INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Comparative Study of ARIMA and Deep Learning for NDVI Forecasting Using Landsat 8 Data

Vidya Mohanty[1]*, **Dayal Kumar Behera**[1], **Amiya Ranjan Panda**[1], **Subhra Swetanisha**[2]

1 School of Computer Engineering, KIIT Deemed to be University, Bhubaneswar, India
2 School of Computing, Trident Academy of Technology, Bhubaneswar, India

Check for updates

OPEN ACCESS

Received: 03-01-2025

Accepted: 17-03-2025

Published: 10-04-2025

Citation: Mohanty V, Behera DK, Panda AR, Swetanisha S (2025) Comparative Study of ARIMA and Deep Learning for NDVI Forecasting Using Landsat 8 Data. Indian Journal of Science and Technology 18(11): 922-936. https://doi.org/10.17485/IJST/v18i11.18

* **Corresponding author**.

2081040@kiit.ac.in

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objectives: This study is about developing an accurate and efficient NDVI forecasting framework using remote sensing data from the Landsat 8 satellite. The objective is to study the vegetation changes over time and compare the performance of classical statistical models (ARIMA) with deep learning models (LSTM, RNN) in predicting NDVI trends. The study focuses on the Kendrapara district of Odisha, India, and investigates both weekly and monthly NDVI variations to improve forecasting granularity. **Method:** The proposed methodology involves acquiring Landsat 8 imagery for the study area from 2015 to 2022. The NDVI values are extracted, and two time-series datasets (weekly and monthly NDVI) are generated. Data preprocessing techniques, including missing value imputation and noise reduction, are applied to enhance data quality. Three forecasting models that are used in this study ARIMA, LSTM, and RNN—are implemented and trained using an 80:20 train-test split. The models are evaluated based on Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) to determine their predictive performance. **Findings:** The results indicate that ARIMA performs best for short-term NDVI forecasting, with RMSE values of 0.0484 (weekly NDVI) and 0.0645 (monthly NDVI). Deep learning models, particularly LSTM, exhibit superior performance in capturing long-term NDVI trends but show higher error values in short-term predictions. The study highlights the strengths and drawbacks of each model, providing insights into their suitability for different forecasting applications. **Novelty:** This study introduces a detailed comparative analysis of statistical and deep learning models for NDVI forecasting, which is largely absent in existing literature. Unlike previous works that focus on either long-term or annual trends, our research analyzes high-resolution weekly and monthly NDVI variations. Additionally, the study implements advanced preprocessing techniques to mitigate atmospheric distortions and missing data issues, enhancing forecasting accuracy. The findings provide a valuable benchmark for selecting appropriate models for vegetation monitoring and climate change assessments.

Keywords: Remote Sensing; Landsat 8; ARIMA; LSTM; RNN; Vegetation Monitoring; Machine Learning; Climate Change; NDVI; Time-Series Forecasting

1 Introduction

In recent years, the use of time series data has become increasingly prevalent in the field of continuous monitoring and forecasting[1]. The assessment of vegetation health and land cover changes is critical for sustainable environmental management, agricultural planning, and climate change mitigation. Remote sensing techniques, particularly the Normalized Difference Vegetation Index (NDVI), have been widely employed to monitor vegetation dynamics using satellite imagery[2]. NDVI, derived from multispectral satellite data, is a reliable indicator of vegetation density, providing insights into land-use patterns and ecosystem health over time. The increasing generation of satellite data, such as Landsat 8, and advancements in computational techniques have led to the development of various models for NDVI forecasting. However, the accuracy and applicability of these models remain a challenge, necessitating further research in this domain.

Vegetation variation, measured through NDVI, plays a crucial role in regulating land surface temperature (LST), particularly in the context of increasing urbanization and climate change. Studies such as Vohra et al. (2024)[3] have demonstrated that rapid urban expansion leads to rising LST, necessitating improved forecasting models for vegetation monitoring. Similarly, Pande et al. (2024) applied machine learning techniques to predict LST variations using Landsat-8 data, highlighting the importance of integrating NDVI forecasting for sustainable environmental management[4]. In the study[5], Landsat satellite photos from 1989 to 2015 were used to examine the link between LULC, NDVI (Normalized Difference Vegetation Index), and LST in Sivas city centre and its surrounding areas, and UHI intensity was also shown. The greater NDVI values were discovered in dense vegetation regions, and the urban built-up land recorded the highest average temperature during the study period. NDVI and LST were found to have a negative connection, and the average LST was lower in vegetated areas. In[6], researchers used RS data gathered over 26 years in District 1 of Shiraz City to assess the influence of plant distribution, both qualitatively and quantitatively, on land surface temperature (LST) patterns. In recent decades, there has been a rise in the utilization of remote sensing data and technologies in environmental research. Moreover, the study revealed that the quality of vegetation cover declined as the extent of vegetation cover diminished. Jothimani, et al. attempted to extract the LST and analyse its relationship with land-use/land cover and the three indices of remote sensing NDVI, NDWI, and NDBI in the broader Arba Minch region of Ethiopia's Rift Valley. For the two periods, the study was carried out (08 Dec 2013 and 27 Dec 2020). Between 2013 and 2020, the NDBI value significantly improved.

NDVI has been widely used in agriculture to assess crop health, estimate yield, and optimize irrigation strategies. Carreño-Conde et al. (2021) applied predictive modelling to monitor crop dynamics using vegetation indices, demonstrating the potential of NDVI-based forecasting for precision agriculture[7]. However, existing approaches often lack high-resolution short-term forecasts, limiting their applicability for real-time decision-making. This study addresses this gap by developing an NDVI forecasting framework capable of capturing both weekly and monthly vegetation trends, thereby enhancing its utility for agricultural planning.

Advances in satellite technology, particularly Landsat-8 and Landsat-9, have improved the consistency of NDVI measurements, as demonstrated by Zhang et al. (2025), where the reflectivity and continuity of NDVI values were analyzed using Google Earth Engine (GEE)[8]. These advancements ensure the reliability of satellite-derived NDVI data for time-series analysis.

While several satellite datasets, such as MODIS, Sentinel-2B, and LISS-4, are available for NDVI analysis, Landsat-8 has been widely utilized due to its balance between spatial and temporal resolution, making it particularly suitable for agricultural and environmental applications. A comparative study by Jacob et al. highlighted the advantages of Landsat-8 for precision farming, emphasizing its effectiveness in capturing vegetation dynamics with high accuracy [9]. Given these strengths, this study employs Landsat-8 NDVI data to develop a robust forecasting framework.

