

## EXPERIMENT 9

Develop neural network-based time series forecasting model.

**DATE :** 17/04/25

**AIM :**

To write a python program to develop neural network based time series forecasting model.

**PROGRAM :**

```
import numpy as np
from keras.models import Sequential
from keras.layers import LSTM, Dense
from sklearn.preprocessing import MinMaxScaler
import matplotlib.pyplot as plt

# Prepare data
data = df['DAYTON_MW'].values.reshape(-1, 1)
scaler = MinMaxScaler()
scaled_data = scaler.fit_transform(data)

def create_sequences(data, step=24):
    X, y = [], []
    for i in range(len(data) - step):
        X.append(data[i:i+step])
        y.append(data[i+step])
    return np.array(X), np.array(y)

X, y = create_sequences(scaled_data)

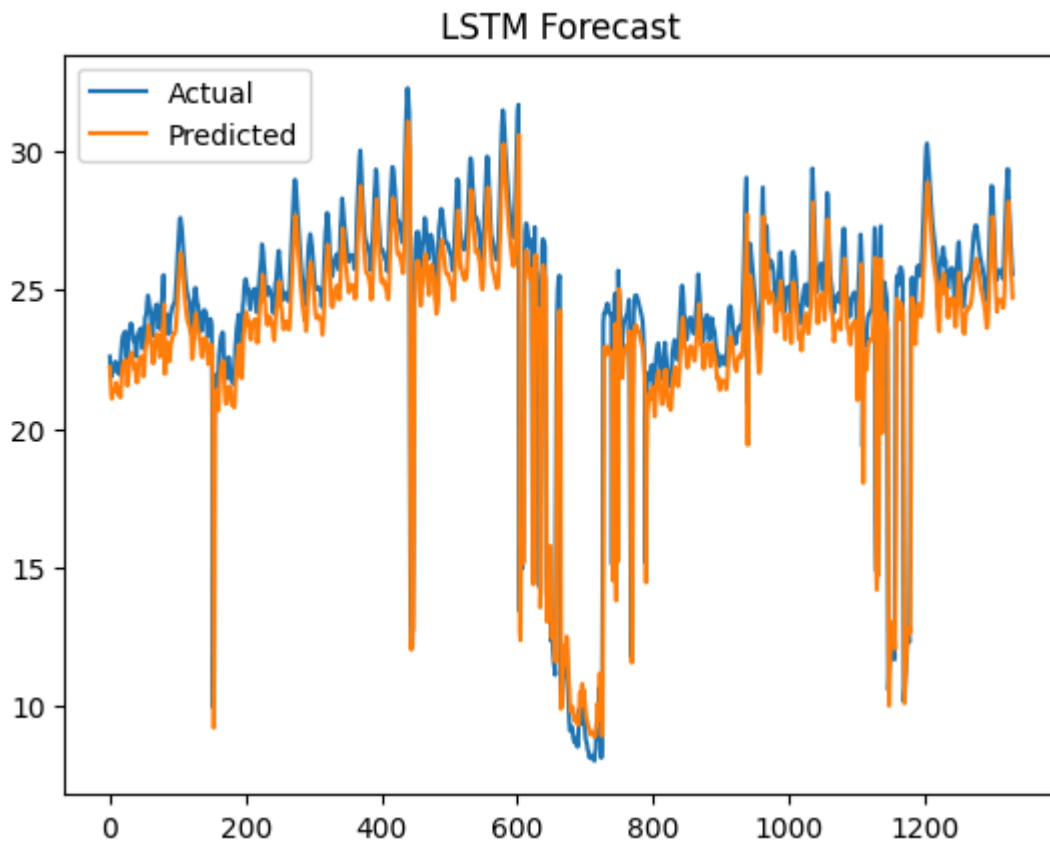
split = int(len(X)*0.8)
X_train, X_test = X[:split], X[split:]
y_train, y_test = y[:split], y[split:]

# Build model
model = Sequential([
    LSTM(50, activation='relu', input_shape=(X.shape[1], 1)),
    Dense(1)
])
model.compile(optimizer='adam', loss='mse')
model.fit(X_train, y_train, epochs=10, batch_size=32)

# Predict and plot
predicted = model.predict(X_test)
predicted = scaler.inverse_transform(predicted)
actual = scaler.inverse_transform(y_test)

plt.plot(actual, label='Actual')
plt.plot(predicted, label='Predicted')
plt.title("LSTM Forecast")
plt.legend()
plt.show()
```

**OUTPUT :**



**RESULT :**

Thus the python program to develop a neural network for time series forecasting is implemented successfully .