Dengue Prediction Using Remote Sensing and Machine Learning

Dr Mohammad Monirujjaman Khan, Syed Niamul Kazbe Rayian, Kazi Farhan Shariar
Dept of Computer Science and Engineering, North South University, Bangladesh
Dept of Computer Science and Engineering, North South University, Bangladesh
Dept of Computer Science and Engineering, North South University, Bangladesh
Email: monirujjaman.khan@northsouth.edu, syed.rayian@northsouth.edu, farhan.shariar@northsouth.edu

***Abstract*— Our project focuses on predicting dengue outbreaks four weeks in advance to anticipate and mitigate potential calamities. By leveraging machine learning techniques and incorporating remote sensing data such as NDVI, temperature, rainfall, humidity, wind, solar radiation, and dew, we aim to forecast dengue occurrences effectively. The dataset underwent a thorough examination for missing values, followed by basic statistical analysis and visualization to understand feature relationships. After log-transforming the features for improved analysis, stratified sampling was used to create training, validation, and testing datasets. This comprehensive approach aims to enhance our ability to predict and respond to dengue outbreaks.**

Keywords— LSTM, SVR, Random Forest, Bidirectional LSTM, Lasso, Linear Regression, SimpleRNN, NDVI, TRMM, EVI, hyperparameter.

I. INTRODUCTION

Dengue is a viral infection caused by the dengue virus, which is primarily transmitted to humans through the bite of infected Aedes mosquitoes, primarily Aedes aegypti. It is a significant public health concern in many tropical and subtropical regions around the world, particularly in urban and semi-urban areas. The symptoms of dengue can vary widely, ranging from mild flu-like symptoms to severe and potentially life-threatening manifestations. "Arboviral disease is a general term used to describe infections caused by a group of viruses spread to people by the bite of infected arthropods (insects) such as mosquitoes and ticks. These infections usually occur during warm weather months, when mosquitoes and ticks are active." [1] While there is a wide array of viruses, not all possess the ability to induce arboviral diseases. An arbovirus necessitates the capacity to reside within an arthropod vector, be conveyed through a bite, and subsequently reproduce to prompt illness in humans. Numerous arboviruses are present in mammals or birds. When an insect bites one of these animals, it becomes infected and subsequently transmits the ailment to humans. Dengue is the most common arboviral disease, as it is transmitted by Aedes mosquitoes, and infections are more common in Asia, South America, and Africa. About half of the world's population is at risk of dengue, and dengue affects approximately 129 countries. It has been estimated that about 100 to 400 million cases are reported every year, and the American region alone has reported about 2.8 million cases and 101,280 deaths. [2] This year, in 2023, record-breaking dengue cases have occurred in the countries of Asia, where Bangladesh is facing its worst dengue fever outbreak in five years and has seen a huge spike in hospitalizations. Thailand, Nepal, India, Pakistan, Malaysia, and Cambodia are also seeing a surge in dengue cases and reporting much higher case numbers compared with 2022. This year, from January 1 to August 7, 2023, the Ministry of Health and Family Welfare of Bangladesh reported a total of 69,483 laboratory-confirmed dengue cases and 327 related deaths, with a case fatality rate (CFR) of 0.47%. Of these, 63% of cases and 62% of deaths were reported in the month of July 2023. Although dengue is endemic in Bangladesh, the current dengue surge is unusual in terms of seasonality and the early sharp increase in comparison to previous years, where the surge started around late June. The CFR so far this year is relatively high compared to previous years for the full-year period. The pre-monsoon Aedes survey shows that the density of mosquitoes and the number of potential hotspots are at their highest levels in the past five years. The higher incidence of dengue is taking place in the context of an unusual episodic amount of rainfall, combined with high temperatures and high humidity, which have resulted in an increased mosquito population throughout Bangladesh. [3] Remote sensing is the process of collecting information about the Earth's surface or other objects from a distance, typically using specialized sensors and instruments on aircraft, satellites, drones, or other remote platforms. It involves the capture of data without physical contact with the objects or areas being observed. Remote sensing is widely used in various fields, including environmental monitoring, agriculture, forestry, urban planning, disaster management, and geology, among others. It relies on sensors and instruments capable of detecting various forms of electromagnetic radiation, such as visible light, infrared, and microwave. These sensors are often mounted on satellites, aircraft, drones, or ground-based platforms. Different types of objects and materials reflect, emit, or interact with electromagnetic radiation in unique ways. By analyzing the electromagnetic spectrum, remote sensing systems can gather information about the properties and characteristics of the target objects or surfaces. Furthermore, remote sensing systems collect data by recording the electromagnetic radiation reflected or emitted by the Earth's surface or other objects. This data is typically captured in the form of images or digital measurements. However, once the data is collected, it is processed and analyzed to extract useful information. Various techniques, including image processing, spectral analysis, and pattern recognition, are used to interpret the data and generate meaningful results. Remote sensing and machine learning are often combined to enhance the analysis and interpretation of remote sensing data. Machine learning techniques can automate and improve the extraction of valuable information from large and complex remote sensing datasets. One of the primary applications of machine learning in remote sensing is image classification. Machine learning models, such as convolutional neural networks (CNNs), can be trained to classify pixels or image segments into predefined classes or categories. For example, satellite images can be classified to identify land cover types like forests, urban areas, water bodies, and agriculture. An assessment of remotely sensed environmental variables on dengue epidemiology in Central India by Devojit Kumar Sarma, et al. Here, the study area was Bhopal, the capital city of Madhya Pradesh state in

Central India. The city is surrounded by dense to open scrub forests, and the major land use types are sparse to high-density built-ups, water bodies, agricultural croplands, and a range of fallow/wastelands, which is a preferable condition for Aedes spreading. The paper stated that dengue is diagnosed primarily based on clinical manifestations (high fever, headache, retro-orbital pain, myalgia, arthralgia, rash, and hemorrhagic manifestations) and laboratory diagnosis and the collection of epidemiological data. Then, for the meteorological data extraction, the parameters were extracted from the National Aeronautics and Space Administration (NASA) Langley Research Center Prediction of Worldwide Energy Resource Project, which is based on the Modern-Era Retrospective Analysis for Research and Applications version 2 (MERRA-2) dataset merged with Goddard Earth Observing System Model (GEOS) version 5.12.4. The multi-sensor satellite images were extracted through Google Earth Engine and ArcGIS 10.2 software to assess the change in land use and land cover (LULC) and vegetation index. Landsat-8 Operational Land Imagers (OLI) and Thermal Infrared Sensors (TIRS) images (S1 Table) were extracted for preparing LULC maps. The Normalized Difference Vegetation Index (NDVI) is a numerical quantity derived from reflectance measured in the near-infrared (NIR) and red spectral bands. Collected epidemiological and geographical data were converted into spatial layers and aggregated to the ward boundaries of Bhopal city using ArcGIS 10.2 software. The ward-wise cases from 2012 to 2019 were analyzed and normalized with the wards' total population (as per the 2011 census). As a result, spatial distribution showed the wards located in the southern and eastern parts of the city to be significant dengue hotspots. A data-driven epidemiological prediction method for dengue outbreaks using local and remote sensing data by Anna L. Buczak, et al. Temperature is frequently employed due to its impact on biological factors like the mosquito's extrinsic incubation period. Since mosquitoes rely on water for their life cycle, researchers have explored the utility of rainfall information in forecasting disease outbreaks. This rainfall data can be obtained locally or through satellite measurements. The Tropical Rainfall Measuring Mission (TRMM) satellite data have been used to derive rainfall measurements in remote and resource-limited regions, and these measurements have been used for predictions for disease outbreaks. Commonly used leaf area indices are the Normalized Difference Vegetation Index (NDVI) and the Enhanced Vegetation Index (EVI). Both are available from satellite sensors such as the Advanced Very High-Resolution Radiometer (AVHRR) and the Moderate Resolution Imaging Spectrometer (MODIS). NDVI is closely related to photosynthesis, while EVI is closely related to leaf display. Because climate effects, such as the El Nino Southern Oscillation (ENSO), can indicate near-term future rainfall anomalies, the Southern Oscillation Index (SOI) and various sea surface temperature anomalies (SSTA) have also been used as indicators of future disease outbreaks. Rainfall data with a 0.25-degree resolution was collected from the satellite measurements of the NASA Tropical Rainfall Measuring Mission (TRMM). Spatial aggregation of rainfall data from 0.25-degree grid cells to districts was performed by first computing a weight equivalent to the estimated proportions of a district comprised by each grid cell. It is then used with each district's set of weights and the gridded rainfall values for a given week to determine a single rainfall amount for each district during each week. The dengue prediction was formulated using the following: Definition of spatiotemporal resolution and data preprocessing to fit that resolution. Division of the data set into disjoint training, validation, and test subsets. Rule extraction from training data using fuzzy association rule mining (FARM). Automatic building of classifiers from the rules extracted in the previous step. Choice of the best classifier based on its performance on the validation data set. Computation of predictions on the test data using the classifier from the previous step. Calculation of performance metrics. As a result, three different prediction models (classifiers) were automatically built using the methodology developed. The first two are weekly, i.e., we predicted either high or low dengue incidence for a given future week (T+3 and T+4). The third prediction encompassed a four-week period, specifically four to seven weeks from the time of prognosis (T+4 to T+7). This is a single prediction for whether the dengue incidence rate will be low or high over the entire four-week period.

