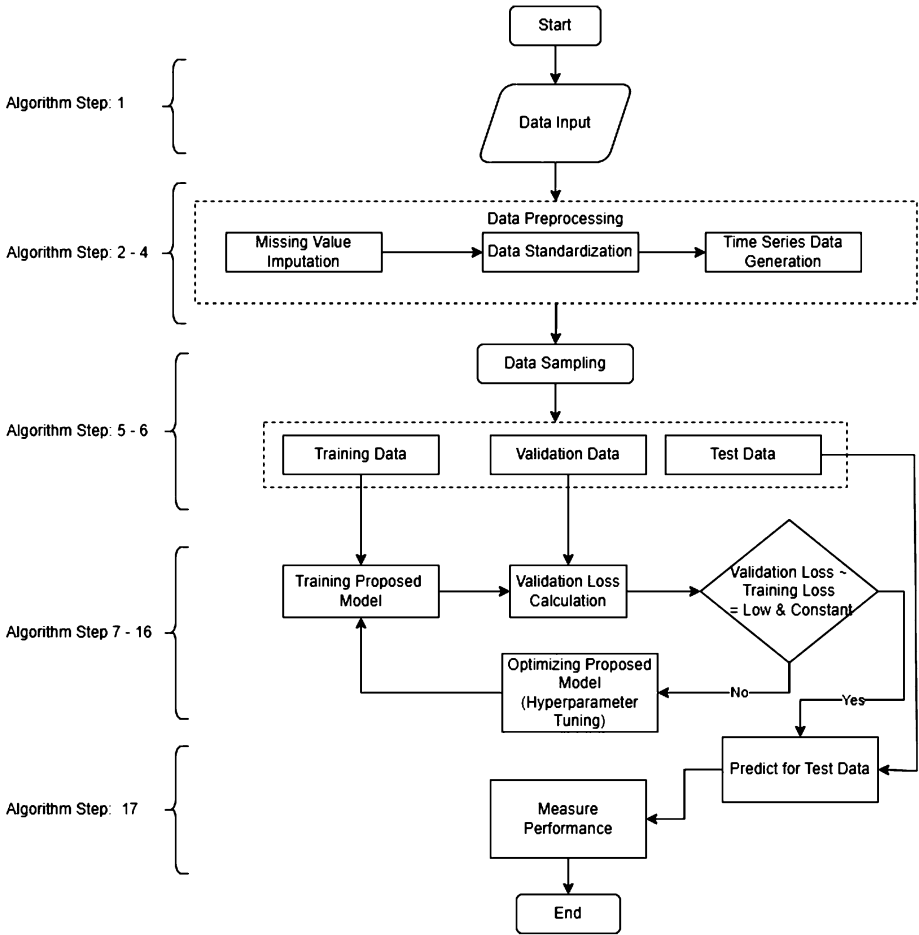


Figure 4 Proposed sConvLSTM1D model working framework for sunspot-number time-series prediction.

Equation 15 refers to the formula of Mean Absolute error, which is the simplest measure for performance evaluation. In this equation, “D” represents the complete dataset,

$$MAE = \sum_{j=1}^D |(y_j - \hat{y}_j)|. \tag{15}$$

Equation 16 defines the mean absolute scaled error, where e_i signifies residual error, also known as forecast error, n is the seasonal period, and T corresponds to the number of data points contained within the time-series data and the summation part in the denominator is the mean absolute error,

$$MASE = mean \left(\frac{|e_i|}{\frac{1}{T-n} \sum_{t=n+1}^T |Y_t - Y_{t-n}|} \right). \tag{16}$$

Table 1 Descriptions of four variants of sunspot data (SILSO World Data Center).

Name	Time Span	Frequency	Total Features	Feature for Analysis	Total Samples
Daily Total SSN	01 Jan 1818 – 31 Jan 2023	Daily	8	Daily SSN	74,906
Monthly Mean Total SSN	Jan 1749 – Jan 2023	Monthly	7	Monthly Mean Total SSN	3289
13-month Smoothed Monthly Total SSN	Jan 1749 – Jan 2023	Monthly	7	Monthly Smoothed Total SSN	3289
Yearly Mean Total SSN	1700 – 2022	Yearly	5	Yearly Mean SSN	323

Table 2 Statistical data description of four variants of sunspot data.

Name of Data	Standard Deviation	Average	Min	25%	50%	75%	Max	Trend	Seasonality
Daily Total SSN	77.2	78.6	−1	15	58	123	528	4015	365
Monthly Mean Total SSN	67.7	81.6	0	24	67.2	122.4	398.2	132	12
13-month Smoothed Monthly Total SSN	63.2	81.2	−1	25	71.3	118.8	285	132	12
Yearly Mean Total SSN	61.9	78.4	0	24.5	65.3	115	269.3	11	1

MAPE is a data-independent performance measure used to calculate the fractional error. The related formula is presented in Equation 17, where ϵ is a small positive arbitrary number chosen to prevent undefined outcomes in cases where y_j equals zero and other symbols have their meanings as mentioned above,

$$MAPE = \frac{1}{m} \sum_{j=1}^{m-1} \frac{|\hat{y}_j - y_j|}{\max(\epsilon, y_j)}. \quad (17)$$

3. Data Analysis and Experimental Setup

3.1. Data

Data have been obtained from the SIDC, Royal Observatory, Belgium (SILSO World Data Center) website in four variants described in Table 1. The “Total Features” characteristic corresponds to the total number of features present in the raw data, which includes other related descriptions along with actual observations of sunspot numbers such as fractional year, standard deviation, definitive indicator, number of observations utilized to compute the value, etc., whereas “Feature for Analysis” represents the actual number of SSN utilized for the processing. Further, statistical descriptions of all four variants of the sunspot data are explained in Table 2.

The additive-seasonal decomposition for all four dataset variants is depicted in Figure 5, where the original data are split into the three components: trend, seasonality, and noise. Although the decomposition has been performed on the complete dataset for all four variants,

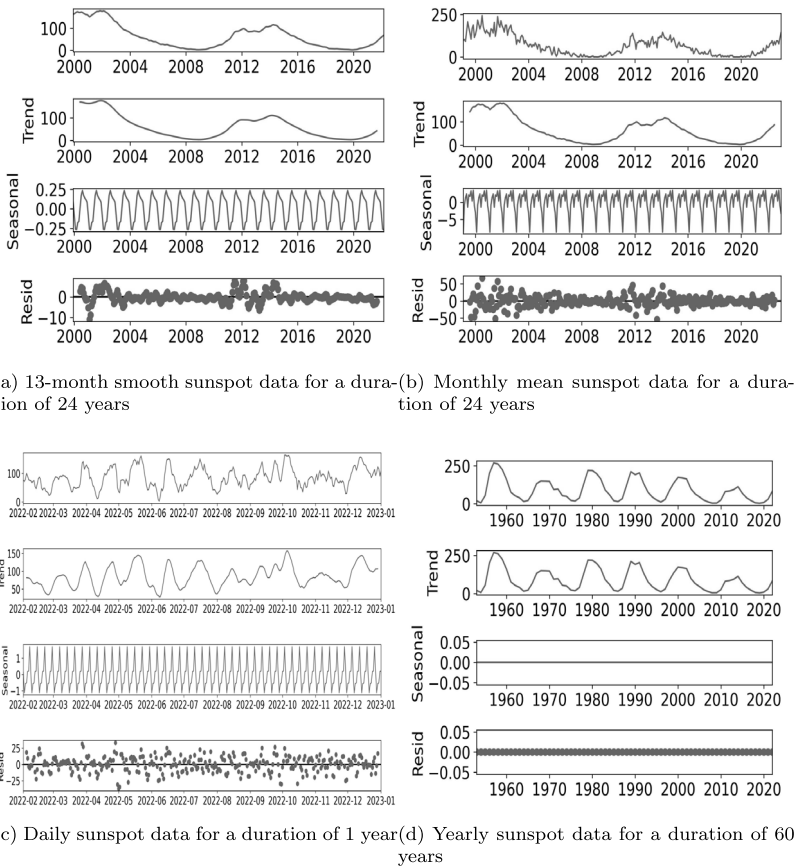


Figure 5 Additive decomposition of sunspot data with different frequencies.

the seasonal decomposition depicted in Figure 5 is for a slice of data for a specific duration for visualization. From Figure 5, it can be deduced that the data follow the cyclic trend with seasonality within each cycle and some associated noise, except for the yearly data, which lack seasonality and noise. A study by Chattopadhyay, Jhaharia, and Chattopadhyay (2011) propounds that the SSN data are stationary concerning time owing to the sinusoidal decaying nature of ACF.

3.2. Experimental Setup

This section presents the analysis of experiments performed utilizing five basic deep-learning models and the proposed model.

3.2.1. Data Pre-processing

Data pre-processing has been performed over each variant of the dataset to obtain cleaned data for time-series analysis. During the data-pre-processing phase, the feature corresponding to the sunspot data from each variant of the dataset is obtained separately and missing

Table 3 Hyper-parameter description of deep-learning models.

Models	Layers	Units	Filter	Activation Function	Kernel	Dropout	Optimizer
RNN	1	16	–	relu	–	0.25	adam
CNN	1	–	16	relu	1	0.25	adam
LSTM	1	16	–	relu	–	0.25	adam
BiLSTM	1	8	–	relu	–	0.25	adam
GRU	1	16	–	relu	–	0.25	adam
Proposed Model	2	32, 16	–	swish	12, 12	0.1	rmsprop

values are replaced with the average of previous and next observations. Then, data standardization is performed using z-score normalization. Time-series data have been created out of univariate data using a lag of 11 as it was found on the experimental basis that the data with a lag of 11 performs better than lags of 1, 6, 22, 66, and 132 using trial and error. It was also suggested by Chattopadhyay, Jhaharia, and Chattopadhyay (2011) in their study that the highest autocorrelation coefficient occurs at lags of 11, which was supported by our experimental results. Then, the dataset is trifurcated into training, validation, and a test set in the ratio of 81%, 9%, and 10%, respectively.

3.2.2. Hyper-parameter Setting of Traditional and Proposed Model

The proposed model has been contrasted with five DL models, namely RNN, CNN, GRU, LSTM, and BiLSTM, with their hyper-parameters described in Table 3 for evaluating the efficiency of the proposed model over basic DL models. The batch size is fixed at 66, and the “adam” optimizer has been utilized for all basic models. Patience is set to 20, and each model is trained for 250 epochs with early stopping criteria enabled based on validation loss, min_delta equal to 0, and loss being calculated over “mean squared error”.

The software and hardware specifications utilized for carrying out all the experiments related to this study are as follows:

- **Software Specification:** All the analyses conducted in the study were performed on a Windows 11 Home Operating System in the Python programming language version 3.10.9, which is highly performing and open source. The environment has been created using jupyter version 3.5.3. Tensorflow has been utilized with version 2.11.0 for developing the models and performing the experiments.
- **Hardware Specification:** All experiments reported here used a PC with i3-1115G4 processor (11th Gen Intel® Core™), 3.00 GHz CPU, and 8 GB of RAM.

