A
Project Report
On
"Household Power Consumption Time Series Data Analysis using LSTM "

Prepared by

Dhruvin Gohil (17DCS016)

Karan Zaveri (17DCS072)

Under the guidance of

Mohammed Bohara

Assistant Professor

A Report Submitted to

Charotar University of Science and Technology

for Partial Fulfillment of the Requirements for the

7th Semester Programming in Python (CS441)

Submitted at

CHARUSAT

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CSE

DEPSTAR

At: Changa, Dist: Anand – 388421

OCT 2020

CERTIFICATE

This is to certify that the report entitled "Household Power Consuption Time Series Data Analysis using LSTM" is a bonafied work carried out by Mr. Dhruvin Gohil (17DCS016) and Mr. Karan Zaveri (17DCS072) under the guidance and supervision of Assistant Prof. Mohammed Bohara for the subject CS441 Programming in Python (CSE) of 7th Semester of Bachelor of Technology in DEPSTAR at Faculty of Technology & Engineering – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Mohammed Bohara
Assistant Professor
CSE
DEPSTAR, Changa, Gujarat.

Dr. Amit Ganatra
Principal, DEPSTAR
Dean, FTE
CHARUSAT, Changa, Gujarat.

TABLE OF CONTENTS

LIST OF FIGURES

Acknowledgement

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations.

I would like to extend my sincere thanks to all of them. I am highly indebted to Asst Prof. Phenil Buch for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

I would like to express my gratitude towards member of DEPSTAR for their kind cooperation and encouragement which help me in completion of this project.

1. ABSTRACT

LSTM (long-short term memory) Networks is one of the RNNs (recurrent neural networks). Long Short Term Memory (LSTM) is among the most popular deep learning models used today. It performs better than normal RNNs in processing and predicting time series related data. At present, LSTM has achieved considerable success on many issues and has been widely used. Based on the excellent performance of LSTM Networks in time series.

2. INTRODUCTION

Time Series is a collection of data points indexed based on the time they were collected. Most often, the data is recorded at regular time intervals. What makes Time Series data special? Forecasting future Time Series values is a quite common problem in practice. Predicting the weather for the next week, the price of Bitcoins tomorrow, the number of your sales during Christmas and future heart failure are common examples.

Time Series data introduces a "hard dependency" on previous time steps, so the assumption that independence of observations does not hold.

Time series analysis refers to the analysis of change in the trend of the data over a period of time. Time series analysis has a variety of applications. One such application is the prediction of the future value of an item based on its past values. In this, we will see how we can perform time series analysis with the help of a recurrent neural network (RNN).

3. DESCRIPTION

Deep Learning and time-series data analysis with the aim to build the simplest Long Short-Term Memory (LSTM) recurrent neural network for the data and visualization of data as analysis of data.

4. METHODOLOGY

In order to classify the data, we need to collect the data. In our case, the dataset was provided by the website that has hosted the competition. The data needs to experience different pre-processing steps which makes it more machine reasonable than its past structure.

The common methodology for classification involves-

a. Collection of data
b. Analyzing Dataset
c. LSTM Prediction

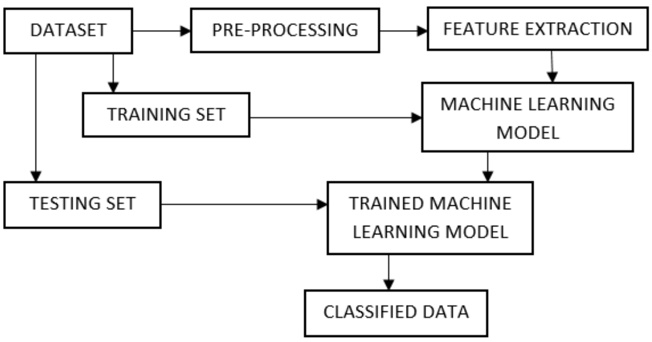


Figure 4.1: Methodology

4.1 Collection of Data

http://archive.ics.uci.edu/ml/datasets/Individual+household+electric+power+consumption