II. METHODOLOGY

In this section, all methods and materials, the dataset feature's description, block diagram, flow diagram, and evaluation matrices of the system are discussed.

A. Dataset

The main goal of the system is to predict dengue cases in advance based on various attributes. Remote sensing data from Bangladesh is used for the development of the system. The dataset has a dimension of 1461 rows vs. 9 columns, which is relatively small in the field of machine learning. Table 1 shows the nine different attributes: year, month, week, case, rainfall, temperature, humidity, wind, and solar radiation that have been considered for the final outcome.

Year	Month	Week	Case	Rainfall	Temperature	Humidity	Wind	Solarradiation
2020	Jan	1	3	0	21.6	67.5	7.6	168.9
2020	Jan	1	17	2	21.8	73.5	11.2	153.2
2020	Jan	1	20	18	19.8	90.5	18.4	105.2
2020	Jan	1	39	0	20.5	82	9.4	127

Table 1: Dengue Dataset of proposed machine learning

For range values, a log transformation method has been applied for scaling the dataset. Log transformation helps to normalize the distribution of data, it stabilizes the variance of the data, which is an assumption of many statistical tests, by reducing the variability across the range of values, log transformation helps to satisfy the assumption of homoscedasticity (constant variance) in regression analysis. The formula is as follows: $\ln xt = \ln b + at$.

Features	Description
Year	The calendar year in which the data was recorded.
Month	The month of the year when the data was collected.
Week	The week number within the year when the data was recorded.
Case	The number of reported Dengue cases.
Rainfall	The amount of precipitation measured during the recorded period.
Temperature	The recorded temperature during the specified period.
Humidity	The level of moisture content in the air during the recorded period.
Wind	The speed and direction of the wind during the recorded period.
Solarradiation	The amount of solar radiation received during the recorded period.

Table 2: Dataset description of proposed machine learning system

Table 2 shows the nine features of the dataset which have been considered for the proposed model to predict the dengue cases and the description of the features. Eight of the features are numerical, where Year and Week are discrete, others are continuous, and only Month is nominal categorical.

Block Diagram. Figure 2 shows the block diagram of the machine learning system. The dengue dataset has been used in the system, which contains all the attributes and values. First, the dataset underwent a thorough examination for any missing values, which were found to be absent. Following the handling of missing data, basic statistics were computed for the cleaned dataset. Subsequently, visualization techniques were employed to enhance understanding of the relationships between the features. Among the attributes, only one was categorical, and it underwent label encoding before undergoing log transformation. All other features were subjected to log transformation for analysis. To gauge the relationships between attributes, correlation analysis was conducted using a correlation matrix. Performing the correlation matrix, it was indicated that using dengue case as output, the relation of it was closer to Week, Temperature, Humidity and Solar radiation.

Next, the features have been assigned to make the prediction, and the target value has been set so that the model can predict. Then, the dataset was split for training validation and testing. Random sampling has been used for the split, but this creates an imbalance between training and testing split. So, stratified sampling has been applied with a training-validation size of 80% and a testing size of 20%. After that, standardization has been applied to do the scaling of the features. Few of the models have been implemented using the scikit-learn library and the others are using keras library.

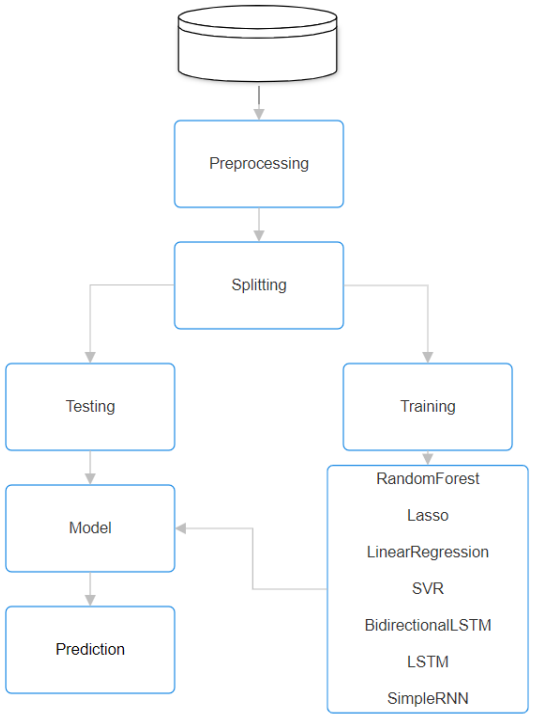

Figure 1: Block Diagram

B. Machine Learning Models

2.1. RandomForest Flowchart. Figure 2 shows the flow diagram of the whole random forest model. It is a collection of some decision trees. The process is the same as the decision tree. It preprocesses the data and selects some random samples from the dataset for training. For every selected sample, it forms a decision tree. First, the random forest model has been trained without fine-tuning. Then, grid search has been used with 5-fold cross-validation and different parameter combinations such as the number of trees in the random forest (n_estimators), what function to use for the number of features to consider at every split, levels in the tree, and method of selecting samples for training each tree.

2.2 Linear Regression Flowchart.
Figure 3 shows the flow diagram of the whole linear regression model. The process for training a linear regression model is straightforward. Initially, the data is preprocessed to handle missing values, outliers, and feature scaling if necessary. Subsequently, a portion of the dataset is randomly sampled for training the linear regression model. During training, the model learns the coefficients for each feature to fit a linear equation that best represents the relationship between the independent variables and the target variable. Unlike decision trees, linear regression does not involve forming multiple models; instead, it focuses on finding the optimal coefficients through techniques like ordinary least

squares (OLS) or gradient descent. After the initial training, hyperparameter tuning can be performed using methods like grid search or randomized search with cross-validation to find the best combination of parameters such as regularization strength, choice of solver, or feature selection. This iterative process aims to enhance the model's predictive performance and generalization capabilities. To work out the regression line the following values need to be calculated:

$$a = \bar{y} - b\bar{x} \text{ and } b = \frac{S_{xy}}{S_{xx}}$$

The easiest way of calculating them is by using a table.

