A Comparative Study of Deep Learning and Machine Learning models for Forecasting Electricity Production

by

Peter Sunny Shanthveer Markappa

This thesis has been submitted in partial fulfillment for the degree of Master of Science in Artificial Intelligence

in the
Faculty of Engineering and Science
Department of Computer Science

May 2022

Declaration of Authorship

This report, A Comparative Study of Deep Learning and Machine Learning models for Forecasting Electricity Production, is submitted in partial fulfillment of the requirements of Master of Science in Artificial Intelligence at Munster Technological University Cork. I, Peter Sunny Shanthveer Markappa, declare that this thesis titled, A Comparative Study of Deep Learning and Machine Learning models for Forecasting Electricity Production and the work represents substantially the result of my own work except where explicitly indicated in the text. This report may be freely copied and distributed provided the source is explicitly acknowledged. I confirm that:

- This work was done wholly or mainly while in candidature Master of Science in Artificial Intelligence at Munster Technological University Cork.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at Munster Technological University Cork or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this project report is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: Peter Sunny Shanthveer Markappa

Date: 08-May-2022

MUNSTER TECHNOLOGICAL UNIVERSITY CORK

Abstract

Faculty of Engineering and Science
Department of Computer Science

Master of Science

by Peter Sunny Shanthveer Markappa

Electricity is the key factor for the economic development of a country, Electric Production companies strive hard to meet the requirement. The production of electricity is directly proportional to its consumption and this is related to different factors like an increase in the industries which can be the main reason for the increase in the population of immigrants that leads to consumption of more electricity. Hence, Electricity Industry makes strategic decisions to manage their production and consumption. Over the past decade, Ireland is giving more attention to moving towards green energy. This research aim is to forecast the production of electricity in Ireland and analyze several aspects that may affect the prediction for a better forecast using traditional, machine learning, and deep learning algorithms to get a better prediction model, prior to that data pre-processing is the foremost task to be considered before implementation and evaluation of the model. Finally, the performance of the implemented models will be evaluated using statistical error evaluation metrics.

Acknowledgements

I am thankful to Dr. Mohammed Hasanuzzaman for his sincere efforts in guiding me to pursue a Research Project. ...

Contents

List of Figures

Abbreviations

ARIMA	Autoregressive Integrated Moving Average
MAPE	Mean Absolute Percentage Error
ANFIS	Adaptive Neuro-Fuzzy Inference System
SVM	Support Vector Regression
LS-SVM	Least Square Support Vector Machine
SMA	Simple Moving Average
WMA	Weighted Moving Average
SES	Simple Exponential Smoothing
HL	Holt Linear Trend
HW	Holt -Winter
CMA	Centered Moving Average
MSE	Mean Square Error
RMSE	Root Mean Square Error
GGM	Group Grey Model

Dedicated to Almighty God, My Parents for their blessings and prayers,

My brother John and Kamlesh for standing beside me all the way in this course,

Cousin Khyati, Apoorva, Akash and Sister-in-law Divya for enduring moral support,

Friends (Archit, Rajeev, Shekar) for their moral support throughout the course and Ravindra Kulkarni for guidance,

Friends in India Veeresh, Ramesh, Praveen.

Finally to my Nephew 'David'

...

Chapter 1

Introduction

1.1 Background and Motivation

The Electricity Production by any company or the Government is volatile market for them, they take risk to manage this production and the supply as it cannot be stored like other commodities, they can use the batteries to store the electricity but the installation and maintenance is high. So the companies/government who produces the electricity strive hard to produce it in right amount and make supply, to generate the electricity in the right amount organisation invest their time in money to forecast it by looking into the past data. Although domain knowledge in very crucial in this industry for accurate and reliable forecasting which will provide crucial decision for their production, along with analysing the data the generation of electricity depends on various other external factors which has to be considered into an account while making prediction.

Time series prediction can be done using many methods, both classical and modern, In this thesis the data set from EIRGrid Group is used which contains the data of every 15 minutes, the data was available of last two months and another data set from SEAI is also used which is having the data of every month production of electricity from January 2010 till march 2022.

For methodology comparison, it must be assumed that variables in the data set have predictive power. Whether or not this is correct can be determined by how well our results are in general and across methods. As a result, we can see how different method implementations affect the results. As the base of the method, LSTM is used using Recurrent Neural Network along for both fifteen minutes power production and monthly production data sets. The main methodology hypothesis is combined with LSTM, Auto Regressive and modern machine learning framework like XGBOOST, to make the forecasting more efficient. Different error metrics are used to evaluate the prediction such as RMSE, MAE,MAPE for both the data set, all the results are described in the methodology section.

Chapter 2

Research Proposal

2.1 Literature Review

Much research has been done to compare these strategies with other strategies. Souhaib Ben Taieb and Gianluca Bontempi are two researchers who frequently discuss this topic. The article NN5 Review and Compare Strategies for Predicting Time Series in Anticipation of Multiple Steps Based on NN5 Prediction Competition compares and summarizes different methods from different sources [1].

Time Series Forecasting is accomplished by making scientific predictions based on time-stamped historical data. This process entails developing models through statistical analysis and then using them to make observations that will drive future decision-making. Among them are several models for forecasting observational data. ARIMA is the most popular model [2] [3] for forecasting, this model was used to forecast the wind speed using root mean square error, the mean absolute percentage error, and the mean absolute error [3], it is also used to forecast the consumption of electricity by using Mean Absolute Percentage Error (MAPE) to forecast the accuracy [2] [4]. Adaptive Neuro-Fuzzy Inference System (ANFIS) was used to predict the demand for electricity by collecting the historical data from the Electricity Distribution Company of Ghana [5] and the performance of ANFIS was assessed by comparing Support Vector Regression (SVM), Least Square Support Vector Machine (LS-SVM) and ARIMA, the predictive accuracy of ANFIS was improved which was relied on quality data for training and reliable setting of hyper-parameters tunning [6]. Electricity consumption forecasting was conducted using Simple Moving Average (SMA), Weighted Moving Average (WMA), Simple Exponential Smoothing (SES), Holt Linear Trend (HL), Holt -Winter (HW), and Centered Moving Average (CMA). CMA produced the lowest Mean Square Error (MSE) and Root Mean Square Error (RMSE) whereas the decrease in the population produced HW forecasted low trend at the same time CMA increased. [2] [5]. Grouped Grey Model (GGM(1,1)) was used in modeling medium-term forecasting of consumption of electricity. [6].

Apart from the traditional methods for forecasting, the prevalent technique Deep Learning was used in this, the feature is extracted automatically then learn from its filters before giving designated results [7]. Due to repetitive behavior in the usage of electricity by the customers tends most of the statistical / Machine learning models give high accuracy [8]. Loading the entire electricity data set can result in out-of-date historical information, which can affect the effective cluster status at a given time stamp. To address this issue, a network-based model was developed in which the data set is divided into different time-stamps. Customers' load profiles are subdivided into smaller groups with homogeneous electricity consumption, allowing forecasting tasks to be performed more efficiently. Researchers taught that clustering data and running it through an auto-regressive model would improve forecasting accuracy, but this was not the case when the data series were either stationary or quasi-stationary, and this conditional was not validated in real-world applications. [8] [9].

Carbon emission can be reduced when consumption of electricity is reduced researchers have researched many hybrid methodologies hybrid methodologies [7] by splitting the data into groups the same process was approached using network-based [8] where short-term load forecasting was implemented by dividing the time series model into trend item, period item, holiday item, and error item by applying smoothing algorithm because the of inconsistency of data in the smart grid [10]. Extensive research was conducted on forecasting methods researcher proposed that considering the time lag of economic factors on load, changed the economic situation on the electricity forecasting [11]. Modern power grids are confronted with the difficulty of rising renewable energy penetration, which is stochastic in nature and necessitates excellent demand forecasting in order to deliver the best possible power supply. While many recent studies have looked into both macro and micro scale short-term load forecasting (STLF), a thorough examination of the effects of electrical demand aggregation size on STLF is rare, especially with large sample sizes, where it is critical for optimal sizing of residential micro-grids, demand response markets, and virtual power plants. Furthermore, using this methodology study was conducted of STLF with five aggregation levels (3, 10, 30, 100, 479) with a sample size (1,4,15,47,159) per level. On applying different Deep Learning methods for STLF and after fine-tuning with methodological sensitivity analysis with early stopping, the test result reveals that MAPE of (247-3.31%) was achieved which was close to country-level highest aggregation, one step ahead of Deep Neural Network in this experiment achieved the highest performance which was followed by Bi-directional Gated recurrent unity which is fully connected layers (Bi-GRU-FCL) [12].

