**Aim:** Write a program to build a Long Short-Term Memory (LSTM) model for sequential data using TensorFlow.

**Description:** Long Short-Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) capable of learning order dependence in sequence prediction problems. This model is widely used in tasks such as time series forecasting, natural language processing, and more.

**Objective:** To implement an LSTM model using TensorFlow that can learn from sequential data (e.g., sine wave or time series) and predict future values.

**Steps Overview:** Import necessary libraries.

 Generate or load sequential dataset.

 Preprocess the data for LSTM input.

 Build the LSTM model using TensorFlow.

 Compile and train the model.

 Evaluate the model and visualize the predictions.

**Implementation:***import numpy as np*

*import matplotlib.pyplot as plt*

*import tensorflow as tf*

*from tensorflow.keras.models import Sequential*

*from tensorflow.keras.layers import LSTM, Dense*

*# 1. Generate synthetic sequential data (sine wave)*

*def generate\_sine\_wave(seq\_length=50, total\_points=1000):*

*x = np.arange(total\_points)*

*y = np.sin(0.02 \* x)*

*return y*

*# 2. Prepare the dataset*

*def prepare\_dataset(data, seq\_length):*

*X, y = [], []*

*for i in range(len(data) - seq\_length):*

*X.append(data[i:i+seq\_length])*

*y.append(data[i+seq\_length])*

*X = np.array(X)*

*y = np.array(y)*

*return X, y*

*# Parameters*

*seq\_length = 50*

*data = generate\_sine\_wave()*

*X, y = prepare\_dataset(data, seq\_length)*

*# Reshape input to be [samples, time steps, features]*

*X = X.reshape((X.shape[0], X.shape[1], 1))*

*# 3. Build LSTM model*

*model = Sequential([*

*LSTM(50, activation='relu', input\_shape=(seq\_length, 1)),*

*Dense(1)])*

*# 4. Compile the model*

*model.compile(optimizer='adam', loss='mse')*

*# 5. Train the model*

*model.fit(X, y, epochs=10, batch\_size=32)*

*# 6. Predict and visualize*

*predicted = model.predict(X)*

*plt.figure(figsize=(10,5))*

*plt.plot(y, label='Actual')*

*plt.plot(predicted, label='Predicted')*

*plt.legend()*

*plt.title("LSTM Sequential Prediction")*

*plt.xlabel("Time Steps")*

*plt.ylabel("Value")*

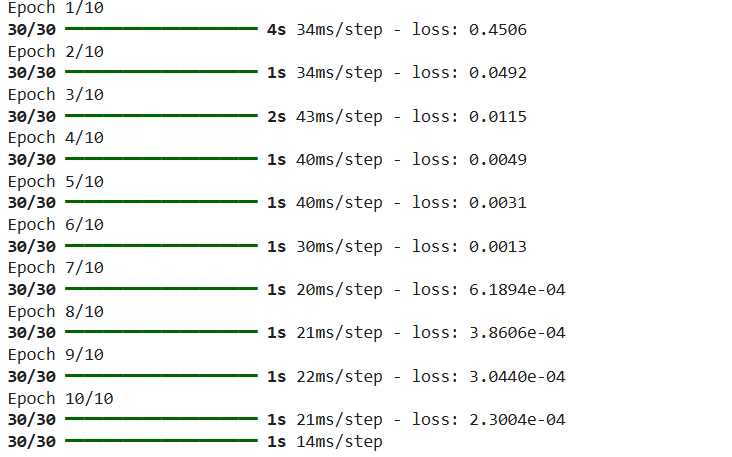
*plt.show()*

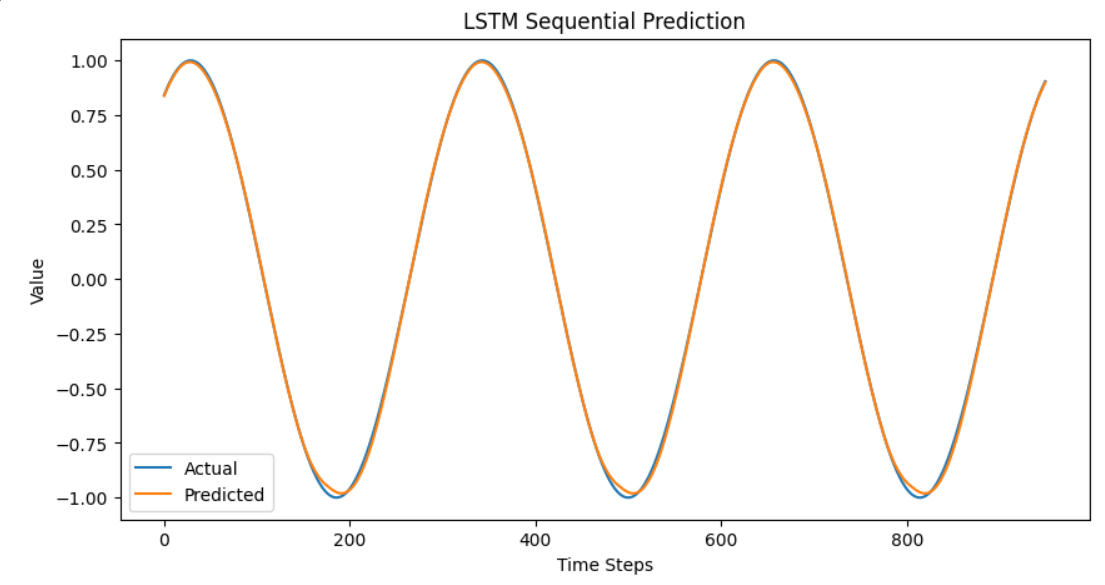
**Output:*** A plot showing the actual vs predicted values from the LSTM model.*

* Training logs for each epoch showing the loss decreasing.*

***Details of the Dataset Used:***

* ***Type:*** *Synthetic Time Series Data*
* ***Function Used:*** *y = sin(0.02 \* x)*
* ***Total Points:*** *1000*

****

****

**Conclusion:**

The LSTM model built using TensorFlow was able to learn from a sequential sine wave dataset and provide predictions that follow the expected pattern. This basic implementation can be extended to real-world datasets such as stock prices, weather data, or text sequences.