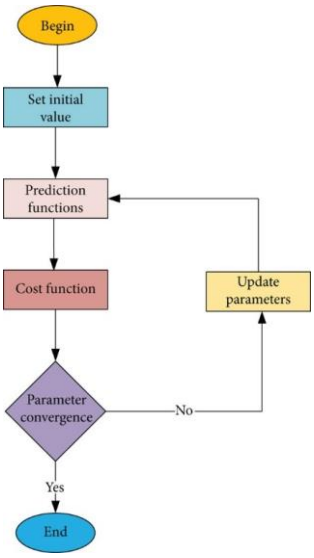

Figure 3: Linear Regression Flowchart

2.3 SVR Flowchart. Figure 4 shows the flow diagram of the whole Support Vector Regression model. The training process for a Support Vector Regression (SVR) model involves several steps. Initially, the data undergoes preprocessing, which may include handling missing values, scaling features, and encoding categorical variables. Following this, a subset of the dataset is randomly selected for training the SVR model. During training, the SVR algorithm constructs a hyperplane or set of hyperplanes in a high-dimensional space, aiming to minimize the error between the predicted and actual target values while respecting a specified margin. Unlike traditional linear regression, SVR considers a margin of tolerance around the regression line, allowing some deviation in predictions. After the initial training, hyperparameter tuning becomes crucial to optimize model performance. Techniques like grid search or randomized search with cross-validation are commonly employed to determine the optimal parameters, such as the choice of kernel function, regularization parameter (C), and kernel coefficient (gamma). This iterative process iteratively refines the model to enhance its predictive accuracy and generalization ability. The regression equation in the feature space can be approximated by:

$$z(a, w) = (w \cdot \phi(a) + c)$$

where, w defines the weight vector, c is a constant, $\phi(a)$ is the feature function and $(w \cdot \phi(a))$ the dot product therein.

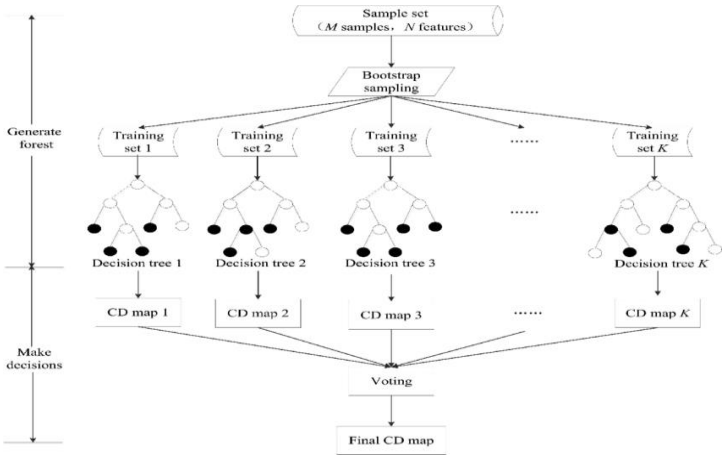

Figure 2: RandomForest Flowchart

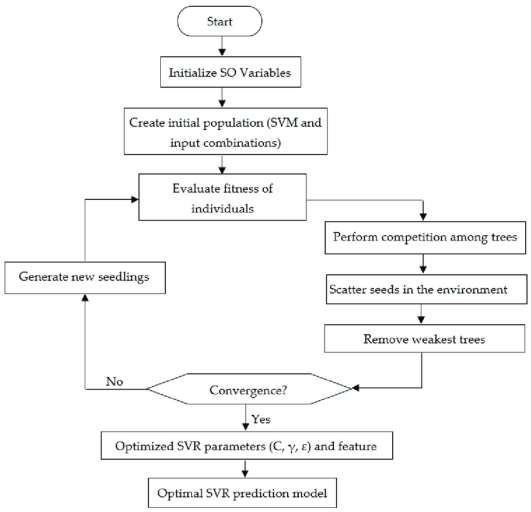

Figure 4: SVR Flowchart

2.4 BidirectionalLSTM Flowchart. Figure 5 shows the flow diagram of the whole BidirectionalLSTM model. Training a Bidirectional Long Short-Term Memory (LSTM) model involves several key steps. Initially, the dataset is preprocessed to handle missing values, scale features, and possibly encode categorical variables. Subsequently, a portion of the dataset is randomly sampled for training the Bidirectional LSTM model. This model architecture consists of two LSTM layers: one processes the input sequence in a forward manner, while the other processes it in reverse.

This bidirectional processing allows the model to capture dependencies in both directions, potentially improving its ability to learn complex temporal patterns. During training, the model learns to predict the target variable based on the sequential input data.

Hyperparameter tuning is often performed to optimize model performance, including parameters such as the number of LSTM units, the choice of activation functions, dropout rates, and learning rates. Techniques like grid search or randomized search with cross-validation are commonly used for this purpose. By iteratively adjusting these parameters, the Bidirectional LSTM model can be fine-tuned to achieve better predictive accuracy and generalization on unseen data.

2.5 LSTM Flowchart. Figure 6 shows the flow diagram of the whole LSTM model. Training a Long Short-Term Memory

(LSTM) model follows a systematic approach. Initially, the dataset undergoes preprocessing, including steps such as handling missing values, feature scaling, and potentially encoding categorical variables. Following this, a subset of the

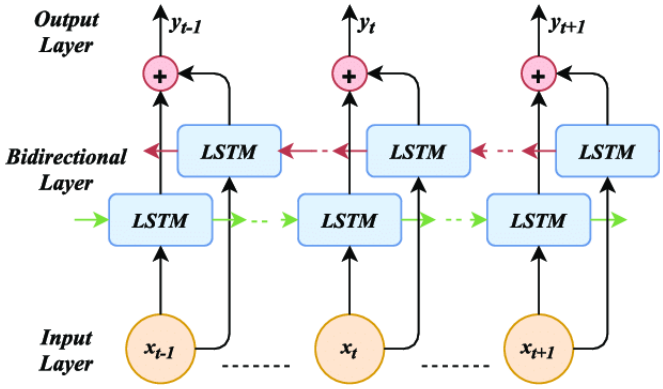

Figure 5: BidirectionalLSTM Flowchart

During training, the LSTM model learns to predict the target variable based on the sequential input data. Hyperparameter tuning plays a crucial role in optimizing model performance, with parameters such as the number of LSTM units, activation functions, dropout rates, and learning rates often adjusted. Techniques like grid search or randomized search with cross-validation are commonly employed for this purpose. By iteratively fine-tuning these parameters, the LSTM model can improve its predictive accuracy and generalization ability on unseen data.

2.6 SimpleRNN Flowchart. Figure 7 shows the flow diagram of the whole SimpleRNN model. Training a Simple Recurrent Neural Network (SimpleRNN) model involves several essential steps. Initially, the dataset is preprocessed, which may include handling missing values, scaling features, and encoding categorical variables. Subsequently, a portion of the dataset is randomly sampled for training the SimpleRNN model. This model architecture typically consists of one or more SimpleRNN layers, capable of capturing sequential dependencies in the input data. During training, the SimpleRNN model learns to predict the target variable based on the sequential input data. Hyperparameter tuning is crucial for optimizing model performance, with parameters such as the number of SimpleRNN units, activation functions, dropout rates, and learning rates often adjusted. Techniques like grid search or randomized search with cross-validation are commonly used for this purpose. By iteratively fine-tuning these parameters, the SimpleRNN model can enhance its predictive accuracy and generalization ability on unseen data.

$$h = \sigma(UX + Wh\text{-}1 + B)$$

$$Y = O(Vh + C)$$

Unlike Deep neural networks where we have different weight matrices for each Dense network in RNN, the weight across the network remains the same. It calculates state hidden state H_i for every input X_i. By using the above formulas. Here S is the State matrix which has element si as the state of the network at timestep i. The parameters in the network are W, U, V, c, b which are shared across timestep.

dataset is randomly selected for training the LSTM model. This model architecture typically consists of one or more LSTM layers, each capable of capturing long-term dependencies in sequential data.

