# Big Data Analytics for Demand Response in Smart Grid

## A Thesis

*submitted for the award of the degree of*

## Doctor of Philosophy

in

**Computer Science and Engineering Department**

Submitted by

**Sanju Kumari**
(Reg no: 901603015)

Under the Guidance of

**Dr. Neeraj Kumar**
Professor

**Dr. Prashant Singh Rana**
Associate Professor


**Thapar Institute of Engineering and Technology, Patiala,**
**Punjab - 147004, India**
**November 2022**

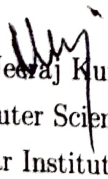
# Certificate

I hereby certify that the work, which is being presented in the thesis, entitled "Big Data Analytics for Demand Response in Smart Grid" partial fulfillment of the requirements for the award of the degree of Doctor of Philosophy and submitted to the institution is an authentic record of my work carried out under the supervision of Dr. Neeraj Kumar and Dr. Prashant Singh Rana. I have cited the reference about the text(s)/figure(s)/table(s) from where they have been taken.
The matter presented in this thesis has not been submitted either in-part or full to any other University/Institute for the award of any other degree.

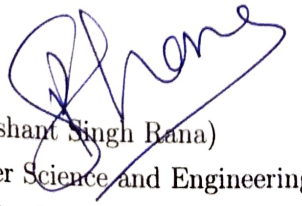
(Sanju Kumari)
Registration No. 901603015

This is to certify that the above statements made by the candidate are correct and true to the best of my knowledge.

Verified by:



(Dr. Neeraj Kumar,)
Computer Science and Engineering Department
Thapar Institute of Engineering and Technology, Patiala, Punjab, India.



(Dr. Prashant Singh Rana)
Computer Science and Engineering Department
Thapar Institute of Engineering and Technology, Patiala, Punjab, India.

*Dedicated to Shivababa*

# Acknowledgements

I thought this day would never arrive without support of my guides, family, friends and children as well as meditation. Now that it has, my heart overflows with gratitude for those who have been with me throughout the ordeal. I want to express my deep gratitude to Dr. Prashant Singh Rana and Dr. Neeraj Kumar, my research supervisors, for their patient guidance, enthusiastic encouragement, and useful critiques of this research work. Their guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better team to mentor my PhD study.

I want to express my sincere gratitude to Dr. Prashant Singh Rana and Dr. Neeraj Kumar accepting me as their student. I deeply thank them for his constant support and for giving me the freedom to explore my intellectual curiosity without objection. There advice and encouragement were always important guiding lights for when I lost my focus. Sincere thanks must also go to Dr. Sahaj Saxena and Dr. Singara and Dr. Gita for their continuous support by always willing to answer my many questions. Their motivation and enthusiasm are contagious. I can only hope that I have been able to absorb some of his magical intuition for doing interesting collaborative research. I want to thank the members of my thesis committee: Dr. Maninder Singh, Dr. Seema bawa and Dr.Rajiv Kumar and Dr. Mandi. They generously gave their time to offer insightful comments towards improving my work.

The contribution of my husband and parents friends, those from before, cannot be left unsaid. Their visits and telephonic conversations were always a source of immense joy. To those who I was fortunate enough to be friend during my stay here, I thank you for sharing this journey with me. Their presence and continued support has helped me sail through this. A heartfelt thank you to my seniors who were like elder siblings, usually showering love and care but also keeping us in check, as and when needed.

I thank my fellow labmates for not only their stimulating discussions and valuable suggestions; but also for all the fun we shared. I truly appreciate my colleagues, including those from other disciplines, for enriching me by sharing their experiences. We've all been there for one another and have taught ourselves and each other many tools. I know that I could always ask for advice and opinions on any issue that I may be dealing with. I'm thankful to our bhajan-mandali friends at TIET residential area. My life would have been very dull without their fun interactions.

Numerous faculty members have always been very kind with their words of encourage-

ment. Crossing paths with them while walking on campus has ever brought a smile to my face. I wish to give them my sincerest thanks. The office staff has always been available and ready for the assistance of any kind. The security guards are diligent and very co-operative. They have made my stay at the campus a pleasurable one.

I wish to thank my family for always having my back. Irrespective of what they were dealing with on an individual level, they were always available for a conversation. I owe a lot to my parents, who encouraged and helped me at every stage of my personal and academic life, and longed to see this achievement come true. My husband has been my best friend all my life; I love his dearly and thank him for his care and concern. My sincere thanks to wife of my guides for giving me mentle support till completion of my PhD. I would not have made it this far without them.