Attribute Information:

1. Date: Date in format dd/mm/yyyy
2. Time: time in format hh:mm:ss
3. Global_active_power: household global minute-averaged active power (in kilowatt)
4. Global_reactive_power: household global minute-averaged reactive power (in kilowatt)
5. Voltage: minute-averaged voltage (in volt)
6. Global_intensity: household global minute-averaged current intensity (in ampere)
7. Sub_metering_1: energy submetering No. 1 (in watthour of active energy). It corresponds to the kitchen, containing mainly a dishwasher, an oven and a microwave (hot plates are no t electric but gas powered).
8. Sub_metering_2: energy submetering No. 2 (in watthour of active energy). It corresponds to the laundry room, containing a washing-machine, a tumble drier, a refrigerator and a light.
9. Sub_metering_3: energy submetering No. 3 (in watthour of active energy). It corresponds to an electric water-heater and an air-conditioner.

4.2 Visualization

```
[ ]  df.Global_active_power.resample('D').sum().plot(title='Global_active_power resampled over day for sum')
     #df.Global_active_power.resample('D').mean().plot(title='Global_active_power resampled over day', color='red')
     plt.tight_layout()
     plt.show()

     df.Global_active_power.resample('D').mean().plot(title='Global_active_power resampled over day for mean', color='red')
     plt.tight_layout()
     plt.show()
```

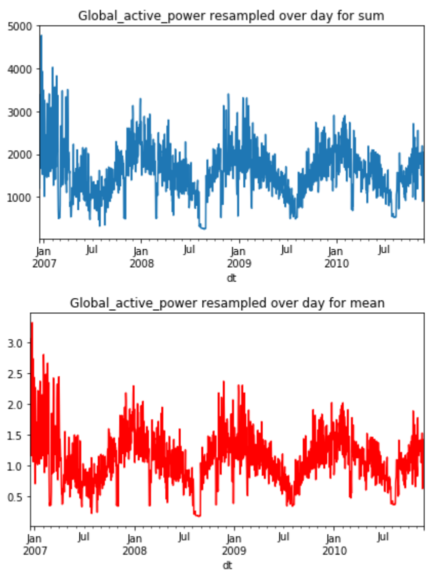
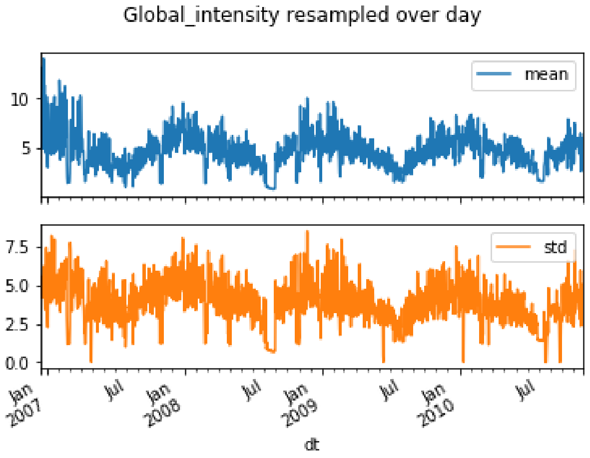


Figure 4.2.1: Global Active Power resampled by (a) Day for sum (b) Day for mean

```
[ ] ### Below I show mean and std of 'Global_intensity' resampled over day
    r = df.Global_intensity.resample('D').agg(['mean', 'std'])
    r.plot(subplots = True, title='Global_intensity resampled over day')
    plt.show()
```




```
[ ] ### Below I show mean and std of 'Global_reactive_power' resampled over day
    r2 = df.Global_reactive_power.resample('D').agg(['mean', 'std'])
    r2.plot(subplots = True, title='Global_reactive_power resampled over day', color='red')
    plt.show()
```