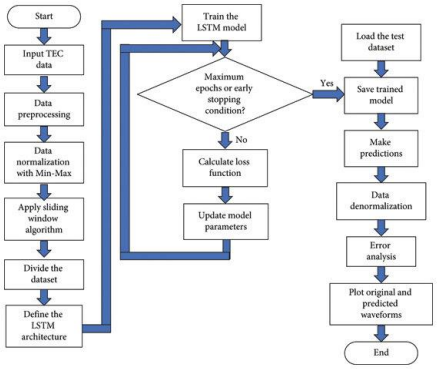

Figure 6: Long Short-Term Memory Flowchart

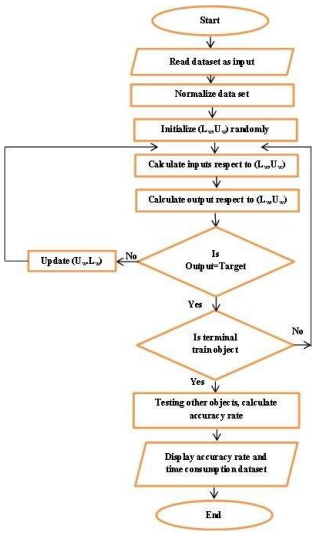

Figure 7: SimpleRNN Flowchart

C. *Dataset Preprocessing*

In our dataset, we employ seven different models, each presenting its own challenges. For instance, the random forest model cannot handle null values, necessitating thorough pre-processing of our dataset. Our dataset is balanced, with no null values, missing values, or duplicates, which we verified at the outset.

Since models do not operate on categorical values and only understand numerical data, we applied label encoding to convert categorical values into numerical ones, followed by log transformation for enhanced analysis.

We utilized MinMaxScaler for feature scaling. Initially, we applied feature scaling, then split our data into training and testing sets with an 80:20 ratio to ensure effective model evaluation.

III. Results and Analysis

The models' functions, model predictions, analysis, and final results are discussed in this section.

A. *Data Visualization*

3.1. Histogram. Figure 7 to 13 shows the scatter graph of the training and validation set. Scatter graph portrays the ratios of the dataset. From Year vs Dengue Cases, it has been

observed that in 2020 there are 23.5%, 2021 has 10.9%, 2022 has 10.6%, and in 2023 there are 55.0% of dengue cases. During the month from June to November, the spread of dengue case increases, and similarly, this pattern is observed after 24th week. There was spike of dengue cases in every year when the humidity was more than 60. In the case of rainfall, 80% of the dengue arise when there is no rainfall or very minimal rainfall. Solar Radiation and temperature provide same kind of pattern where it is observed that aedes mosquito increases when these two variable increases which contributes to more dengue cases. 95% of the dengue cases occur when there is wind less than 50.

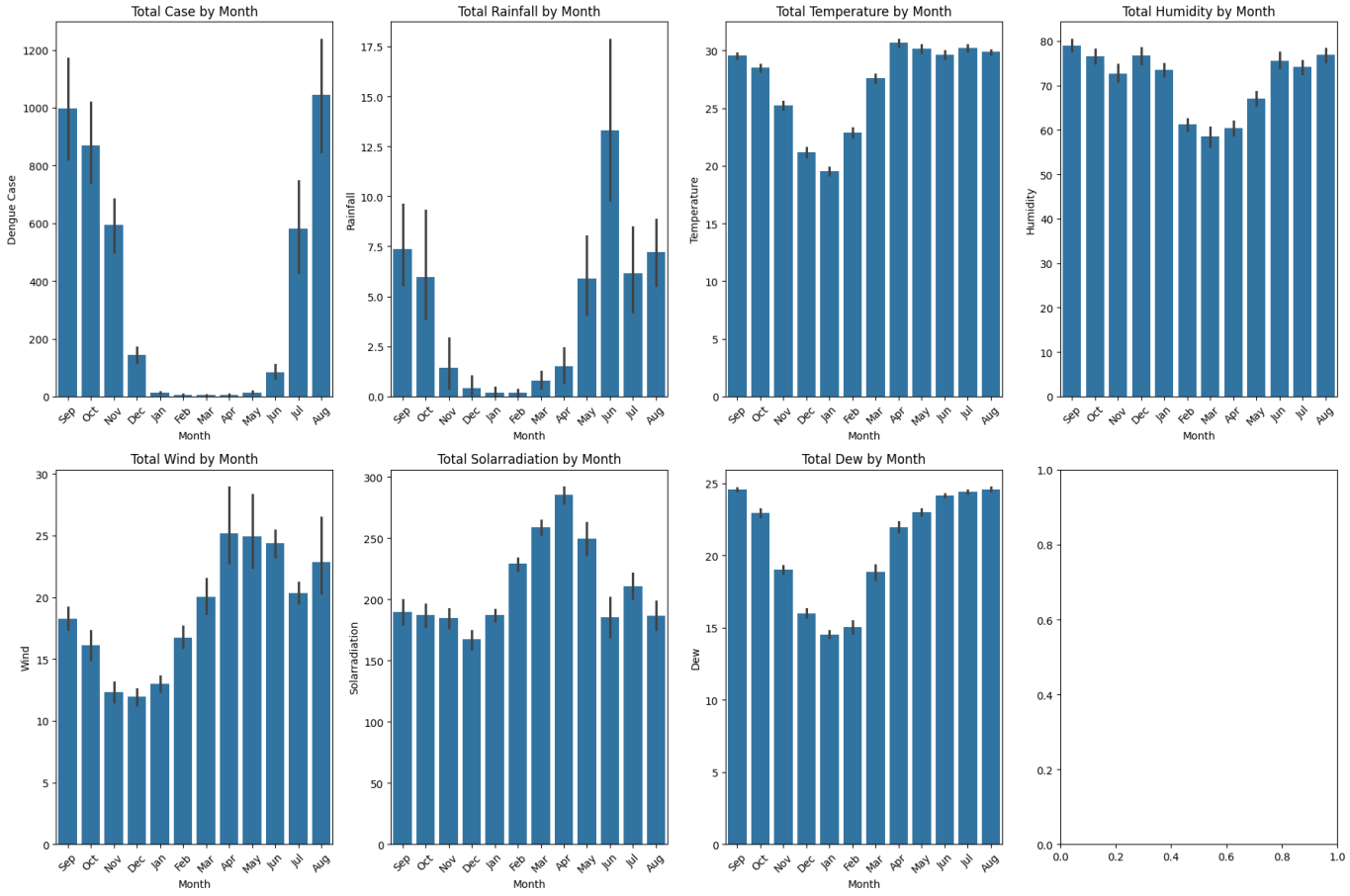

Figure 8: Histogram of training and validation set

B. *Evaluation Matrices*

3.2. Evaluation Matrices. The evaluation metrics for assessing the performance of machine learning models vary based on the task and the nature of the data. Commonly used evaluation metrics for regression models include Mean Squared Error (MSE), R-squared (R2) score, and Mean Squared Logarithmic Error (MSLE). MSE measures the average squared difference between the predicted and actual values, providing an overall assessment of prediction accuracy. R2 score quantifies the proportion of the variance in the target variable that is predictable from the independent variables, with higher values indicating a better fit of the model to the data. MSLE evaluates the mean squared logarithmic difference between the predicted and actual values, which is particularly useful when the target variable spans a wide range of values. By considering these evaluation metrics, practitioners can comprehensively assess the performance of regression models and make informed decisions about model selection and refinement.

