

Deep learning in Time Series

Business Objective

Deep Learning has become a fundamental part of the new generation of Time Series Forecasting models, obtaining excellent results. While in classical Machine Learning models - such as autoregressive models (AR) or exponential smoothing - feature engineering is performed manually and often some parameters are optimized also considering the domain knowledge, Deep Learning models learn features and dynamics only and directly from the data. Thanks to this, they speed up the process of data preparation and are able to learn more complex data patterns in a more complete way.

Till now, in this sequence of time-series projects, we have covered the machine learning topics such as Autoregression modelling, Moving Average Smoothing techniques, ARIMA model, Multiple linear regression, Gaussian process, and ARCH-GARCH models.

In this project, we will demonstrate how deep learning in time series can be used. There are four major models which will be built,

- Multi-Layer Perceptron (MLP)
- Convolutional Neural Network (CNN)
- Long Short-Term Memory (LSTM)
- Hybrid CNN – LSTM

Data Description

The dataset is “Call-centres” data. This data is at month level wherein the calls are segregated at domain level as the call centre operates for various domains. There are also external regressors like no of channels and no of phone lines which essentially indicate the traffic prediction of the inhouse analyst and the resources available.

The total number of rows are 132 and number of columns are 8:

- Month, healthcare, telecom, banking, technology, insurance, no of phonelines and no of channels.

Aim

We aim to build four deep learning models such as MLP, CNN, LSTM, and a hybrid CNN-LSTM model on the given time series dataset.

Tech stack

- Language - Python
- Libraries - pandas, numpy, matplotlib, TensorFlow

Approach

1. Import the required libraries and read the dataset
2. Perform descriptive analysis
3. Data pre-processing
 - Setting date as Index
 - Setting frequency as month
4. Exploratory Data Analysis (EDA) -
 - Data Visualization
5. Set the training format
 - Reshape the input data
6. Perform train-test split
7. MLP model
 - Define learning rate, number of epochs, and optimizer.
 - Build a sequential model with dense layers
 - Fit and train the model
 - Make predictions on the test data
 - Plot the results
8. CNN model
 - Reshape the data in three dimensions
 - Define learning rate, number of epochs, and optimizer.
 - Build a sequential model with convolution, max-pooling layers
 - Fit and train the model
 - Make predictions on the test data
 - Plot the results
9. LSTM model
 - Define learning rate, number of epochs, and optimizer.
 - Build a sequential model with LSTM layer and dense layers
 - Fit and train the model
 - Make predictions on the test data
 - Plot the results
10. CNN-LSTM model
 - Reshape the data in four dimensions
 - Define learning rate, number of epochs, and optimizer.
 - Build a sequential model
 - Fit the model
 - Make predictions on the test data
 - Plot the results

Modular code overview

```
input
|_CallCenterData.xlsx

src
|_Engine.py
|_ML_Pipeline
|   |_CNN_LSTM.py
|   |_CNN_Model.py
|   |_LSTM.py
|   |_MLP.py

lib
|_DeepLearning.ipynb

output
|_plots folder
|   |_visualization plots
|_models folder
|   |_models saved in pickle format
```

Once you unzip the modular_code.zip file, you can find the following folders within it.

1. Input
2. Src
3. Output
4. Lib

1. Input folder - It contains all the data that we have for analysis. The following csv is used.
 - CallCenterData.xlsx
2. Src folder - This is the most important folder of the project. This folder contains all the modularized code for all the above steps in a modularized manner. This folder consists of:
 - Engine.py
 - ML_PipelineThe ML_pipeline is a folder that contains all the functions put into different python files which are appropriately named. These python functions are then called inside the engine.py file.

3. Output folder - The output folder contains two subfolders. In the plots folder, you will find all types of visualization plots, whereas the model folder, contains the four models saved in pickle format.
4. Lib folder - This is a reference folder. It contains the original IPython notebook that we saw in the videos. The ppt used during the videos is also present here.

Project Takeaways

1. Introduction to Time series
2. Understand the basics of time series
3. Importing the dataset and required libraries
4. Data pre-processing
5. Exploratory Data Analysis (EDA)
6. What is deep learning, and how can it be used for time series?
7. What is a Multi-Layer Perceptron?
8. What are Convolutional Neural Networks?
9. What is Long-Short Term Memory?
10. Concept of hybrid CNN-LSTM
11. Building an MLP model
12. Building a CNN model
13. Building an LSTM model
14. Building a CNN-LSTM model
15. How to make predictions on the test data and plot the results