The orthodox method ARIMA performs well with the linear aspect but down the line, it fails to account for non-linearity aspects furthermore, it is visa-versa with a neural network that accounts for non-linearity aspects ignoring the linearity aspect with the time series analysis [2] [13]. To overcome the energy maintenance problem in Ukraine and Kazakhstan (Maksat) conducted research using classical statical ‘ad hoc’ along with advance ensemble methods and neural network to predict considering the case of wholesale energy transmission company [14].

Widely use classical time series technique ARIMA or ARMA, applies Box-Jenkins Methodology, and they predict the future values based on the linear combination of the previous values and disturbance. Alternative to those models Exponential smoothing approach is the next powerful model in time series for univariate data, compared to ARIMA and ARMA this model is used in various different fields of studies due to its flexibility and also reliability to forecast with low expenses.

The simplest amongst all time-series methods is SMA in which m number of data from a series consists of n number of data is the average values of m of the previous values whereas SES gives more weights to the recent data than the previous ones. The process of smoothing the data in SES is the same as in SMA, later used in forecasting the future value. Whereas if the trend in time-series is linear then HTL is a useful method as it will double the exponential smoothing, which adds the second exponential smoothing parameter to smooth the trend in time-series data. furthermore, to this method, HW is triple exponential smoothing where in here HW will add the seasonal component when compared to HLT, this technique could be used during the seasonal data in time-series. Sometimes time-series data exhibits seasonal fluctuations in such circumstances it is necessary to discard the fluctuation to leave deseasonalized data. to use this deseasonalized data first step is to proceed with SMA of one year later the result is averaged using CMA, in this process specific seasonal for each month is computed by dividing the actual data by CMA. The third step is to calculate the seasonal index of each month m as the mean of specific seasonal for month m throughout the year. To remove the seasonal variation whole actual data for each month is divided by the seasonal index of the particular month.

2.2 Research Aim

The aim of this thesis is to make comparative analysis of different models of Machine Learning and Recurrent Neural Network

2.3 Research Questions

The purpose of this project is to get a better understanding of electricity production forecasting by answering the following research questions.

1. Is it possible to forecast a country's electricity production using Deep Learning and machine learning algorithms based on various factors when compared with traditional ones?
2. How much of an impact does each factor have on fluctuations in the consumption of electricity in different seasons?
3. Is it true that different optimisation techniques can enhance the performance of the model?

2.4 Research Objectives

The base goal of this research is to forecast the production of Ireland Electricity and analyze the performance of various models along with the impact of seasonal consumption factors on the projection.

2.5 Methods used for Time Series Forecasting

Forecasting or forecasting future value has been done in some form for a long time. The motivation is straightforward. It is very useful in the financial business to predict the future by correctly predicting the cost and profits of a project, or by correctly predicting future stock prices.

Classic time series prophecies are frequently linked to economic theory on the subject. For example, if the Efficient Market Hypothesis assumes that the stock price is a random walk, then predicting tomorrow's price using a random walk will be today's price (Adland, Alizadeh). That is, if the current price correlates with the previous day's price, i.e., the factor is adjusted to the previous price value in the trend. When a time series correlates with itself, this is referred to as autocorrelation.

This autocorrelation can be explained by employing an autoregressive model, which is common in traditional prediction methods. These models are either univariate, predicting only price, or multivariate, predicting a small set of carefully selected variables. They operate by forecasting the incremental steps and then using the predicted values to forecast the next value.A recursive approach to multi-level prediction is used in this case.

Chapter 3

Research Methodology

3.1 Data

The first step in forecasting is gathering information; the collected data must be specific to the research that has been conducted, and it is critical to ensure that the completed data is collected during this stage; additionally, care must be taken to collect it legally or from open-source citations.
In general, data can be of three types

1. First-Party Data – if the data is collected directly from any organization by the researcher

2. Second Party Data – if the data is shared by a second organization with the researcher

3. Third-Party Data- if the data is aggregated and sold or rented by the organization where it does not have any connection data owner.

For this thesis two type of data were collected, one is from EIRGrid, this data consists of every fifteen minutes power generation data of two which is in MW(Mega Watts), this data from EIRGrid is from whole Ireland including other Islands, and the other data is Collected from SEAI, which is montly data from January 2010 to March 2022 that is in Giga Watts. The data that is used from the EIRGrid is been constantly used by the company on day-to-day make the forecasting, that public data of only two is made visible.

3.1.1 Descriptive Statistics

If the forecasting is done on every data time series data, it is difficult to acquire the huge amount of data because for whole year the data available will be 365/366 observations, and also most of the time series changes over a period of time. The electricity requirement during the period of 2010 to 2015 in Ireland was completely different from the present day or year requirement, and the monthly data it is even more difficult as only 12 information will be accumulated over period of one year. For this thesis fifteen minutes data is also considered for larger amount of data, but the data that is been used is of only two month, so with this type of data it is difficult to identify the seasonality also such data fluctuates very much, as the forecasting electricity also dependents on many other external factors like seasons, timing of usage, it can analysed that the amount of electricity used during the winter month from November to January will not be the same June to August. so the data that is gathered has information from mid of January to mid of March which make the prediction accuracy to decrease due to the algorithm will not be able to follow the seasonality.

From figure 3.1 which is snipped from the jupyter notebook, we can see that total there are 5668 information with average fluctuation of 765 mega watts, 75 percentage the electricity production is more than five thousand range and the maximum amount of electricity generated is 6884 mega watts of electricity.

let us look into another line plotting for better understanding of the data. From the figure 3.3 it can be seen that one peek, and from the Trend it can said that

	AG_MW
count	5668.000000
mean	4785.447601
std	765.016435
min	2584.000000
25%	4176.750000
50%	4791.000000
75%	5343.250000
max	6884.000000

FIGURE 3.1: Description of the Data

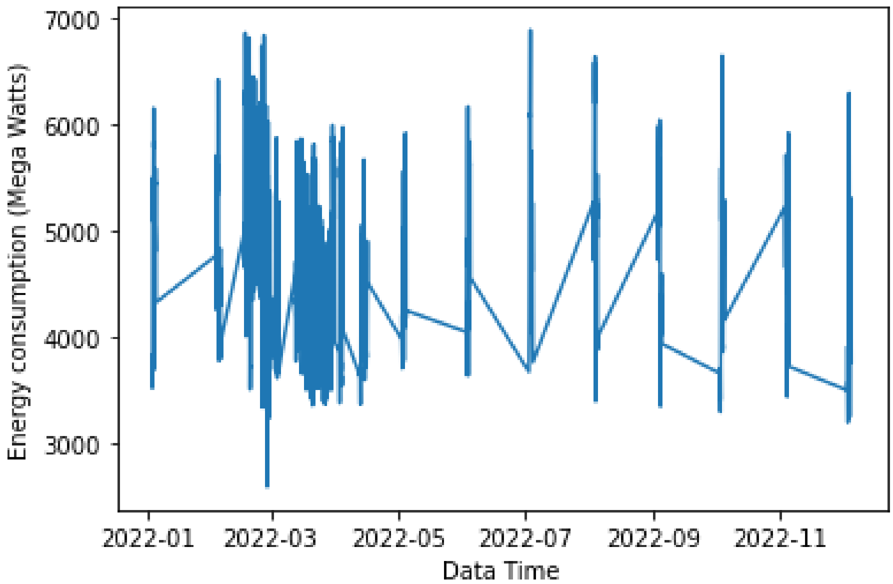

FIGURE 3.2: Plotting the Data with Time

there is constant form of wave everyday, that is the it reaches to the peek once in a day, so it can said that the maximum amount of consumption of electricity will be in night and in day the peek falls, but another factor also has to be considered before assuming anything that during working hours even though electricity is not consumed by homes in day time but most of business are open, but it can be concluded that there is one particular pattern in the trend. Despite this pattern it can be seen that there is lot of noise in the the data which not explained by the daily seasonality for this I tried to model using other variables in the data sets and feature engineering.