Formula for Mean Squared Error*:*

$$\text{MSE} = \frac{1}{\text{n}}\sum(y_{\text{i}} - p_{\text{i}})^2$$

Formula for R-Squared:

$$R^2 = 1 - \frac{Unexplained\ Variation}{Total\ Variation}$$

Formula for Mean Squared Logarithmic Error*:*

$$MSLE(y, \hat{y}) = \frac{1}{N}\sum_{i=0}^{N-1}(\log_e(1 + y_i) - \log_e(1 + \hat{y}_i))^2$$

C. *Correlation Matrix*

3.3 Correlation Matrix. Figure 17 shows the correlation matrix of the features in the dataset. The correlation matrix indicates how features are interrelated with each other. The group is the main target feature for relation to dengue cases. If the value of the group is greater than 0.5, the patients have dengue. From the correlation matrix, it has been observed that the higher the values of Week and humidity, there is more chance of increasing dengue cases.

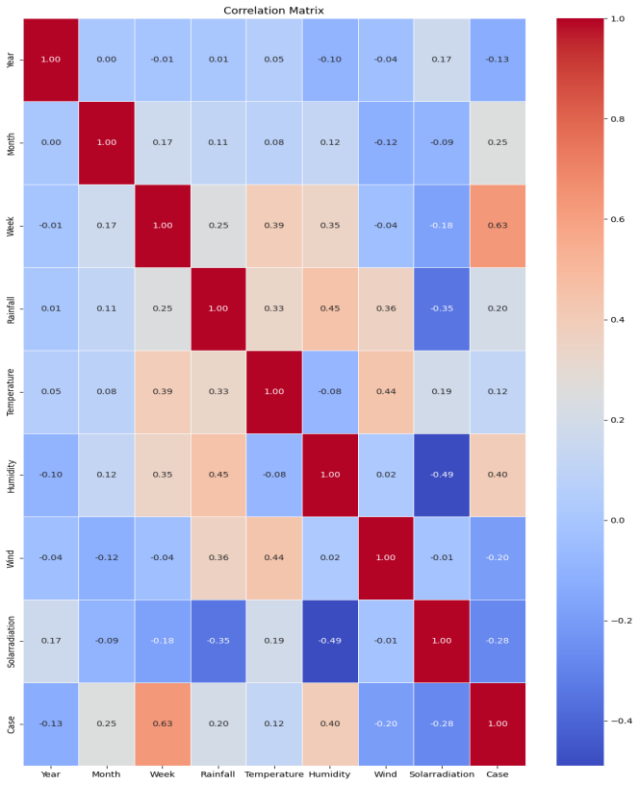

Figure 9: Correlation matrix

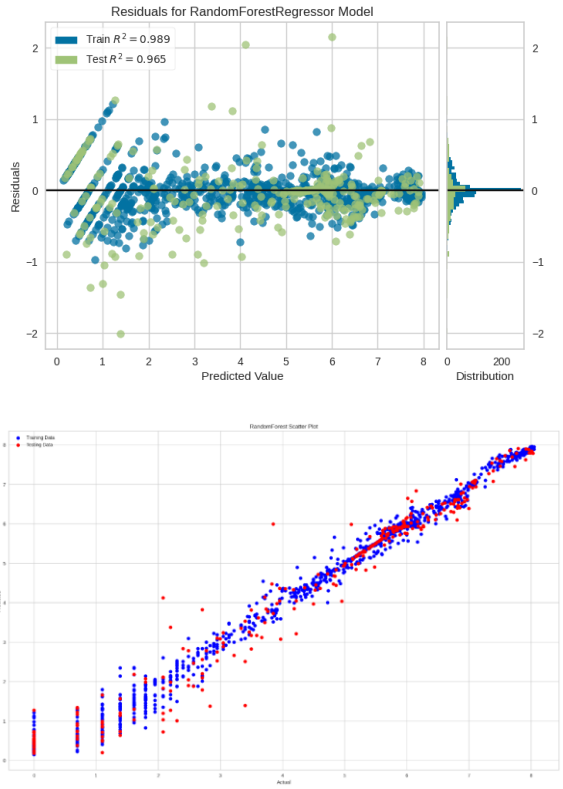

Figure 10: Random Forest Model

D. Model

3.4. RandomForest Model. Table *3* shows the random forest model's prediction after hyper tuning. The Random Forest model achieved a very low mean squared error (MSE) of 0.0604 on the training data, indicating that it fits the training data very well. Additionally, the R2 score of 0.9892 suggests that the model explains approximately 98.92% of the variance in the training data, indicating a high level of accuracy. On the testing data, the model performed slightly worse compared to the training data, with a higher MSE of 0.1907. However, this MSE is still relatively low, indicating reasonable performance. The R2 score of 0.9642 on the testing data suggests that the model explains approximately 96.42% of the variance in the testing data, indicating a good level of generalization and predictive capability. Overall, the Random Forest model appears to be performing well and has the potential to make accurate predictions on unseen data.

Evaluation	Training	Testing
MSE	0.0604	0.1907
R^2 score	0.9891	0.9642

Table 3: Random Forest Model Evaluation

Evaluation	Training	Testing
MSE	4.4284	4.1513
R^2 score	0.2056	0.2217

Table 4: Lasso Model Evaluation

3.2.2 Lasso Model. Table *4* shows the Lasso model's prediction after hyper tuning. The Lasso Regression model achieved a relatively high mean squared error (MSE) of 4.4284 on the training data, indicating a significant amount of error between the actual and predicted values. The R2 score of 0.2057 suggests that only approximately 20.57% of the variance in the training data is explained by the model, indicating poor performance in capturing the variability of the training data. On the testing data, the model performed slightly better compared to the training data, with a lower MSE of 4.1513. However, this MSE is still relatively high, indicating that the model's predictions deviate significantly from the actual values. The R2 score of 0.2217 on the testing data suggests that only approximately 22.17% of the variance in the testing data is explained by the model, indicating limited predictive capability. The model demonstrates poor generalization performance, as evidenced by the similar MSE scores on both the training and testing datasets.