4. Results and Analysis

The observations made while executing DL and the proposed models are presented here. First, the training of the different models was verified using the graphs of the training loss of the models concerning the number of epochs. The evaluation of the efficacy of different DL models on the monthly mean of total SSN was analyzed using the box plot presented in Figure 6, where each box contains 24 iterated results of specific performance metrics

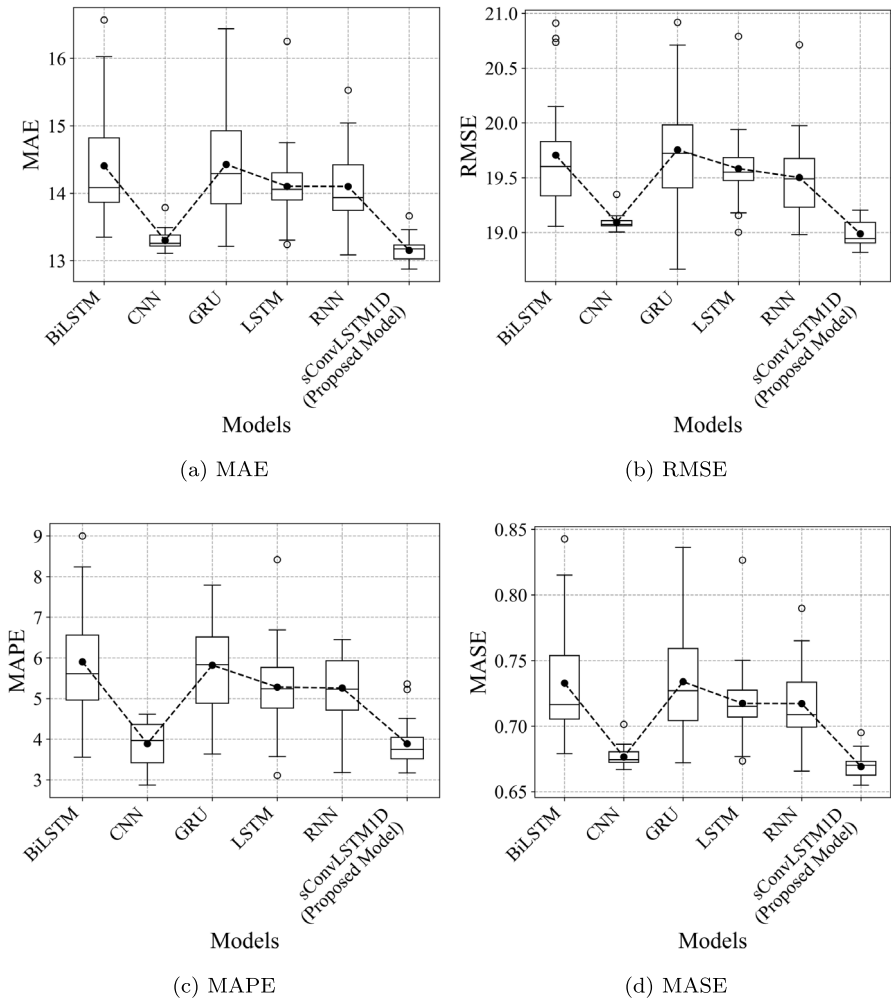


Figure 6 Performance measures of traditional and proposed DL models over the monthly mean of total SSN.

obtained from 24 independent iterations. The graph shows that the mean value of measures obtained for the proposed sConvLSTM1D model is better than others for all four measures utilized for evaluation.

The performances of GRU and BiLSTM are comparatively less efficient with a mean of RMSE of 19.75 and 19.71, respectively, as depicted in Figure 6b. The performance of DL models concerning the proposed model is depicted in Figure 7 for 13-month smoothed SSN data, which validates that the proposed model outperforms the traditional approaches with the second most efficient performance observed using CNN. The proposed model’s mean of RMSE reached 5.69, as illustrated in Figure 7b.

The plot of actual vs. predicted for different models on monthly mean SSN is depicted in Figure 8, illustrating the pattern-capturing capacity of all the models. The scatter plot depicts the relation between actual and predicted values along with the trend line, which is almost at a 45° angle with the value of R^2 , showing that the proposed

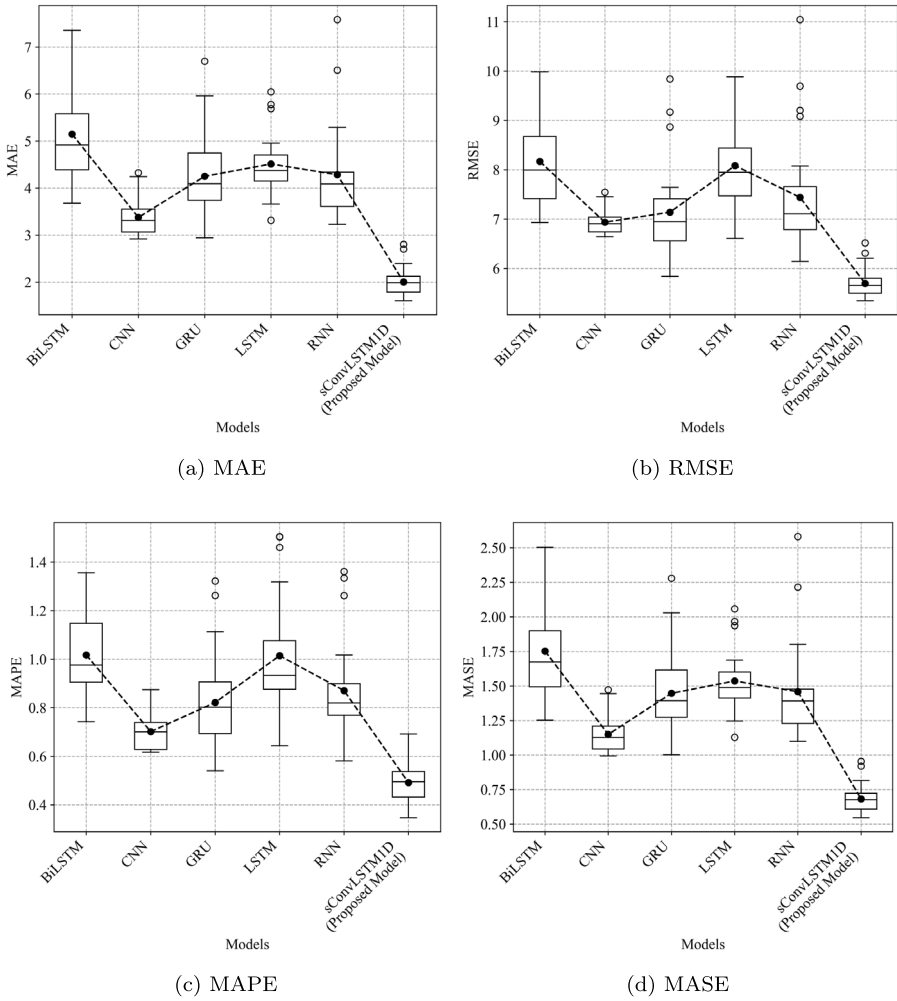


Figure 7 Performance measures of traditional and proposed deep-learning models over the 13-month smoothed SSN.

model has a better R^2 -value, which is nearer to unity. The spread of the data points along the linear trend line is less for the proposed model, characterizing its unbiased prediction nature. The scatter plot of actual vs. predicted for monthly average of SSN is depicted in Figure 9. The predicted values concerning the actual values are depicted in Figure 10, which shows almost coinciding lines representing efficient predictions for all models.

The representation of the predicted values corresponding to actual values for the test data slice of 13-month smoothed sunspot data is in Figure 11, which shows that the data points are highly aligned with the linear-trend line with a high value of R^2 illustrating less error for this variant of data among all four variants of the data. For this variant of SSN data, the value of R^2 is the highest for the proposed model among all six models. From Figure 12, it can be observed that despite the similar performance of the CNN model with the proposed

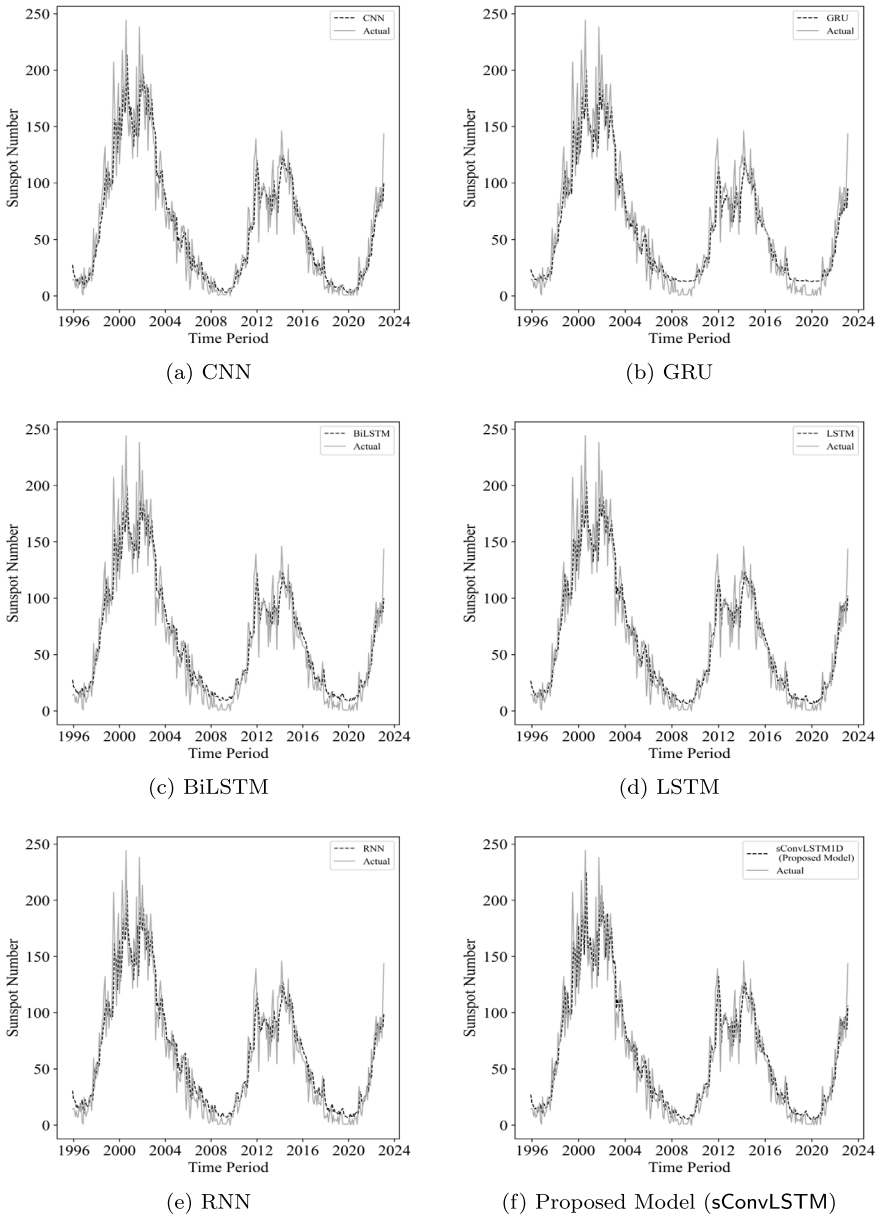


Figure 8 Actual vs. predicted values of monthly mean of total SSN.

model, the standard deviation of the proposed model is small, illustrating the consistency of the model, and the proposed model is more effective from other traditional models as per three performance metrics except for the CNN model. It is more suitable than the other traditional models based on RMSE, as depicted in Figure 12b. The better performance of CNN is due to the more significant number of samples available for training while performing analysis over daily SSN data. However, Wibawa et al. (2022) already observed that CNN is

