**CS365 – Deep Learning**

**PROJECT**

**Multi-Step Air Quality Forecasting Using Robust Deep Learning Model**

**DONE BY-**

**Diksha Barnwal (2101AI14)**

**N. Haritha Reddy (2101AI20)**

**Overview:**

* The goal of this project is to perform multistep time series forecasting for predicting PM2.5 concentration based on various environmental factors.
* The dataset is split into training and testing sets, and a Long Short-Term Memory (LSTM) neural network is employed for the forecasting task.
* The workflow involves data preprocessing, sequence creation, model architecture definition, training, testing, and evaluation.

**Data Preprocessing:**

* Data Import:

Four datasets of different cities (B, G, S, T), train and test, are loaded from CSV files containing environmental data.

* Column Dropping:

The 'Unnamed: 0' column is dropped as it seems to be an unnecessary index column.

**Data Splitting:**

* Feature Selection:

Selected features for training and testing include weather conditions, temperature, pressure, humidity, wind speed, wind direction, and concentrations of various pollutants (PM10, NO2, CO, O3, SO2).

* Target Variable:

PM2.5 concentration is chosen as the target variable.

**Sequence Preparation:**

* Time Series Sequences:

Function create\_sequences is defined to create input-output sequences for training and testing.

* Sequence Parameters:

n\_steps\_in represents the number of time steps as input, and n\_steps\_out is the number of time steps to predict ahead.

**Model Architecture:**

1. LSTM Model: A stacked LSTM model with two layers is defined using Keras.

* Layer 1: LSTM with 100 units, activation 'relu', and return sequences set to True.
* Layer 2: LSTM with 100 units, activation 'relu', and return sequences set to False.
* Output Layer: Dense layer with 1 unit.

1. Compilation: The model is compiled using the Adam optimizer and Mean Squared Error loss.

**Model Training**:

* Training Data:

The model is trained on the training sequences for 50 epochs with a batch size of 90.

* History Tracking:

Training history is stored for later analysis.

**Model Evaluation:**

* Testing Data:

The trained model is used to predict PM2.5 concentrations on the testing data.

* Evaluation Metrics:

Mean Squared Error (MSE) and Mean Absolute Error (MAE) are calculated to assess the model's performance.

* Visualization:

The model's architecture is visualized using the plot\_model function.

**Results:**

* Predictions:

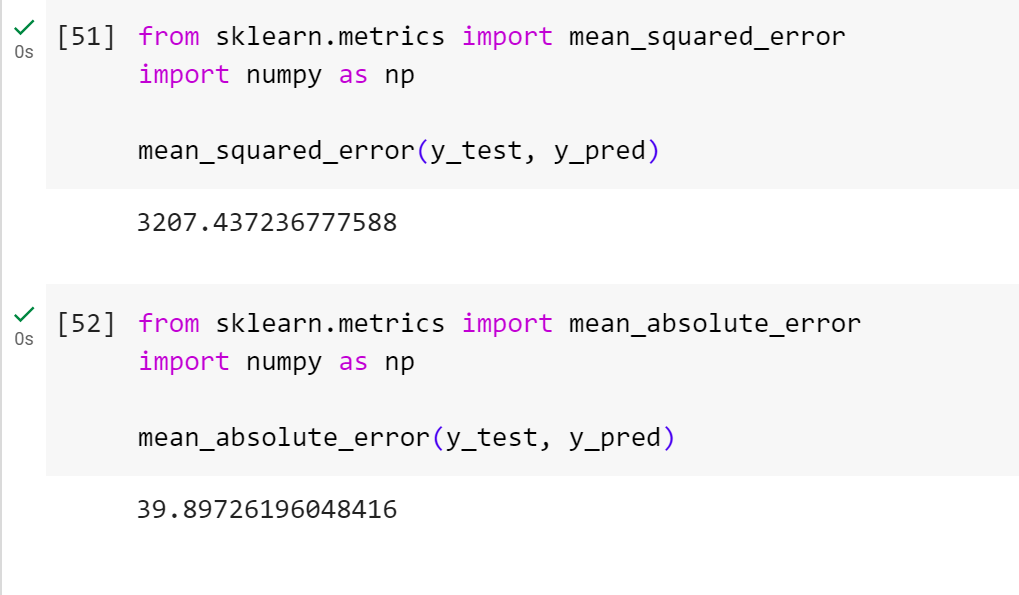
The model predicts PM2.5 concentrations for the testing data.

