ISSN No:-2456-2165

Power Grid Energy Consumption

Radhika Mittal
Department of Computer Science and Engineering,
Jaypee Institute of Information Technology Sector-62,
Noida

Ria Singh, Harsh Pooniwala
Department of Computer Science and Engineering,
PES University Electronic City Campus,
Bengaluru

Abstract:- With the increase in technological advancement, the need for an energy conservation system is increasing. It is necessary to build an effective and predictive system now that existing energy consumption models emphasize on and forecast accuracy. Consumption of energy has increased many folds and now has become a concerning issue. The utilization of energy has increased with human development and growth. The main reasons for these problems are uncontrolled power usage, including excessive consumption, lack of optimal design, and energy wastage.

The purpose of this work is to predict the future trend in power usage for any system that monitors and requires this information in real-time. The best model to accomplish this goal was evaluated using recurrent neural networks (RNN) and long short-term memory (LSTM). Results from experiments demonstrate great accuracy and require fewer computer resources during model training than competing models.

Keyword:- *Power Consumption, Short-term load forecasting, electricity markets, spot prices, Recurrent Neural Networks (RNN).*

I. INTRODUCTION

India ranks third in both the production and consumption of electricity worldwide. As of March 31, 2020, India's national electric grid has an installed capacity of 370.106 GW. Large hydroelectric facilities are included in the category of renewable power plants, which make up 35.66% of India's installed capacity. The gross power produced by utilities in India for the 2018–19 fiscal year was 1,372 TWh, and the nation as a whole (utilities and non–utilities) produced 1,547 TWh. 2018–19 had a gross power usage of 1,181 kWh per person.

Day-ahead spot power is available 365 days a year, 24 hours a day. It generally consists of 24 hourly auctions that take place concurrently one day in advance. Spot electricity prices are known to exhibit the highest level of price volatility of any commodity. Seasonality, Mean Reversion, Volatility, and Jumps/Spikes are four stylized facts that are present in the electricity price series. As such, it is highly pertinent for simulating spot power prices with lead times of a few hours or days. In a power exchange with a two-sided auction, the spot electricity price is cleared in the market. the point at which the entire supply and demand curves for each area of the electricity market intersect at a specific hour.

II. LITERATURE SURVEY

Observing the previous methods used in various research papers we have concluded that currently to solve the situation of Power Grid Energy Consumption using predictive methods based on machine learning algorithms. The main approach is to observe the data pattern and create a model based on that.

In [1], the strategy used the most is based on a group of models called Autoregressive Integrated Moving Average (ARIMA) models. Several significant groups of research topics can be addressed by ARIMA models. The model identification procedure, a statistical estimate of parameters, and technical features of ARIMA models are all covered in detail. Examples are used to make the technical talk more understandable.

In [2], the book is divided into four chapters. The first describes the basic venues for price determination exchanges and power pools—in deregulated, competitive energy markets. A detailed description of electricity contracts and the method for establishing spot prices is provided. The chapter ends with an up-to-date survey of market solutions implemented in different parts of the world, with a particular emphasis on European and North American structures.

In [4], one of the key methods used by policymakers throughout the world to predict energy consumption in developing nations is forecasting. Most of the early research conducted in Turkey employed various econometric modeling techniques. However, time-series forecasting seems to produce superior outcomes since the anticipated economic and demographic factors typically differ from the realizations. In this work, we estimated Turkey's projected primary energy consumption employing the Autoregressive Integrated Moving Average (ARIMA) and seasonal ARIMA (SARIMA) methods, from 2005 to 2020.

In [5], this paper proposes an Artificial Neural Network (ANN)-based short-term load forecasting method that takes electricity price into account as one of the key system load characteristics, highlighting the significance of taking pricing into account when predicting loading in today's electricity markets. Testing is done using historical load data from the Ontario Hydro system as well as price data from a nearby system, demonstrating the effectiveness of the suggested strategy.

In [6], the inverse-leverage effect on several regional Indian power markets was examined using an ARIMA-GARCH model of the Indian spot electricity price series. Equations for the conditional mean and conditional variance were estimated.

In [7], the paper suggests an LSTM- based model named TL-MCLSTM for forecasting multi-step short-term power consumption to extract rich feature representations from three channels: power usage, time/location, and consumer behavior channels with a multiple output strategy.

In [8], the paper solves the problem of short-term load for residential households. A density-based clustering is used to check the differences in the loads. This paper suggests the DBSCAN clustering algorithm and LSTM neural network to predict the best energy consumption taking the lifestyle inconsistency of each household under consideration.

In [9], The paper studied the multi-step time series forecasting problem with the solution being different modeling ideas according to the different data patterns. It also conducted experiments to verify the proposed method and compared it with the results produced by the ARIMA/GRNN model.

In [10], this paper uses data from smart meters. This work proposes the S2S RNN algorithm for building-level energy forecasting. From using energy load forecasting to predicting the output value for the consecutive next input, this methodology uses RNN, GRU, and LSTM. The results show that the GRU S2S-o model outperformed the LSTM S2S-o model for all the prediction length cases.

In [11], the paper suggests short-term load and wind power forecasting using NN- based PIs to address the shortcomings of handling uncertainty. Instead of using traditional PIs, a method called LUBE method was used to construct PIs to transform the multi-objective problem into a single-objective problem.

III. PROPOSED SOLUTION

First, the data is preprocessed which includes cleaning the data with methods such as outlier removal, datatype correction etc. Simultaneously we conduct exploratory data analysis. The null values, if present, are also dealt with.

The data is then tested for time series to check for patterns and seasonality. This is done by projecting the data as a graphical representation, conducting tests such as KPSS Test, ADF Test, and checking for autocorrelation and partial autocorrelation.

Thereafter, based on the conclusions from the tests determined the suitable model for the data. The data is divided into training and testing data sets and machine learning models are run on them to gain outputs.