Several studies have explored time-series forecasting models for NDVI prediction. Traditional statistical models such as the AutoRegressive Integrated Moving Average (ARIMA) and its variants have been extensively used due to their effectiveness in capturing linear patterns in time-series data [10]. However, these models often struggle with non-linear and complex vegetation dynamics influenced by climatic variability [11]. Deep learning models involving Long Short-Term Memory (LSTM) and Recurrent Neural Networks (RNN) have shown promising results in capturing long-term dependencies and non-linear patterns in remote sensing data[miller2024deep].

Recent advancements in deep learning have significantly improved the accuracy of satellite image-based forecasting. Moskolaï et al. [12] reviewed various deep learning architectures for time-series prediction using satellite imagery, highlighting their advantages in capturing complex, non-linear vegetation patterns. Inspired by these developments, this study employs LSTM and RNN models alongside ARIMA to compare their effectiveness in NDVI forecasting, offering a novel benchmark for vegetation trend analysis.

Despite these advancements, limited studies compare classical statistical methods with deep learning approaches on the same dataset, making it difficult to determine the most suitable forecasting model. Accurate NDVI forecasting is essential for long-term vegetation monitoring, particularly in the context of land use and climate change. Recent studies, such as Raza et al. (2024) [13], have explored CA-ANN-based predictive modeling to analyze LULC changes and their correlation with land surface temperature (LST) and NDVI. These findings highlight the importance of integrating advanced forecasting models to predict

Another key limitation in existing studies is the lack of high-resolution temporal analysis. While several works have analyzed annual or seasonal NDVI trends [14], very few have investigated short-term variations at weekly or monthly scales, which are crucial for real-time environmental monitoring [15]. Moreover, NDVI data is often affected by atmospheric noise, cloud cover, and missing values, yet most studies do not incorporate robust preprocessing techniques to handle these inconsistencies [16].

Furthermore, although deep learning models like LSTM and RNN have been applied to NDVI forecasting, these methods often lack interpretability and require large datasets for effective training [17]. On the other hand, ARIMA-based models, while interpretable, are constrained by their inability to handle complex non-linear relationships [18]. A comprehensive performance comparison across different models using standardized evaluation metrics (e.g., RMSE, MAE) is currently missing in the literature.

Based on the above discussion, the major gaps in existing NDVI forecasting studies are:

1. Lack of comparative studies between statistical (ARIMA) and deep learning models (LSTM, RNN) for NDVI forecasting.
2. Limited research on high-resolution forecasting (weekly and monthly NDVI trends) as most studies focus on annual or seasonal changes.
3. Insufficient pre-processing techniques to address atmospheric distortions and missing data in satellite images.
4. Limited benchmarking of forecasting models using standardized performance metrics for real-world applications.

To address the loopholes, this study proposes a comparative analysis of ARIMA, LSTM, and RNN models for NDVI time-series forecasting using Landsat 8 data over the Kendrapara district of Odisha, India. The major contributions of this work are:

- Development of a high-resolution NDVI forecasting framework, incorporating weekly and monthly temporal scales.
- Implementation of data pre-processing techniques, including noise reduction and missing value imputation, to improve forecasting accuracy.
- Comprehensive performance comparison of statistical and deep learning models using standardized evaluation metrics (MAE, RMSE).
- Identification of the most suitable model for NDVI forecasting based on empirical analysis and real-world applicability.

Section 2 provides the necessary context for this paper. Section 3 and 4 explains the proposed paradigm, as well as the results of the experimentation. Section 5 concludes with a discussion of the findings.

2 Methodology

2.1 NDVI

In order to offer a quantitative measurement of vegetation, the Normalized Difference Vegetation Index (NDVI) [19] measures the difference between near-infrared light, which vegetation strongly reflects, and red light, which vegetation absorbs. The NDVI value is mostly ranging between -1 and +1. It's highly probable that there is water in the ROI when we see negative values. Areas with an NDVI score near +1 are more likely to have thick green foliage. It is therefore plausible that a place with a low NDVI could be urbanized. The interpretation of NDVI values is displayed in Table 1. The index that remote sensing experts utilize the most frequently is the NDVI. The NDVI calculation formulas are presented in Equation 1.

Table 1. Summary of NDVI Values

NDVI	Conclusion
NDVI < 0	Water in the Region of Interest
NDVI Close to 0	Urbanized Area
NDVI Close to 1	Thick/ dense green leaves

$$NDVI = \frac{Band_5 - Band_4}{Band_5 + Band_4} \tag{1}$$

Where, $Band_5$ = Near-Infrared Band

$Band_4$ = Visible Red Band

Landsat 8 produces near-infrared and red light in B_5 and $B_{4,}$ respectively. A farmer might employ NDVI for precision farming and biomass measurement. Forest availability and leaf area index are measured using the NDVI in forestry. In addition, NASA believes that NDVI is a reliable indication of drought conditions. NDVI is being utilised in a number of different fields, as we can see.

2.2 ARIMA Model

Auto Regressive Integrated Moving Average is the abbreviation for ARIMA [20,21]. The ARIMA regression model is widely used for time series forecasting by analysing the relationship between a dependent variable and other changing factors. Instead of focusing on actual values, it predicts future trends by examining differences between data points. ARIMA models help identify variability and ensure non-stationary data is converted into a stationary format for analysis. The model consists of three key components: autoregression (AR), integration (I), and moving average (MA).

AR (Auto Regression): The AR model illustrates a random process by using Auto Regressive (AR). An auto regression is a regression in which the variable is regressed against itself. In other words, to predict future values, we use lagged values of the target variable as input variables. Equation 2 is what an autoregressive model of order p will look like.

$$X_t = \theta_0 + \theta_1 X_{t-1} + \theta_2 X_{t-2} + \theta_3 X_{t-3} + \cdots + \theta_p X_{t-p} \tag{2}$$

The present observed value of "X" is a linear function of its previous p values in the preceding equation. After training, the regression coefficients [θ_0,θ_p] are calculated. The analysis of auto-correlation function (ACF) and partial auto correlation function (PACF) plots is a basic way of determining optimal p values. Using ACF and PACF plots, we can determine historical value dependencies, which in turn help us determine p in AR

I (Integrated): Integrated denotes any differentiation that must be done to make the data steady. The data can be subjected to a Dickey-Fuller test (code provided below) to determine stationarity, and then alternative differencing factors can be tested.

