Load Forecasting Using LSTM Model

Amal Alhammadi[1], Jawaher Alshamsi[2], Maryam Almajidi[3], Reema Alzarooni[4]

[1]Engineer, [2]Senior Engineer, [3]*Actg. Asst. Manager*, [4]*Actg. Manager*
Transmission Protection Department, Dubai Electricity and Water Authority (PJSC), Dubai, UAE

Abstract - *This paper represents the application of Long-Short Term Memory (LSTM) deep network study that is focused on load forecasting with long-short term memory. The LSTM enables generating a forecast for future loads and the requirement for operating a combination of generators in order to meet electrical demand. To achieve an accurate load forecasting, the load data for each generator is separated and used to train an LSTM model. The objective of the study is to develop an efficient and accurate method of forecasting the load by utilizing data for the past 4 years (i.e. from 2018 to 2021) in LSTM model to predict the future outcomes based on the past time series. Essentially, the final forecast is estimated using numerical code expressions of the peak load data collected in the past 4 years.*

Key Words: LSTM Model, Load Forecasting, Generators, Electricity, Time series, Coding, Peak load data, RNN.

1. INTRODUCTION

Load forecasting has become a necessity in the power industry. For the energy utilities to maintain the continuity of their services with minimum interruptions, certain measures ought to take place such as outage planning, energy storage management, market supply and demand, power plants maintenance schedules, and so on. All of which are dependent on load forecasting [1]. LSTM offers the tools that could assist the power industry to maintain effective services by forecasting the future demand based on past data. LSTM model is one of the types of Recurrent Neural Networks (RNN) that has shown outstanding results in sequence learning problems [2]. This paper aims to utilize the LSTM model to forecast load demand with the use of past data of 4 consecutive years. It will first discuss the gathered data from the different generators, then their behavior due to the varied load, and lastly the coding used for LSTM model to achieve the desired results.

1.1 The Generators

The data for the load is gathered from four types of generators, which are Coal Old, Coal New, Gas, and Peaker generators. The power plant **Coal Old and Coal New** have similar operation systems and function as thermal power stations that burn coal to generate electricity. A Coal-Fired Power Station is a type of fossil fuel power station in which it is usually crushed and then burned in a coal-fired boiler [3]. As figure 1 indicates, steam is generated from the boiled water, which is used to spin the turbines that turn generators. Therefore, mechanical, thermal and electrical energies are generated due to chemical energy stored in the coal [4]. Coal that has been ground into a fine powder is blown into a boiler and burned. Due to the high heat, the water runs in the pipe of the boiler will steam and cause a high pressure which will turn the blades of the turbine that's connected to the shaft.

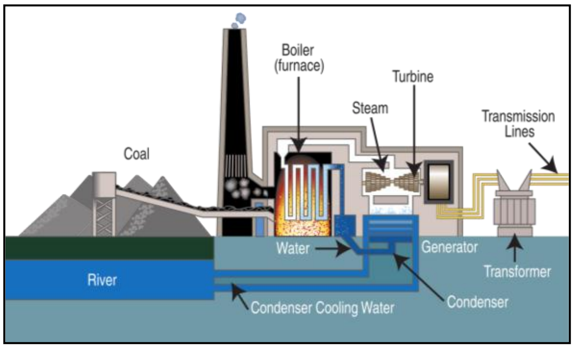

Figure -1: Coal-fired power station [5]

As a result of that, the generator will spin and produce electricity [3].

The Gas Power Plants generate electricity using natural gas as their fuel. All the natural gas power plants can generate electricity with different purposes. Gas turbine can be used in all-natural gas plants by adding natural gas and stream of air that will expands and causes the rotation of the magnet in the generator which will produce electricity. The emissions of the natural gas plants is more significant than a nuclear power plant and it is significantly contribute to climate change regardless the air quality improvement [6].

Peaker power plant is a mixture of load power plants to supply energy to grid. The base load power plants supply the reliable amount of energy that is in daily bases, called the base load. Examples of base load power plants include hydroelectric, coal or nuclear. Also, the Peaker Power Plant is a power plant that only turns on during times when electricity is at its peak demand in which it operates differently when compared to other power plants that generates the electricity. Moreover, the cost that Peaker Power Plant reaches its maximum and provide electricity into the grid when energy demand is at its maximum, in factthe power supplied by Peaker Power Plant is at higher price per kWh because they only supply power for few hours per year with a fast rate response [7]. Finally, the energy source used for Peaker Plants is fossil fuel, which is natural gas [7].