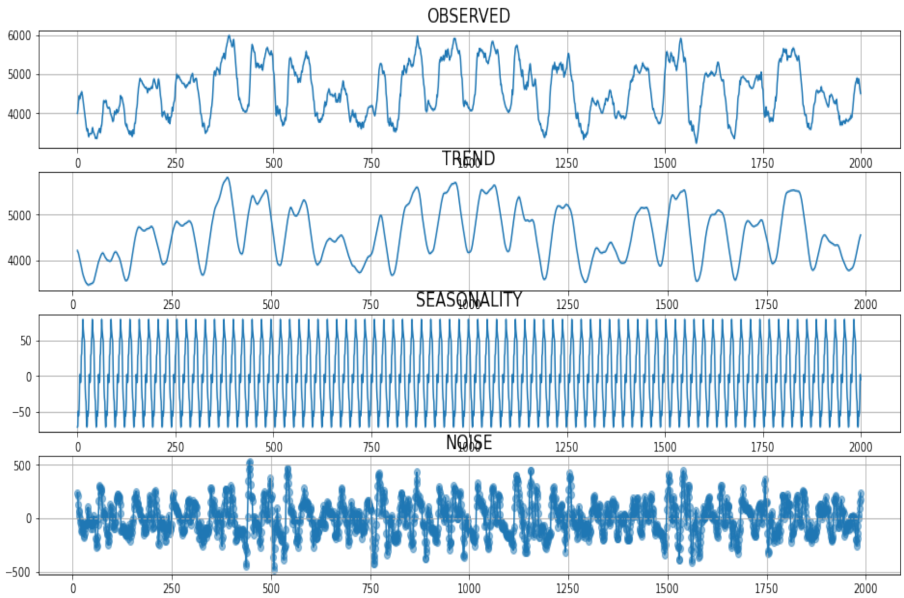

FIGURE 3.3: Plotting the Data with Time

Looking into figure 3.4, we can say with high confidence that our data is auto-regressive and that we can improve the performance of our model using lags after plotting this auto correlation graph. In other words, the Energy prediction can be explained using the values from the previous hour and day.

3.1.2 Stationarity

All observations in the stationary dataset are not seasonal or trendy and are independent of the time series in which they were observed (Hyndman and Athanasopoulos, 2018). You can improve the stability of the temporary time series by differentiating the values so that each value represents the difference from the previous day: y'(t) = y(t) - y(t-1). Some predictive methods may improve performance when using different datasets, while other methods have built-in capabilities to explain trends and seasonality. B. A multivariate model with dummy variables that take into account time-specific events, seasons, or days of the week. Knowing the steady-state properties of a dataset will help you develop better predictive models. Therefore, before you build your model, you need to determine if your dataset contains such functionality.

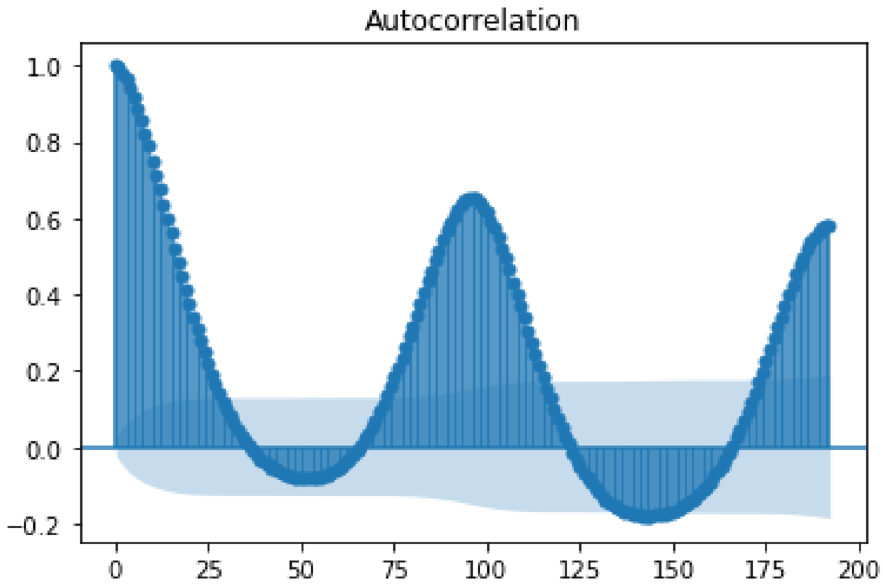

FIGURE 3.4: Auro Corellation Graph (192 lags-Two Days)

Before moving into next part of the thesis I also wanted to check Kwiatkowski-Phillips Schmidt-Shin Test and Agumented Dickey-Fuller Test which gives different information on the data.

3.1.3 Unit Root Test

There are many test in statistics to identify whether the consumption of energy in stationary or not like Kwiatkowski Phillips Schmidt Shin(KPSS) Test, Augmented Dickey Fuller (ADF) and Phillips-Perron. For this fifteen minutes I will be implementing KPSS and ADF test metrics as it as proved to show better performance with ARIMA model on average(Hyndman, 2014) [15]. These tests, in general, struggle with size distortion and may reject the null hypothesis even if it is true (Zivot, Wang) [16].

When the test statistic exceeds the critical value of 0.463 at 5 percentage, the null hypothesis (H0) of the KPSS test is satisfied and rejected. The KPPS test produces a test statistic of 30-32 with many lags, indicating that the null hypothesis was rejected at the 1 percentage level. This confirms the autocorrelation shown in Figure 3.4's autocorrelation plot. It does, however, contradict KoekebakkerKoekebakker, Steen; Adland, Roar; Sødal, Sigbjør.(2006) in terms of stationary freight rates [17].

According to the Auto co-relation plot, energy consumption needs to be differentiated to be more steady. Therefore, if the purpose of this analysis was inference, you might have distinguished the data. In our case, we want to make the most accurate predictions possible, so it is still arguable whether we need to distinguish the data to allow for more accurate predictions. Therefore, if model constancy is an important concern, test with or without differentiation and choose the model with the best performance.

The Kwiatkowski-Phillips-Schmidt-Shin Test (KPSS) checks for the stationarity of a timeseries by testing the null hypothesis that the data is stationary about a trend. A p-value above 0.05 indicates the data is trend-stationary. We'll set up metric tables for our differenced data and view the results. The large p-value is

```
Test Statistic          0.178888
p-Value                 0.100000
Number of Lags          2.000000
Critical Value (10%)    0.347000
Critical Value (5%)     0.463000
Critical Value (2.5%)   0.574000
Critical Value (1%)     0.739000
dtype: float64
```

FIGURE 3.5: KPSS Metrics:

above 0.05, which means our differenced data is trend-stationary. The null hypothesis is accepted.

```
Dickey-Fuller Metrics:

Test Statistic              -3.573080
p-Value                      0.006297
Number of Lags               0.000000
Number of Observations      10.000000
Critical Value (1%)         -4.331573
Critical Value (5%)         -3.232950
Critical Value (10%)        -2.748700
dtype: float64
```

FIGURE 3.6: Dickey-Fuller Test Metrics:

The second of our stationarity tests is the Augmented Dickey-Fuller (ADF) Test. This test assumes the data is non-stationary as the null hypothesis. A p-value below 0.05 indicates a stationary timeseries. This is opposite the KPSS test so care must be taken when making conclusions. Here we see the small p-value is below 0.05 so the null hypothesis is rejected.

Both ADF and KPSS tests verify that the data is stationary so a second-order differencing is not needed.

3.2 Data Preparation

Pre-processing is one of the data mining steps where the data is cleaned which helps to understand the data clearly. Real-world data will never be perfect and cannot be used directly for prediction. In this section, we examine the raw data, dependent variables, and their relationships with other variables in the raw data before delving into the mission values.

3.2.1 Handling Missing Values

The purpose of dealing with missing values is to make available data available in different ways while managing the trade-off between data quality and the number of observations. Depending on the type of data, their distribution, purpose, and the model used, processing the missing observations can have different effects on the results. In order to deal with the missing data, where 4 of the energy information was NaN is filled with padding method, with very less amount of missing values the data could be deleted to deal with it but as this information is of time series every bit of information will impact on model performance.

3.3 Feature Selection

This is a vital section for any high dimensional data which usually requires feature selection before implementation of any methods, it enables the learning algorithm to train faster which will help in improving the accuracy also reduce the complexity of the model and computational cost. Statistical-based feature selection is one of the methods in which the relationship between input variable and output is identified which has the strongest relationship with the target variable. This methodology could be effective and fast but it all lies on the data type of those variables.