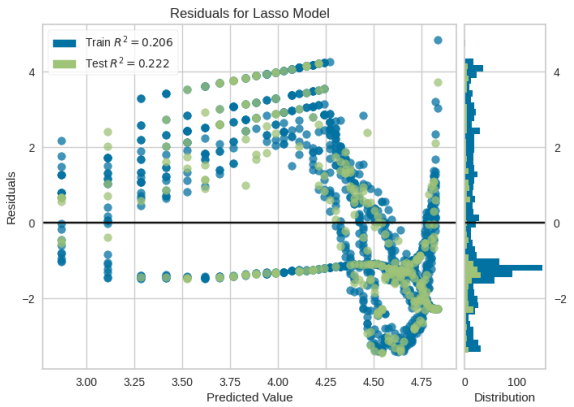

Figure 11: Residual for Lasso Model

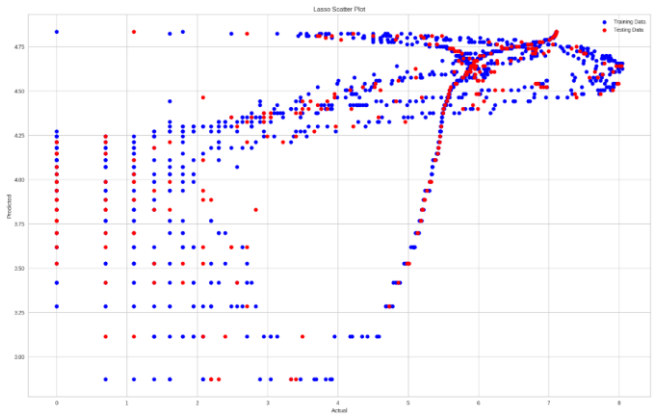

Figure 12: Lasso Model

3.2.3 Linear Regression Model. Table *5* shows the Linear Regression model's prediction after hyper tuning. The Linear Regression model achieved a moderate mean squared error (MSE) of 2.8350 on the training data, indicating a moderate amount of error between the actual and predicted values. The R2 score of 0.4915 suggests that approximately 49.15% of the variance in the training data is explained by the model, indicating moderate performance in capturing the variability of the training data. On the testing data, the model performed slightly better compared to the training data, with a lower MSE of 2.6224. However, this MSE is still relatively moderate, indicating that the model's predictions deviate to some extent from the actual values. The R2 score of 0.5084 on the testing data suggests that approximately 50.84% of the variance in the testing data is explained by the model, indicating moderate predictive capability. Overall, the Linear Regression model appears to be performing reasonably well and may be suitable for predicting dengue cases in this scenario.

3.2.4 SVR Model. Table *6* shows the SVR model's prediction after hyper tuning. The SVR model achieved a low mean squared error (MSE) of 0.2829 on the training data, indicating a small amount of error between the actual and predicted values. The high R2 score of 0.9493 suggests that approximately 94.93% of the variance in the training data is explained by the model, indicating strong performance in capturing the variability of the training data. On the testing data, the model performed slightly worse compared to the training data, with a higher MSE of 0.3512. However, this MSE is still relatively low, indicating that the model's predictions deviate moderately from the actual values. The R2 score of 0.9342 on the testing data suggests that approximately 93.42% of the variance in the testing data is explained by the model, indicating strong predictive capability. Overall, the SVR model appears to be performing very well and is well-suited for predicting dengue cases in this scenario.

3.2.5 BidirectionalLSTM Model. Figure *14* shows the BidirectionalLSTM model's prediction for different numbers of epoch. As the number of epochs increases, there is a notable improvement in both training and testing performance metrics, including MSE and R2 scores. With fewer epochs, such as 50, the model exhibits higher MSE and lower R2 scores, indicating poorer performance in Table *7*. However, as the number of epochs increases, the model's performance improves significantly.

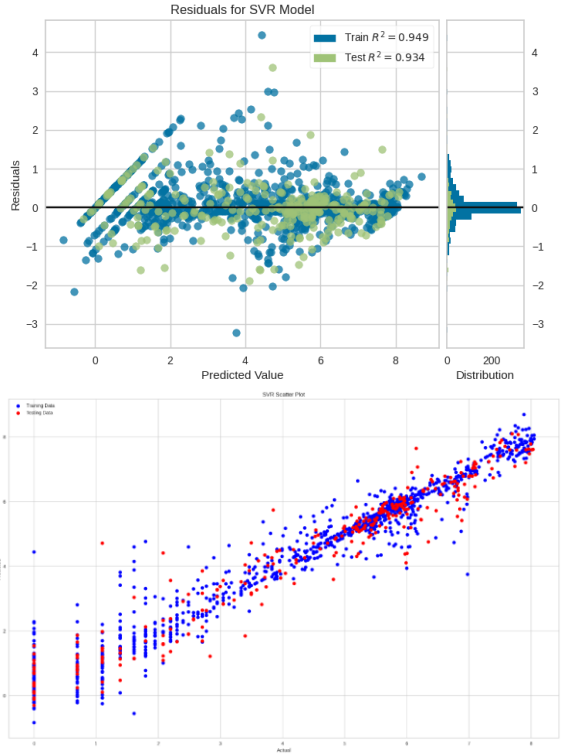

Figure 13: SVR Model

Evaluation	Training	Testing
MSE	2.8350	2.6223
R^2 score	0.4914	0.5083

Table 5: Linear Regression Model Evaluation

Evaluation	Training	Testing
MSE	0.2828	0.3511
R^2 score	0.9492	0.9341

Table 6: SVR Model Evaluation

Specifically, with 500 epochs, the model achieves the lowest MSE of 0.01095 on the training data and 0.3794 on the testing data, along with the highest R2 scores of 0.9980 and 0.9289 for the training and testing datasets, respectively. This suggests that the Bidirectional LSTM model benefits from additional training epochs, leading to better learning and predictive capability.

3.2.6 LSTM Model. Figure *16* shows the LSTM model's prediction for different numbers of epoch. As the number of epochs increases, there is a noticeable improvement in both training and testing performance metrics, including MSE and R2 scores. With fewer epochs, such as 50, the model exhibits higher MSE and lower R2 scores, indicating poorer performance in Table *8*.

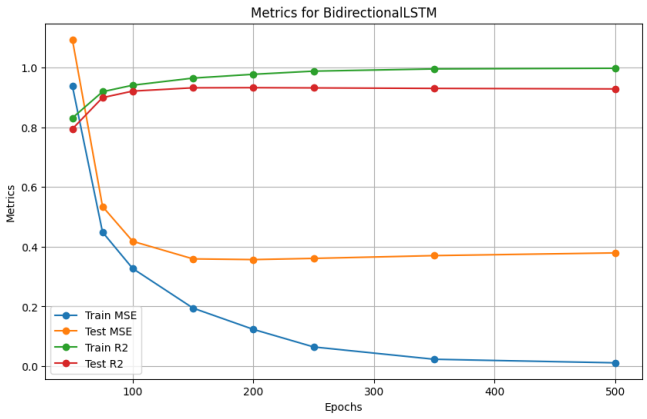

Figure 14: BidirectionalLSTM Model

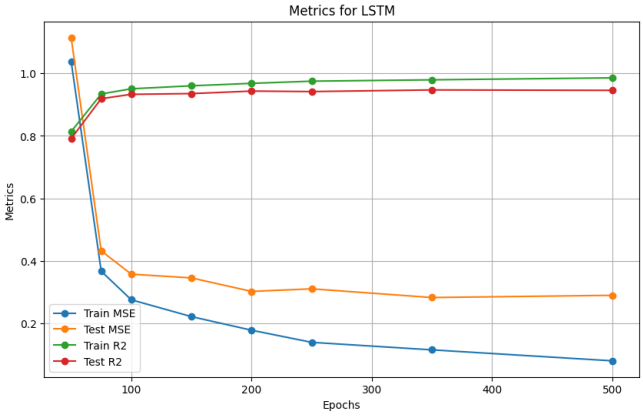

Figure 16: LSTM Model

However, as the number of epochs increases, the model's performance improves significantly. Specifically, with 500 epochs, the model achieves the lowest MSE of 0.07964 on the training data and 0.2895 on the testing data, along with the highest R2 scores of 0.9857 and 0.9457 for the training and testing datasets, respectively. This suggests that the LSTM model benefits from additional training epochs, leading to better learning and predictive capability.