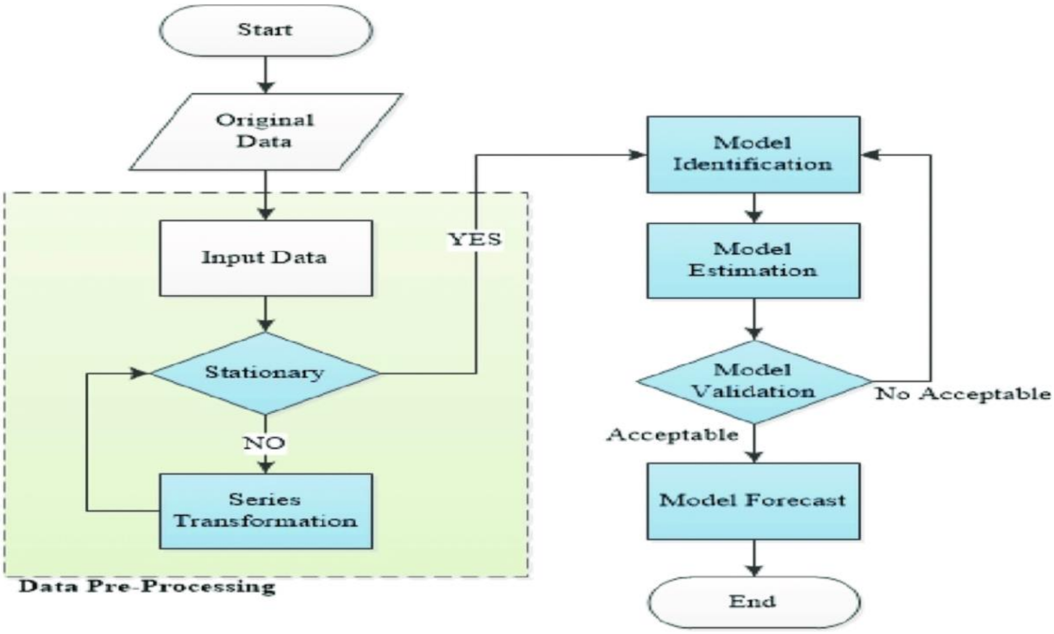

Fig. 1 Proposed Methodology

A. Dataset Used

Under the Ministry of Power, Power System Operation Corporation Limited (POSOCO) is a fully owned Government of India company. Prior to then, Power Grid Corporation of India Limited controlled it entirely. It was established in March 2009 to take care of PGCIL's power management duties.

The POSOCO weekly energy reports were used as the source for the dataset. The dataset contains energy consumption on an hourly basis from 2002 to 2018. There were 145k total observations in the dataset, none of which were null or zero. The dataset has 2 columns:

- Date-time,
- PJME_hourly consumption in megawatts.

As a result, the dataset has undergone pre-processing to become time series data.

ISSN No:-2456-2165

B. Preprocessing Data

For the two columns, the dataset consisted of 145266 entries. As these are the two necessary columns we will start with the removal of null values, if present.

After data cleaning, the attribute DateTime was divided into the following attributes:

- date,
- hour,
- dayOfWeek,
- quarter,
- month,
- year,
- dayOfYear,
- dayOfMonth,
- weekOfYear.

By analyzing data in a graphical and visual context, data visualization aims to interpret it and shows trends, patterns, and connections that are otherwise latent. Following data preprocessing, we performed Exploratory Data Analysis.

We analyze the cumulative load of power consumption. The below graph displays the distribution of PJME Load.

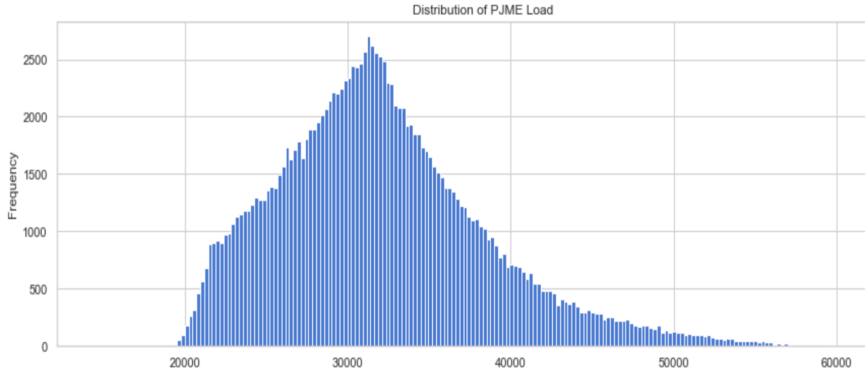

Fig. 2: Distribution of PJME Load

After that we also visualize the electric power consumption made on a daily basis.

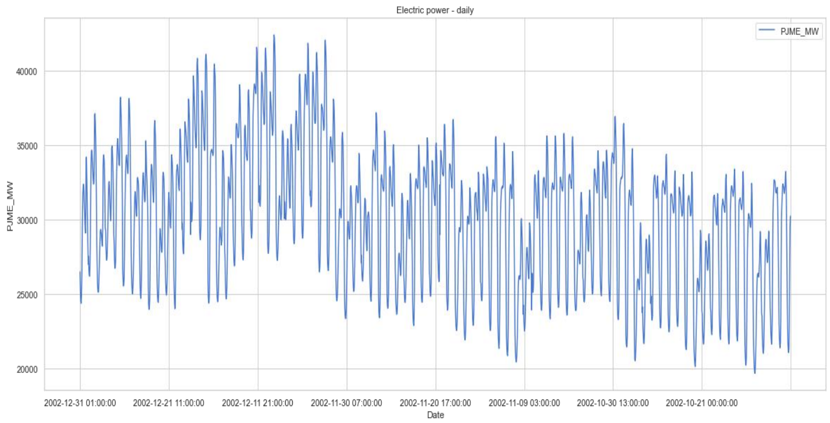

Fig. 3 Daily Electric Power Consumption

In addition, to this we observe the energy consumption within a week, from Monday to Sunday:

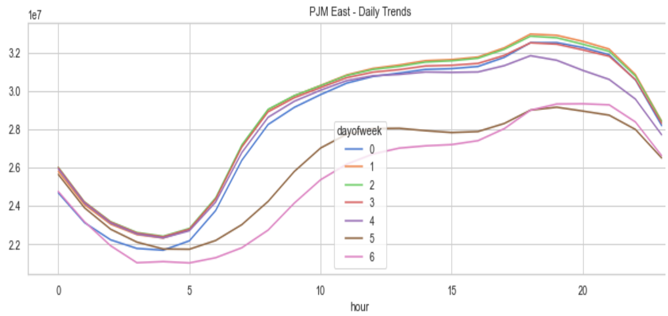

Fig. 4: PJME Load daily distribution within a week

The graph plots the amount of electricity used by the hour, dayOfWeek, year, weekOfYear and month.

IV. TIME SERIES ANALYSIS

After loading the dataset, the data's contents were examined. We verified that the date and time were formatted correctly in the Datetime field. The dataset includes values from January 1, 2002, to August 3, 2018.

The provided time plot displays the amount of electricity consumed between 2002 and 2018. This is to confirm that the data is in the correct format.

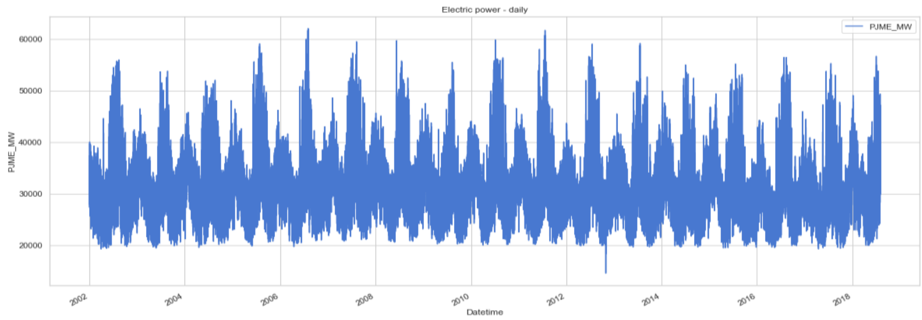

Fig. 5: Electric Power-Daily

The addition of the quarter column came next. The seasonal plotting that would be demonstrated later required this. The time series was transformed into an average monthly format via resampling. As a result, the time plot became smoother. The use cases now include monthly billing.

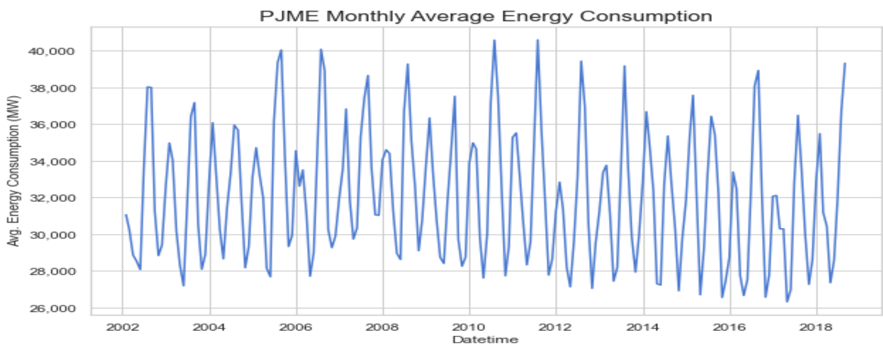

Fig. 6: PJME monthly average energy consumption

A. *Lag Plots*

To view a time series' fundamental characteristics, a lag plot is utilized. A collection of data from time step y(t+i) with time t and lag I is shown against data from a subsequent time step y in this form of scatter plot (t). If the time series data is random, sinusoidal, autocorrelated, or contains outliers, the pattern in a lag plot will demonstrate this. These patterns also reveal the best models to use for the data.

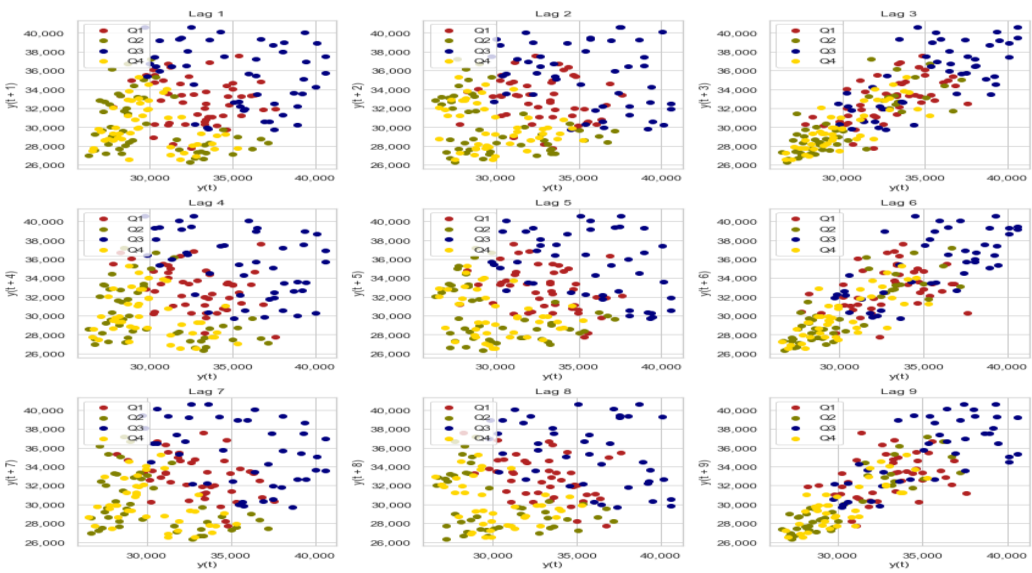

Fig.7: Lag Plots

ISSN No:-2456-2165

Due to the positive linear connection in the figure, lags 3, 6, and 9 exhibit autocorrelation at regular intervals. This finding eliminates randomness and indicates that the data is seasonal, or non-stationary.