Below chart provides hierarchy of feature selection techniques

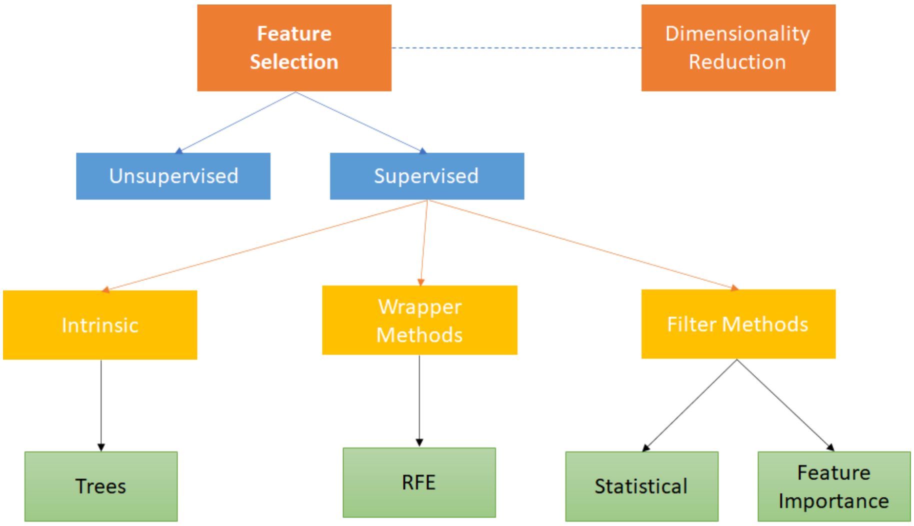

Figure 3.7: Feature Selection Hierarchy

With respect to this these the models used here are uni-variate which has built in feature selection, and the main aim is of comparison of methods, feature selection is not much focused.

3.4 Aggregation

It is critical to consider the aggregation of the data set, which changes the coarseness of the data. If forecasting or predicting the upcoming week is required, both historical daily data and weekly data could be used. One method for aggregating data is to take the mean of the observations in the aggregated period, while the second method is to sum the change of all observations. Forecasting with either method yields different results.

Aggregating the data set from daily observations to weekly or monthly can have an impact on forecasts in both positive and negative ways. On the one hand, aggregating will save computational time because the data set will be smaller. Furthermore, aggregation can provide models with access to data that has been registered at a different level of granularity. This can assist the forecast in making use of data that it would not otherwise have access to. Aggregating, on the other hand, means losing information that could be useful in making a precise forecast.

3.5 Implementation and Evaluation

3.5.1 Implementation

Different experiments were made on both of this data by using Long Short-Term Memory(lstm) which is Recurrent Neural Network(RNN). LSTM addresses the major short memory issues that plague recurrent neural networks. LSTMs use a set of "gates", each with its own RNN, to hold, forget, or ignore data points based on a probabilistic model.
LSTMs also aid in the solution of vanishing gradient explosion and disappearance problems. These issues arise as a result of the neural network's repeated weight adjustments. The gradient increases and decreases as the epoch repeats. With each adjustment, the network's slope becomes stronger in either direction. This combination results in a gradient that is either too large or too small. Traditional RNNs have significant drawbacks such as gradient explosions and disappearances, but the LSTM architecture significantly mitigates these issues.
The model then uses the prediction to predict the next value in the sequence. Each prediction introduces a flaw into the model. To avoid a gradient explosion, the input and output values of the pregate are (usually) "squeezed" using the sigmoid and tanh activation functions. The LSTM architecture is depicted in the diagram below.

The Min-Max scaler is used to normalize the input features/variables. All features will be transformed into the range [0,1], which means that the minimum and maximum value of a feature/variable will be 0 and 1, respectively.

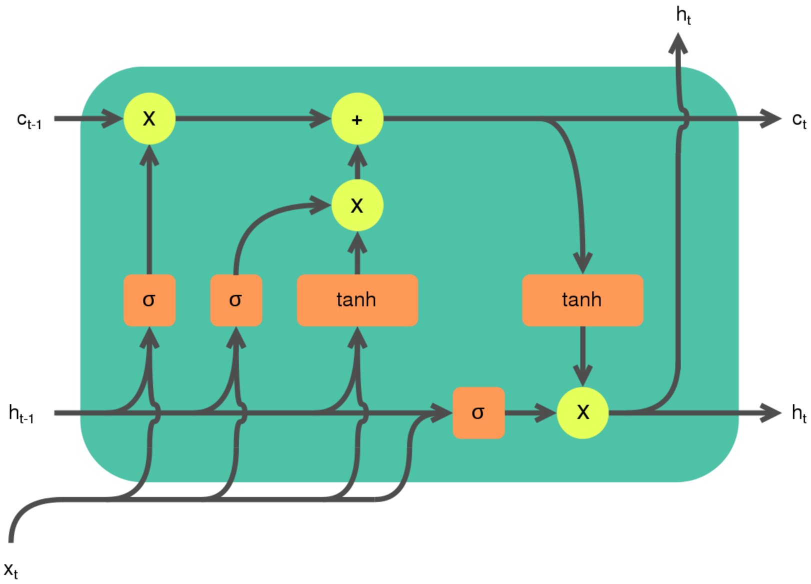

FIGURE 3.8: LSTM Architecture

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

FIGURE 3.9: Min-Max Scaler

The question always arises Why to normalize prior to model fitting? The basic concept of normalization / standardization is always the same. Variables measured at different scales do not contribute equally to model fit and trained functions and can introduce bias. To address this potential issue, feature-related normalization B.MinMax scaling included. This is especially useful for some ML models, such as: B. Multilayer perceptron (MLP). Backpropagation is more stable and faster if the input features are minimax scaled (or generally scaled) instead of using the original unscaled data. This must also be considered that, Scaling has little effect on tree-based models, but non-tree models like SVMs and LDAs are frequently affected.

```
given the Array = : [0.55232558 0.55813953 0.57651163 0.57767442]
Predict the y = : [[0.56511628]]
```

FIGURE 3.10: One Hour Prediction

Considering all the last four energy production values from the data set which sums up to one hour next fifteen minutes production was made which predicted

0.5611628 means that the model was predicting nearer to the real value. Next experiment was made by taking the lag of fourty eight that make prediction of twelve hours, that is data of last twelve hours was taken to predict next value. For this the data was pushed into neural network with Input of hundred neurons and output of one, the activation function which was used for input is 'tanh' and for the output is 'sigmoid' the optimizer used is adam and for the error calculation mean square error was used.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm_4 (LSTM)                (None, 100)               40800

dense_6 (Dense)              (None, 1)                 101

=================================================================
Total params: 40,901
Trainable params: 40,901
Non-trainable params: 0
_________________________________________________________________
```

FIGURE 3.11: Model Summary

After running for the 10 epochs the loss is dropped to 3.52 but there was slightly variation after epoch 4.

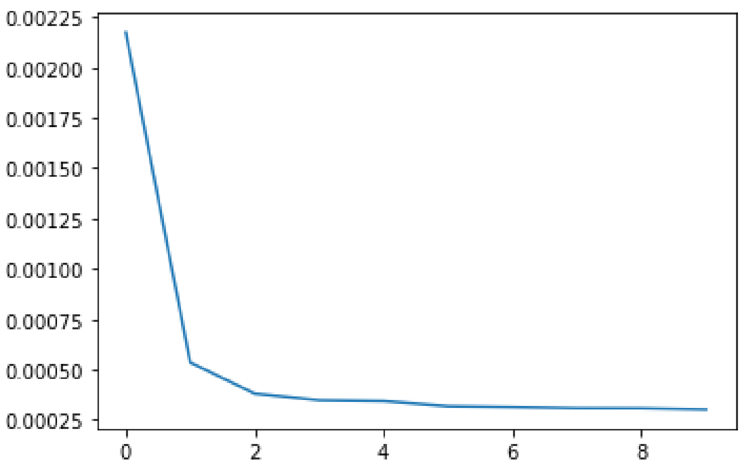

FIGURE 3.12: Loss-Plotting

From the figure 3.12 and figure 3.13 it can be seen that even though the loss as fallen for the iteration but the model has not performed well as the data is getting over fitted. Over fitting is Machine Learning is common thing that occurs where the training data learns and becomes over confident that it cannot make prediction on the test data. Finally the mean squared error for this model was 1626.116

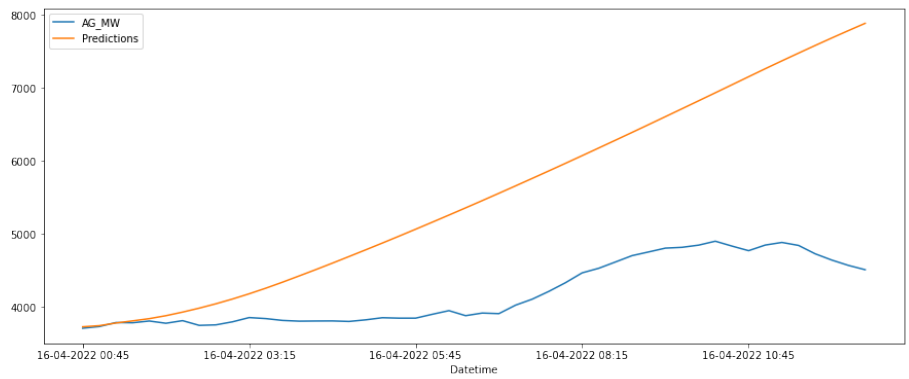

FIGURE 3.13: Prediction-Plotting