Evaluation	Epoch	Training	Testing
MSE	50	0.9386	1.0942
	75	0.4489	0.5348
	100	0.3270	0.4182
	150	0.1943	0.3593
	200	0.1230	0.3569
	250	0.0641	0.3609
	300	0.0228	0.3702
	500	0.0109	0.3794
R^2 score	50	0.8316	0.7948
	75	0.9194	0.8997
	100	0.9413	0.9215
	150	0.9651	0.9326
	200	0.9779	0.9330
	250	0.9884	0.9323
	300	0.9958	0.9305
	500	0.9980	0.9288

Table 7: BidirectionalLSTM Model Evaluation

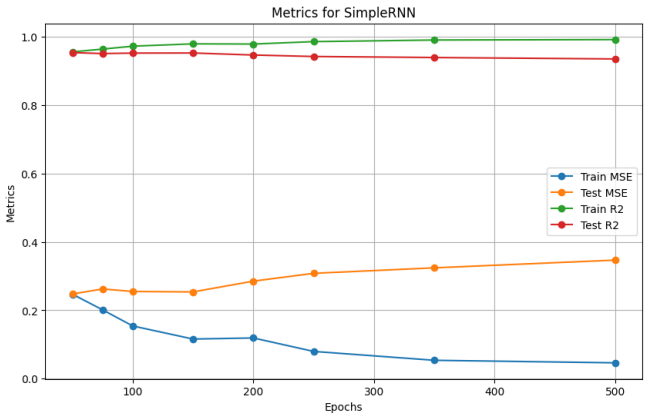

Figure 15: SimpleRNN Model

Evaluation	Epoch	Training	Testing
MSE	50	1.0386	1.1146
	75	0.3665	0.4317
	100	0.2750	0.3571
	150	0.2212	0.3451
	200	0.1776	0.3019
	250	0.1388	0.3102
	300	0.1147	0.2823
	500	0.0796	0.2894
R^2 score	50	0.8137	0.7910
	75	0.9342	0.9190
	100	0.9506	0.9330
	150	0.9603	0.9352
	200	0.9681	0.9433
	250	0.9750	0.9418
	300	0.9794	0.9470
	500	0.9857	0.9457

Table 8: LSTM Model Evaluation

Evaluation	Epoch	Training	Testing
MSE	50	0.2471	0.2476
	75	0.2013	0.2623
	100	0.1541	0.2550
	150	0.1159	0.2537
	200	0.1190	0.2852
	250	0.0796	0.3080
	300	0.0537	0.3241
	500	0.0463	0.3467
R^2 score	50	0.9556	0.9535
	75	0.9638	0.9508
	100	0.9723	0.9521
	150	0.9792	0.9524
	200	0.9786	0.9465
	250	0.9857	0.9422
	300	0.9903	0.9392
	500	0.9916	0.9349

Table 9: SimpleRNN Model Evaluation

3.2.7 SimpleRNN Model. Figure *15* shows the SimpleRNN model's prediction for different numbers of epoch. As the number of epochs increases, there is a general trend of improvement in both training and testing performance metrics, including MSE and R2 scores. With fewer epochs, such as 50, the model exhibits relatively higher MSE and lower R2 scores, indicating suboptimal performance. However, as the number of epochs increases, the model's

performance tends to improve significantly. Specifically, with 500 epochs, the model achieves a relatively low MSE of 0.04632 on the training data and 0.3467 on the testing data, along with relatively high R2 scores of 0.9917 and 0.9350 for the training and testing datasets, Table *9*, respectively.

E. Model Comparison

3.3.1. Comparison Table. Table 10 shows the comparison table of the models. The table clearly indicates that Random Forest is the best model among the other models in the system. It has the best mean square root and R-square score. The Random Forest model achieved the lowest MSE of 0.1907 and a high R2 score of 0.9642, indicating excellent performance in predicting dengue cases. The Lasso regression model exhibited a significantly higher MSE of 4.1513 and a relatively low R2 score of 0.2217, suggesting poor performance compared to other models. This could indicate that Lasso regression may not be the most suitable model for this dataset. Linear regression performed moderately well, with an MSE of 2.6224 and an R2 score of 0.5084, indicating reasonable predictive capability. SVR (Support Vector Regression) demonstrated a relatively low MSE of 0.3512 and a high R2 score of 0.9342, indicating strong predictive performance. Bidirectional LSTM, LSTM, and SimpleRNN models, which are recurrent neural network (RNN) variants, showed comparable performance with MSE values ranging from 0.2320 to 0.3027 and R2 scores ranging from 0.9432 to 0.9565. These models generally outperformed linear regression and Lasso regression, suggesting that RNN-based models might be better suited for this dataset's predictive task.

Model	MSE	R^2 score
RandomForest	0.1907	0.9642
Lasso	4.1513	0.2217
LinearRegression	2.6223	0.5083
SVR	0.3511	0.9341
BidirectionalLSTM	0.3027	0.9432
LSTM	0.2749	0.9484
SimpleRNN	0.3467	0.9564

Table 10: Comparison Table

IV. Conclusions

The dengue prediction project, utilizing remote sensing and machine learning, holds paramount significance for public health in Bangladesh. This innovative initiative aims to enhance the accuracy of dengue outbreak forecasts, enabling timely and targeted responses to mitigate the impact of the disease. Given the rising dengue-related fatalities in the country, the project addresses a critical healthcare priority. Beyond its health implications, the project reflects a commitment to sustainable practices by reducing reliance on environmentally harmful methods traditionally associated with disease control. By fostering resource efficiency and contributing to climate resilience, the initiative aligns with Bangladesh's broader goals of promoting public health and environmental sustainability. In essence, this project signifies a crucial step forward in utilizing technology for the betterment of public health in the specific context of Bangladesh, with a dual focus on immediate health outcomes and long-term environmental stewardship.

A. Limitations

The dengue prediction project, utilizing remote sensing and machine learning in the context of Bangladesh, faces a notable challenge due to the limitations of outdated and inaccurate datasets. The success of the project hinges on the availability of reliable data for training and refining the predictive models. In Bangladesh, where data quality has been a concern, efforts must be directed towards improving the accuracy and timeliness of the datasets. Despite this limitation, the project signifies a crucial step forward in addressing the pressing issue of dengue outbreaks in the country. By acknowledging and actively working to rectify these limitations, the project not only contributes to advancing dengue prediction but also lays the groundwork for enhancing data quality standards in public health initiatives in Bangladesh. Additionally, errors in coding and testing processes can impact the reliability of results. Addressing these limitations is paramount to the project's success. Collaborative efforts to enhance data accuracy, thorough code review, and meticulous testing procedures are essential steps. Recognizing these challenges underscores the commitment to refining the project methodology, ensuring its robustness, and contributing to the broader advancement of predictive modeling in public health within the Bangladeshi context.

B. Future Improvement

The current dengue prediction model, leveraging the amalgamation of remote sensing and machine learning in Bangladesh, represents a commendable step forward in preemptive healthcare strategies. However, recognizing the evolving nature of infectious diseases, particularly dengue, there is a clear pathway for enhancing the model's efficacy. At present, the model adeptly analyzes historical data to predict dengue outbreaks based on past occurrences. A pivotal future improvement lies in extending the predictive capacity of the model to anticipate outbreaks for the next 4 to 6 weeks. This evolution would empower healthcare authorities with more comprehensive foresight, enabling them to implement timely and targeted interventions to curb the spread of dengue. To achieve this enhancement, the model's dataset must evolve to incorporate real-time data streams. This includes integrating up-to-the-minute information on climate, vector density, and demographic factors. Concurrently, refining the machine learning algorithms to process this dynamic data efficiently becomes paramount. Collaborative efforts with meteorological agencies, public health institutions, and technology experts are instrumental in enriching the dataset and fine-tuning the predictive algorithms. By fostering a multidisciplinary approach, the model can adapt to the intricacies of dengue dynamics, providing a more accurate and actionable forecast. In summary, the future improvements envisioned for the dengue prediction model in Bangladesh involve a transition from retrospective analysis to prospective forecasting. By extending the prediction horizon to 4 to 6 weeks, the model can become an even more invaluable tool in the country's public health arsenal, contributing significantly to the mitigation of

dengue outbreaks and the overall well-being of its populace.