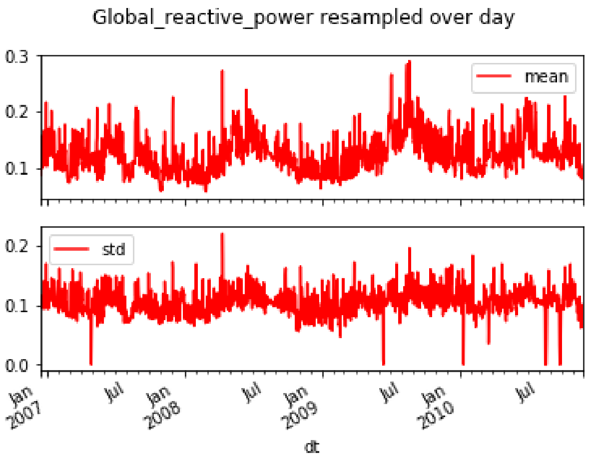


Figure 4.2.2: (a) Global Intensity resampled over day
(b) Global Reactive Power resampled over day

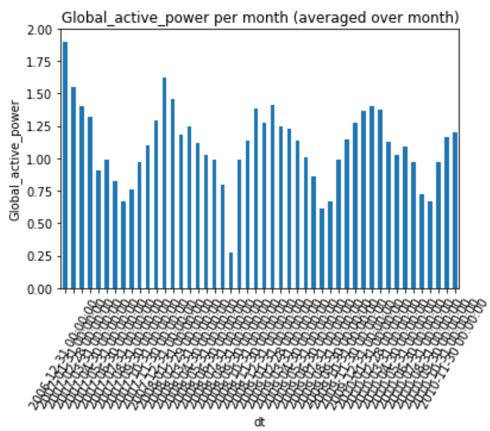


Figure 4.2.3: Global active power per month (Average over month)

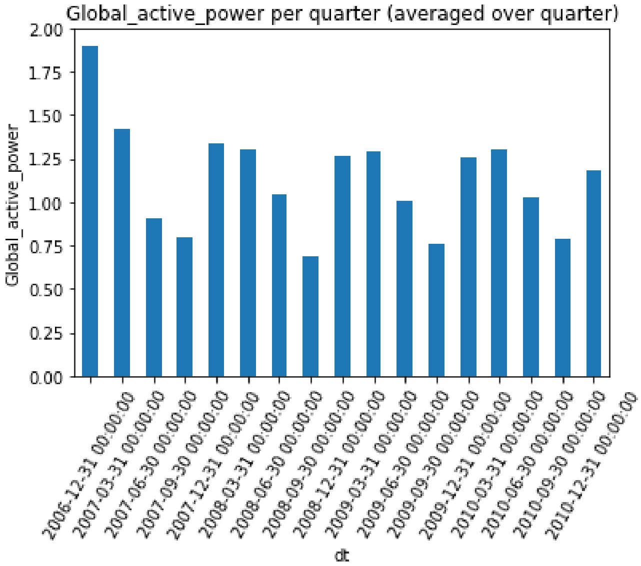


Figure 4.2.4: Global active power per month (averaged over month)

It is seen from the above plots that the mean of 'Voltage' over month is pretty much constant compared to other features.