```
Mean Square Error 1626.116499506423
```

FIGURE 3.14: MSE

which is lot higher figure 3.14.

Similar experimentation with same number of layers using 'relu' activation function for the input layers and using adam optimiser for monthly energy production data with loss function used was mean squared error and 'sequential Model'.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 100)               40800

 dense (Dense)               (None, 1)                 101

=================================================================
Total params: 40,901
Trainable params: 40,901
Non-trainable params: 0
_________________________________________________________________
```

FIGURE 3.15: Modal Summary

For this time as the data is very less this model was made to run for fifty epochs, from the figure 3.16 we can see that there is huge amount of fluctuation in the loss graph, some times fluctuations is needed so that we can understand that the modelling is training better.

For this model lags of 12 was taken that is precious one year data was taken to make next prediction. Even though the data is getting over fifteen in the beginning but it can also be observed that the model has maintain constant gap between

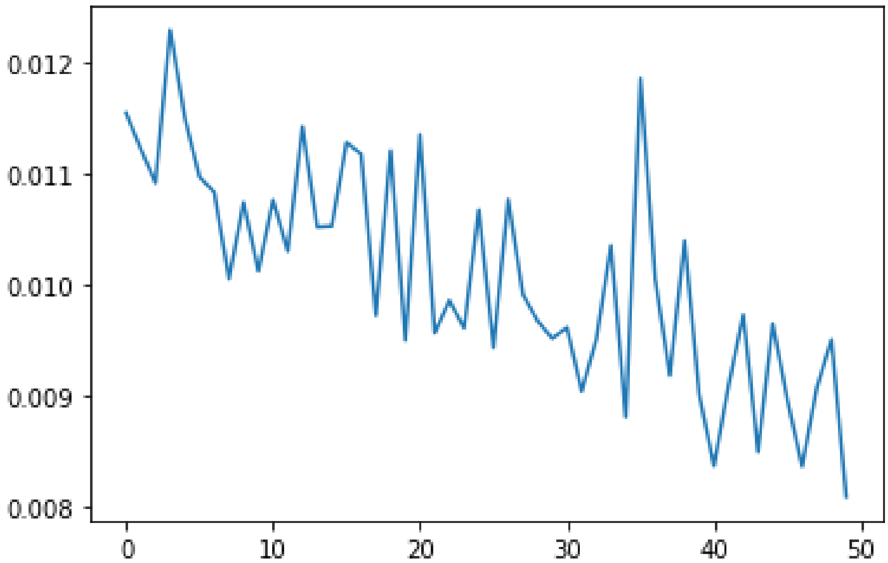

FIGURE 3.16: Loss-Plotting

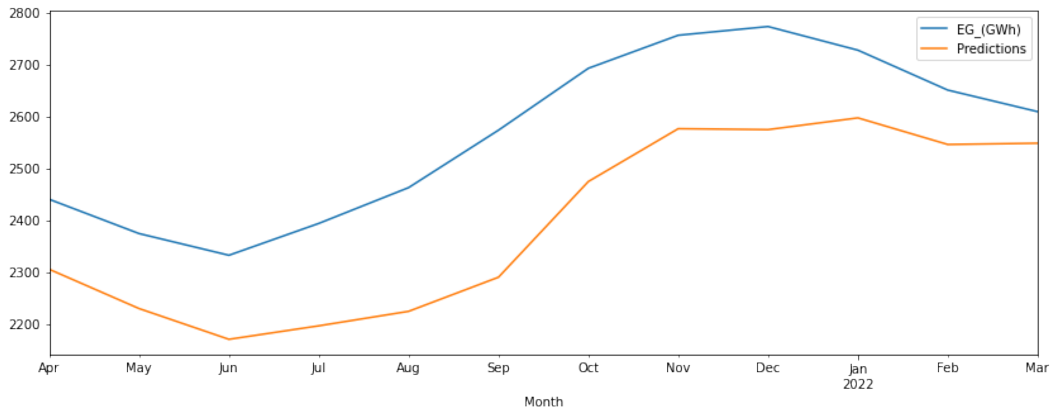

FIGURE 3.17: Prediction-Plotting

```
180.9974425656647
```

FIGURE 3.18: MSE

test value and prediction and after the month of December it the gap started to reduce. Mean Square Error was used for this model which gave value of 180.997 which is better than the model that was build for fifteen minutes data for same architecture and activation functions.

To improve above model one more dense layer of hundred neurons was added which makes now input layer with two hundred neurons, for both the input layer and hidden layer activation function 'tanh' was used and for output layer 'Sigmoid' is used as activation function, where like base model 'adam' optimizer and loss function 'mean square error' was applied.

```
Model: "sequential_9"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_11 (LSTM)              (None, 200)               161600

 dense_12 (Dense)            (None, 100)               20100

 dense_13 (Dense)            (None, 1)                 101

=================================================================
Total params: 181,801
Trainable params: 181,801
Non-trainable params: 0
_________________________________________________________________
```

FIGURE 3.19: Modal Summary

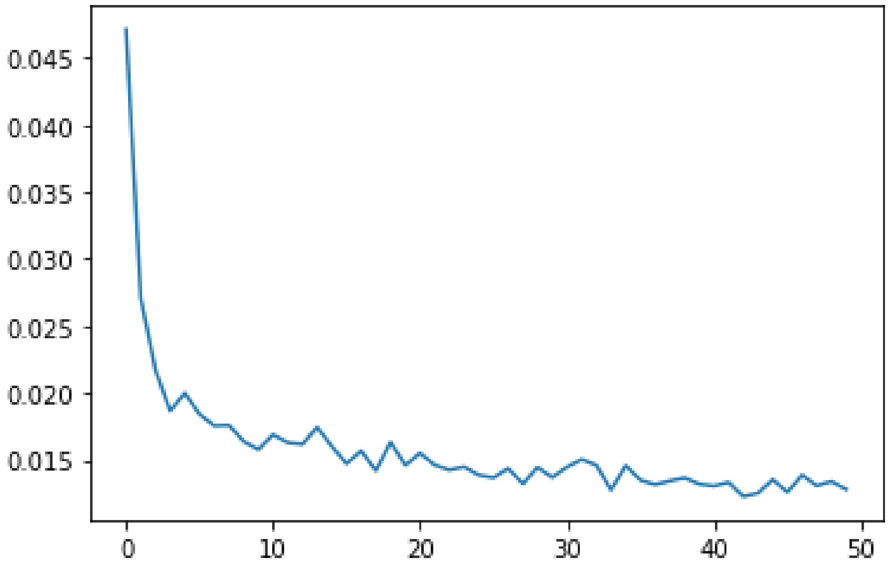

FIGURE 3.20: Loss-Plotting

```
Mean Square Error 60.75039691794017
```

FIGURE 3.21: MSE

From the above figure 3.20 the loss has taken steep fall after first epoch and with gradually it reduced to 0.128 by the end of fifty epochs, with slight fluctuation, from this it can be said that the model may perform better with the real data. With referring figure 3.22 we can see that the prediction is converging that is gap between the real data and predicted was more during month of June but it slightly reduced, that is the we can observe a cross point during month of September in