MA (Moving Average): Future values are predicted by moving average models using past prediction error rather than past values, as in a regression-like model. The following Equation 3 illustrates a moving average model:

$$k_t = \theta_0 + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \theta_3 e_{t-3} + \cdots + \theta_q e_{t-q} \tag{3}$$

In this MA(q) model, 'e' represents the random residual differences between the model and the target variable, commonly known as error. When differencing is combined with autoregression and a moving average model, it forms an ARIMA model. The non-seasonal ARIMA model, denoted as ARIMA(p, d, q), is expressed in Equation 4.

$$k_t^{'} = c + \theta_1 k'_{t-1} + \theta_2 k'_{t-2} + \theta_3 k'_{t-3} + \cdots + \theta_p k^{'}{}_{t-p} + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \theta_3 e_{t-3} + \cdots + \theta_q e_{t-q} \tag{4}$$

Where k_t ' is differenced series, 'p', 'd', 'q' represents number of AR terms, number of first order differences, number of MA terms respectively.

Using Backshift Operator :
ARIMA (1,1,1) can be written as

$$k_t^{'} = c + \theta_1 k'_{t-1} + \theta_1 e_{t-1} + e_t$$

$$k_t^{'} = k_t - k_{t-1}$$

$$k_{t-1}^{'} = k_{t-1} - k_{t-2}$$

$$k_t - k_{t-1} = c + \theta_1 \left(k_{t-1} - k_{t-2}\right) + \theta_1 e_{t-1} + e_t$$

$$k_{t-1} = Bk_t$$

$$k_{t-2} = B^2 k_t$$

$$e_{t-1} = Be_t$$

$$k_t - Bk_t = c + \theta_1 Bk_t - \theta_1 B^2 k_t + \theta_1 e_{t-1} + e_t$$

$$=> k_t - Bk_t - \theta_1 Bk_t + \theta_1 B^2 k_t = c + \theta_1 Be_t + e_t$$

$$=> k_t \left(1 - \theta_1 B\right) - Bk_t \left(1 - \theta B\right) = c + \left(1 + \theta_1 B\right) e_t$$

$$=> \left(1 - \theta_1 B\right)\left(k_t - Bk_t\right) = c + \left(1 + \theta_1 B\right) e_t$$

$$=> \left(1 - \theta_1 B\right)\left(1 - B\right) k_t = c + \left(1 + \theta_1 B\right) e_t$$

ARIMA(p,d,q) can be written as
$\left(1 - \theta_1 B - \ldots - \theta_p B^p\right)\left(1 - B\right)^d k_t = c + \left(1 + \theta_1 B + \ldots + \theta_q B^q\right) e_t$

Forecasting :

Point forecasts are derived by expanding the ARIMA equation, shifting time indices, and substituting future values with forecasts while setting future errors to zero. Residuals replace past errors for accurate predictions.

ARIMA(3, 1, 1)

$$\left(1 - \widehat{\theta_1} B - \widehat{\theta_2} B^2 - \widehat{\theta_3} B^3\right)\left(1 - B\right) k_t = \left(1 + \widehat{\theta_1} B\right) e_t$$

$$\left(1-\left(1+\widehat{\theta_1}\right)B+\left(\widehat{\theta_1}-\widehat{\theta_2}\right)B^2+\left(\widehat{\theta}-\widehat{\theta_3}\right)B^3+\widehat{\theta_3}B^4\right]k_t=(1+\widehat{\theta_1}B)e_t$$

Applying backshift operator,

$$k_t-\left(1+\widehat{\theta_1}\right)k_{t-1}+\left(\widehat{\theta_1}-\widehat{\theta}\right)k+\left(\widehat{\theta_2}-\widehat{\theta_3}\right)k_{t-3}+\widehat{\theta_3}k_{t-4}=e_t+\widehat{\theta_1}e_{t-1}$$

$$k_t=\left(1+\widehat{\theta_1}\right)k_{t-1}-\left(\widehat{\theta_1}-\widehat{\theta_2}\right)k_{t-2}-\left(\widehat{\theta_2}-\widehat{\theta_3}\right)k_{t-3}-\widehat{\theta_3}k_{t-4}+e_t+\widehat{\theta_1}e_{t-1}$$

For second step, replace t with T+1.

$$k_{t+1}=\left(1+\widehat{\theta_1}\right)k_t-\left(\widehat{\theta_1}-\widehat{\theta_2}\right)k_{t-1}-\left(\widehat{\theta_2}-\widehat{\theta_3}\right)k_{t-2}-\widehat{\theta_3}k_{t-3}+e_{t+1}+\widehat{\theta_1}e_t$$

2.3 RNN Model

In a deep neural network, there may be several hidden layers. Each hidden layer can be identified by the weights and biases that it possesses. There are variations in both the weights and the biases of these hidden layers. Since every one of these layers operates in a distinct and independent manner, it is impossible for us to merge them all together. When it comes to these hidden layers, we need to ensure that they all have the same weights and biases so that they may be bound together.

We are now able to integrate these layers because the weights and bias of all of the hidden layers have now become consistent with one another. It is possible to bring all of these hidden layers together under a single recurring layer.

RNN Model Recurrent Neural Network (RNN) [22] is a neural network model in which we fed the output from the previous state as input to the current state. The RNN overcomes the issues of traditional neural networks using the help of hidden layers. The length of inputs of this model are variable for a sequence data, i.e. {a_1, a_2, …, a_n} where each t ∈ {1, 2, 3, …, T} and the hidden state Z_t of the RNN updated Equation 5.

$$Z_t=f\left(Z_{t-1},X_t\right) \tag{5}$$

Where f is the activation function. Y_t is the current state. Z_{t-1} is the previous state. X_t is the input state. Training of RNN is similar to back propagation over time. The architecture is represented in Figure 1 and the output of each sequence layer can be expressed from the equation 6 to Equation 9. O_i represents output for the layer i=1 to n, where O_0 is the initial random values. w is the initial input weight matrix and $w^{'}$ is the output weight from each layer. $w^{''}$ is the final weight matrix from the last output layer, which acts as an input to the softmax layer g . $\hat{y}$ is the predicted value.

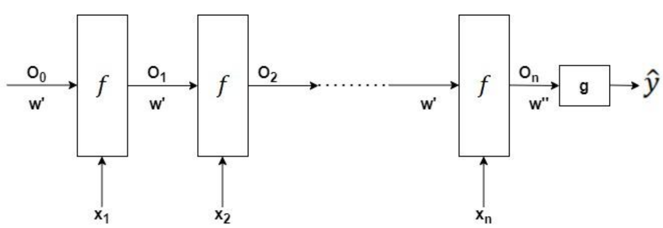


Fig 1. RNN Memory Blocks

The gradients of RNN vanish for long sequences of data and become difficult for optimization of this model.

$$O_1=f\left(wa_1+w^{'}O_0\right) \tag{5}$$

$$O_2=f\left(wa_2+w^{'}O_1\right) \tag{6}$$

$$O_n=f\left(wa_{n-1}+w^{'}O_{n-1}\right) \tag{7}$$

$$\hat{y}=g\left(w^{''}O_n\right) \tag{8}$$

2.4 LSTM Model

Long short-term memory (LSTM)[23] is a subtype of artificial recurrent neural network (RNN) architecture used in the field of deep learning to predict time series data. RNN predicts the future by learning past occurrences in a time series manner and contains a loop to output and input using built-in memory. The RNN's vanishing gradient problem makes the system infeasible in deep neural network, and hence its loss function becomes zero. This gradient problem of RNN can be overcome through gates using the LSTM model.