Needless to say, each of the generators have different constraints associated with it, such as Fuel Cost ($/MWh), Maintenance Cost ($/h), Engine Cost ($/switch), Minimum and Maximum Generation Level (MW), Ramp Up and Ramp Down (MW/h), Minimum Upper and Lower Time (h) are the constraints and have been collected for each power plant. The details of which are shown in table-1.

Table -1: Power Plant Properties

Constraints	CoalOld	CoalNew	Gas	Peaker
FuelCost ($/MWh)	30	25	45	60
MaintenanceCost ($/h)	20	35	50	10
EngineCost ($/switch)	900	650	200	80
MinGenerationLevel (MW)	120	90	40	3
MaxGenerationLevel (MW)	800	250	200	120
RampUp (MW/h)	30	40	80	800
RampDown (MW/h)	50	50	100	1900
MinimumUpperTime (h)	4	3	5	1
MinimumLowerTime (h)	3	1	4	1

The Power Plant properties are entered in the python coding program as shown in figure 2.

Load Data

```
plantProps = pd.read_csv('plantProps.csv').to_numpy()
print(plantProps)
plantProps = plantProps[:,1:]

loadProfile = pd.read_csv('loadProfile.csv').to_numpy()
loadProfile = loadProfile[:,1]
nHours = loadProfile.shape[0]
Time = np.arange(1,nHours+1)
```

```
[['FuelCost ($/MWh)' 30 25 45 60]
 ['MaintenanceCost ($/h)' 20 35 50 10]
 ['EngineCost ($/switch)' 900 650 200 80]
 ['MinGenerationLevel (MW)' 120 90 40 3]
 ['MaxGenerationLevel (MW)' 800 250 200 120]
 ['RampUp (MW/h)' 30 40 80 800]
 ['RampDown (MW/h)' 50 50 100 1900]
 ['MinimumUpperTime (h)' 4 3 5 1]
 ['MinimumLowerTime (h)' 3 1 4 1]]
```

```
print('Number of Hours=',nHours,'\n')
```

```
Number of Hours= 168
```

Figure -2: Constraints per generator in Python program

1.2 Generators Behavior

The data of each of the generators versus time in a week (total of 168 hours) are collected, table 2, and entered in python codes specifically performed for the past four years (2018 to 2021). The total power generated versus time will be implemented from the results in python coding program.

Table -2: Sample of Data Gathered from 2018 to 2021

Day	Month	Year	Seconds	Demand
1	1	2018	0	56234
1	1	2018	1600	461234
1	1	2018	3600	106345
1	1	2019	1800	52473
1	1	2019	3600	53197
1	1	2019	5400	53991
1	1	2020	0	66037
1	1	2020	1800	70021
1	1	2020	3600	73805
1	1	2021	0	68246
1	1	2021	1800	67648
1	1	2021	3600	67591

In order to visualize the behavior of each power generated for each plant versus time (in hours), the codes in figure 3 has been inserted in Python taking in consideration the values of the constraints for each power plant.

```
#RampUp and RampDown Constraits
m.c1=pyo.Constraint(m.K_1N, rule=lambda m,k:
                    m.CoalOld[k]-m.CoalOld[k-1]<=30)

m.c2=pyo.Constraint(m.K_1N, rule=lambda m,k:
                    m.CoalOld[k]-m.CoalOld[k-1]>=-50)

m.c3=pyo.Constraint(m.K_1N, rule=lambda m,k:
                    m.CoalNew[k]-m.CoalNew[k-1]<=50)

m.c4=pyo.Constraint(m.K_1N, rule=lambda m,k:
                    m.CoalNew[k]-m.CoalNew[k-1]>=-40)

m.c5=pyo.Constraint(m.K_1N, rule=lambda m,k:
                    m.Gas[k]-m.Gas[k-1]<=100)

m.c6=pyo.Constraint(m.K_1N, rule=lambda m,k:
                    m.Gas[k]-m.Gas[k-1]>=-80)

m.c7=pyo.Constraint(m.K_1N, rule=lambda m,k:
                    m.Peaker[k]-m.Peaker[k-1]<=1900)

m.c8=pyo.Constraint(m.K_1N, rule=lambda m,k:
                    m.Peaker[k]-m.Peaker[k-1]>=-800)

#Meet Hourly Demand Constrain

m.c9=pyo.Constraint(m.K, rule=lambda m,k:
                    m.CoalOld[k]+m.CoalNew[k]+m.Gas[k]+m.Peaker[k]>=GenerationTarget[k])

#Generation upper limit constraints
m.c10=pyo.Constraint(m.K, rule=lambda m,k:
                     m.CoalOld[k]<=800) #define max gen level for coal old

m.c11=pyo.Constraint(m.K, rule=lambda m,k:
                     m.CoalNew[k]<=250) #define max gen level for coal new

m.c12=pyo.Constraint(m.K, rule=lambda m,k:
                     m.Gas[k]<=200) #define max gen level for Gas

m.c13=pyo.Constraint(m.K, rule=lambda m,k:
                     m.Peaker[k]<=120) #define max gen level for peaker
```