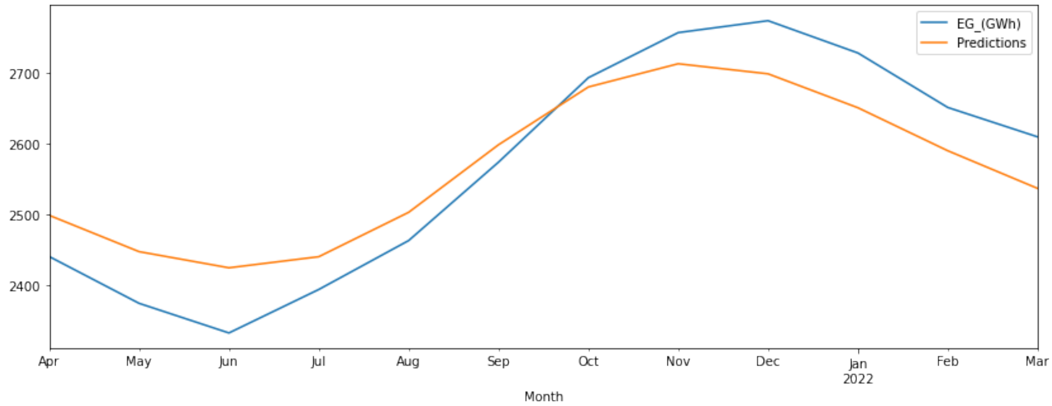

FIGURE 3.22: Prediction-Plotting

which prior to this the prediction values were slightly higher than the actual value where as the predicted values are now lower when compared to real data, it means that model is predicting higher values for first half and lower for second half. But when with plotting and difference between the two line remains almost constant, adding more layers with dropout with more number of epochs can reduce the difference between the true value and forecast value.

The Mean Square Error for base architecture for monthly energy prediction data set was 180.997 and for second architecture it is 60.750 figure 3.21 which is almost three hundred percent less, from this it can be said that by increasing the number of layers and epochs will reduce the error.

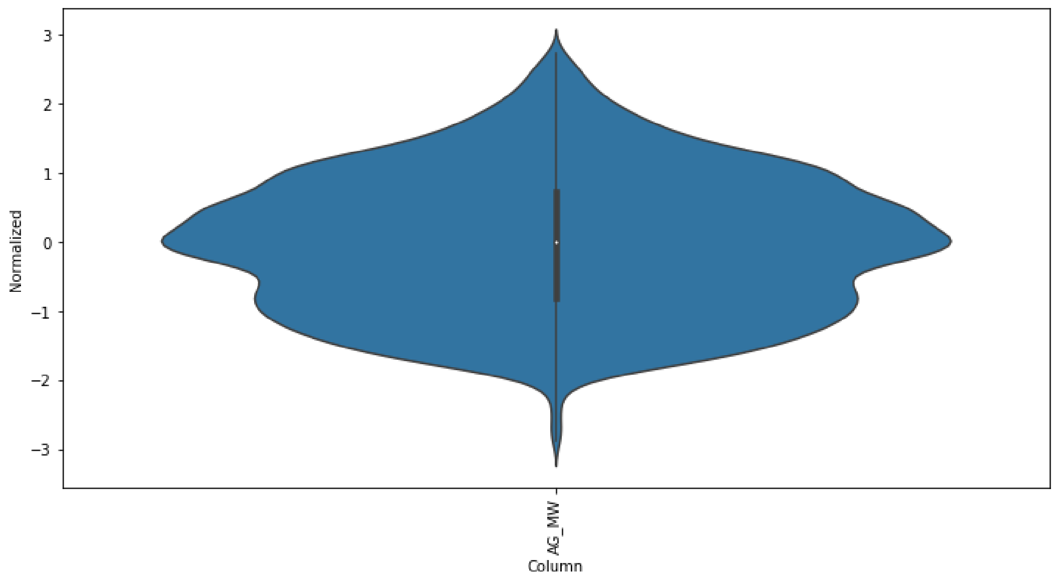

FIGURE 3.23: Violin Plot

For the fifteen minutes power production data sets the different approach was made by normalising the data the figure 3.23 depicted shows the violin plot after normalising, instead of MinMaxScaler for standardising RobustScaler was used to remove the outliers as this method works better for such data, the neural network of hundred neurons input layer follower by two hundred and three hundred dense layer with all of this layers having 'tanh' as the activation function, followed by the output layer with 'sigmoid' as the activation and adam optimiser using the mean square error as the loss function, summary of this architecture figure 3.24.

```
Model: "sequential_5"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_7 (LSTM)               (None, 100)               40800

 dense_6 (Dense)             (None, 200)               20200

 dense_7 (Dense)             (None, 300)               60300

 dense_8 (Dense)             (None, 1)                 301

=================================================================
Total params: 121,601
Trainable params: 121,601
Non-trainable params: 0
_________________________________________________________________
```

FIGURE 3.24: Modal Summary

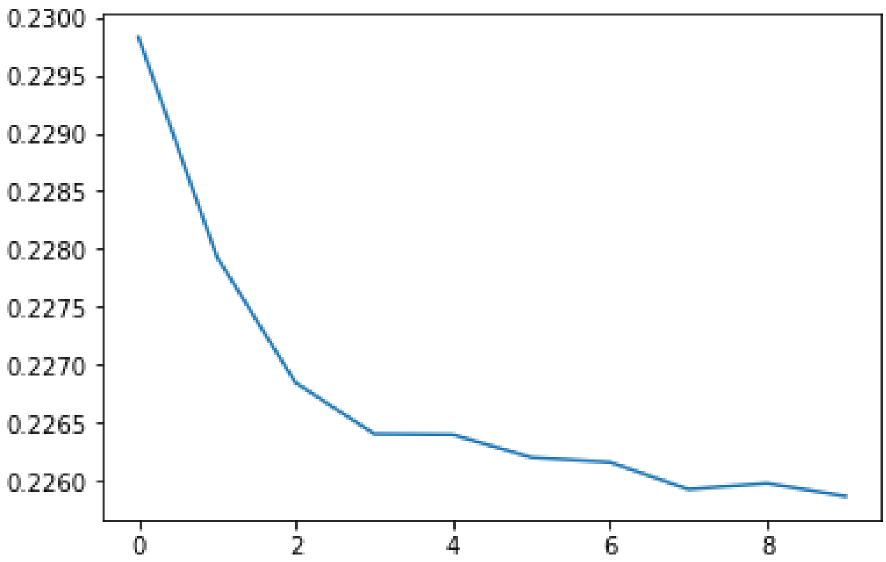

FIGURE 3.25: Loss-Plotting

From figure 3.25 we can say that we have reduced error from 1626.116 (figure 3.14)

```
Mean Square Error 0.922065373587615
```

FIGURE 3.26: MSE

from previous network architecture and by applying 'Robust Scaler' for standardising the data instead of 'MinMaxScaler', also by increasing the hidden layer. From the above experimentation it can said that due to problem with outliers in this fifteen minutes data the error was higher even with different modulation in network architectures. so with the all the methods experimented on LSTM the conclusion for this approach will abbreviated in the final conclusion section.

On the monthly energy production data with bit change in the network and different error function was used in this approach, by keeping the input layer with hundred neurons and increasing the hidden layer from two to three from previous approach, with first hidden layer with one hundred fifty neurons followed by a 'dropout' of 0.2 and second hidden layer with two hundred neurons and third layer with four hundred neurons by adding another dropout of 0.2 between them keeping 'tanh' as activation function and 'sigmoid' as activation for final output layer and running it for fifty epochs.

