Time Series Forecasting with LSTM & Integrated Gradients

This project implements a time-series forecasting model using Long Short-Term Memory (LSTM) networks in TensorFlow/Keras and explains model predictions using Integrated Gradients (IG). The project is suitable for energy consumption forecasting and other multivariate time-series tasks.

Features

- LSTM-based time-series forecasting
- Support for multivariate time-series data
- Integrated Gradients for explainable AI
- CPU-optimized TensorFlow execution
- Modular and extensible project structure

Technologies Used

- Python 3.10+ (tested on Python 3.13)
- TensorFlow / Keras
- NumPy
- Pandas
- Scikit-learn

Project Structure

data/energy.csv – Dataset
models/lstm_model.h5 – Trained LSTM model
dataset.py – Data loading and preprocessing
train.py – Model training script
explain.py – Integrated Gradients implementation
main.py – Program entry point

Dataset Description

The dataset is a multivariate time-series stored in CSV format. All features must maintain consistent order, scale, and time intervals.

Model Training

Run the training script to train the LSTM model. The trained model is saved in the models directory for later inference and explanation.

Integrated Gradients Explanation

Integrated Gradients is used to explain individual predictions by attributing importance to each timestep and feature. The output has the shape (1, timesteps, features).

Notes

TensorFlow may display informational CPU optimization logs. These messages are normal and do not indicate errors. When adding new data, retraining the model is recommended.

Author

Anowin Deepak