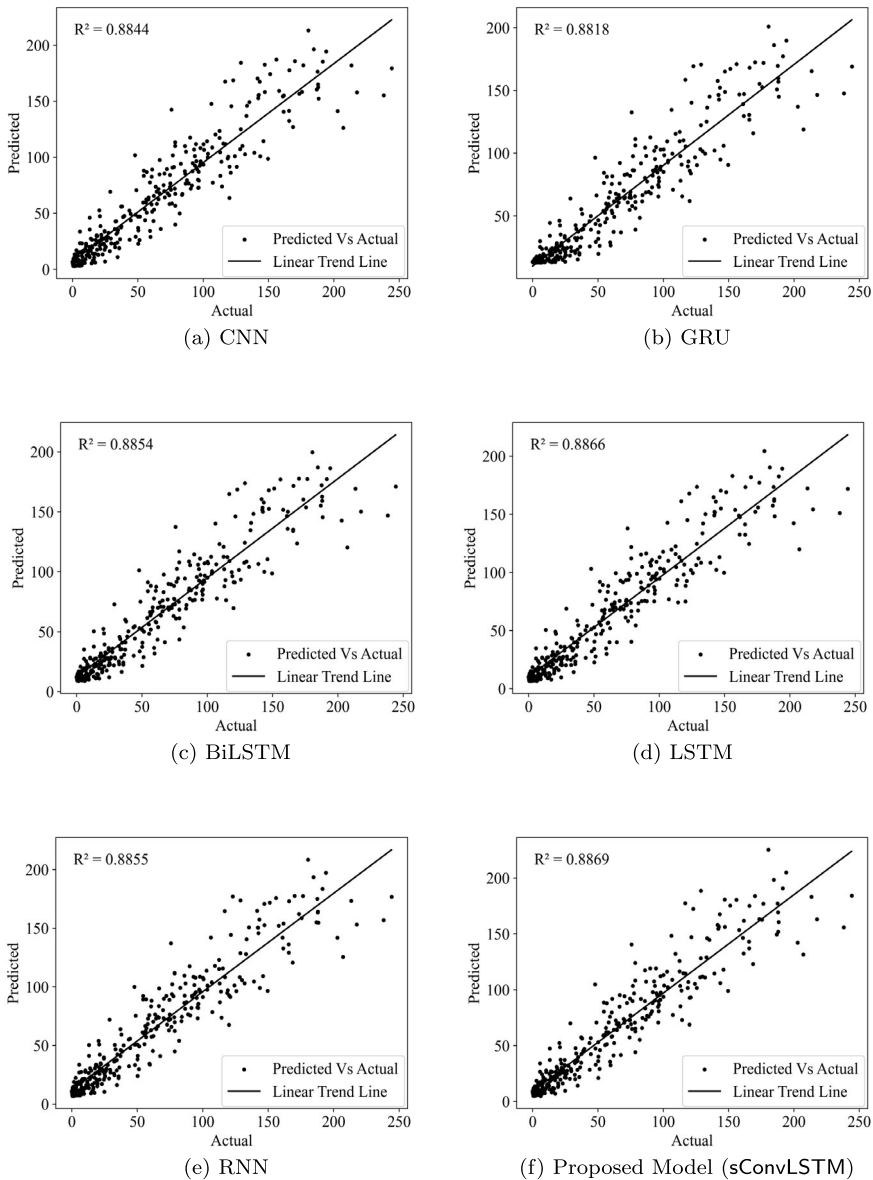


Figure 9 Actual vs. predicted values of monthly mean of total SSN.

more suitable for time-series predictions. Scatter plots with the corresponding value of R^2 are illustrated in Figure 13 for the best predictions of each model out of 24 iterations, which shows that the proposed model has better R^2 ; that is 0.9377, while the CNN model has R^2 -value equal to 0.9277, which is similar but lower than the proposed model. The GRU model has the worst performance due to being most scattered along the linear-trend line, representing high variance. Figure 14 depicts the trend capturing capacity of all models for daily SSN. The comparative view of all models based on the fourth variant of SSN data with

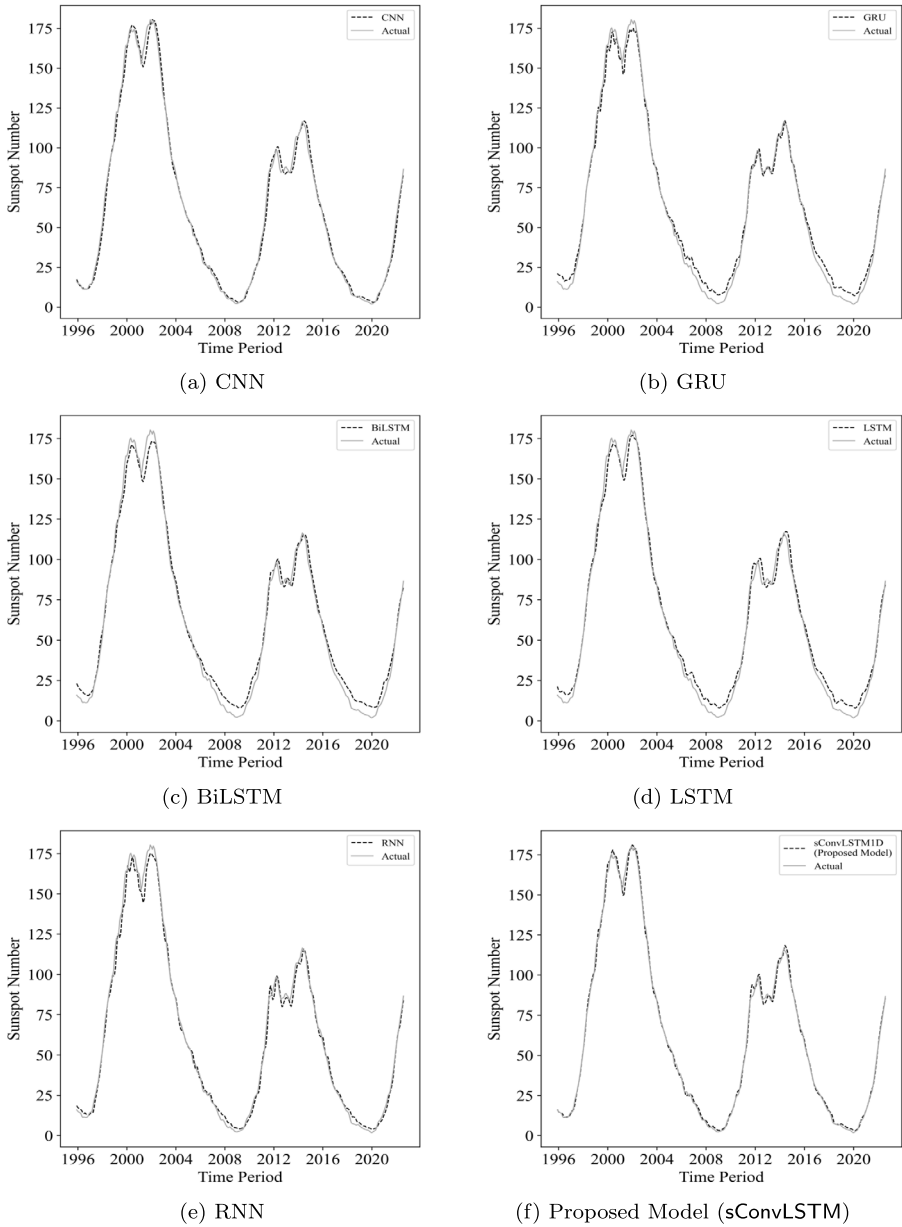


Figure 10 Actual vs. predicted values of 13-month smoothed SSN.

yearly frequency over the four different performance metrics is depicted in Figure 15 where it can be observed that despite the smaller number of available samples for SSN data at yearly frequency, the proposed model is performing far better than all other traditional models. The mean of RMSE reached 14.55, as depicted in Figure 15b. The summarized version of the comparative performance measures over all four variants of SSN data is represented

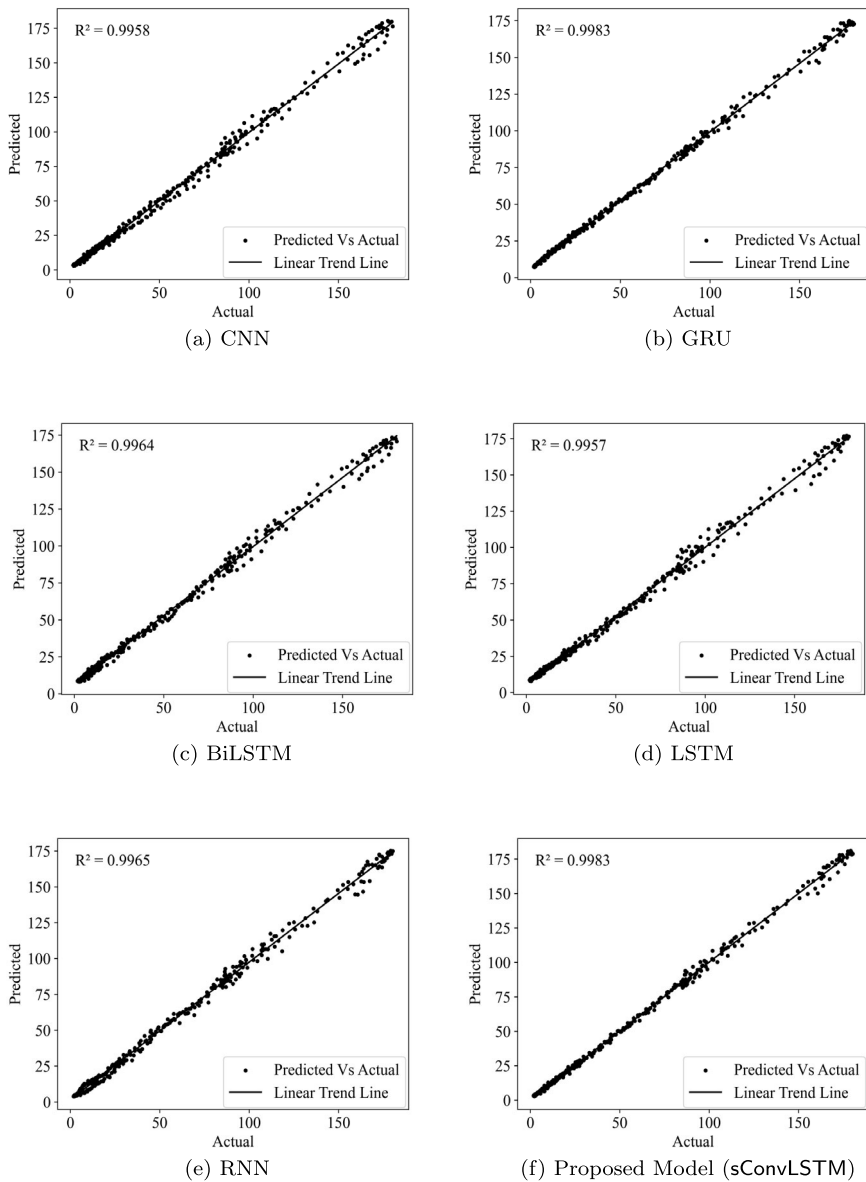


Figure 11 Actual vs. predicted values of 13-month smoothed SSN.

in Table 4. Apart from this, the error measures obtained in the literature have also been included in Table 5 for a comparative analysis of the efficiency of different models from the literature and the proposed model. Figure 16 and Fig 17 represent the actual vs. predicted line and scatter plots of all models respectively. The dotted line in Figure 18 is the forecast plot of SC25 utilizing the proposed model. The Figure 19 to Figure 26 presents the training and validation loss of all the models on all four variants of the SSN data.