Figure -3: Codes for power generated for each plant versus time

Upon running the codes, Chart 1 shows the results of the maximum and minimum loads, which can be read and compared with maximum and minimum generation level that is taken from table-2 of power plant properties for each power plant for a week (168 hours). It is shown that for all the four power plants the demand (load) level is nearly close to the generation level. In fact, during week days which can be taken as full load condition the peak value for the maximum load level shown in coal old power plant is about 600 MW with maximum generation level of 800 MW and the load is reduced over the weekend which can be considered less load condition almost 150MW with minimum generation level of 120MW.

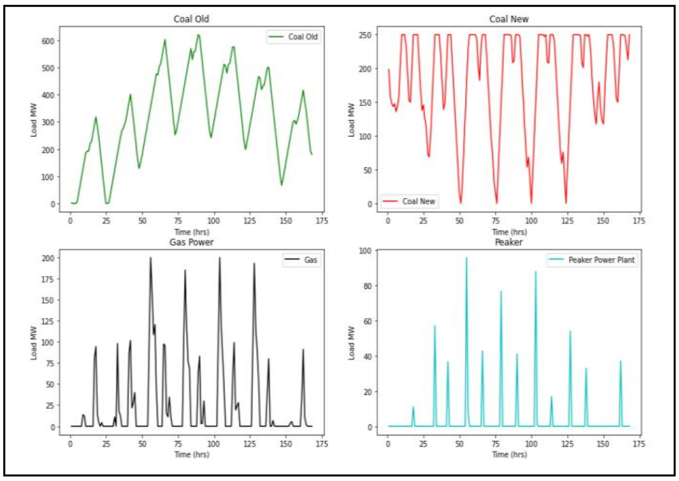

Chart -1: Power generated for each plant versus time

Moreover, in case of coal new the maximum load level is about 250MW with maximum generation level of also 250MW and this can be concluded that these maximum peaks are due to the utilizing of electricity during day hours more than night time and at working days condition, however the minimum load level can be viewed from plot as about 65MW with minimum generation level of 90MW. Finally, in the case of Gas Power Plant, the maximum peaks are shown with maximum load of 200MW with maximum generation level of 200MW and the peaks are getting reduced during less load condition which is 30MW as shown from the plot and 40MW minimum generation level that is to less usage of electricity during some parts of the day and without working days or school days condition. Moreover, the same scenario applies for Peaker Power Plant in which during working hours and week days the maximum load it reaches is about 95MW with maximum generation level of 120MW and the least load it reaches is about 7MW in which the minimum generation level in this case is 3MW. The maximum generation levels and the peaks of the maximum load level for each generator are almost near each other, also the minimum generation level value is close the minimum peaks of the load level. Finally, as the results indicate that the generation and the load are almost near each other which will result in having the desirable behavior as shown in Chart 2.

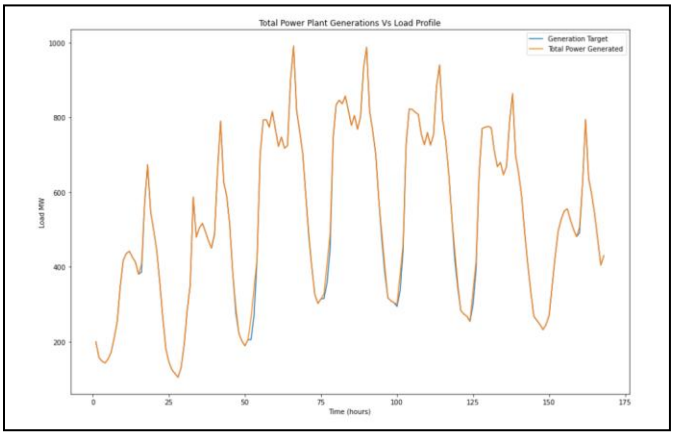

Chart -2: Total Power Plant Generated vs Load Profile

2. Logic of LSTM model and Outcome

Unlike the traditional types of Neural Network models that only rely on previous N histories, LSTM is capable of learning long-term dependencies and capable to capture nonlinear patterns in time series data [8]. Hence, it is a great candidate to predict subsequent demand every half-hour.

In this design, LSTM deep network developed according to the following steps:

Step 1: Load Python Libraries

Step 2: Load Data and prepare Data

Step 3: Design a Network

Step 4: Train a Network

Step 5: Test a Network

The evaluation of the forecasting model accuracy can be done by measuring Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE) [9].