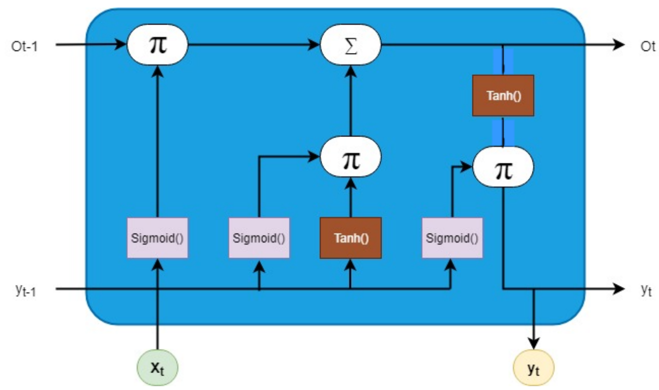


Fig 2. Long Short-Term Memory Block

Figure 2 shows the architecture of the LSTM block. Basically, the LSTM model consists of four neural network layers having three sigmoid() gates activation functions $\alpha_{1,}$ α_2 and $\alpha 3$ and two output tanh() activation functions η_1 and η_2 as in Figure 2 with the order from left to right. The Π symbol represents multiplications of elements, whereas the symbol Σ represents element-wise additions. The symbol (.) bullet represents the concatenation operation. The cell state is the basic LSTM component that goes from the previous state of memory (O_{t-1}) to the current state of memory (O_t). Information in the model flows from the start state to the output through forget gate, input gate and output gate and is controlled by layer α_1 as shown in equation 9. W_f represents the concatenation of weights in forget gates. Information flows in the input gate is represented in equation 11 to Equation 14. The results of output gate are modelled through equation 14 and equation 15. All the notations are taken as per the Figure 2 .

$$f_t = \alpha_1 \left(W_f \cdot (y_{t-1}, a_t] + b_f \right) \tag{9}$$

$$I_t = \alpha_2 \left(W_I \,.\, (y_{t-1}\,, a_t] \,+\, b_I \right) \tag{10}$$

$$\widetilde{O}_t = \left(W_O \,.\, (y_{t-1}\,, a_t] + b_O \right) \tag{11}$$

$$O_t = \left((f_t * O_{t-1}) + (I_t * \widetilde{O}_t) \right) \tag{12}$$

$$C_t = \alpha_3 \left(W_o \,.\, (y_{t-1}\,, a_t] \,+\, b_o \right) \tag{13}$$

$$y_t = C_t * \eta_2 \ (O_t) \tag{14}$$

3 Proposed Model

Proposed model is depicted in Figure 3. First Landsat 8 imagery of Kendrapara district of Odisha is acquired over a time period from 2015 to 2022. This data was collected from Google Earth Engine. Then the shape file of the region of interest is created from which the ROI is clipped. Then the NDVI[24] of ROI is calculated followed by mean NDVI for day of the year. This mean NDVI data can be represented in two ways: finding monthly mean NDVI for five years and finding weakly mean NDVI over the years. 80-20 train test split applied on both the dataset. Different time series models are fitted to train data and validated with test data. Finally, forecasting is done using different Time series model (ARIMA, LSTM, and RNN). The detail approach is discussed in following sections.

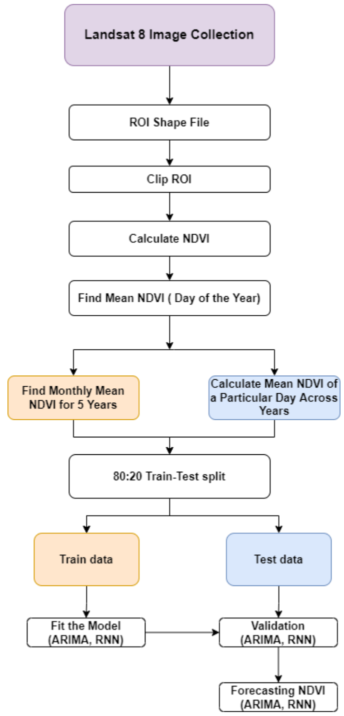


Fig 3. Proposed Model

3.1 Study Area and Data Collection

This study focuses on Kendrapara district in Odisha, India, located at 20.4969° N, 86.4289° E, and bordered by Bhadrak, Jajpur, Jagatsinghpur, Cuttack, and the Bay of Bengal. Originally part of Cuttack, Kendrapara became a separate district on April 1, 1993. Satellite imagery from 2015 to 2022 was analyzed using Google Earth Engine to examine the region's spatial characteristics. Figure 4 depicts the general layout of our research space.

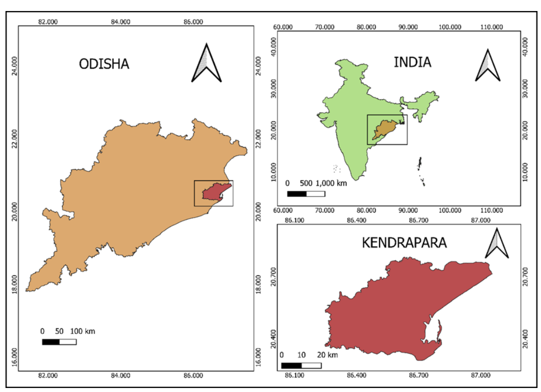


Fig 4. Location of Study Area

The Landsat 8 satellite image is collected in Google Earth Engine editor by setting the image collection "LANDSAT/LC08/C01/T1" with filter date in between '2015-01-01' and '2022-01-01'. Then NDVI is calculated by using the band B4 and B5. Two sequences are generated by using the ui.Chart.image.doySeries function with mean as the reducer. The first series contains the weekly mean NDVI over all the years. The second series contains the monthly mean NDVI for the years 2015 to 2021. The sample of both the series is shown in Figure 5.

day	ndvi
2020-01-02	0.155
2020-01-05	0.132
...	...
2020-12-27	0.137
2020-12-30	0.149

(Weekly mean NDVI)

day	ndvi
2015-01-31	0.1520
2015-02-28	0.1900
...	...
2021-11-30	0.2465
2021-12-31	0.1305

(Monthly mean NDVI)

Fig 5. Sample of the two series

3.2 Pre-processing

Series 1 contains 53 weeks NDVI mean, whereas series 2 includes 84 months' NDVI. After setting the frequency to "M" the missing values are filled with the front value. Series 1 and Series 2 are shown in Figure 6 and Figure 7, respectively.