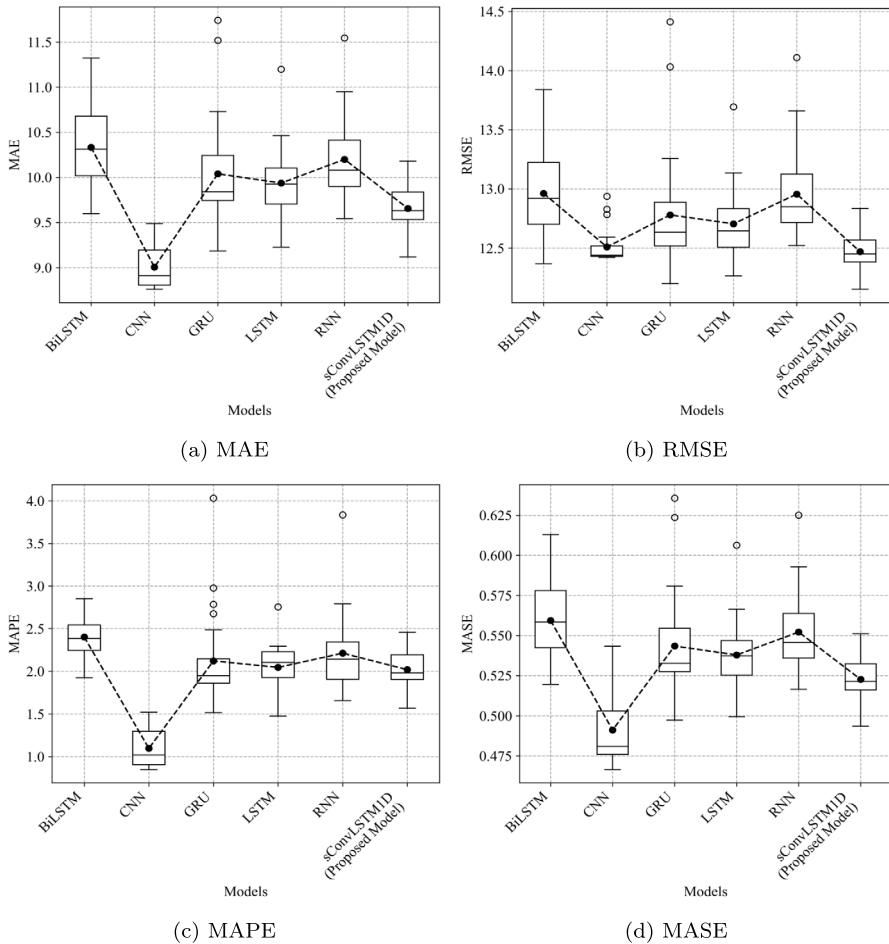


Figure 12 Performance measures of traditional and proposed deep-learning models over the daily SSN.

4.1. Non-parametric Statistical Test

Average ranks have been obtained by implementing the Friedman test over all four variants of data over every evaluation parameter. The Friedman test has been carried out in Table 4 with the ranking of different datasets over different performance measures as all are independent. The final results of the Friedman test, along with Holm’s adjustment and unadjusted p-value, can be seen in Table 6 (Demšar, 2006).

From Table 6, it is deduced that the proposed sConvLSTM1D model has the best ranking with 1.1875. The unadjusted p-value obtained for CNN is insufficient to disprove the null hypothesis. Hence, Holm’s is utilized to adjust the p-value. After applying Holm’s, the value reached 0.05, sufficient to disprove the null hypothesis, proving the proposed model’s effectiveness over the traditional models. The Friedman statistic exhibits a distribution that conforms to a χ^2 -distribution with five degrees of freedom, which is 61.1. The p-values, depicted in Table 6, are derived by employing post-facto methods on the outcomes of the Friedman procedure.

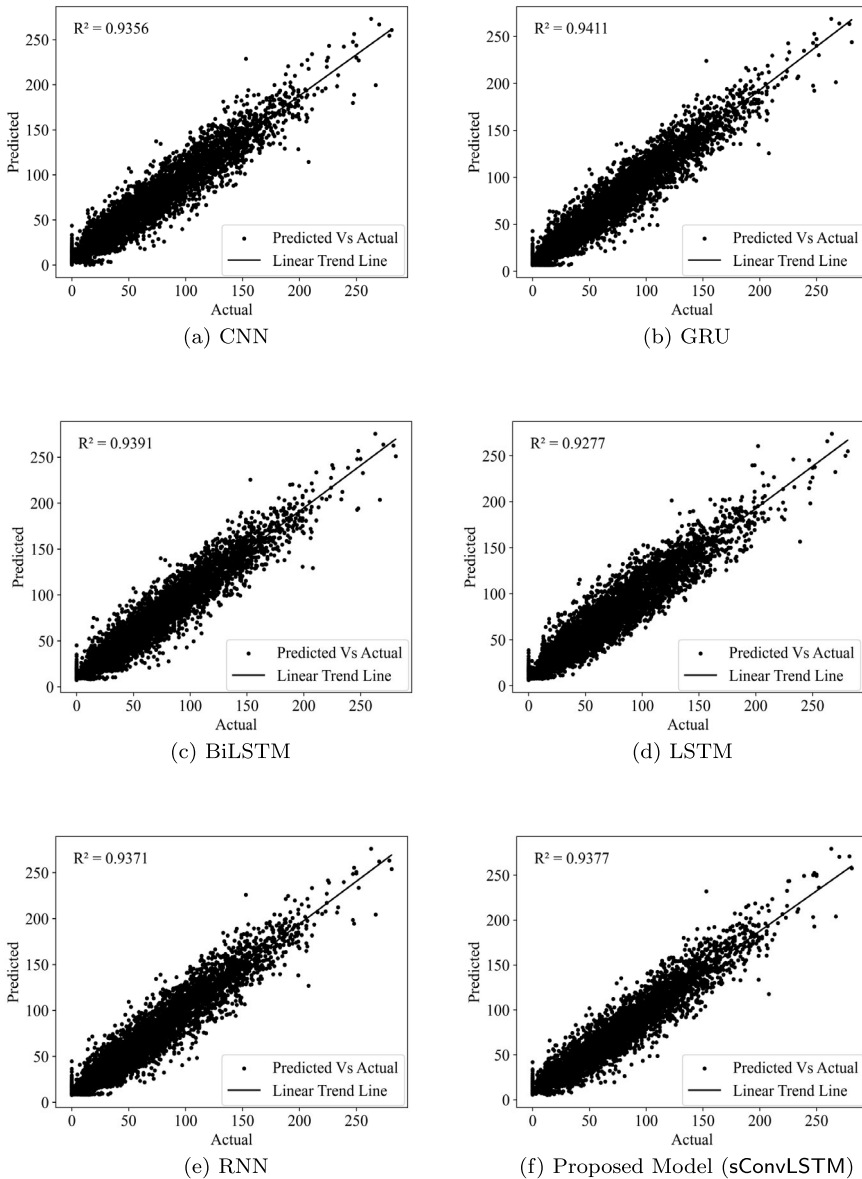
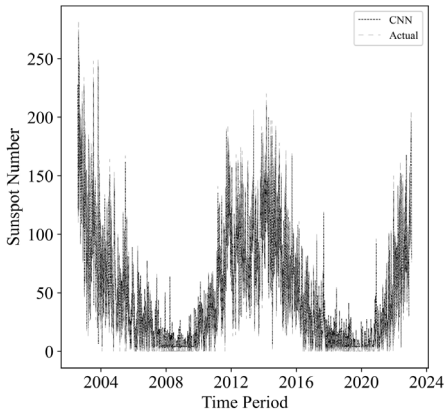
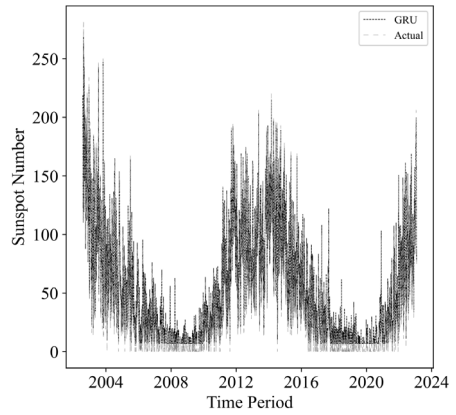


Figure 13 Actual vs. predicted values of daily SSN.

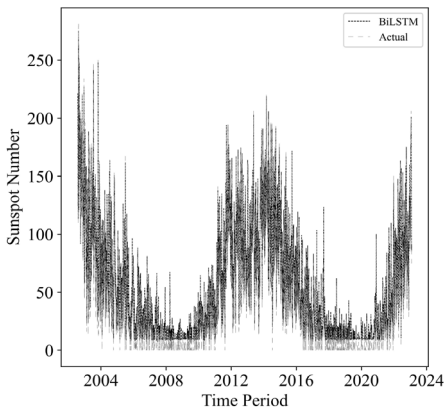
Based on the Friedman ranking of the proposed model (sConvLSTM1D) concerning other traditional deep-learning models depicted in Table 6 and comparative analysis of performance measures with state-of-the-art models illustrated in Table 5, it can be deduced that the sConvLSTM1D model has more accurate predictions.



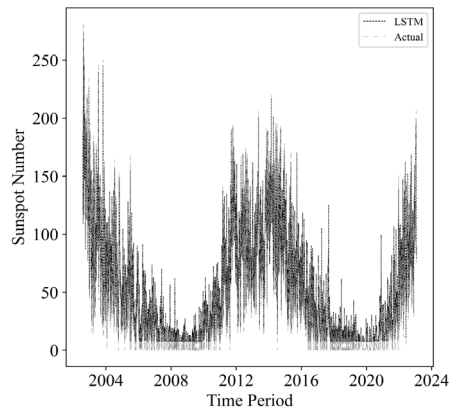
(a) CNN



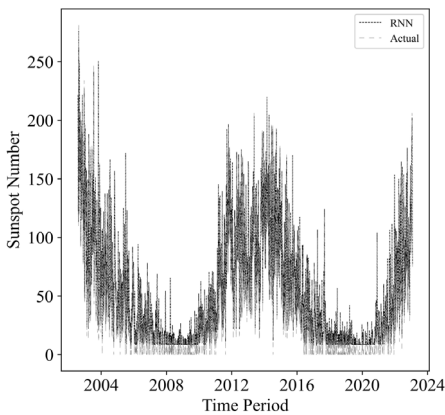
(b) GRU



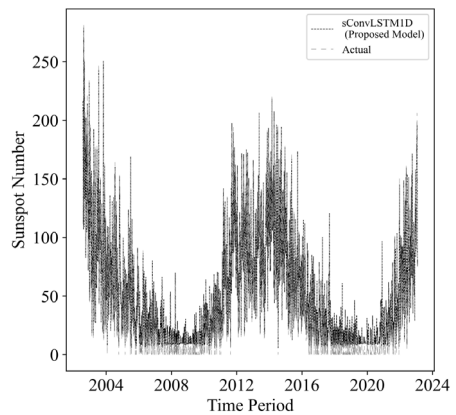
(c) BiLSTM



(d) LSTM



(e) RNN



(f) Proposed Model (sConvLSTM)

Figure 14 Actual vs. predicted values of daily SSN.

