Big Data Analytics

A Project Submitted

By

Batch 2

Under the Guidance of

Kunal

BMS INSTITUTE OF TECHNOLOGY AND MANAGEMENT

YELAHANKA BANGALORE

Phase 4 Documentation

Project: Big Data Analytics

Phase 4 Submission by: Batch 2
Year: 4th Year, Computer Science and Engineering (CSE)

1. Introduction to Testing

Testing in machine learning and deep learning projects is a crucial phase to evaluate the effectiveness and generalization capabilities of the developed models. In this project, we applied extensive testing methodologies for both classical machine learning models (Random Forest, Linear Regression) and deep learning models (LSTM) to ensure robust performance on unseen rainfall data. Testing is aimed at validating model accuracy, reliability, interpretability, and ability to generalize well beyond the training dataset.

2. Evaluation Metrics:

Different metrics were used to assess models based on the nature of the problem — regression and classification.

For Regression (Linear Regression, LSTM):

Mean Squared Error (MSE): Measures the average squared difference between the predicted and actual values. Lower MSE values indicate better performance.

R² Score (Coefficient of Determination): Indicates how well the model explains variance in the target variable. A score closer to 1 indicates a better fit.

For Classification (Random Forest Classifier):

Accuracy:The proportion of correct predictions out of total predictions.

Confusion Matrix:A matrix showing true positives, false positives, true negatives, and false negatives, useful for visual performance.

Classification Report:Provides precision, recall, f1-score, and support — enabling in-depth model assessment for each class label.

3. Train-Test Splitting

To ensure that models are not overfitting, we used the train_test_split() function from Scikit-learn. A typical 80-20 split was used, ensuring that models are trained on one portion of the dataset and evaluated on another, unseen portion. This provides a strong indicator of real-world performance.

4. Visual Testing & Data Understanding

Correlation Heatmap:

We used seaborn.heatmap() to visualize the correlation matrix between rainfall features. This helped in feature selection and multicollinearity analysis.

Boxplots and Histograms:

Used to observe the spread and outliers in the dataset, allowing early identification of anomalies or data preprocessing issues.

PCA Scatter Plot:

Principal Component Analysis (PCA) reduced high-dimensional rainfall data to 2D/3D scatter plots, enabling better visualization and interpretation of clusters or patterns in the data.

5. Anomaly Detection Testing

An important aspect of this project was identifying abnormal rainfall patterns. Anomaly detection was implemented using:

Residual Analysis:Residual = actual value - predicted value. High residuals indicate outliers or unexpected patterns.

Threshold-Based Detection:Residuals above a fixed threshold (e.g., 100mm) were marked as anomalies.

Visualization:Anomalies were plotted using line plots with red markers to visually inspect their occurrences.

This technique helped identify periods with unusually high or low rainfall, aiding real-world applications like drought or flood warnings.

6. LSTM Model Testing

For the LSTM (Long Short-Term Memory) model used in time series forecasting:

Input Reshaping: Data was reshaped into 3D ([samples, time steps, features]) for compatability with lstm layers

Epoch-based Training: Model was trained using multiple epochs, and performance was tested on the validation set.

Loss Visualization: The loss values during training and validation were plotted to detect overfitting or underfitting.

Prediction Plots: Comparisons between actual and predicted sequences over time were visualized to interpret accuracy.

7. Model Comparison

Multiple models were trained and tested:

Linear Regression had interpretability and low training time but struggled with non-linearities.

Random Forest provided better performance and was more resistant to noise.

LSTM excelled at time-dependent patterns but required more tuning and resources .