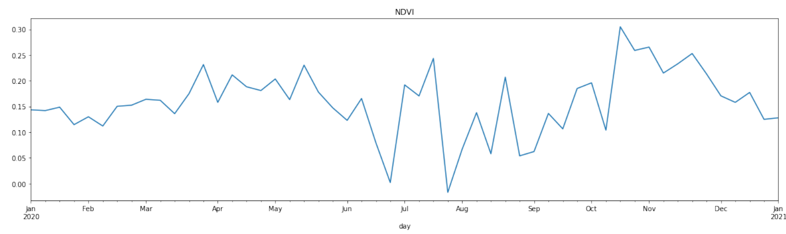


Fig 6. Weekly Mean NDVI Series

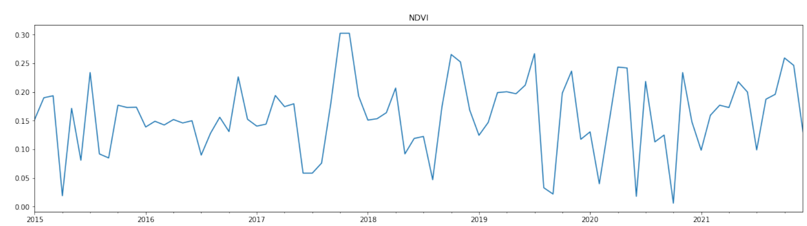


Fig 7. Monthly Mean NDVI Series

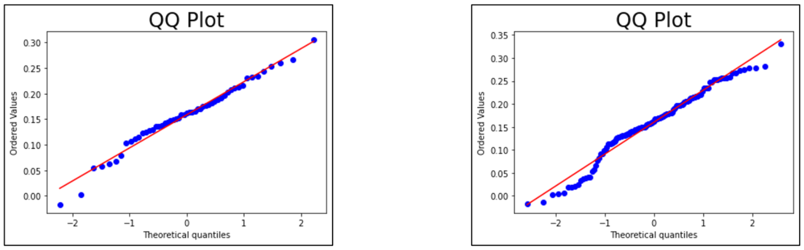
From the QQ plot as shown in Figure 8, it is evident that the data follows a normal distribution.

The series is divided into 80:20 train test ratios. Initial 80% of the data is used for training and the rest 20% of the data is used for testing or validating the model. In series 1, out of 53 numbers of samples 42 samples are used for training, and 11 samples are used for testing. In series 2, 67 samples are used for training, and the rest 17 samples are used for testing.

As the training sample is less to give more data to the testing sample we have taken 80-20 set.

3.3 Stationary Test

The stationary test determines whether the statistical moments are changing in the series or not concerning time. Augmented Dickey-Fuller indicates that the data is stationary as the P-value is less than 0.05 in both series. Figure 9 and Figure 10 represent the rolling mean and standard deviation of weekly mead NDVI and Monthly mean NDVI, respectively. Visually it is also clear that there is no change in the first- and second-order moments over the timestamp.



(a) **(b)**

Fig 8. QQ Plot of (a) Weekly Mean NDVI (b) Monthly Mean NDVI

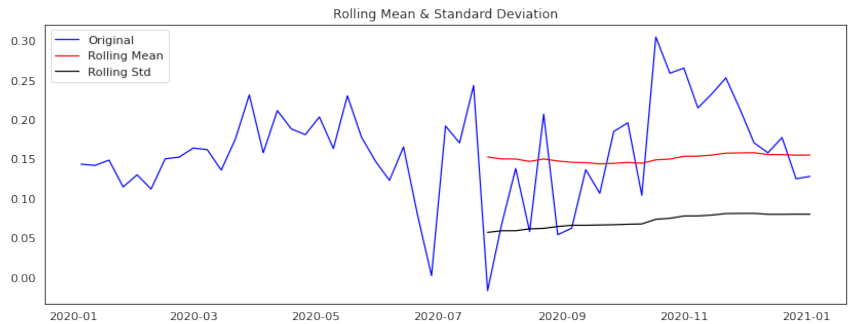


Fig 9. Rolling Mean & Standard Deviation of Weekly Mean NDVI

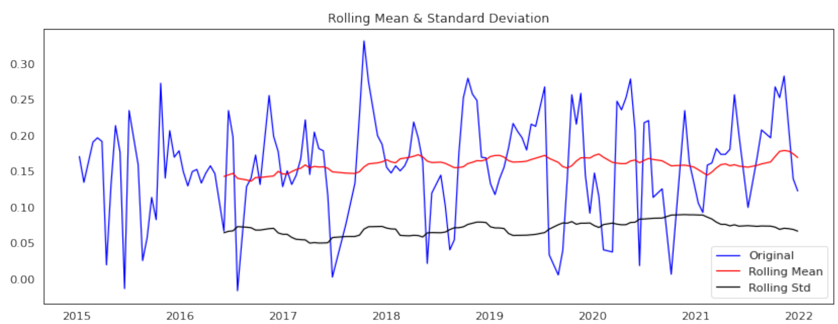


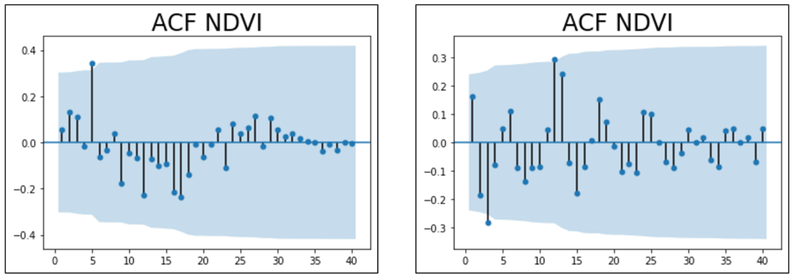
Fig 10. Rolling Mean & Standard Deviation of Monthly Mean NDVI

3.4 Order of ARIMA Model

The order of the ARIMA model is determined by three parameters (p, d, q), where p, d, and q represent the number of AR, differencing and MA terms, respectively. These parameters are set on a hit-and-trial basis by considering the lower Akaike information criterion (AIC) value. A lower AIC value indicates the best fit of the data. Initially, the order p is set by looking into the PACF plot. The Partial Autocorrelation function reveals the significant lag values. ACF and PACF plot of weekly and monthly mean NDVI are shown in Figure 11 and Figure 12 respectively.