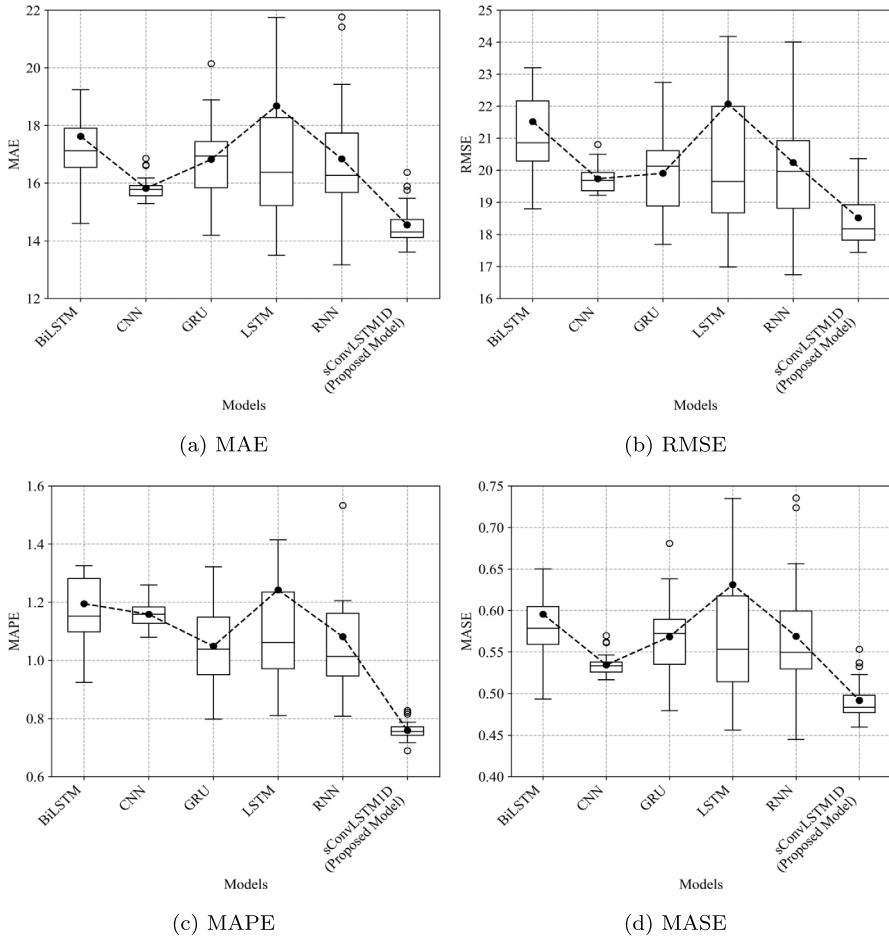


Figure 15 Performance measures of traditional and proposed DL models over the yearly mean of total SSN.

4.2. Prediction of Solar Cycle 25

As per the study of Pesnell and Schatten (2018), the anticipated peak of SC25 is 2025.2 ± 1.5 year, while Pala and Atici (2019) predicted that the SC25 will reach its peak in 2023.2 ± 1.1 year with a maximum of 167.3. A forecast of SC25 has also been made by Upton and Hathaway (2018) using the Advective Flux Transport model, noting the resemblance of the pattern of the SC25 with that of the SC24 and establishing it as the smallest cycle in the previous century. We partially agree with the statement as it is also a weak solar cycle, but in our study, it has been observed that SC25 will be slightly stronger than SC24. As per our iterated one-step-ahead forecast from the static model over the yearly mean of total SSN, it has been observed that the peak value of SSN in SC25 will be 140.8 in 2024, whereas the span of the present cycle will be up to the year 2030 with a minimum value of 16.1. Considering the minimum value of the yearly average of total SSN, which was 3.6 observed in 2019, this cycle will also be 11 years. Similarly, dynamo-based forecasting carried out by Labonville, Charbonneau, and Lemerle (2019) suggests that SC25 would be

Table 4 Performance comparison of different models over different performance measures (*Bold value* represents the best performance in the respective section for each error measure).

	Models	MAE	RMSE	MASE	MAPE	R^2
Daily Total Sunspot Number	LSTM	9.94 (± 0.42)	12.70 (± 0.31)	0.54 (± 0.02)	2.04 (± 0.28)	0.931 (± 0.003)
	GRU	10.04 (± 0.63)	12.78 (± 0.53)	0.54 (± 0.03)	2.12 (± 0.55)	0.930 (± 0.005)
	CNN	9.07 (± 0.39)	12.51 (± 0.14)	0.49 (± 0.02)	1.13 (± 0.36)	0.933 (± 0.002)
	RNN	10.20 (± 0.48)	12.96 (± 0.38)	0.55 (± 0.03)	2.21 (± 0.46)	0.929 (± 0.004)
	BiLSTM	10.33 (± 0.46)	12.96 (± 0.37)	0.55 (± 0.02)	2.40 (± 0.26)	0.928 (± 0.004)
	Proposed Model (sConvLSTM1D)	9.65 (± 0.25)	12.47 (± 0.16)	0.52 (± 0.01)	2.02 (± 0.22)	0.934 (± 0.002)
Monthly Mean Total Sunspot Number	LSTM	14.10 (± 0.59)	19.58 (± 0.34)	0.71 (± 0.03)	5.28 (± 1.12)	0.878 (± 0.004)
	GRU	14.43 (± 0.85)	19.75 (± 0.50)	0.73 (± 0.04)	5.82 (± 1.13)	0.876 (± 0.006)
	CNN	13.30 (± 0.15)	19.09 (± 0.07)	0.68 (± 0.008)	(3.89 ± 0.57)	0.884 (± 0.0008)
	RNN	14.10 (± 0.53)	19.50 (± 0.38)	0.72 (± 0.03)	5.26 (± 0.83)	0.879 (± 0.004)
	BiLSTM	14.40 (± 0.86)	19.71 (± 0.52)	0.73 (± 0.04)	5.90 (± 1.33)	0.876 (± 0.007)
	Proposed Model (sConvLSTM1D)	13.15 (± 0.18)	18.99 (± 0.12)	0.67 (± 0.01)	3.89 (± 0.54)	0.885 (± 0.001)
13-month smoothed Total Sunspot Number	LSTM	4.51 (± 0.63)	8.08 (± 0.83)	1.54 (± 0.21)	1.01 (± 0.23)	0.976 (± 0.005)
	GRU	4.25 (± 0.94)	7.14 (± 0.97)	1.45 (± 0.32)	0.82 (± 0.20)	0.982 (± 0.005)
	CNN	3.38 (± 0.40)	6.94 (± 0.26)	1.15 (± 0.14)	0.70 (± 0.08)	0.983 (± 0.001)
	RNN	4.28 (± 0.99)	7.43 (± 1.20)	1.46 (± 0.34)	0.87 (± 0.20)	0.978 (± 0.007)
	BiLSTM	5.14 (± 1.05)	8.17 (± 0.90)	1.75 (± 0.36)	1.02 (± 0.17)	0.976 (± 0.005)
	Proposed Model (sConvLSTM1D)	2.00 (± 0.31)	5.69 (± 0.30)	0.68 (± 0.11)	0.49 (± 0.089)	0.988 (± 0.001)
Yearly Mean Total Sunspot Number	LSTM	18.68 (± 7.24)	22.07 (± 7.51)	0.63 (± 0.24)	1.24 (± 0.54)	0.796 (± 0.18)
	GRU	16.82 (± 1.42)	19.91 (± 1.36)	0.57 (± 0.05)	1.05 (± 0.13)	0.850 (± 0.02)
	CNN	15.82 (± 0.42)	19.73 (± 0.44)	0.53 (± 0.01)	1.16 (± 0.04)	0.853 (± 0.007)
	RNN	16.84 (± 2.06)	20.24 (± 2.24)	0.57 (± 0.07)	1.08 (± 0.21)	0.844 (± 0.04)
	BiLSTM	17.62 (± 2.06)	21.52 (± 2.26)	0.60 (± 0.07)	1.19 (± 0.17)	0.824 (± 0.04)
	Proposed Model (sConvLSTM1D)	14.55 (± 0.78)	18.51 (± 0.94)	0.49 (± 0.03)	0.76 (± 0.03)	0.87 (± 0.01)

weaker than the preceding cycle with a short duration and a peak in the first half of 2025. Similar to our prediction, Li et al. (2018) also forecast that SC25 will be of higher intensity than SC24 in terms of amplitude and reach its peak in October 2024 with an anticipated value of 168.5 ± 16.3 . SC25 has also been predicted by Kakad, Kakad, and Ramesh (2017) based on Shannon Entropy estimates, suggesting a 63 ± 11.3 peak for the smoothed SSN. Utilizing an optimized LSTM model, Zhu et al. (2023) predicted that SC25 will reach its peak in January 2025 with a maximum value of 213, while Han and Yin (2019) predicted that the maximum value of sunspots will reach approximately 228.8 ± 40.5 at 2023.9 ± 1.6 year. Another study on the prediction of SC25 by Zhu, Zhu, and He (2022) anticipates the peak of SSN in July 2025 \pm two months with a SSN peak amplitude 143.6 ± 8.7 using LSTM with $F_{10.7}$. According to the study of Okoh et al. (2018) using hybrid regression and a neural-network method for the SC25, the peak of SSN will be 122.1 ± 18.2 in January 2025 \pm six months. Another prediction comes from Du (2022) based on the rising rate of the solar cycle, which means the growth rate of solar activity in the early phase of the solar cycle. Du (2022) predicted that the SC25 will peak with an SSN value of 135.5 ± 33.2 in December 2024. A summary version of the comparative analysis of the peak of SSN for SC25 is presented in Table 7.

Table 5 Comparative analysis of different state-of-the-art models with the proposed model (Bold value represents the best performance in the respective section for each error measure).

Data	Reference	Approach	RMSE	MAE	MASE
Daily Total Sunspot Number	Peguero and Carrasco (2023)	LSTM	40.1	–	–
	This article	Proposed Model (sConvLSTM1D)	12.47	9.65	0.52
Monthly Mean Total Sunspot Number	Pala and Atici (2019)	LSTM 6 part	35.9	–	–
	Panigrahi et al. (2021)	PHM-MAX-ARIMA-ETS-SVM	22.72	16.55	0.905
	Hasoon and Al-Hashimi (2022)	DNN-LSTM	20.33	13.75	–
	Lee (2020)	EMD+LSTM	60 > x > 30	–	–
	Nghiem et al. (2022)	CNN-LSTM	26.10	18.74	–
	Dang et al. (2022)	DL+XgBoost	25.70	19.82	–
	This article	Proposed Model (sConvLSTM1D)	18.99	13.15	0.67
Yearly Mean Total Sunspot Number	Panigrahi et al. (2021)	PHM-MAX-ARIMA-ETS-SVM	25.46	18.991	0.607
	Elgamal (2020)	Deep LSTM	23.47	–	–
	Kumar, Sunil, and Yadav (2023)	β SARMA + LSTM	–	–	0.5367
	This article	Proposed Model (sConvLSTM1D)	18.51	14.55	0.49

Table 6 Average Friedman rankings of the algorithms.