My research would have been impossible without the support of Dr. Prashant Singh Rana as well as Dr. Neeraj Kumar. My wholehearted thanks to them for sharing their domain knowledge, without which this research would have been an uphill task.

# Abstract

The power industry will depend on Smart Grid (SG) to a great degree in the future. It provides qualitative and quantitative services for better management of energy. Electrical devices such as Advanced Metering Infrastructure (AMI) and Smart Meter (SM) produce large data which is called Big Data. These Big Data is in the form of time series data that requires complex data analytics for prediction of consumption of energy. Prediction of consumption of energy using Big Data analytics can help to balance the supply and demand of energy which is one of the challenging task of SG. The researchers have covered these topics however, they have not tuned the parameters with optimization algorithm such as Genetic Algorithm (GA) for time series data. They have not analysed the prediction of energy using the Prophet model, data anomaly detection techniques and filtering techniques with respect to large time series data in SG.

In the first scheme, GA is applied for tuning the parameters of Long Short Term Memory (LSTM). GA is an evolutionary process which is used for optimization. LSTM memorises values over arbitrary intervals which are capable to manage time series data. GA is combined with LSTM in order to process hyper-parameters such as hidden layers, epochs, data intervals, batch size and activation functions. Hence, GA creates a new vector for optimum solution that provides minimum error. These methods provide better results when compared with existing benchmarks. Moreover, GA-LSTM is used in a multi-threaded environment which will increase the speed of convergence.

In the second scheme, various filtering techniques are used to predict the energy forecast which can improve the quality of service to the users. The filtering techniques ' primary task is to handle non-linearity in the input dataset. Various filtering techniques reduce the redundant data for energy consumption prediction. Five different filtering techniques such as Butterworth, Smoothing, Kalman, Frequency, and Filtfilt have been used to preprocess the five different power consumption datasets. LSTM model was used on the processed data for the power consumption prediction.

In third scheme, Auto Regressive Integrated Moving Average (ARIMA) and Prophet model is used for energy prediction. ARIMA is mainly used by professionals who have prior knowledge of the intricacies of the model. If a single parameter in the equation is incorrect, the entire result will be affected. However, the Prophet model uses a Bayesian curve fitting method and does not require prior knowledge of datasets. It automatically finds seasonal trends from the data. The Prophet model incorporates seasonal trends such as holidays and weekends, whereas the ARIMA model incorporates both seasonal

and non-seasonal trends with time-series data. It provides great precision compared to any other method.

The fourth scheme, uses anomaly detection techniques for the large datasets. Different anomaly techniques are compared and tested as preprocessing techniques with LSTM, ARIMA and Prophet models and the results are analysed with different performance metrics. Different anomaly techniques such as forest, K-NN, Histogram, SVM, SOS, and OSVM have been used and compared with preprocessing algorithm on different datasets.The novelty of the work lies in the preprocessing techniques on the LSTM, ARIMA and Prophet model where different anomaly techniques have been compared.

**Keywords:** *ARIMA, Big Data, Big Data Analytics, Energy Consumption Prediction, Ensamble Learning, Genetic Algorithm, LSTM, Filtering Techniques, Prophet Model and Smart Grid*

# Table of Contents

# List of Figures

# List of Tables

# List of Abbreviations

| | |
|---|---|
| **AMI** | Advance Metering Infrastructure |
| **ANNs** | Artificial Neural Networks |
| **ADR** | Anciallary Demand Response |
| **ADT** | Anomaly Detection Techniques |
| **BD** | Big Data |
| **BW** | Butter Worth |
| **DNN** | Deep Neural Network |
| **DL** | Deep Learning |
| **DR** | Demand Response |
| **DSM** | Demand Side Management |
| **ECP** | Energy Consumption Prediction |
| **EDR** | Emergency Demand Response |
| **EDR** | Economic Demand Response |
| **FF** | Filtfilt Filtering |
| **FSS** | Frequency Swept Signal |
| **GA-LSTM** | Genetic Algorithm- Long Short Term Memory |
| **GA** | Genetic Algorithm |
| **IC** | Intelligent Compression |
| **IF** | Isoltion Forest |
| **KF** | Kalmen Filtering |
| **KNN** | K-Nearest Neighbour |
| **LF** | Load Forecasting |
| **LR** | Linear Regression |
| **LSTM** | Long Short Term Memory |
| **MA** | Multithreading Approach |
| **MWh** | Megawatts Hour |
| **M2M** | Machine-To-Machine |
| **MAE** | Mean Absolute Error |
| **MSE** | Mean Square Error |
| **OSVM** | Open Support Vector Machine |
| **PJM** | Pennsylvania-New JerseyMaryland Interconnection |
| **RA** | Random Approach |
| **RFR** | Random Forest Regressor |

| | |
|---|---|
| **RNN** | Recurrent Neural Network |
| **SVR** | Support Vector Regressor |
| **SG** | Smart Grid |
| **SOS** | Stocastich Outlier Selection |
| **SF** | Smoothing Filtering |
| **TDR** | Time-Domain Reflectometry |