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_2 (LSTM)               (None, 100)               40800

 dense_2 (Dense)             (None, 150)               15150

 dropout (Dropout)           (None, 150)               0

 dense_3 (Dense)             (None, 200)               30200

 dropout_1 (Dropout)         (None, 200)               0

 dense_4 (Dense)             (None, 400)               80400

 dense_5 (Dense)             (None, 1)                 401

=================================================================
Total params: 166,951
Trainable params: 166,951
Non-trainable params: 0
_________________________________________________________________
```

FIGURE 3.27: Modal Summary

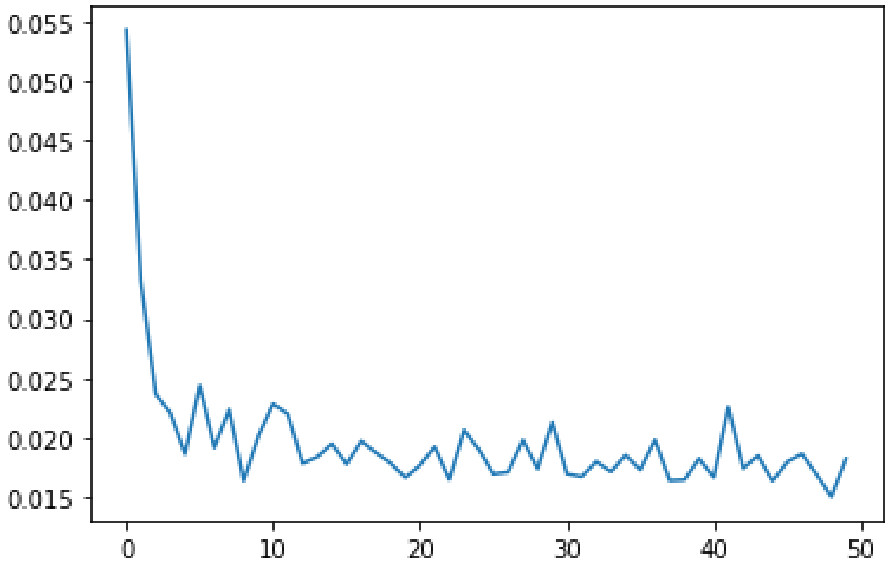

FIGURE 3.28: Loss-Plotting

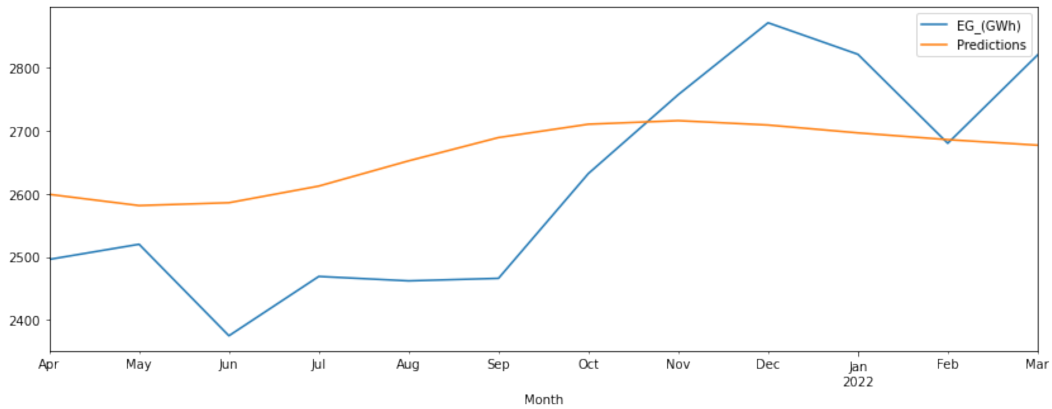

FIGURE 3.29: Prediction-Plotting