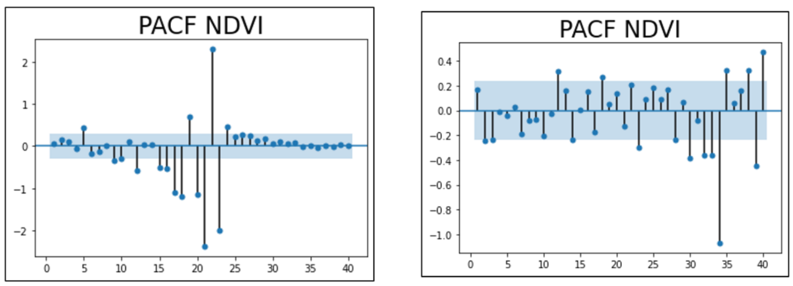
4 Results and Discussion

The effectiveness of the proposed forecasting models—ARIMA, LSTM, and RNN—was assessed using Mean Absolute Error (MAE)[(25)] and Root Mean Square Error (RMSE)[(26)] as performance metrics. Two NDVI time-series datasets, weekly mean NDVI and monthly mean NDVI, were used for model training and validation. Table 2 summarizes the error metrics obtained for each model. A lower value indicates a better model.



(a) **(b)**

Fig 11. ACF Plot of (a) Weekly Mean NDVI (b) Monthly Mean NDVI

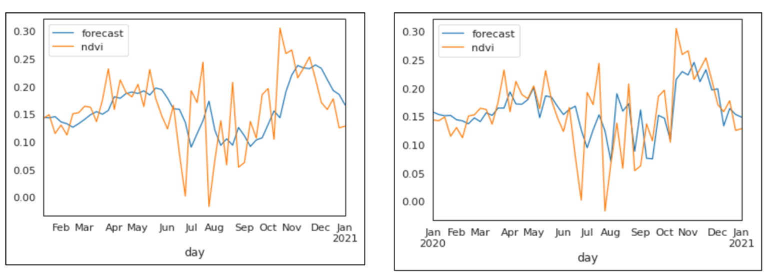


(a) **(b)**

Fig 12. PACF Plot of (a) Weekly Mean NDVI (b) Monthly Mean NDVI

Results of ARIMA

In the first series, the results of ARIMA (1, 1, 1) and ARIMA (5, 0, 5) are shown in Figure 13, where the in-sample lagged values are used for prediction. Comparing the model ARIMA (5, 0, 5) gives significant results.



(a) **(b)**

Fig 13. PACF Plot of (a) Result of ARIMA (1, 1, 1) (b) Result of ARIMA (5, 0, 5)

Figure 14 reflects the forecast result of the test data. Rather than using lagged dependent variables, in-sample forecasts are used here. The confidence intervals for the forecasts are (1 - alpha) %, highlighted in the shaded area. The alpha value is set to 0.5.

In the second series, the results of ARIMA (1, 1, 1) and ARIMA (3, 0, 3) are shown in Figure 15, where the in-sample lagged values are used for prediction. Comparing the model ARIMA (3, 0, 3) gives significant results.

Figure 16 reflects the test data's forecast result in which the alpha value is set to 0.5.

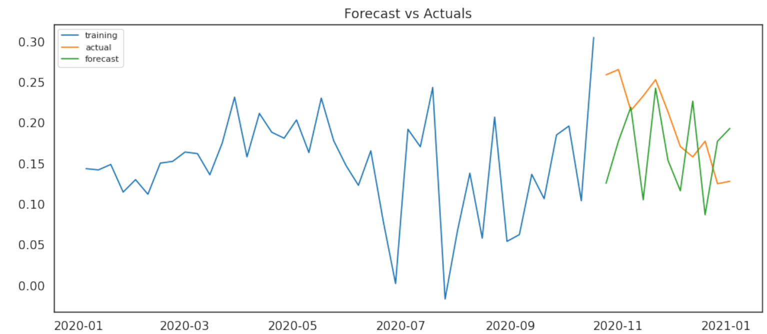
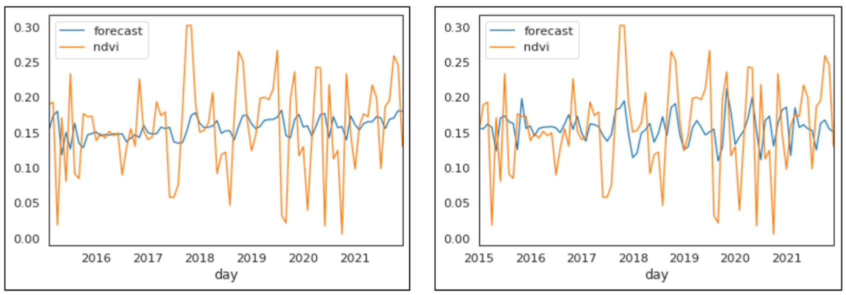


Fig 14. Forecast Result of the Test Data



(a) **(b)**

Fig 15. Forecast Results of (a) ARIMA (1, 1, 1) (b) ARIMA (3, 0, 3)

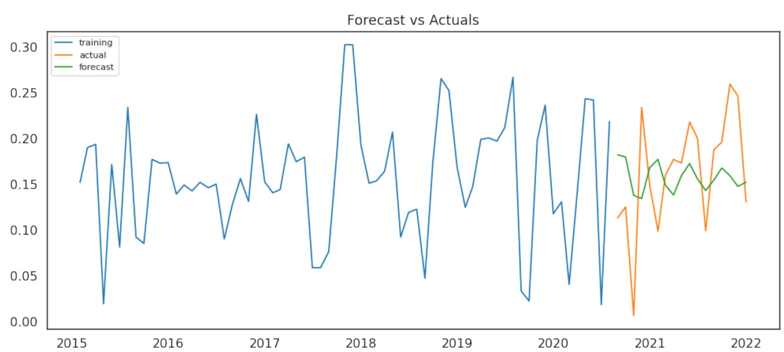


Fig 16. Forecast Result of the Test Data having Alpha = 0.5

Results of Deep Learning Model

The series is also trained using LSTM [23] and RNN [22]. In both cases, a simple architecture is taken to build the model. The first hidden layer consists of 100 neurons, and the output layer consists of one neuron for predicting the target variable. The summaries of the LSTM and RNN models are shown in Figure 17.