Algorithm	Ranking	$z = (R_0 - R_i)/SE$	p	Holm
LSTM	4.50	5.01	0.000001	0.013
GRU	3.88	4.06	0.000048	0.025
CNN	1.94	1.13	0.256839	0.050
RNN	3.94	4.16	0.000032	0.017
BiLSTM	5.56	6.61	0	0.010
Proposed Convolutional LSTM Model	1.19	–	–	–

5. Conclusions

In this work, an attempt is made to improve the prediction of SC25 by reducing the error while testing the model using a novel stacked model made up of ConvLSTM1D layers with an embedded layer of a “repeat vector” within it, followed by dropout and a fully connected layer. The model’s effectiveness is validated on the four variants of the SSN data obtained from the SILSO, Royal Observatory, Belgium, with the four different frequencies of sampling of the SSN on the solar surface and different statistical characteristics. Friedman ranking has been carried out as a non-parametric statistical test to assess the effectiveness of the proposed sConvLSTM1D model with five traditional models namely LSTM, GRU, CNN, RNN, and BiLSTM, which resulted in the rejection of the null hypothesis for all other models suggesting the better performance of the proposed model. This ensures that the proposed model composed of stacked ConvLSTM1D layers provides more accurate predictions than the traditional deep-learning models. After validating the efficacy of the model, the forecast

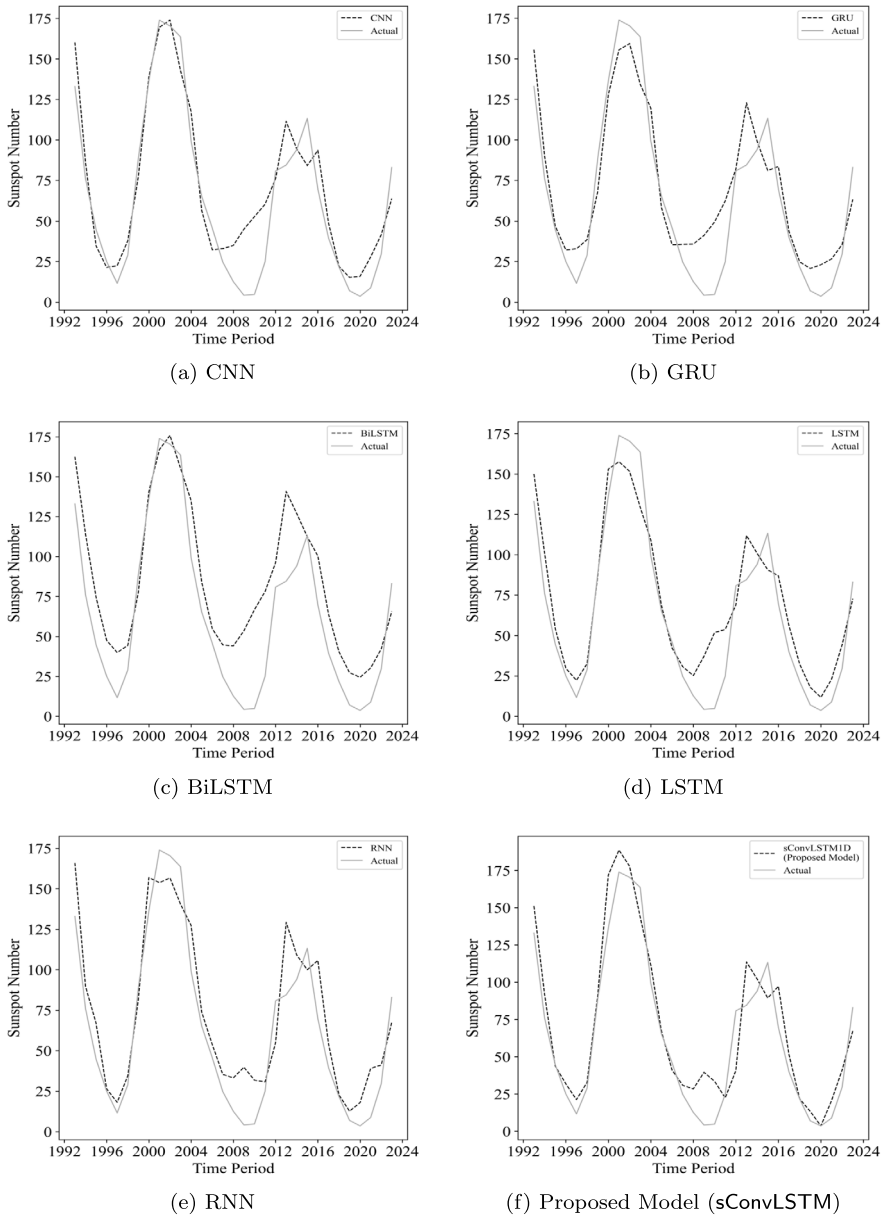
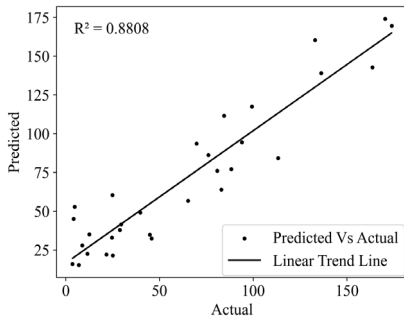
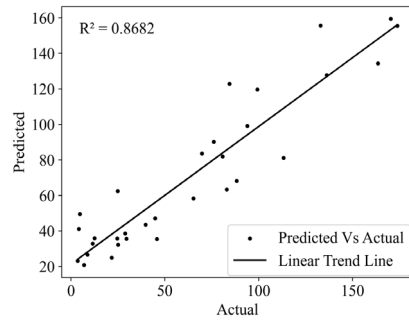


Figure 16 Actual vs. predicted values of yearly mean of total SSN.

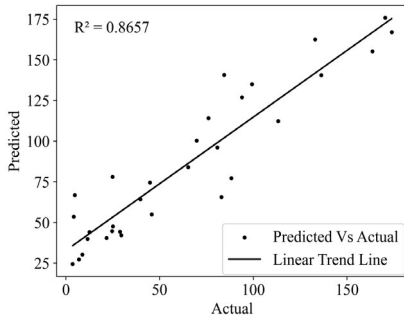
for the ongoing SC25 has been carried out with the prediction of the peak value of SSN to be reached as 140.8 and 2024 as the year in which the peak is achieved. Apart from this comparative analysis of the peak and timing for SC25, a literature search has been carried out to find the differences and resemblances in the predictions. Observing the forecast, it is also deduced that SC25 is a weak cycle based on the temporal duration and amplitude, and it will last up to 2030. The future scope for improving the SSN prediction model includes



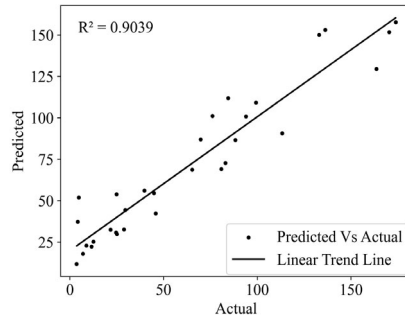
(a) CNN



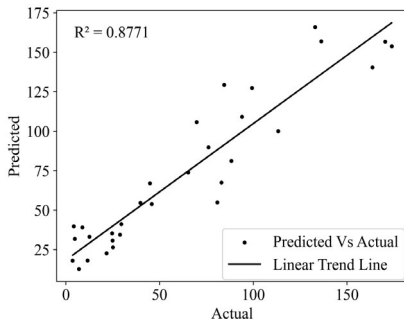
(b) GRU



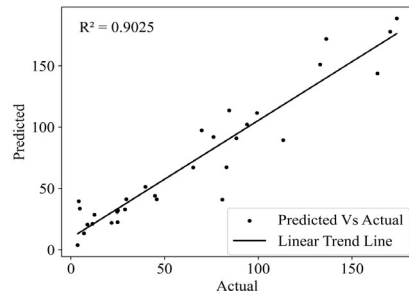
(c) BiLSTM



(d) LSTM



(e) RNN



(f) Proposed Model (sConvLSTM)

Figure 17 Actual vs. predicted values of yearly mean total SSN.

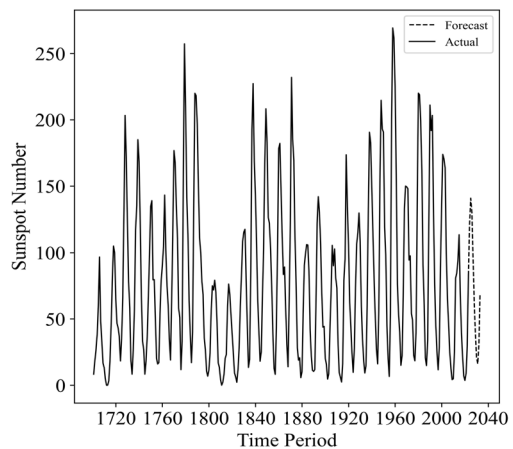
improvement to overcome the problem of residuals present in it. Furthermore, the forecasting model can also be adjusted with different orientations and numbers of layers for better performance and to reduce the error. The decomposition of the dataset can also be utilized for separate training and prediction of different components of the dataset.

Table 7 Comparison of peak of SSN for SC25 and its corresponding time.

Reference	Time Period	Peak SSN
Kakad, Kakad, and Ramesh (2017)	–	63 ±11.3
Singh and Bhargawa (2017)	June 2024	102.8 ±24.6
Pesnell and Schatten (2018)	2025.2 ±1.5	–
Okoh et al. (2018)	January 2025 ±0.5 year	122.1 ±18.2
Li et al. (2018)	October 2024	168.5 ±16.3
Pala and Atici (2019)	2023.2 ±1.1	167.3
Covas, Peixinho, and Fernandes (2019)	2022–2023	57 ±17
Labonville, Charbonneau, and Lemerle (2019)	2025	–
Han and Yin (2019)	2023	228.8 ±40.5
McIntosh et al. (2020)	–	229 ±76
Gonçalves, Echer, and Frigo (2020)	2024	117
Lee (2020)	December 2024	100
Du (2020)	2024.1 ±0.8 year	–
Wang, Li, and Guo (2021)	2023	114.3
Zhu, Zhu, and He (2022)	July 2025 ±2 months	143.6 ±8.7
Du (2022)	December 2024	135.5 ±33.2
Prasad et al. (2022)	August 2023 ±2 months	171 ±3.4
Zhu et al. (2023)	January 2025	213
This article	2024	140.844

Appendix: Loss Plots of Different Models for Different Datasets

Figure 18 Forecast of SC25 based on the proposed stacked ConvLSTM1D model over yearly mean of sunspot number.



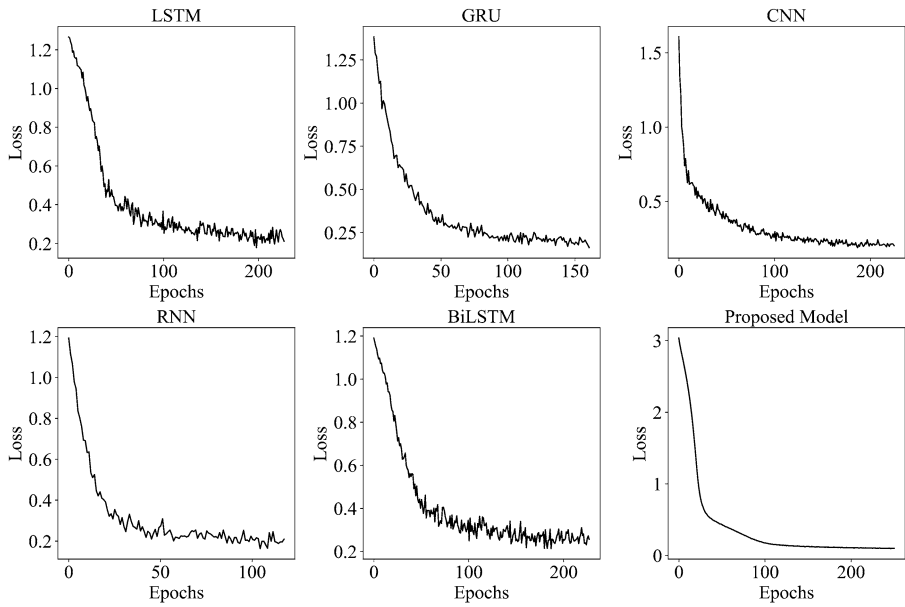


Figure 19 Training loss of all models on the yearly mean of SSN.