To further confirm this conclusion, autocorrelation and partial autocorrelation plots are created. Autocorrelation is the degree of relationship between a specific time series, a self-lagged version, and successive time periods.

Subsequently, it quantifies the relationship between the current variable and available historical data.

As it examines the time series for randomness, it is comparable to the lag plot. For the data, autocorrelations are computed with various delays. For all delays, a whole random time series exhibits autocorrelations close to zero.

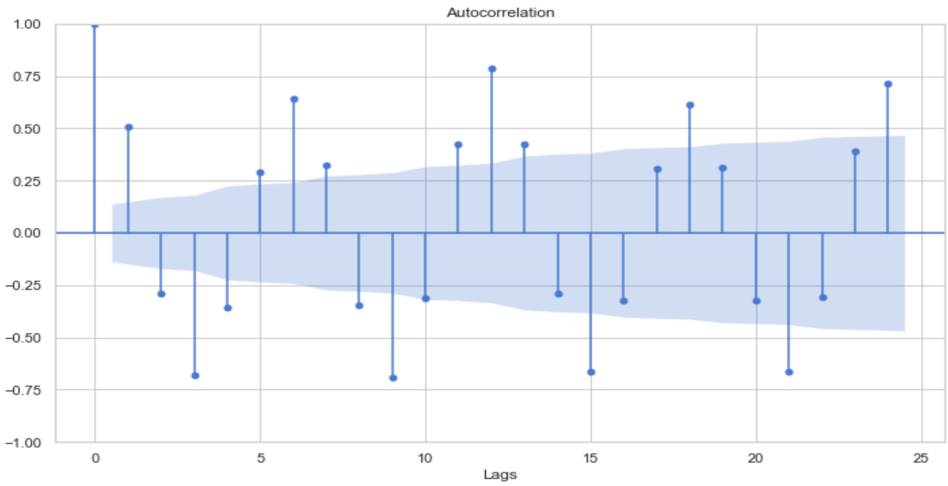

Fig. 8: Autocorrelation Plot

The correlation between two variables that assumes knowledge of and consideration for the values of some other set of variables is known as partial autocorrelation.

Data at various delays can likewise be calculated using partial autocorrelation, but with indirect correlations subtracted. The time series regression model is defined using this graphic. The most important autocorrelations for analysis are the first ones.

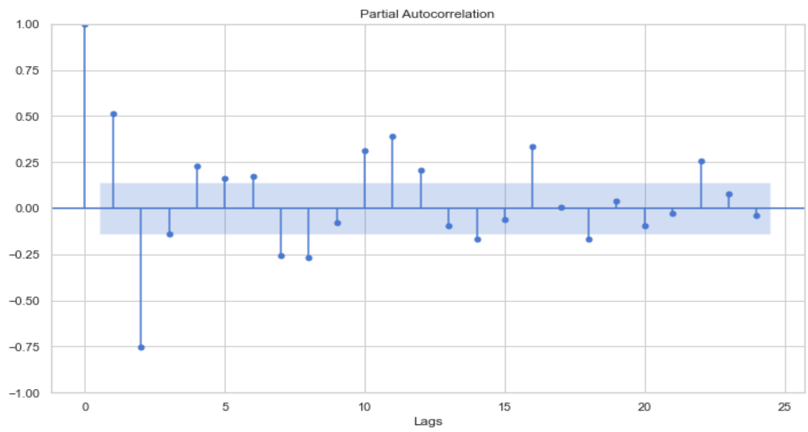

Fig. 9: Partial Autocorrelation Plot

Only the direct association between a timestep and its lag is depicted in the partial correlation graph above. The indirect relationships are negligible and approach zero.

The time series will need to go through a process known as differencing because we are developing a regression model. Any seasonality or tendency is eliminated by doing this. Additionally, this will stabilize the data and prepare it for modeling.

When the difference between successive steps is computed, the mean of a time series is stabilized. This lessens seasonality and pattern. When data becomes stationary, its statistical characteristics remain constant across time. The differenced data, which is more stable around the mean, is displayed in the plot below.

ISSN No:-2456-2165

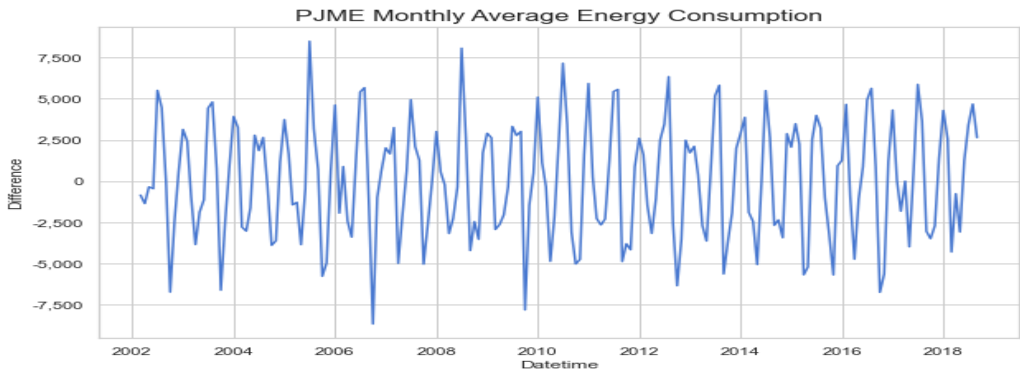

Fig. 10: PJME Monthly Average Energy Consumption

B. *Tests to confirm stationarity*

Two tests are performed to verify stationarity:

- Kwiatkowski-Phillips-Schmidt-Shin test- This is intended to compare the unit root alternative to the null hypothesis (H0), which predicts that the time series would seem stationary around a deterministic trend. The data are trend-stationary if the p-value is greater than 0.05. For our differenced data, a metric table is created, and the findings are as follows:

Test Statistic	0.099818
p-Value	0.100000
Number of Lags	27.000000
Critical Value (10%)	0.347000
Critical Value (5%)	0.463000
Critical Value (2.5%)	0.574000
Critical Value (1%)	0.739000
dtype: float64	

Fig. 11: Results of KPSS Test

- Augmented-Dicky-Fuller test- This is yet another test for detecting series stationarity. This test suggests that the data are not stationary, which is the null hypothesis. The time series is thought to be stationary if the p-value is less than 0.05. This is the inverse of the KPSS test.