MAE uses the same scale as the data being measured, the mean absolute error is a common measure of forecast error in time series analysis, MAE is calculated as:

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} = \frac{\sum_{i=1}^{n} |e_i|}{n}$$

It is the average of the absolute errors e_i, where y_i is the prediction and x_i the true value. Note that alternative formulations may include relative frequencies as weight factors.

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 10 Issue: 01 | Jan 2023 www.irjet.net p-ISSN: 2395-0072

Different from MAE, MAPE measures the error proportion to the absolute value, it expresses the error as a percentage and can be calculated using the following equation.

$$MAPE = \frac{100\%}{n} \sum_{t=1}^{n} \left| \frac{A_t - F_t}{A_t} \right|$$

Where n is the total number of forecast values, A_t and F_t denote the actual and forecast value at time t, respectively.

The Mean Average Percentage Error (MAPE) is 5.325 which represent the average deviation between the forecasted value and actual values and the result found acceptable since it is less than 10% of average demand, and Mean Average Error (MAE) is 631.490 and the result is accepted since it within the ± 5% of the average demand.

The obtained graphs in figure 6 shows the actual versus predicted load demand next half hour (x*1800 second).

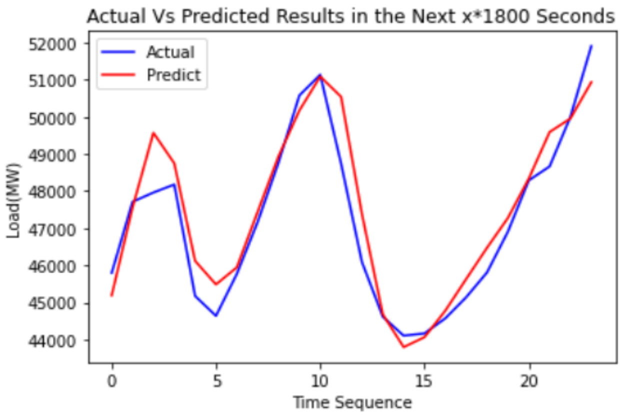

Chart -3: Actual Vs Predicted Load Demand

3. CONCLUSION

As a conclusion the objective to develop an efficient and accurate method of forecasting the load by utilizing data for the past 3 years using LSTM model was achieved through developed model in Python which enabled to predict the future outcomes based on the past time. Moreover, the series maximum generation levels and the peaks of the maximum load level for each generator are almost near each other, also the minimum generation level value is close the minimum peaks of the load level. Finally, as the results indicate that the generation and the load are almost near each other.

REFERENCES

[1] T. Ciechulski and S. Osowski, "High Precision LSTM Model for Short-Time Load Forecasting in Power Systems," *Energies,* vol. 14, no. 2983, pp. 1-15, 21 May 2021.

[2] J. M. Jiménez, L. Stokes, C. Moss, Q. Yang and V. N. Livina, "Modelling energy demand response using long short-term memory neural networks," *Energy Efficiency,* vol. 13, pp. 1263-1280, 24 July 2020.

[3] H. Bambhaniya, "What is Coal Power Plant and How Does it Work?," 23 September 2021. [Online]. Available: https://www.engineeringchoice.com/what-is-coal-power-plant/. [Accessed 16 September 2022].

[4] J. P. Ciferno, T. E. Fout, A. P. Jones and J. T. Murphy, "Capturing Carbon from Existing Coal-Fired Power Plant," *Chemical Engineering Progress,* vol. 105, no. 4, pp. 33-41, April 2009.

[5] University of Calgary, "Coal fired power plant," [Online]. Available: https://energyeducation.ca/encyclopedia/Coal_fired_power_plant#cite_note-8. [Accessed 10 November 2022].

[6] University of Calgary, "Natural Gas Power Plant," Energy Education, [Online]. Available: https://energyeducation.ca/encyclopedia/Natural_gas_power_plant. [Accessed 20 September 2022].

[7] Edgecom Energy Inc., "What is a Peaker Plant," Edgecom Energy, 28 October 2021. [Online]. Available: https://www.edgecomenergy.ca/resources1/what-is-a-peaker-plant. [Accessed 21 September 2022].

[8] H. Abbasimehr and R. Paki, "Improving time series forecasting using LSTM and attention models," *Journal of Ambient Intelligence and Humanized Computing,* vol. 13, no. 1, pp. 673-691, 2022.

[9] C. Tian, J. Ma, C. Zhang and P. Zhan, "A Deep Neural Network Model for Short- Term Load," *Energies,* vol. 11, no. 12, pp. 1-13, 2018.