```
Root Mean Squared Error 140.03764539507168
Mean Absolute Error 123.86248934268967
```

FIGURE 3.30: Error function values

From the above approach on the monthly energy production data sets this architecture did not produced the better result even after adding dropout between the layers, the model is forecasting value constantly where as the data is showing high amount of fluctuation, this architecture can be improvised like by normalising the data and by scaling it using Robust like previous used for fifteen minutes datasets.

For the eirgrid fifteen minutes LightGradientBoostingMachine or LightGBM model

was applied to analyse how machine learning model will forecast with deep learning model. This model was developed by Microsoft and outperforms the standard Extreme Gradient Boosting (XGBoost) in terms of training speed and accuracy. LightGBM extends the gradient boosting algorithm with a kind of automatic feature selection, focusing on boosting samples with larger gradients. This can significantly improve training speed and improve predictive performance. As a result, LightGBM has become the de facto algorithm for machine learning competition when working with tabular data for regression and classification predictive modeling tasks. Therefore, along with Extreme Gradient Boosting, there is some responsibility for the growing popularity and widespread acceptance of the popular gradient boosting method (XGBoost).
After observing the Auto correlation in the dataset (figure 3.4) the lags of 192 was taken that is of two days with each feature value of fifteen minutes energy production, on the LightGBM the Regressor model was applied, the data is splitted into hour, day and month, and mean absolute error was used for the loss function.

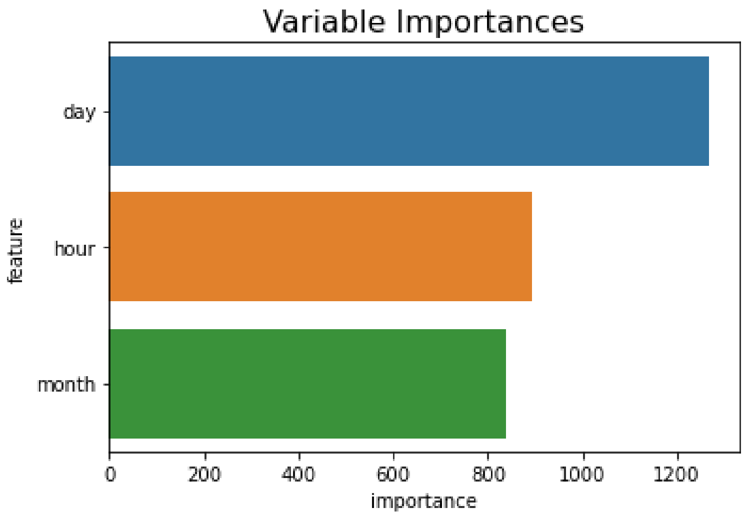

Figure 3.31: Variable Importance

From the figure 3.32 which represent the plotting of real data vs prediction data, it can be seen that the prediction line is almost keeping less difference with real data, only during time period of 04-15 the data is getting over fitted later on the difference was reduced. This model had the used Mean Absolute Error for loss function which gave the value of 397.108 which is almost same with previous

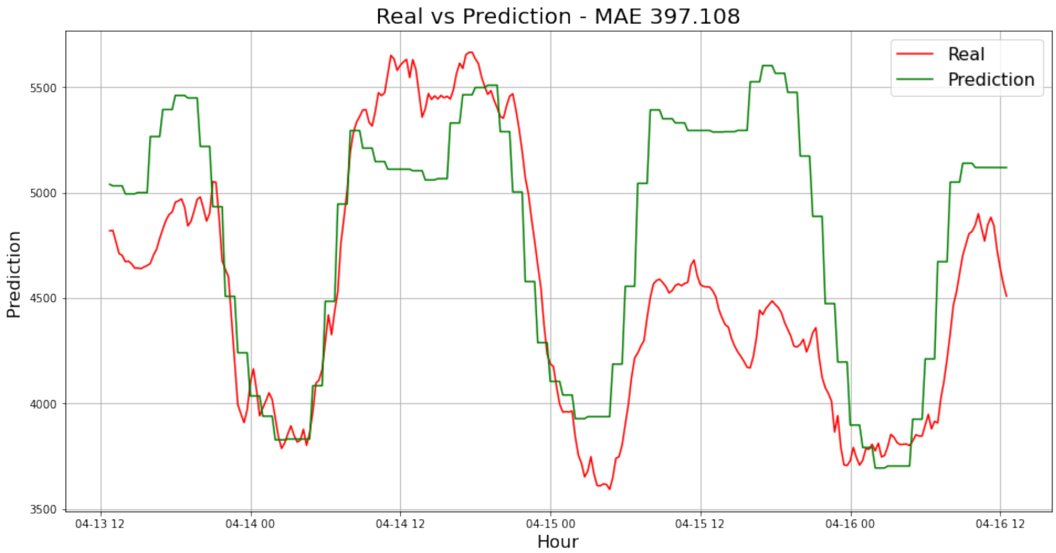

FIGURE 3.32: Two Days Prediction-Plotting

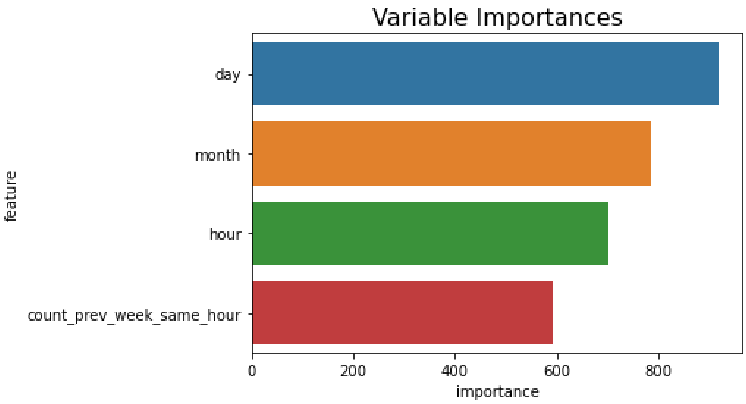

FIGURE 3.33: Variable Importance chart of one week

model LSTM implementation, improve this model importance on the each variable was verified, as this LightGBM give more importance to variables, from the figure 3.32 it can observed that this model is giving more importance to the variable of day, so from lags of two days it was increased to one week, comparing figure 3.32 and figure 3.34 line graph, we can see that forecasting line was on straight line when compared to real data was going down, from another figure at day 04-16 this model is predicting nearer value to the real data, along with energy consumption data value the forecasting line also went down on this particular date.

So with this implementation we can say that lightGBM model gives better prediction when given importance to the particular variable, so with date as the

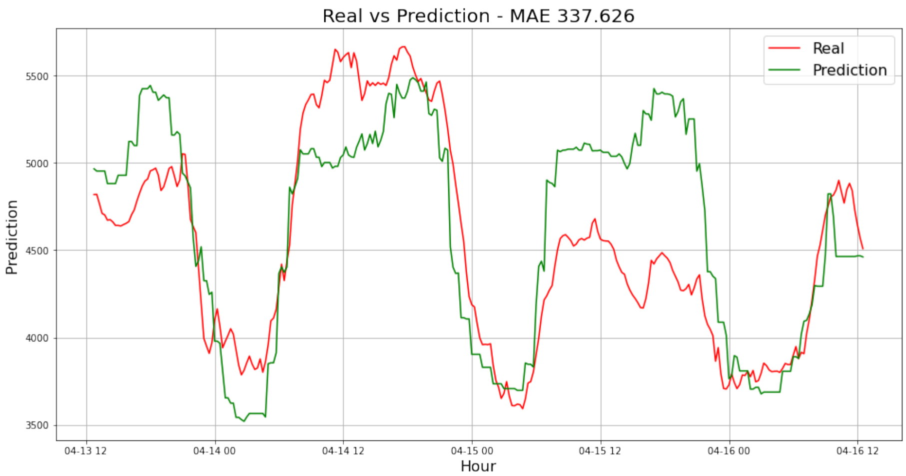

FIGURE 3.34: One week Prediction-Plotting

important variable in this dataset model performed better with error rate from 397.108 to 337.626, so it can be even made better in by scaling and normalisation.

3.5.2 Evaluation

In the previous section all the comparison of each individual models with different architecture, loss function for both the data set made. To finally evaluate the better performance with different ways of the model for two different data sets.

From above charts it can said that model with using 'RobustScaler' as prepossessing with two hidden layer with second layer of two hundred neurons and third with three hundred neurons which have input layer of hundred neurons with activation function of 'tanh' outperformed other LSTM-RNN models with mean square error of 0.92, for the same data set the LightGBM also performed better with less error rate when more number days were considered.

For the monthly datasets network architecture of input layer with two hundred neurons followed by hundred neurons with dense layer both with activation function of 'tanh' performed better with MinMaxScaler techniques, by producing mean square error rate of 60.75.

3.5.3 Conclusion

From implementation of models for both the data sets activation function 'tanh' performed better than the activation function 'relu' in this case, with 'sigmoid' as the output layer activation function.

The forecasting on electricity production is dependent on the two main reasons, data and external factor, like the with with two month records of every fifteen from mid of January to mid of March will not be able to forecast with accuracy because amount of electricity used in the month of march is completely different from January, so if the data of one year with every hour / fifteen minutes consumption can solve this problem, where as at the same time monthly energy consumption dataset can give better forecasting but this can be used to identity only on the monthly basis not daily or hourly basis.

There are many external factors that will easily make under perform the forecasting one among them is government policies to increasing in the number of electrical vehicles in coming days, which drastically increases the consumption of energy which the data will would not be having record in the prior.

3.6 Future Enhancement

Future Enhancement for the above two datasets can be made

1. For EIRGrid (fifteen minutes energy production dataset) Both Robusta and minmax scaler technique can be applied to enchance the better performance, as we have seen that the performance after applying Robusta technique gave better result when compared to minmax scaler, as in the implementation section it has been elaborated that robusta deals outlier easily.

2. Different Normalisation techniques can be applied on LighGBM and LSTM RNN before feeding the data into algorithm, as we have observed the ranage of variation in the real time datasets line plotting in various methods applied.

3. For monthly dataset increasing the performance can be increased with by adding layers, also it has to be considered that electricity production changes over the time period, different external factors has to be considered.

4. Different optimiser can be used like 'Adagrad' which has high gradient with lower learning rate.

Bibliography

[1] S. B. Taieb and R. J. Hyndman. Forecasting: Principles and practice.

[2] M. Elsaraiti, G. Ali, H. Musbah, A. Merabet, and T. Little, "Time series analysis of electricity consumption forecasting using arima model," in *2021 IEEE Green Technologies Conference (GreenTech)*, 2021, pp. 259–262.

[3] M. Elsaraiti, A. Merabet, and A. Al-Durra, "Time series analysis and forecasting of wind speed data," in *2019 IEEE Industry Applications Society Annual Meeting*, 2019, pp. 1–5.

[4] P. K. Jain, W. Quamer, and R. Pamula, "Electricity consumption forecasting using time series analysis," in *Advances in Computing and Data Sciences*, M. Singh, P. K. Gupta, V. Tyagi, J. Flusser, and T. Ören, Eds. Singapore: Springer Singapore, 2018, pp. 327–335.

[5]

[6] V. B. Getanda, P. K. Kihato, P. K. Hinga, and H. Oya, "Electricity consumption modeling and medium-term forecasting based on grouped grey model, ggm(1,1)," in *2020 IEEE PES/IAS PowerAfrica*, 2020, pp. 1–5.

[7] S. Chan, I. Oktavianti, and V. Puspita, "A deep learning cnn and ai-tuned svm for electricity consumption forecasting: Multivariate time series data," in *2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2019, pp. 0488–0494.

[8] E. G. Tajeuna, M. Bouguessa, and S. Wang, "A network-based approach to enhance electricity load forecasting," in *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, 2018, pp. 266–275.

[9] R. Azimi, M. Ghayekhloo, and M. Ghofrani, "A hybrid method based on a new clustering technique and multilayer perceptron neural networks for hourly solar radiation forecasting," *Energy Conversion and Management*, vol. 118, pp. 331–344, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0196890416302497

[10] H. Hou, "Short term load forecasting based on bayesian forecasting model," in *2020 International Conference on Wireless Communications and Smart Grid (ICWCSG)*, 2020, pp. 200–203.

[11] C. Luan, X. Pang, Y. Wang, L. Liu, and S. You, "Comprehensive forecasting method of monthly electricity consumption based on time series decomposition and regression analysis," in *2020 2nd International Conference on Industrial Artificial Intelligence (IAI)*, 2020, pp. 1–6.

[12] A. Shaqour, T. Ono, A. Hagishima, and H. Farzaneh, "Electrical demand aggregation effects on the performance of deep learning-based short-term load forecasting of a residential building," *Energy and AI*, vol. 8, p. 100141, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666546822000052

[13] V. Jadhav and V. Ligay, "Forecasting energy consumption using machine learning," *ResearchGate*, 2016.

[14] M. Kalimoldayev, A. Drozdenko, I. Koplyk, T. Marinich, A. Abdildayeva, and T. Zhukabayeva, "Analysis of modern approaches for the prediction of electric energy consumption," *Open Engineering*, vol. 10, no. 1, pp. 350–361, 2020. [Online]. Available: https://doi.org/10.1515/eng-2020-0028

[15] I. F. J. B. Iztok Fister, Xin-She Yang and D. Fister. (2018) Forecasting: Principles and practice. [Online]. Available: https://otexts.com/fpp2/

[16] E. Zivot and J. Wang, *Unit Root Tests*. New York, NY: Springer New York, 2003, pp. 105–127. [Online]. Available: https://doi.org/10.1007/978-0-387-21763-5_4

[17] R. S. S. Koekebakker, Steen; Adland. (3) Forecasting: Principles and practice.

Appendix A

Code Snippets

All the codes and datasets are available on this git repository

For further references see or go to the next url: `https://github.com/psun6789/Time_Series`