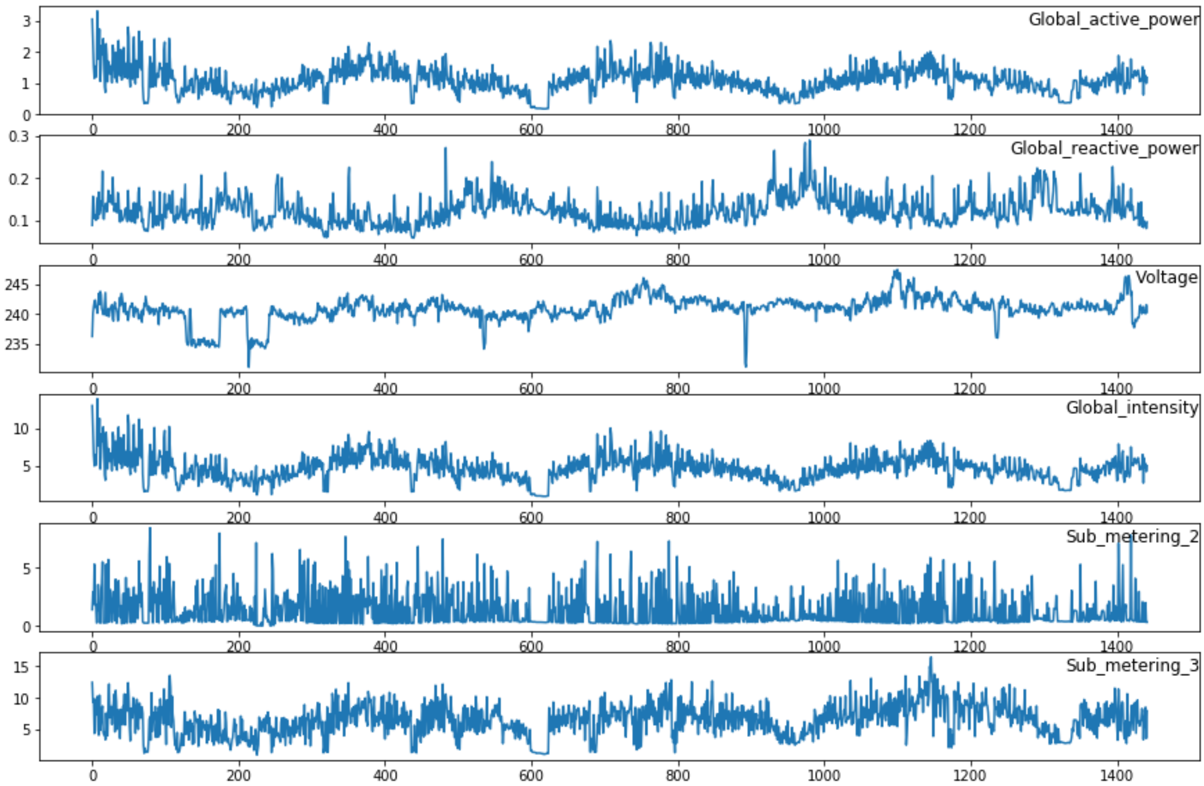


Figure 4.2.5: Compare the mean of different features resampled over day.