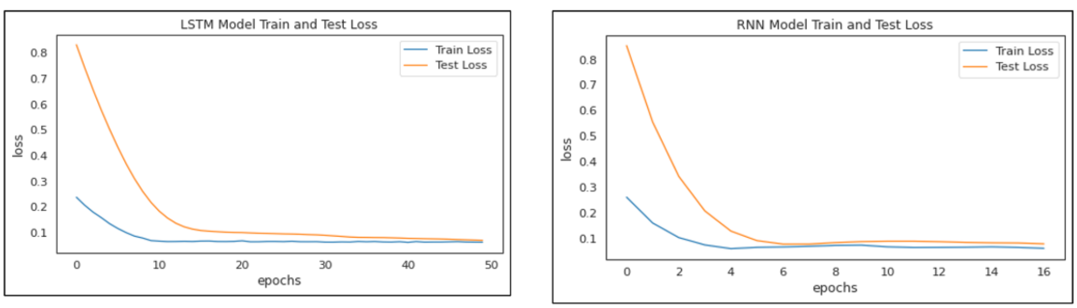
The trainable parameters in LSTM and RNN are 40,901 and 10,301, respectively.

The model is compiled with the mean_squared_error function and adam optimizer. The number of epochs and batch size are set to 50 and 10, respectively. The train and test loss that occurs in LSTM and RNN are depicted in Figure 18.

Several studies, such as the work by Zhang et al., have found that ARIMA-based models excel in short-term NDVI forecasting, particularly when dealing with relatively small datasets [xu2020application]. The performance of the ARIMA, LSTM, and RNN models in our study is consistent with the results reported in the existing literature.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 100)               40800

 dropout (Dropout)           (None, 100)               0

 dense (Dense)               (None, 1)                 101

=================================================================
Total params: 40,901
Trainable params: 40,901
Non-trainable params: 0
_________________________________________________________________
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 simple_rnn (SimpleRNN)      (None, 100)               10200