Dickey-Fuller Metrics:	
Test Statistic	-6.098900e+00
p-Value	9.945211e-08
Number of Lags	1.400000e+01
Number of Observations	1.840000e+02
Critical Value (1%)	-3.466398e+00
Critical Value (5%)	-2.877380e+00
Critical Value (10%)	-2.575214e+00
dtype: float64	

Fig. 12: Results of ADF Test

The results show p-value less than 0.05. The null hypothesis is thus rejected. Both the KPSS and ADF tests yield stationary data. As a result, no second-order differencing is required.

C. *Seasonal Plots*

- Quarterly Plot- Using a seasonal plot, we have plotted the energy data values for all years against the quarterly intervals(Q1, Q2, Q3, Q4).

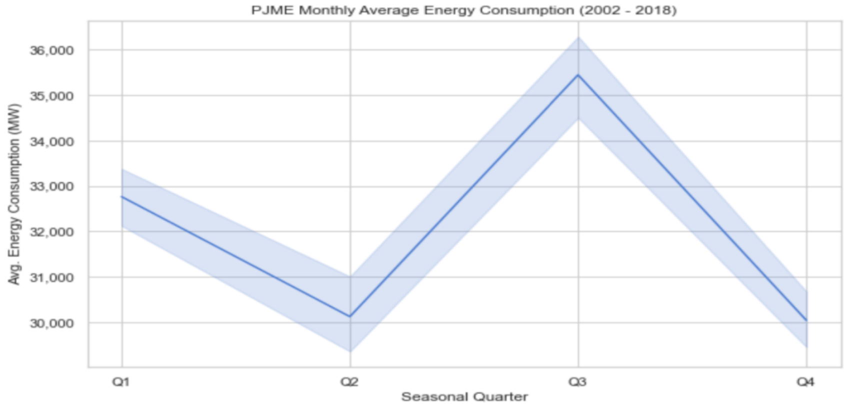

Fig.13: Seasonal Plot

The third quarter shows the maximum energy usage each year. This can be studied further by plotting the data against each month.

- Monthly Plot

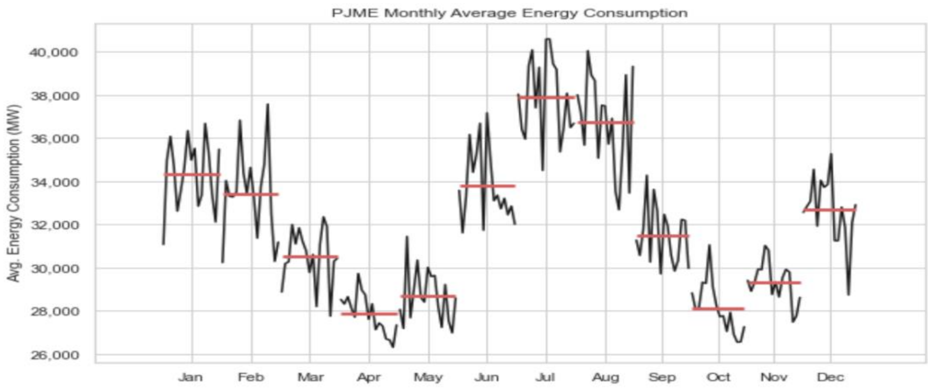

Fig. 14: Monthly Plot

From the above graph, we infer that -

- July is the month when peak average energy usage occurs.
- January has the highest energy usage among winter months.
- The lowest energy consumption is in April and October.

➢ Seasonal Moving Average- To smooth out the time plot, a moving average is created by calculating the mean of time series values over a specified set of past timesteps.

Datetime	MONTHLY_AVG	quarter	MOVING_AVG
2002-01-31	31075.399731	1	NaN
2002-02-28	30239.166667	1	NaN
2002-03-31	28875.256720	1	30063.274373
2002-04-30	28534.731572	2	29216.384986
2002-05-31	28073.653226	2	28494.547173
2002-06-30	33585.919444	2	30064.768081
2002-07-31	38041.896505	3	33233.823059
2002-08-31	38014.021505	3	36547.279152
2002-09-30	31281.468056	3	35779.128689
2002-10-31	28836.814266	4	32710.767942
2002-11-30	29418.240278	4	29845.507533
2002-12-31	32563.034946	4	30272.696497

Fig. 15: Seasonal Moving Average

The consecutive means for the past three monthly averages may be seen in the MOVING_AVG column.This is called the moving average of the third order. The higher the order, the smoother the plot is.