* Error Metrics:

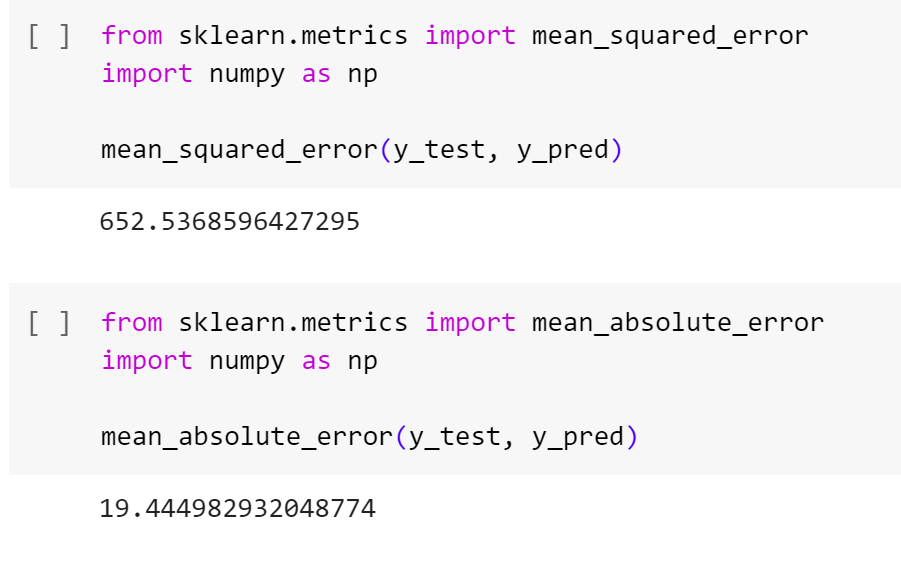
MSE and MAE are computed to quantify the model's accuracy.

**Outputs:**

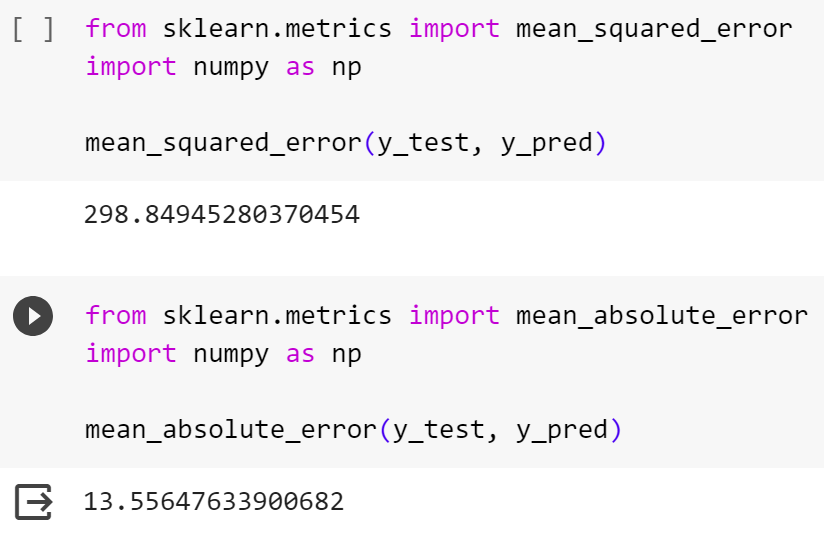
* **City B :**

****

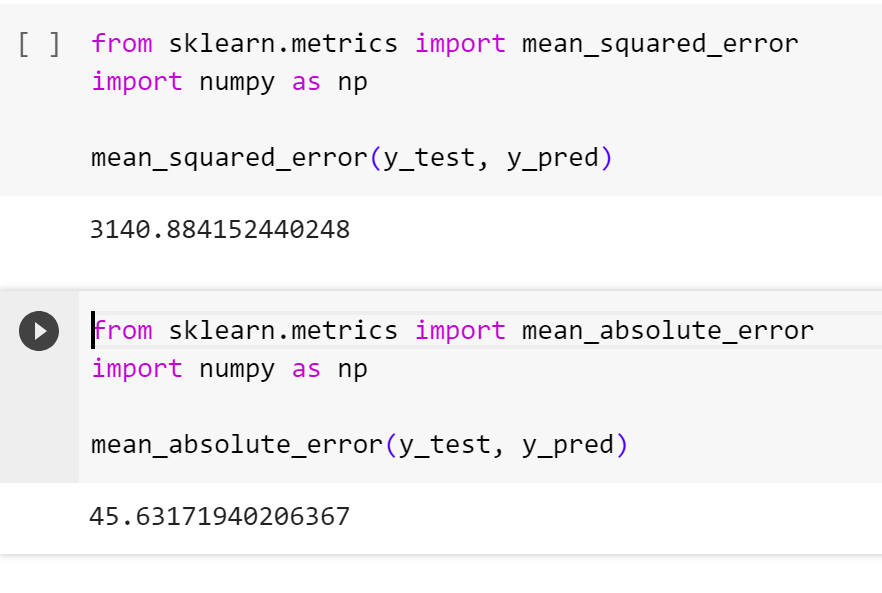
* **City G:**

****

* **City S:**

****

* **City T:**

****

**Thank You**