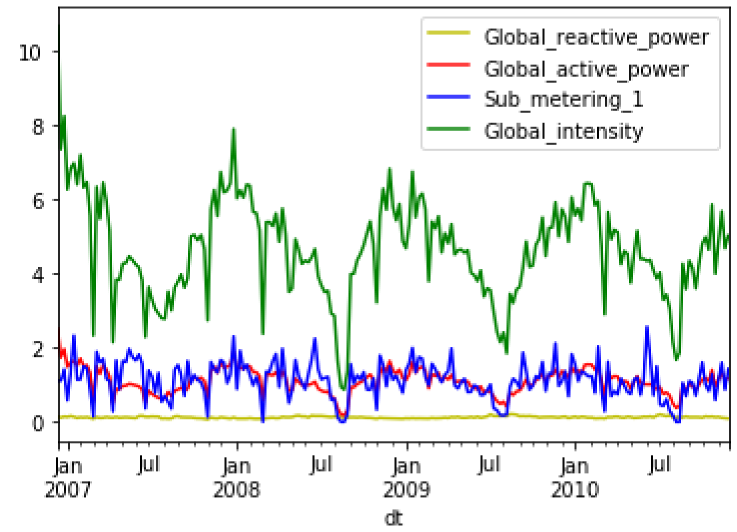


Figure 4.2.6: Resampling over week and computing mean

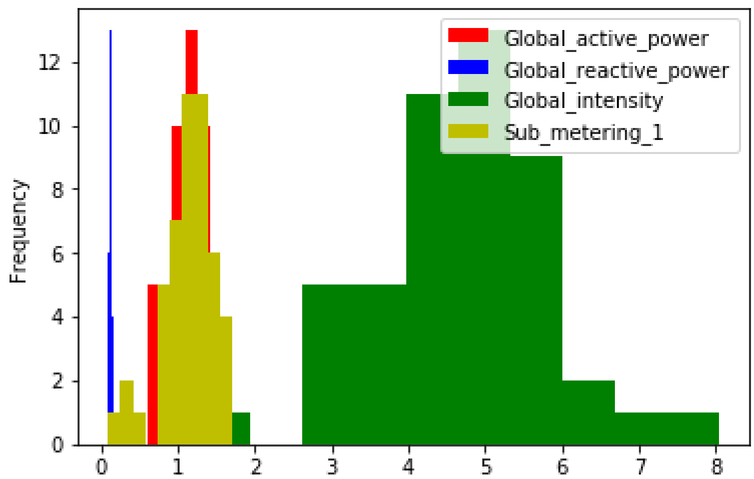


Figure 4.2.7: Histograph plot of the mean of different feature resampled over month

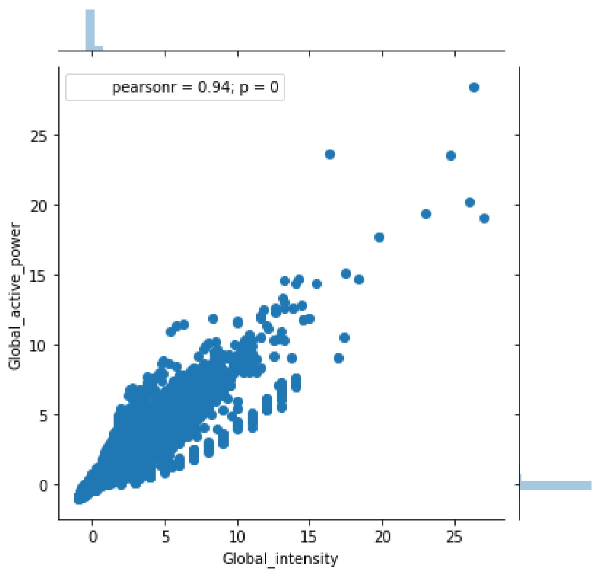


Figure 4.2.8: Correlations between 'Global_intensity', 'Global_active_power'