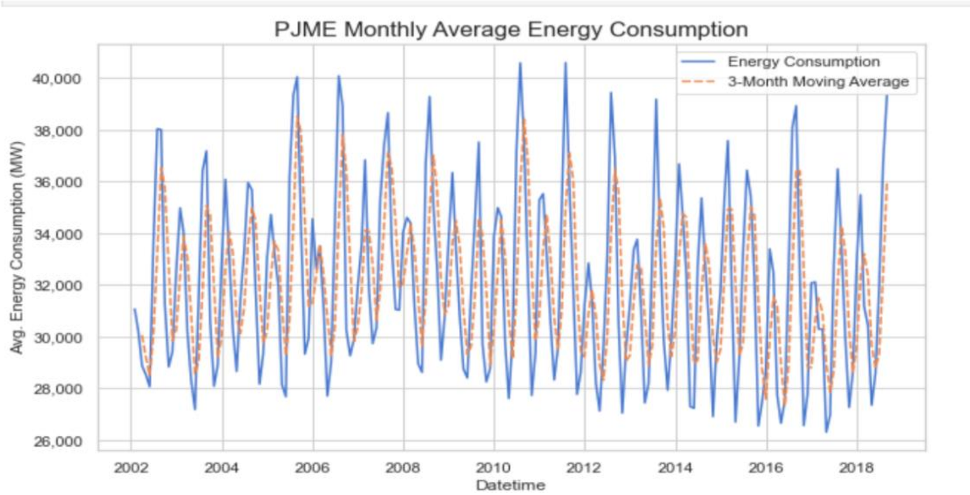

Fig. 16: PJME monthly average energy consumption vs 3-month Moving Average

A plot of a 3-month moving average shows a smoother time series with fewer fluctuations. This method is generally used to estimate the trend of a dataset when performing decomposition.

- Seasonal Trend Decomposition Using Loess- Time Series decomposition is a process used to extract the features for analysis. Seasonal Trend Decomposition using Loess (STL) is a specific type of decomposition that can estimate non-linear relationships. This method is robust to outliers and can handle any kind of seasonality within the data. The function will return three components of the time-series: Season, trend, and residual.

What remains after removing the seasonal trend is known as a residual or remainder component. A residual that displays prominent trend or season characteristics is the result of an incomplete decomposition.. Residual points that deviate significantly can be classified as outliers.

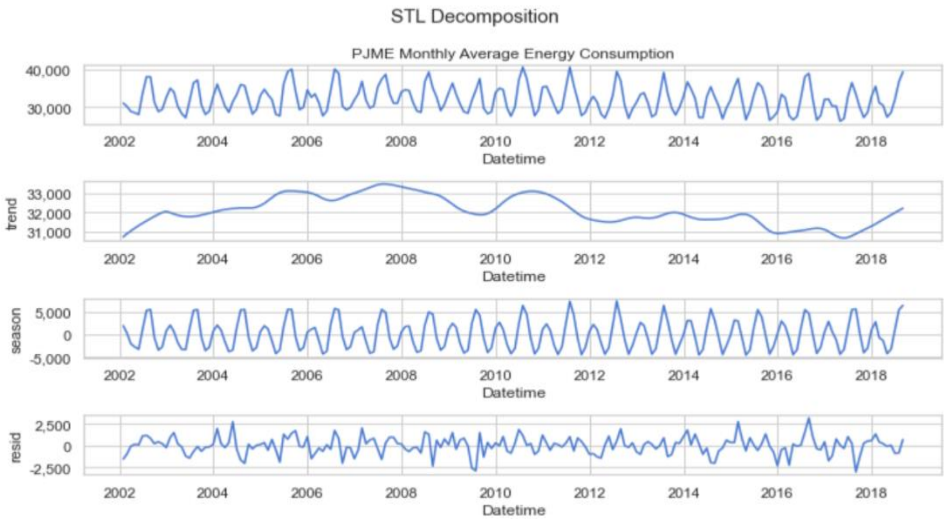

Fig. 17: STL Decomposition

Here we find a seasonal component that follows the same pattern as our quarter and month charts. From 2002 to 2008, the smoothed trend shows a steady increase in average energy use. From 2012 to 2018, the smoothed trend shows a steady decrease in average energy use. The residual component is a random variable with no discernible signal from the trend or season.

V. BUILDING MODELS AND EVALUATION

We proceeded by normalizing energy consumption data using sklearn MinMaxScaler. The range of power consumption values changes after normalization. The values in the previous graph ranged from 0-60000, but the values in the graph after normalization ranged from 0-1.0.

Following that, the data was divided into testing and training sets. This will be further used to train the RNN models.

ISSN No:-2456-2165

```
#create train, test data
seq_len = 20 #choose sequence length

X_train, y_train, X_test, y_test = load_data(df_norm, seq_len)

print('X_train.shape = ',X_train.shape)
print('y_train.shape = ', y_train.shape)
print('X_test.shape = ', X_test.shape)
print('y_test.shape = ',y_test.shape)

print (X_train[0])
```

```
X_train.shape =  (110000, 20, 1)
y_train.shape =  (110000,)
X_test.shape =  (35346, 20, 1)
y_test.shape =  (35346,)
[[0.25184873]
 [0.22338565]
 [0.2113136 ]
 [0.20750026]
 [0.21733909]
 [0.24603392]
 [0.29828295]
 [0.34033498]
 [0.36355209]
 [0.37532919]
 [0.37558201]
 [0.36570104]
```

Fig. 18: Testing and Training Data Information

A. Simple Recurrent Neural Network

A kind of neural network known as RNNs has hidden states and permits the use of previous outputs as inputs. They usually go like:

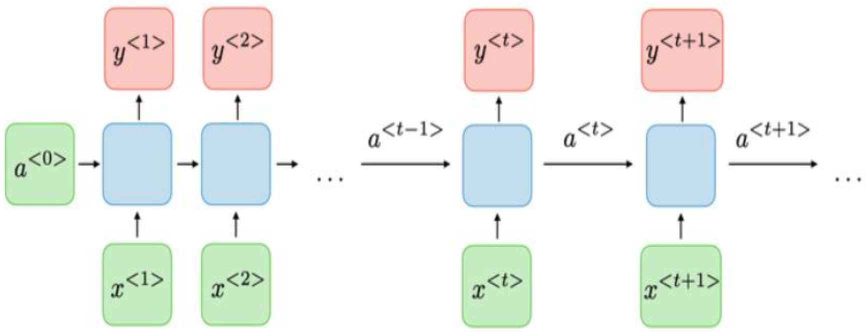

Fig.19: RNN Model

Before sending the input to the middle layer of the neural network, the input layer x receives and analyzes it. There are numerous hidden layers with different activation functions, weights, and biases in the middle layer h. Recurrent neural networks can be used if the neural network has no memory and the individual hidden layer parameters are independent of one another.