ACKNOWLEDGMENT

The authors would like to express their heartfelt gratitude towards their project and research supervisor, Dr. Mohammad Monirujjaman Khan, Associate Professor, Department of Electrical and Computer Engineering, North South University, Bangladesh, for his invaluable support, precise guidance and advice pertaining to the experiments, research and theoretical studies carried out during the course of the current project and also in the preparation of the current report.

Furthermore, the authors would like to thank the Department of Electrical and Computer Engineering, North South University, Bangladesh for facilitating the research. The authors would also like to thank their loved ones for their countless sacrifices and continual support.

REFERENCES

[1] G. Jeon, "Advanced Machine Learning and Deep Learning Approaches for Remote Sensing II," *Remote sensing*, vol. 16, no. 8, pp. 1353–1353, Apr. 2024, doi: https://doi.org/10.3390/rs16081353.

[2] Ganesh N, P. Jain, A. Choudhury, P. Dutta, K. Kalita, and Paolo Barsocchi, "Random Forest Regression-Based Machine Learning Model for Accurate Estimation of Fluid Flow in Curved Pipes," *Processes*, vol. 9, no. 11, pp. 2095–2095, Nov. 2021, doi: https://doi.org/10.3390/pr9112095.

[3] r.k Jeyachitra and S. Manochandar, "Machine Learning and Deep Learning: Classification and Regression Problems, Recurrent Neural Networks,...," *ResearchGate*, Oct. 16, 2023. https://www.researchgate.net/publication/374793041_Machine_Learning_and_Deep_Learning_Classification_and_Regression_Problems_Recurrent_Neural_Networks_Convolutional_Neural_Networks (accessed May 15, 2024).

[4] M. Sharma, A. Kumar, Supriya Muthuraman, and S. Kishore, "Machine learning in remote sensing data—a classification case study," *ResearchGate*, 2023. https://www.researchgate.net/publication/365343443_Machine_learning_in_remote_sensing_data-a_classification_case_study (accessed May 15, 2024).

[5] G. Camps-Valls, "Machine learning in remote sensing data processing," Sep. 2009, doi: https://doi.org/10.1109/mlsp.2009.5306233.

[6] Md. Toyhid Sarwar and Md. Al Mamun, "Prediction of Dengue using Machine Learning Algorithms: Case Study Dhaka," Dec. 2022, doi: https://doi.org/10.1109/icecte57896.2022.10114535.

[7] "Remote Sensing," *Mdpi.com*, 2024. https://www.mdpi.com/journal/remotesensing/special_issues/J938V8W2EM (accessed May 15, 2024).

[8] A. L. Buczak, P. T. Koshute, S. M. Babin, B. H. Feighner, and S. H. Lewis, "A data-driven epidemiological prediction method for dengue outbreaks using local and remote sensing data," *BMC medical informatics and decision making*, vol. 12, no. 1, Nov. 2012, doi: https://doi.org/10.1186/1472-6947-12-124.

[9] Devojit Kumar Sarma *et al.*, "An assessment of remotely sensed environmental variables on Dengue epidemiology in Central India," *PLoS neglected tropical diseases*, vol. 16, no. 10, pp. e0010859–e0010859, Oct. 2022, doi: https://doi.org/10.1371/journal.pntd.0010859.

[10] S. Yin, C. Ren, Y. Shi, J. Hua, H.-Y. Yuan, and L.-W. Tian, "A Systematic Review on Modeling Methods and Influential Factors for Mapping Dengue-Related Risk in Urban Settings," *International journal of environmental research and public health/International journal of environmental research and public health*, vol. 19, no. 22, pp. 15265–15265, Nov. 2022, doi: https://doi.org/10.3390/ijerph192215265.

[11] M. A. Majeed, Z. Mohd, Zed Zulkafli, and Aimrun Wayayok, "A Deep Learning Approach for Dengue Fever Prediction in Malaysia Using LSTM with Spatial Attention," *International journal of environmental research and public health/International journal of environmental research and public health*, vol. 20, no. 5, pp. 4130–4130, Feb. 2023, doi: https://doi.org/10.3390/ijerph20054130.

[12] J. Brownlee, "Time Series Prediction with LSTM Recurrent Neural Networks in Python with Keras - MachineLearningMastery.com," *MachineLearningMastery.com*, Jul. 20, 2016. https://machinelearningmastery.com/time-series-prediction-lstm-recurrent-neural-networks-python-keras/ (accessed May 15, 2024).

[13] Y. Xia and J. Wang, "A One-Layer Recurrent Neural Network for Support Vector Machine Learning," *IEEE transactions on systems, man and cybernetics. Part B. Cybernetics*, vol. 34, no. 2, pp. 1261–1269, Apr. 2004, doi: https://doi.org/10.1109/tsmcb.2003.822955.

[14] S. Y. Fu, Y. C. Tseng, and K. N. Chiang, "Study on Data Effect of Using RNN Model to Predict Reliability Life of Wafer Level Packaging," Oct. 2020, doi: https://doi.org/10.1109/impact50485.2020.9268572.

[15] M.-L. Yin and Hovig Aroush, "Constant-Time Linear Regression Learning and Its Applications on Real-Time R&M Systems," *2022 Annual Reliability and Maintainability Symposium (RAMS)*, Jan. 2022, doi: https://doi.org/10.1109/rams51457.2022.9893939.

[16] Diyana Kinaneva, G. Hristov, Petko Kyuchukov, G. Georgiev, Plamen Zahariev, and Rosen Daskalov, "Machine Learning Algorithms for Regression Analysis and Predictions of Numerical Data," *2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, Jun. 2021, doi: https://doi.org/10.1109/hora52670.2021.9461298.

[17] J. Davis, S. P. Knight, R. Rizzo, O. A. Donoghue, Rose Anne Kenny, and R. Romero-Ortuno, "A linear regression-based machine learning pipeline for the discovery of clinically relevant correlates of gait speed reserve from multiple physiological systems," *2021 29th European Signal Processing Conference (EUSIPCO)*, Aug. 2021, doi: https://doi.org/10.23919/eusipco54536.2021.9616187.

[18] A. Venkatesh and M.S Saravanan, "An Efficient Method for Predicting Linear Regression with Polynomial Regression," *2022 3rd International Conference on Smart Electronics and Communication (ICOSEC)*, Oct. 2022, doi: https://doi.org/10.1109/icosec54921.2022.9952049.

[19] Baidya Nath Saha and Apurbalal Senapati, "Long Short Term Memory (LSTM) based Deep Learning for Sentiment Analysis of English and Spanish Data," *2020 International Conference on Computational Performance*

Evaluation (ComPE), Jul. 2020, doi: https://doi.org/10.1109/compe49325.2020.9200054.
[20] G. N. Kouziokas, "Long Short-Term Memory (LSTM) Deep Neural Networks in Energy Appliances Prediction," Nov. 2019, doi: https://doi.org/10.1109/pacet48583.2019.8956252.
[21] None Yu Wang, "A new concept using LSTM Neural Networks for dynamic system identification," May 2017, doi: https://doi.org/10.23919/acc.2017.7963782.