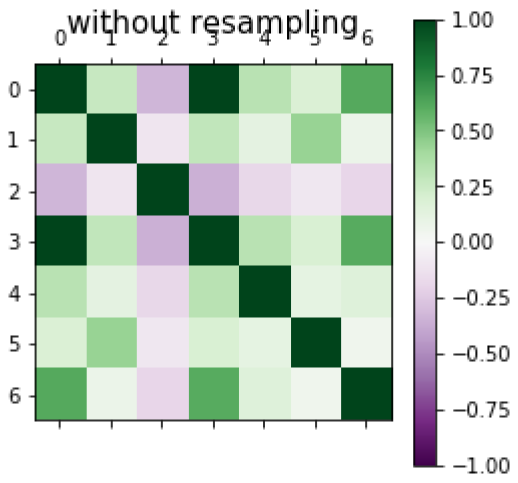


Figure 4.2.9: Correlations among columns

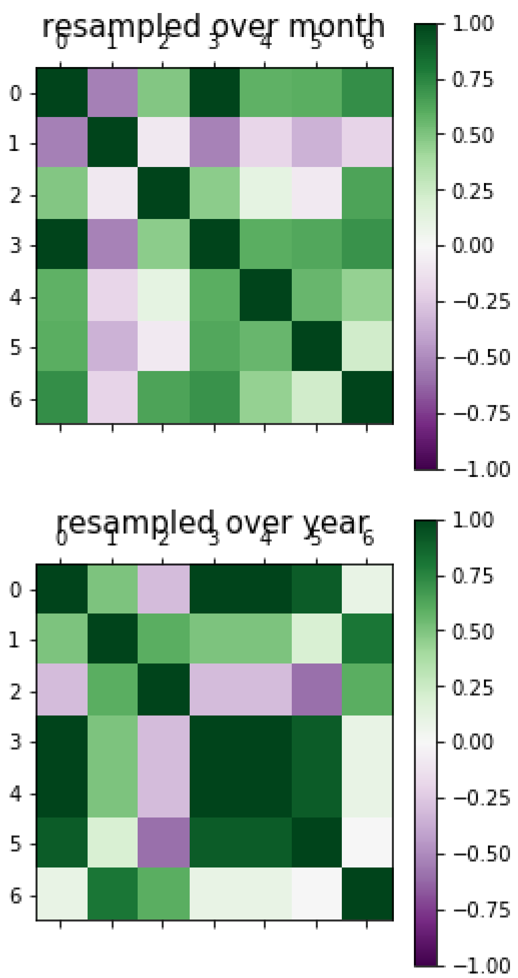


Figure 4.2.10: Correlations of mean of features resampled (a) Over Month (b) Over Year

4.3 Model

4.3.1 Model Architecture

- LSTM with 100 neurons in the first visible layer
- Dropout 20%
- 1 neuron in the output layer for predicting Global_active_power.
- The input shape will be 1 time step with 7 features.
- I use the Mean Absolute Error (MAE) loss function and the efficient Adam version of stochastic gradient descent.
- The model will be fit for 20 training epochs with a batch size of 70.

4.3.2 LSTM

In a traditional recurrent neural network, during the gradient back-propagation phase, the gradient signal can end up being multiplied a large number of times (as many as the number of time steps) by the weight matrix associated with the connections between the neurons of the recurrent hidden layer. This means that, the magnitude of weights in the transition matrix can have a strong impact on the learning process.

These issues are the main motivation behind the LSTM model which introduces a new structure called a memory cell (see Figure 1 below). A memory cell is composed of four main elements: an input gate, a neuron with a self-recurrent connection (a connection to itself), a forget gate and an output gate. The self-recurrent connection has a weight of 1.0 and ensures that, barring any outside interference, the state of a memory cell can remain constant from one timestep to another. The gates serve to modulate the interactions between the memory cell itself and its environment. The input gate can allow incoming signal to alter the state of the memory cell or block it. On the other hand, the output gate can allow the state of the memory cell to have an effect on other neurons or prevent it. Finally, the forget gate can modulate the memory cell's self-recurrent connection, allowing the cell to remember or forget its previous state, as needed.