By standardizing the multiple activation functions, weights, and biases, the recurrent neural network will ensure that each hidden layer has the same characteristics. It will just produce one hidden layer and loop over it as many times as necessary rather than producing several hidden levels.

The data parameters are trained with a batch size of 1000 and in 10 epochs. The loss values that we got from training the data were plotted in the graph as shown below:

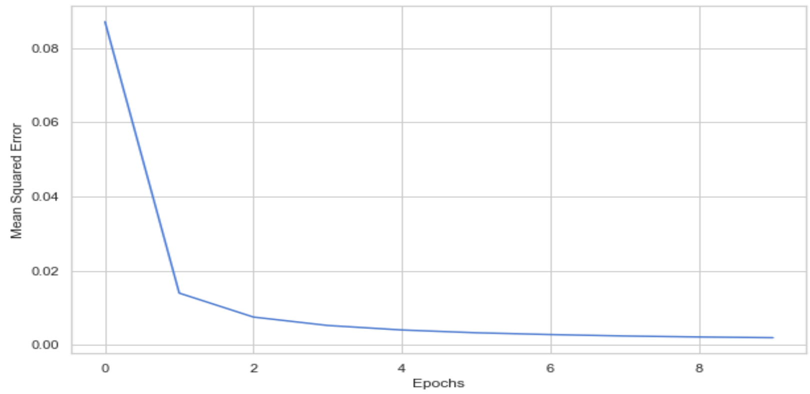

Fig. 20: Loss Distribution Plot for RNN

In a regression model for a dependent variable, the R-squared (R2) statistic measures the percentage of variance that is explained by an independent variable or variables.

```
R2 Score of RNN model =  0.9746711754335387
```

Fig. 21: R-squared value of RNN Model

Since the predicted values are almost identical to the real values, the RNN model is effective in predicting the sequence, as can be shown.

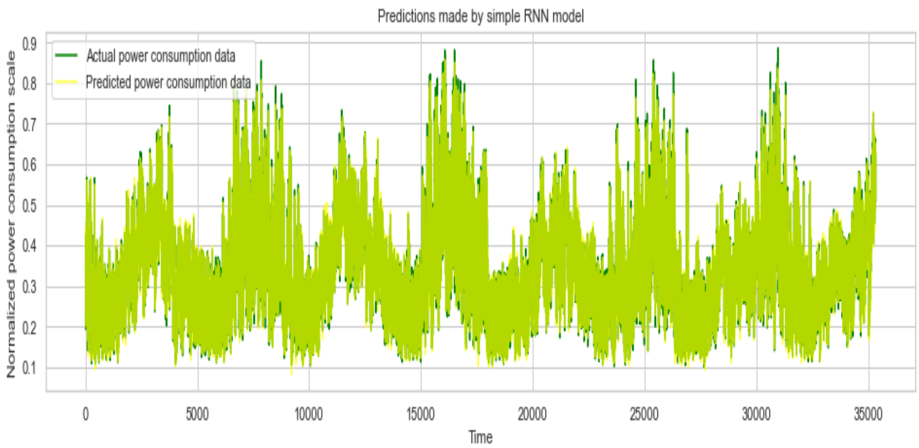

Fig. 22: Predictions made by Simple RNN Model

B. LSTM Model

An enhanced RNN, or sequential network, called a long short-term memory network, permits information to endure. It can fix the vanishing gradient problem with the RNN. For persistent memory, RNN, or recurrent neural network is utilized.

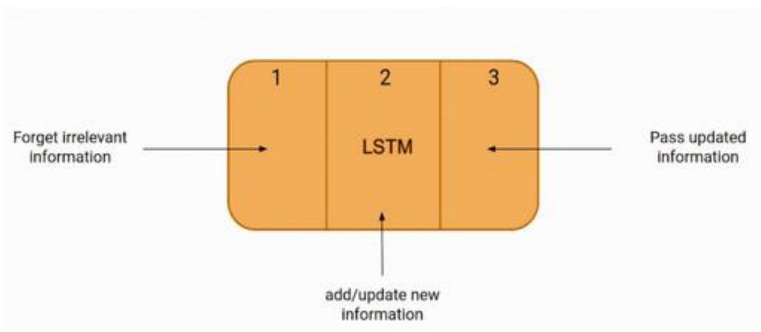

Fig. 23: LSTM Model

ISSN No:-2456-2165

Whether the information from the previous timestamp should be remembered or disregarded depends on the first section. In the second stage, the cell makes an effort to learn new information from the input given to it. The cell changes the data from the current timestamp to the next timestamp in the third and final section.

These three LSTM cell constituents are mentioned by Gates. The first, second, and third components are referred to as the forget gate, input gate, and output gate, respectively.

The data parameters are trained with a batch size 1000 and in 10 epochs. The loss values from the trained data information are plotted in a graph as shown -

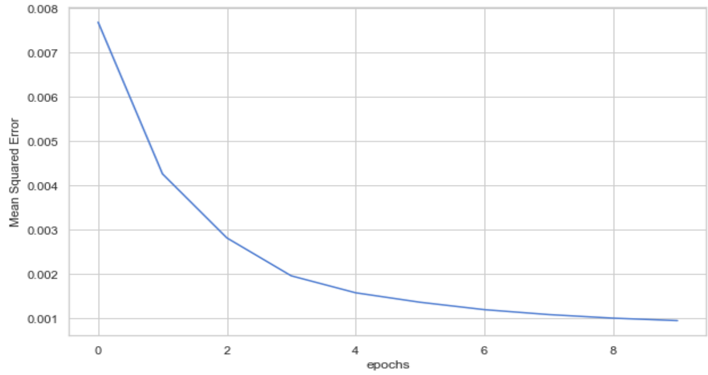

Fig. 24: Loss Distribution Plot for LSTM

R^2 value of the LSTM model calculated was:

```
R^2 Score of LSTM model =  0.9749681282181341
```