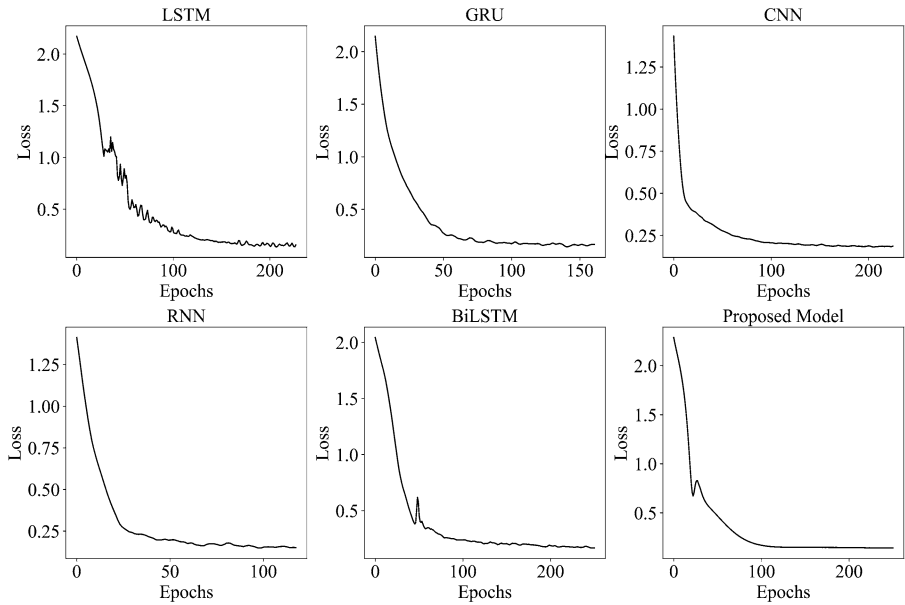


Figure 20 Validation loss of all models on the yearly mean of SSN.

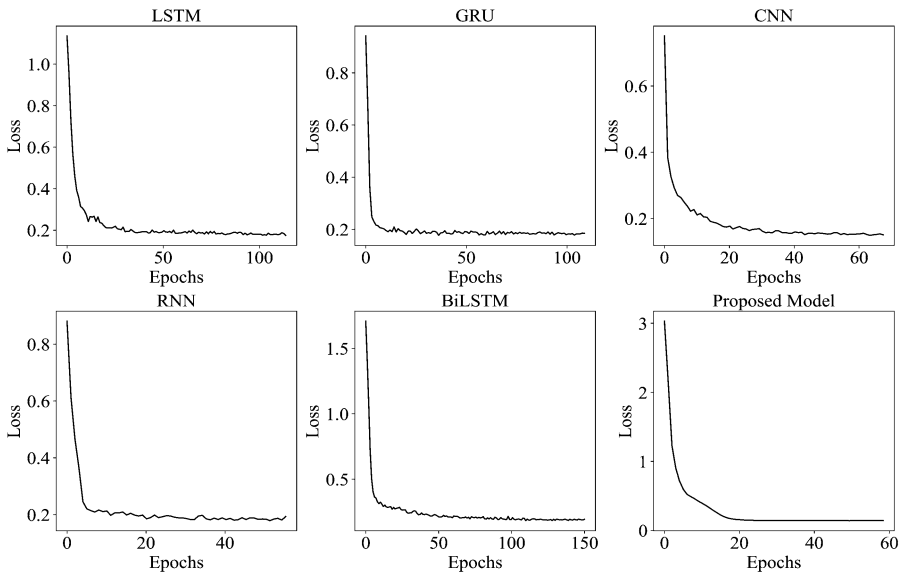


Figure 21 Training loss of all models on the monthly mean of total SSN.

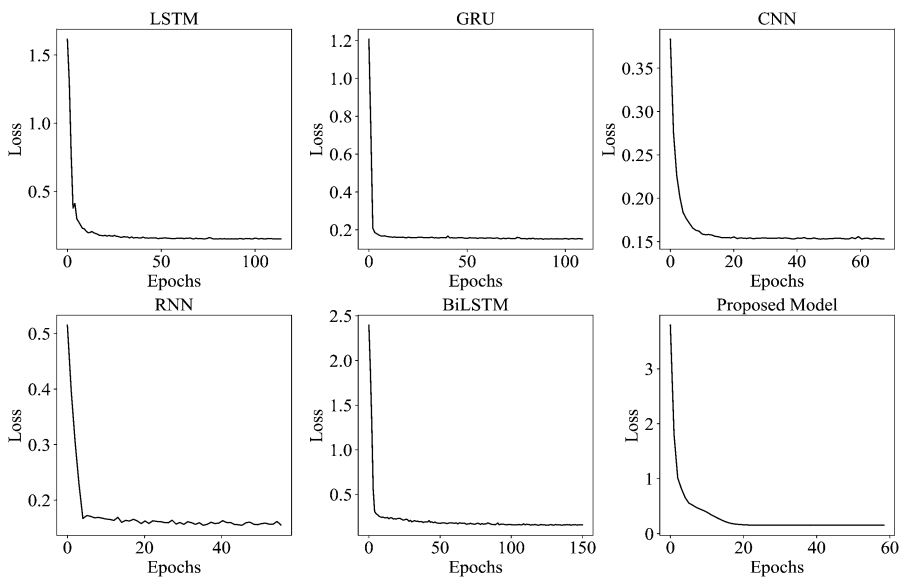


Figure 22 Validation loss of all models on the monthly mean of total SSN.

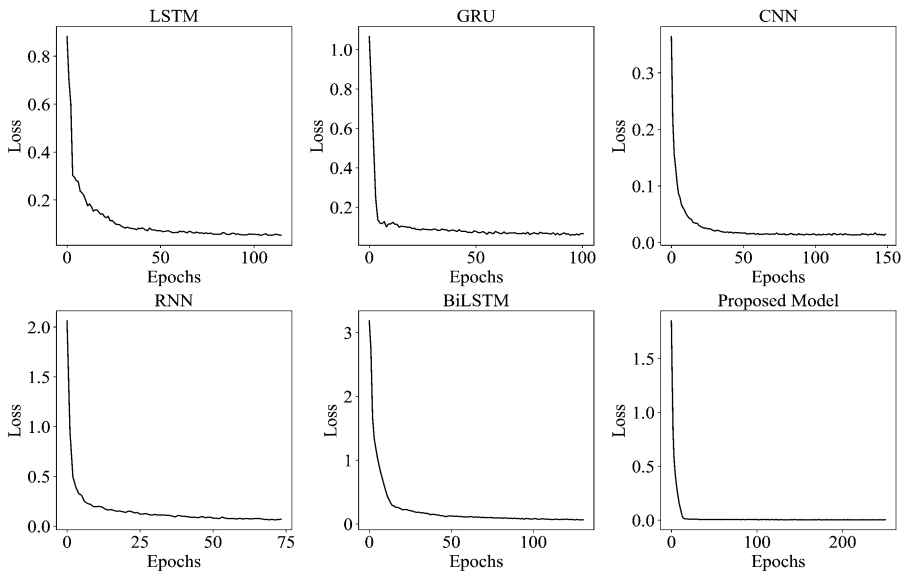


Figure 23 Training loss of all models on the 13-month smooth of total SSN.

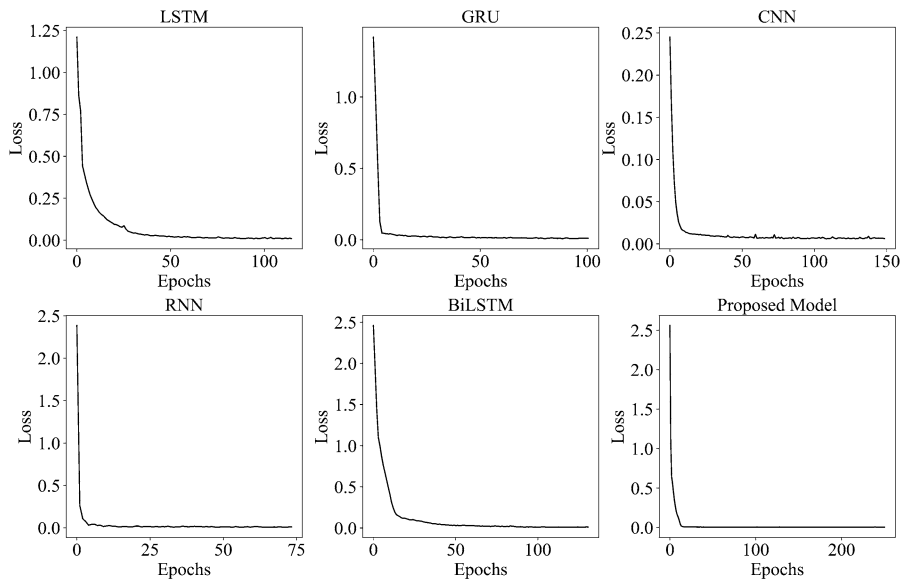


Figure 24 Validation loss of all models on the 13-month smooth of total SSN.

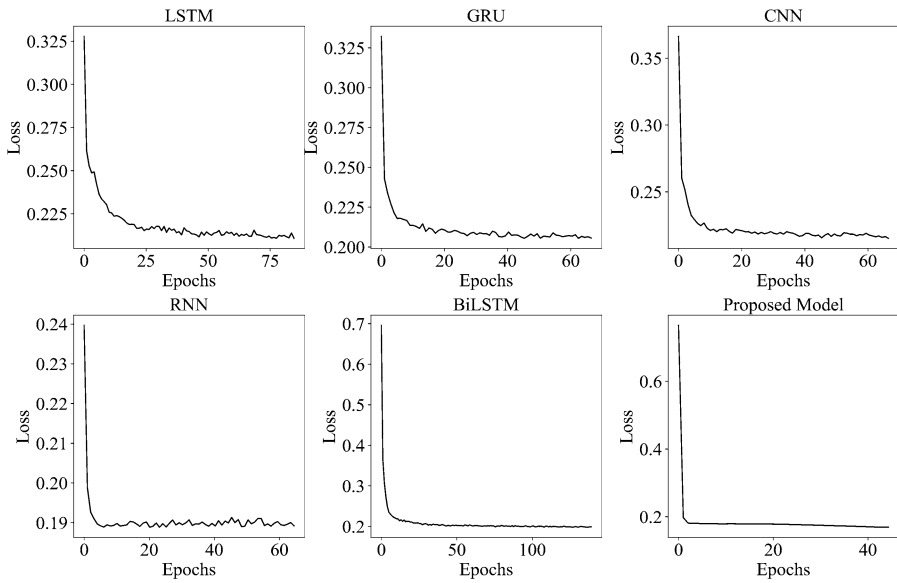


Figure 25 Training loss of all models on the daily SSN.