 dropout_1 (Dropout)         (None, 100)               0

 dense_1 (Dense)             (None, 1)                 101

=================================================================
Total params: 10,301
Trainable params: 10,301
Non-trainable params: 0
_________________________________________________________________
```

Fig 17. Summary of LSTM and RNN Model



(a) **(b)**

Fig 18. Train and Test Loss of (a) LSTM Model (b) RNN Model

Table 2. Error Analysis of Time Series Model

	Weekly mean NDVI (Series 1)		Monthly mean NDVI (Series 2)	
	MAE	RMSE	MAE	RMSE
ARIMA (5,0,5)	**0.0399**	**0.0484**	-	-
ARIMA(3,0,3)	-	-	**0.0522**	**0.0645**
RNN	0.0435	0.0527	0.0549	0.0649
LSTM	0.0686	0.0789	0.0546	0.0661

The results obtained in this study indicate that ARIMA outperforms deep learning models (LSTM, RNN) in short-term (weekly) NDVI forecasting, while LSTM and RNN perform better for long-term (monthly) NDVI predictions. This is consistent with the findings of Carreño-Conde et al. (2021), who demonstrated the importance of NDVI forecasting for agricultural monitoring, but whose work focused primarily on seasonal predictions rather than high-resolution weekly forecasts [carreno2021forecast]. Zhang et al. compared the performance of ARIMA and machine learning models for NDVI forecasting in a semi-arid region [xu2020application].

The superior performance of ARIMA in short-term vegetation forecasting reinforces the need for hybrid forecasting strategies that integrate both statistical and deep learning models. Regarding deep learning models, previous research has indicated that LSTM and RNN are more effective in capturing long-term, non-linear patterns in NDVI data. [fan2023research]

Similarly, our results align with Pande et al. (2024), who applied machine learning-based LST forecasting using Landsat-8 data and found that deep learning models perform well for long-term predictions [pande2024predictive]. However, our study provides a unique perspective by evaluating multiple forecasting approaches on the same NDVI dataset and demonstrating that ARIMA remains a highly competitive choice for short-term forecasting despite the growing preference for deep learning models.

5 Conclusion

This study presents a comparative evaluation of ARIMA, LSTM, and RNN models for NDVI forecasting using Landsat 8 data over the Kendrapara district, focusing on both weekly and monthly NDVI variations. The results demonstrate that ARIMA excels in short-term forecasting, achieving lower error rates compared to deep learning models. This finding challenges the common assumption that deep learning models always outperform statistical methods. Meanwhile, LSTM and RNN models perform better in long-term forecasting, capturing complex non-linear vegetation patterns. The study highlights the importance of selecting forecasting models based on temporal scale rather than assuming a one-size-fits-all approach.

Additionally, this research introduces key innovations by integrating advanced data preprocessing techniques, including noise reduction and missing value imputation, to enhance model accuracy. Unlike previous studies that focus on either ARIMA or deep learning models in isolation, this work directly compares both approaches across different temporal resolutions. The study proposes a hybrid forecasting strategy, where ARIMA is used for short-term monitoring and LSTM is applied for long-term vegetation analysis. This novel integration of methods provides a more comprehensive and adaptable approach to NDVI forecasting.

Moving forward, future research should focus on developing hybrid models that combine statistical and deep learning approaches for enhanced performance. Expanding the dataset by incorporating additional remote sensing sources can further improve model generalization. Moreover, geospatial modeling techniques should be explored to analyze spatial dependencies in NDVI patterns. By addressing these directions, this study advances NDVI forecasting methodologies and provides valuable insights for environmental monitoring, precision agriculture, and climate change adaptation strategies.

References

1) Patil H, Bolla BK, Sabeesh E, Bhumireddy DR. Comparative Study of Predicting Stock Index Using Deep Learning Models. *International Conference on Cognitive Computing and Cyber Physical Systems*. 2023;p. 45–57. Available from: https://arxiv.org/abs/2306.13931.
2) Abd-Elmabod SK, Fitch AC, Zhang Z, Ali RR, Jones L. Rapid urbanisation threatens fertile agricultural land and soil carbon in the Nile delta. *Journal of Environmental Management*. 2019;252. Available from: https://www.sciencedirect.com/science/article/abs/pii/S0301479719313866.
3) Vohra R, Kumar A, Jain R, Hemanth DJ. Analysis and prediction of land surface temperature with increasing urbanisation using satellite imagery. *Heliyon*. 2024;10(22). Available from: https://doi.org/10.1016/j.heliyon.2024.e40378.
4) Pande CB, Egbueri JC, Costache R, Sidek LM, Wang Q, Alshehri F, et al. Predictive modeling of land surface temperature (LST) based on Landsat-8 satellite data and machine learning models for sustainable development. *Journal of Cleaner Production*. 2024;444. Available from: https://doi.org/10.1016/j.jclepro.2024.141035.
5) Karakuş CB. The Impact of Land Use/Land Cover (LULC) Changes on Land Surface Temperature in Sivas City Center and Its Surroundings and Assessment of Urban Heat Island. *Asia-Pacific Journal of Atmospheric Sciences*. 2019;55:669–684. Available from: https://link.springer.com/article/10.1007/s13143-019-00109-w.
6) Fatemi M, Narangifard M. Monitoring LULC changes and its impact on the LST and NDVI in District 1 of Shiraz City. *Arabian Journal of Geosciences*. 2019;12. Available from: https://link.springer.com/article/10.1007/s12517-019-4259-6.
7) Carreño-Conde F, Sipols AE, De Blas CS, Mostaza-Colado D. A forecast model applied to monitor crops dynamics using vegetation indices (NDVI). *Applied Sciences*. 2021;11(4). Available from: https://doi.org/10.3390/app11041859.
8) Zhang Q, Wang D, Fu D, Gong F, He X, Wang Y. Characterization of Landsat-8 and Landsat-9 reflectivity and NDVI continuity based on Google Earth Engine. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. 2024;18:621–641. Available from: https://ieeexplore.ieee.org/document/10745744.
9) Jacob P, Kommuru P, Ruchitha R, Kuracha H, Taruni T. Comparative study of MODIS, LANDSAT-8, SENTINEL-2B, and LISS-4 images for Precision farming using NDVI approach. *E3S Web of Conferences*. 2023;405:1–11. Available from: https://doi.org/10.1051/e3sconf/202340501004.

10) Johnson DM, Rosales A, Mueller R, Reynolds C, Frantz R, Anyamba A, et al. USA crop yield estimation with MODIS NDVI: Are remotely sensed models better than simple trend analyses? *Remote Sensing*. 2021;13(21). Available from: https://doi.org/10.3390/rs13214227.
11) Contreras-Reyes JE, Palma W. Statistical analysis of autoregressive fractionally integrated moving average models in R. *Computational Statistics*. 2013;28:2309–2331. Available from: https://link.springer.com/article/10.1007/s00180-013-0408-7.
12) Moskola WR, Abdou W, Dipanda A, Kolyang. Application of deep learning architectures for satellite image time series prediction: A review. *Remote Sensing*. 2021;13(23). Available from: https://www.mdpi.com/2072-4292/13/23/4822.
13) Raza D, Khushi M, SHU H, Aslam H, Saleem MS, Ahmad A, et al. CA-ANN based LULC prediction and influence assessment on LST-NDVI using multi-temporal satellite images. *Environmental Earth Sciences*. 2024;83. Available from: https://link.springer.com/article/10.1007/s12665-024-11467-8.
14) Sharma I, Tongkumchum P, Ueranantasun A. Regression Analysis of Normalized Difference Vegetation Index (NDVI) to Compare Seasonal Patterns and 15 Year Trend of Vegetation from East to West of Nepal. *Nature Environment and Pollution Technology*. 2021;20(1):267–273. Available from: https://www.researchgate.net/publication/350238754_Regression_Analysis_of_Normalized_Difference_Vegetation_Index_NDVI_to_Compare_Seasonal_Patterns_and_15_Year_Trend_of_Vegetation_from_East_to_West_of_Nepal.
15) Zhang Y, He Y, Li Y, Jia L. Temporal and Spatial Changes and Driving Forces of NDVI From 1982 - 2015 in Qinba Mountains, China. *Research Square*. 2021;p. 1–37. Available from: https://assets-eu.researchsquare.com/files/rs-415051/v1_covered.pdf?c=1631867160.
16) Chu D, Shen H, Guan X, Chen JM, Li X, Li J, et al. Long time-series NDVI reconstruction in cloud-prone regions via spatio-temporal tensor completion. *Remote Sensing of Environment*. 2021;264. Available from: https://www.sciencedirect.com/science/article/abs/pii/S0034425721003527.
17) Gao P, Du W, Lei Q, Li J, Zhang S, Li N. NDVI forecasting model based on the combination of time series decomposition and CNN–LSTM. *Water Resources Management*. 2023;37:1481–1497. Available from: https://link.springer.com/article/10.1007/s11269-022-03419-3.
18) Albeladi K, Zafar B, Mueen A. Time Series Forecasting using LSTM and ARIMA. *International Journal of Advanced Computer Science and Applications*. 2023;14(1):313–320. Available from: https://thesai.org/Downloads/Volume14No1/Paper_33-Time_Series_Forecasting_using_LSTM_and_ARIMA.pdf.
19) Chaubey PK, Kundu A, Mall RK. A geo-spatial inter-relationship with drainage morphometry, landscapes and NDVI in the context of climate change: a case study over the Varuna river basin (India). *Spatial Information Research*. 2019;27(6):627–641. Available from: https://dx.doi.org/10.1007/s41324-019-00264-2.
20) Yan J, Chen X, Chen Y, Liang D. Multistep Prediction of Land Cover from Dense Time Series Remote Sensing Images with Temporal Convolutional Networks. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. 2020;13:5149–5161. Available from: https://ieeexplore.ieee.org/document/9184116?denied=.
21) Panuju DR, Paull DJ, Griffin AL. Change Detection Techniques Based on Multispectral Images for Investigating Land Cover Dynamics. *Remote Sensing*. 2020;12(11). Available from: https://dx.doi.org/10.3390/rs12111781.
22) Lenco D, Gaetano R, Dupaquier C, Maurel P. Land Cover Classification via Multitemporal Spatial Data by Deep Recurrent Neural Networks. *IEEE Geoscience and Remote Sensing Letters*. 2017;14(10):1685–1689. Available from: https://ieeexplore.ieee.org/document/8006221.
23) Reddy DS, Prasad PRC. Prediction of vegetation dynamics using NDVI time series data and LSTM. *Modeling Earth Systems and Environment*. 2018;4:409–419. Available from: https://dx.doi.org/10.1007/s40808-018-0431-3.
24) Borgogno-Mondino E, Lessio A, Gomarasca MA. A fast operative method for NDVI uncertainty estimation and its role in vegetation analysis. *European Journal of Remote Sensing*. 2016;49(1):137–156. Available from: https://dx.doi.org/10.5721/eujrs20164908.
25) Jamali A. Evaluation and comparison of eight machine learning models in land use/land cover mapping using Landsat 8 OLI: a case study of the northern region of Iran. *SN Applied Sciences*. 2019;1. Available from: https://dx.doi.org/10.1007/s42452-019-1527-8.
26) Talukdar S, Singha P, Mahato S, Shahfahad, Pal S, Liou YA, et al. Land-Use Land-Cover Classification by Machine Learning Classifiers for Satellite Observations—A Review. *Remote Sensing*. 2020;12(7). Available from: https://dx.doi.org/10.3390/rs12071135.