Fig. 25: R-squared value of LSTM Model

Predictions made by the model are shown below:

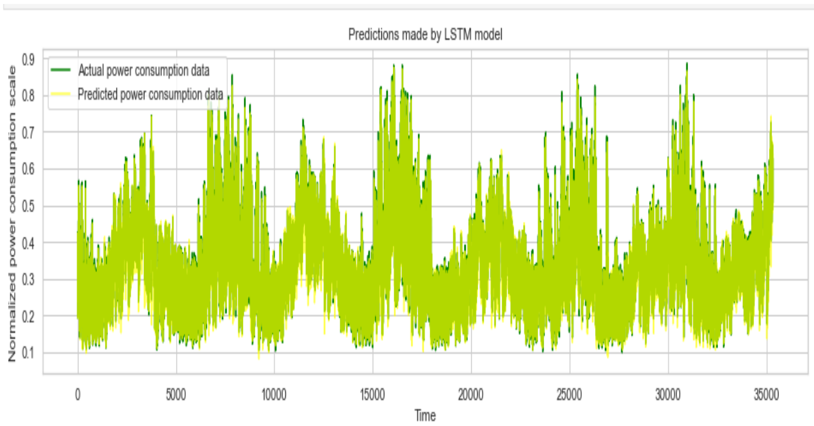

Fig. 26: Predictions made by Simple RNN Model

The below graph compares predictions of simple RNN and LSTM models by plotting data in a single graph.

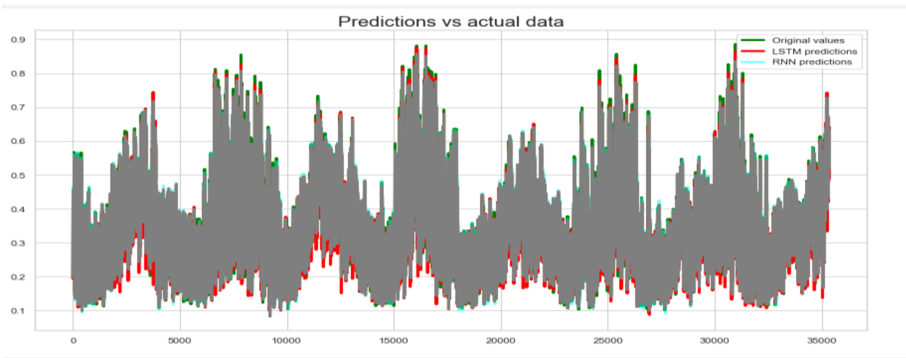

Fig. 27: Predictions vs Actual Data

VI. CONCLUSION

In this project, we performed visualization on the data that had 145366 values. We saw and understood the trends in power consumption from 2002 to 2018. We performed a time series analysis and tested our data for stationarity using KPSS and ADF tests. The daily, monthly, quarterly, and yearly variations in power consumption were visualized and studied. The seasonal variation in the usage helped in better understanding the results. The data pre-processing involves normalization and decomposition. Then using the simple RNN and LSTM models, we predicted the future energy consumption of electricity.

ACKNOWLEDGEMENT

We would like to profess our gratitude to our family and friends for their immense support during this project.

REFERENCES

[1.] Wayne F. Velicer1, Joseph L. Fava, Cancer Prevention Research Center, University of Rhode Island - “Time Series Analysis for Psychological Research”, Volume 2, Issue: 10, 2017

[2.] Weron, R. Modeling and Forecasting Electricity Loads and Prices: A Statistical Approach; John Wiley & Sons Ltd.: West Sussex, UK, 2007; Volume 403

[3.] De Gooijer, J.G.; Hyndman, R.J. 25 years of time series forecasting. Int. J. Forecast. 2006, 22, 443–473

[4.] Ediger, V. Ș.; Akar, S. ARIMA forecasting of primary energy demand by fuel in Turkey. Energy Policy 2007, 35, 1701–1708

[5.] Chen, H.; Canizares, C.A.; Singh, A. ANN-based short-term load forecasting in electricity markets. In Proceedings of the 2001 IEEE Power Engineering Society Winter Meeting (Cat. No. 01CH37194),

[6.] Girish, G P & Sedidi, Vijayalakshmi. (2018). Does Indian Spot Electricity Price Series Exhibit Inverse-Leverage Effect? Theoretical Economics Letters. 08. 234-240. 10.4236/tel.2018.83017.

[7.] X. Shao and C. S. Kim, "Multi-Step Short-Term Power Consumption Forecasting Using Multi-Channel LSTM With Time Location Considering Customer Behavior," in IEEE Access, vol. 8, pp. 125263-125273, 2020, doi: 10.1109/ACCESS.2020.3007163.

[8.] Kong, Weicong & Dong, Z.Y. & Jia, Youwei & Hill, David & Xu, Yan & Zhang, Yuan. (2017). Short-Term Residential Load Forecasting based on LSTM Recurrent Neural Network. IEEE Transactions on Smart Grid. PP. 1-1. 10.1109/TSG.2017.2753802.

[9.] Yunpeng, Liu & Hou, Di & Junpeng, Bao & Yong, Qi. (2017). Multi-step Ahead Time Series Forecasting for Different Data Patterns Based on LSTM Recurrent Neural Network. 305-310. 10.1109/WISA.2017.25.

[10.] L. Sehovac, C. Nesen and K. Grolinger, "Forecasting Building Energy Consumption with Deep Learning: A Sequence to Sequence Approach," 2019 IEEE International Congress on Internet of Things (ICIOT), 2019, pp. 108-116, doi: 10.1109/ICIOT.2019.00029.

[11.] Quan, Hao & Srinivasan, D. & Khosravi, Abbas. (2014). Short-Term Load and Wind Power Forecasting Using Neural Network-Based Prediction Intervals. Neural Networks and Learning Systems, IEEE Transactions on. 25. 303-315. 10.1109/TNNLS.2013.2276053.