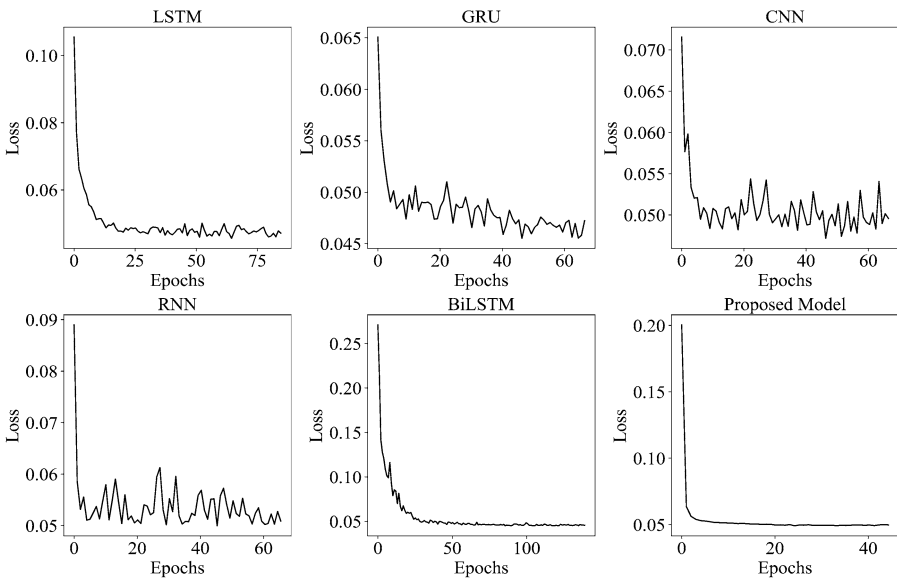


Figure 26 Validation loss of all models on the daily SSN.

Acknowledgments The authors thank the WDC-SILSO, Royal Observatory of Belgium, Brussels, for the sunspot-number data.

Author contributions Abhijeet Kumar developed the methodology, performed the experimental analysis for the article and prepared the original draft. Vipin Kumar did the conceptualisation of the article, and methodology, supervised, reviewed and validated the work, formal analysis.

Declarations

Competing interests The authors declare no competing interests.

References

- Arfianti, U.I., Novitasari, D.C.R., Widodo, N., Hafiyusholeh, M., Utami, W.D.: 2021, Sunspot number prediction using Gated Recurrent Unit (GRU) algorithm. *Indones. J. Comp. Cybern. Sys.* **15**, 141. DOI.
- Bai, Y., Zeng, B., Li, C., Zhang, J.: 2019, An ensemble long short-term memory neural network for hourly PM2.5 concentration forecasting. *Chemosphere* **222**, 286. DOI.
- Benson, B., Pan, W., Prasad, A., Gary, G., Hu, Q.: 2020, Forecasting solar cycle 25 using deep neural networks. *Solar Phys.* **295**, 65. DOI.
- Büyüksahin, Ü.Ç., Ertekin, Ş.: 2019, Improving forecasting accuracy of time series data using a new ARIMA-ANN hybrid method and empirical mode decomposition. *Neurocomputing* **361**, 151. DOI.
- Cantillo-Luna, S., Moreno-Chuquen, R., Celeita, D., Anders, G.: 2023, Deep and machine learning models to forecast photovoltaic power generation. *Energies* **16**, 4097. DOI.
- Chattopadhyay, S., Jhajharia, D., Chattopadhyay, G.: 2011, Trend estimation and univariate forecast of the sunspot numbers: development and comparison of ARMA, ARIMA and autoregressive neural network models. *C. R. Géosci.* **343**, 433. DOI.
- Covas, E., Peixinho, N., Fernandes, J.: 2019, Neural network forecast of the sunspot butterfly diagram. *Solar Phys.* **294**, 24. DOI.
- Dang, Y., Chen, Z., Li, H., Shu, H.: 2022, A comparative study of non-deep learning, deep learning, and ensemble learning methods for sunspot number prediction. *Appl. Artif. Intell.* **36**, 2074129. DOI.
- Demšar, J.: 2006, Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.* **7**, 1.
- Du, Z.: 2020, The solar cycle: predicting the peak of solar cycle 25. *Astrophys. Space Sci.* **365**, 104. DOI.
- Du, Z.: 2022, Predicting the maximum amplitude of solar cycle 25 using the early value of the rising phase. *Solar Phys.* **297**, 61. DOI.
- Elgamal, M.: 2020, Sunspot time series forecasting using deep learning. *Int. J. Comp. Inform. Tech.* **2279**(0764), 9. DOI.
- Gonçalves, Í.G., Echer, E., Frigo, E.: 2020, Sunspot cycle prediction using warped Gaussian process regression. *Adv. Space Res.* **65**, 677. DOI.
- Goodfellow, I., Bengio, Y., Courville, A.: 2016, *Deep Learning*, MIT, Cambridge USA.
- Han, Y., Yin, Z.: 2019, A decline phase modeling for the prediction of solar cycle 25. *Solar Phys.* **294**, 107. DOI.
- Hasoon, S.O., Al-Hashimi, M.M.: 2022, Hybrid deep neural network and long short term memory network for predicting of sunspot time series. *Int. J. Math. Comput. Sci.* **17**, 955.
- Hochreiter, S., Schmidhuber, J.: 1997, Long short-term memory. *Neural Comp.* **9**(8), 1735. DOI.
- Kakad, B., Kakad, A., Ramesh, D.S.: 2017, Shannon entropy-based prediction of solar cycle 25. *Solar Phys.* **292**, 1. DOI.
- Kumar, B., Sunil, Yadav, Y.: 2023, A novel hybrid model combining β SARMA and LSTM for time series forecasting. *Appl. Soft Comput.* **134**, 110019. DOI.
- Labonville, F., Charbonneau, P., Lemerle, A.: 2019, A dynamo-based forecast of solar cycle 25. *Solar Phys.* **294**, 82. DOI.
- Lee, T.: 2020, EMD and LSTM hybrid deep learning model for predicting sunspot number time series with a cyclic pattern. *Solar Phys.* **295**, 82. DOI.
- Li, F., Kong, D., Xie, J., Xiang, N., Xu, J.: 2018, Solar cycle characteristics and their application in the prediction of cycle 25. *J. Atmos. Solar-Terr. Phys.* **181**, 110. DOI.
- Marques, C., Leal-Júnior, A., Kumar, S.: 2023, Multifunctional integration of optical fibers and nanomaterials for aircraft systems. *Materials* **16**, 1433. DOI.
- McIntosh, S.W., Chapman, S., Leamon, R.J., Egeland, R., Watkins, N.W.: 2020, Overlapping magnetic activity cycles and the sunspot number: forecasting sunspot cycle 25 amplitude. *Solar Phys.* **295**, 1. DOI.

- Nghiem, T.-L., Le, V.-D., Le, T.-L., Maréchal, P., Delahaye, D., Vidosavljevic, A.: 2022, Applying Bayesian inference in a hybrid CNN-LSTM model for time-series prediction. In: *2022 Internat. Conf. Multimedia Analy. Pattern Recog. (MAPR)*, **1**, IEEE, Los Alamitos. DOI.
- Okoh, D., Seemala, G., Rabi, A., Uwamahoro, J., Habarulema, J., Aggarwal, M.: 2018, A hybrid regression-neural network (HR-NN) method for forecasting the solar activity. *Space Weather* **16**, 1424. DOI.
- Pala, Z., Atici, R.: 2019, Forecasting sunspot time series using deep learning methods. *Solar Phys.* **294**, 50. DOI.
- Panigrahi, S., Pattanayak, R.M., Sethy, P.K., Behera, S.K.: 2021, Forecasting of sunspot time series using a hybridization of ARIMA, ETS and SVM methods. *Solar Phys.* **296**, 1. DOI.
- Peguero, J., Carrasco, V.: 2023, A critical comment on “can solar cycle 25 be a new Dalton Minimum?”. *Solar Phys.* **298**, 48. DOI.
- Pesnell, W.D.: 2008, Predictions of solar cycle 24. *Solar Phys.* **252**, 209. DOI.
- Pesnell, W.D., Schatten, K.H.: 2018, An early prediction of the amplitude of solar cycle 25. *Solar Phys.* **293**, 112. DOI.
- Prasad, A., Roy, S., Sarkar, A., Panja, S.C., Patra, S.N.: 2022, Prediction of solar cycle 25 using deep learning based long short-term memory forecasting technique. *Adv. Space Res.* **69**, 798. DOI.
- Ramachandran, P., Zoph, B., Le, Q.V.: 2017, Searching for activation functions. DOI. arXiv.
- Ramadevi, B., Bingi, K.: 2022, Time series forecasting model for sunspot number. In: *2022 Internat. Conf. on Intelligent Control. Comp. Smart Power (ICICCCSP)*, **1**, IEEE, Los Alamitos. DOI.
- Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K., Woo, W.-C.: 2015, Convolutional LSTM network: a machine learning approach for precipitation nowcasting. In: Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., Garnett, R. (eds.) *Adv. Neural Inform. Proc. Sys.* **28**, Curran Associates, Red Hook. URL.
- Shi, C., Zhang, Z., Zhang, W., Zhang, C., Xu, Q.: 2022, Learning multiscale temporal-spatial-spectral features via a multipath convolutional LSTM neural network for change detection with hyperspectral images. *IEEE Trans. Geosci. Remote Sens.* **60**, 1. DOI.
- SILSO World Data Center: The International Sunspot Number. International Sunspot Number Monthly Bulletin and online catalogue. URL.
- Singh, A., Bhargawa, A.: 2017, An early prediction of 25th solar cycle using Hurst exponent. *Astrophys. Space Sci.* **362**, 199. DOI.
- Szandafa, T.: 2021, Review and comparison of commonly used activation functions for deep neural networks. *Bio-Inspir. Comput.* **903**, 203. DOI.
- Upton, L.A., Hathaway, D.H.: 2018, An updated solar cycle 25 prediction with AFT: the modern minimum. *Geophys. Res. Lett.* **45**, 8091. DOI.
- Vokhmyanin, M., Arlt, R., Zolotova, N.: 2020, Sunspot positions and areas from observations by Thomas Harriot. *Solar Phys.* **295**, 39. DOI. ADS.
- Wang, Q.-J., Li, J.-C., Guo, L.-Q.: 2021, Solar cycle prediction using a long short-term memory deep learning model. *Res. Astron. Astrophys.* **21**, 012. DOI.
- Wibawa, A.P., Utama, A.B.P., Elmunsyah, H., Pujiyanto, U., Dwiyanto, F.A., Hernandez, L.: 2022, Time-series analysis with smoothed convolutional neural network. *J. Big Data* **9**, 44. DOI.
- Zhang, L., Lu, L., Wang, X., Zhu, R.M., Bagheri, M., Summers, R.M., Yao, J.: 2019, Spatio-temporal convolutional LSTMs for tumor growth prediction by learning 4D longitudinal patient data. *IEEE Trans. Med. Imaging* **39**, 1114. DOI.
- Zhu, H., Zhu, W., He, M.: 2022, Solar cycle 25 prediction using an optimized long short-term memory mode with F10.7. *Solar Phys.* **297**, 157. DOI.
- Zhu, H., Chen, H., Zhu, W., He, M.: 2023, Predicting solar cycle 25 using an optimized long short-term memory model based on sunspot area data. *Adv. Space Res.* **71**, 3521. DOI.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.