The LSTM Time Series Forecasting Model.

Created by:M.yeswanthkumar.

This script demonstrates time series forecasting using an **LSTM (Long Short-Term Memory) model** built with TensorFlow/Keras. The model is designed to predict the next value in a sequence given past observations.

---

1. Data Generation

- The function generate_time_series(n_samples, n_steps) creates synthetic time series data.
- It generates a sine wave with added Gaussian noise to simulate real-world fluctuations.
- The dataset consists of **1000 samples**, each containing **50 time steps**, plus 1 extra time step as the target.

---

2. Data Preprocessing

- The dataset is **normalized** using MinMaxScaler to scale values between 0 and 1, which helps improve model training.
- The input (X_train) consists of the first 50 time steps, while the target (y_train) is the 51st time step.
- The input shape is reshaped to **(samples, time steps, features)**, which is required for LSTM models.

---

3. Building the LSTM Model

- A **stacked LSTM** architecture is implemented:
    1. **First LSTM layer**: 50 units with ReLU activation and return_sequences=True, allowing stacked LSTM layers.
    2. **Second LSTM layer**: 50 units with ReLU activation, capturing sequential dependencies.
    3. **Dense output layer**: A fully connected layer with 1 neuron, predicting the next value in the sequence.
- The model is compiled using:
    - **Optimizer:** adam (adaptive learning rate optimization)
    - **Loss function:** mse (Mean Squared Error)

---

4. Model Training

- The model is trained for **20 epochs** with a batch size of **32**.

- Training loss is minimized over iterations to improve accuracy.

---

5. Making Predictions

- Once trained, the model predicts the **next value in the sequence** for the first sample in the dataset.
- The predicted value is printed to evaluate model performance.

Use Cases & Extensions

This model can be adapted for real-world applications such as:

✅ **Stock Market Forecasting** – Predicting future stock prices.
✅ **Weather Forecasting** – Predicting temperature trends.
✅ **Energy Consumption Prediction** – Forecasting electricity usage.
✅ **Sales Forecasting** – Predicting future sales volume.

Possible Enhancements

- **Using multiple features** (e.g., external factors like seasonality, trends).
- **Adding dropout layers** to prevent overfitting.
- **Increasing epochs & tuning hyperparameters** for better performance.
- **Comparing with ARIMA models** for traditional time series forecasting.