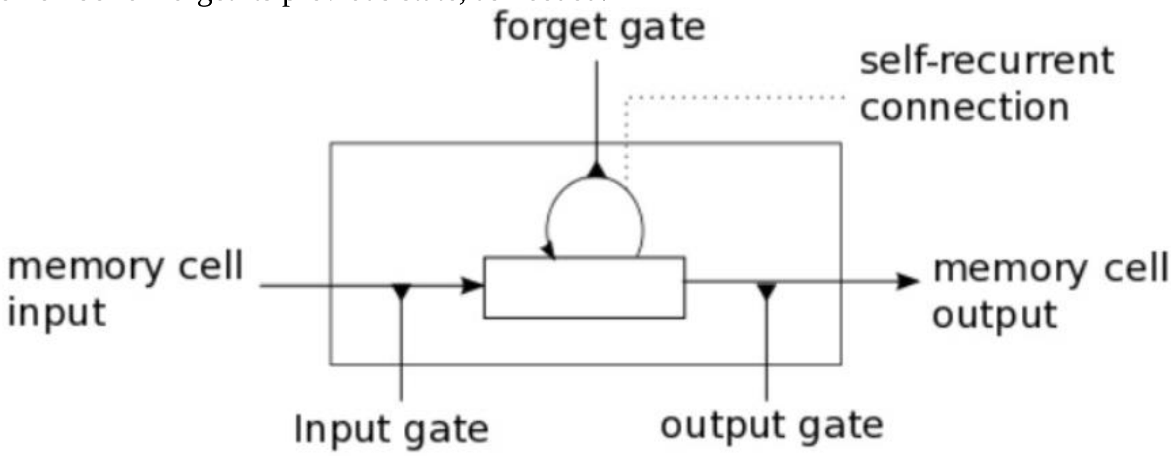
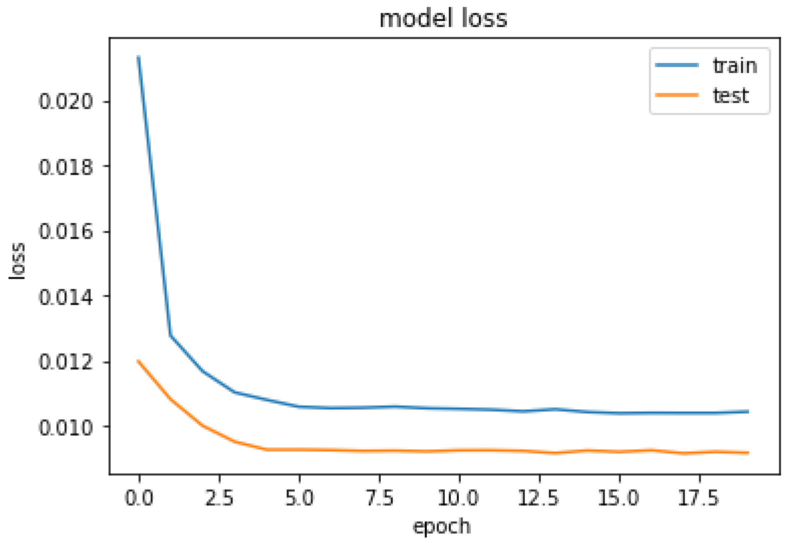


Figure 4.3.2.1: LSTM Architecture

5. EXPERIMENTAL RESULT




```
Test RMSE: 0.616
```

Figure 5.1: Summarized history for loss

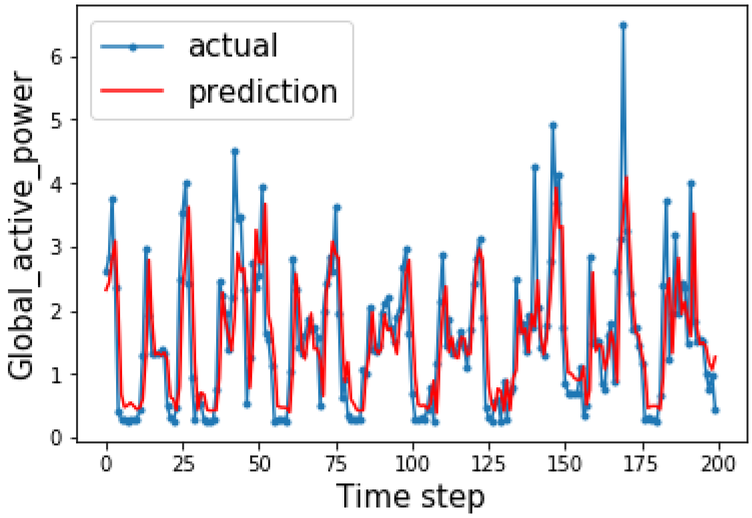


Figure 5.2: Comparison of Global Active Power and Time step of predicted and actual data.

6. REFERENCES

https://www.analyticsvidhya.com/blog/2017/12/fundamentals-of-deep-learning-introduction-to-lstm/

https://en.wikipedia.org/wiki/Long_short-term_memory#:~:text=Long%20short-term%20memory%20%28%20LSTM%29%20is%20an%20artificial,sequences%20of%20data%20%28such%20as%20speech%20or%20video%29.

https://www.tutorialspoint.com/data-analysis-and-visualization-in-python

https://gilberttanner.com/blog/introduction-to-data-visualization-inpython

https://www.coursera.org/learn/python-for-data-visualization