# Chapter 1

# Introduction

The development of new technologies, a large data is generated by various devices which is called Big Data (BD). These data is used in Smart Grid (SG) where it is needed for maintaining communication in networking. SG is used many devices such as AMI, Supervisory Control and Data Acquisition (SCADA) and various sensors [2]. Sensors generate a lot of energy from the home appliances and systems. AMI generates data at every 15 minutes and SCADA generates at every 15 seconds. BD is monitoring and a lot distributing grids. It is capable to forecast and scheduling the loads. AMI and SM are measuring energy consumption data at every five minutes and 15 minutes [3].

Big Data Analytics (BDA) is a process where large data is analysed so that data becomes smooth and readable. It is capable to control Big Data Management (BDM) [4]. It is easy to store, process and extracts data from huge generated data by various sources such as sensor, equipments and devices. We can make our data secure, reliable by using BD tools in Smart Grid (SG). SG provides good solutions of problems such as secure and reliable data. BD is very beneficial in SG modification of customer behaviour, conservation, consumption and DR. The aim of the current work is to predict the energy demand response in SG [5].

In recent years, the Big Data techniques are highly used in demand response management. Demand response management is essential in consumption of energy during peak-load times. Here, peak load time is a period of time when electrical power is required a sustained period (without interruption) which is based on demand. We can say that for a moment high electricity demand which is called peak load. However, reduction of energy in peak load usually results in shifting of the energy to an off-peak period (not being in the period of maximum use or business) [6]. It is possibility that peak load reduction can particulary reduce the load on the system. Basically, demand response consumers have some MW energy and they have ability to manage, reduce power by using for period of time. For managing power houses can offer energy by started generators, by changing production schedule. A group of houses can turn switches off the electrical devices such as freezers for an hour at a time. The grid is essential thing in electrical energy [7]. It is a type of network of transmission lines and used for electric power distribution. It is also

known as the grid related terms such as SG, Power, electricity generation, consumption and distribution. When electrical grid is combined with computer intelligence called smart grid. In the SG millions of sensors will be there to manage the power, voltage, current and other things. The demand side load management perform effectively by changing the way of energy generation and consumption [8, 9].

## 1.1 Background

Electricity devices such as Advanced Metering Infrastructure (AMI), Smart Meter (SM) and home appliances produces huge data which is called Big Data. By using Big Data Analytics (BDA) the supply and demand of energy can be managed in Smart Grid (SG). Therefore, prediction of consumption of energy is necessary part of our work so that supply and demand can be balanced [10]. BDA are very helpful for future predictions of Energy in Smart Grid (SG). Due to increasing modern technologies energy is essential part of every field. Data is growing very rapidly through sensors, social media, internat and other devices. Recently, Smart Meter (SM) measures data every 15 minutes and in some cases five minutes [11]. AMI measures data of remotely consumers at 15 minutes or hourly intervals. For reducing ambiguous data or abnormal data from storage devices is a great challenges. Therefore, due to increasing high data, storage space is effected. There is a more chances of fractions of data which is created by load variations in real time [12]. It is denoted that the chances of increasing time gap which could be reason inaccurate optimization. To overcome of this issues various speed-up operation methods by prediction algorithms such as GA-LSTM, Random Approach and multi-threading could be a future solution. By using various prediction algorithms, ambiguous data in storage devices could be reduced [13].

### 1.1.1 Smart Grid

When electricity flows from one system to another is called grid. It is an interconnected network for supply energy from supplier to consumers. Smart Grid (SG) is an electrical grid which is capable to manage large data. It is an electrical grid which developed technologically [14]. This enables two-way communications between suppliers and consumers. It has quality for self-healing and auto-restore capability. SG has Smart Meter which measures consumed energy and it gives more control to the user for energy consumption and billing [15]. The SG empowers the consumer to manage efficient energy. When electrical grid combines with In formation Technology (IT) is called intelligent grid because it handles large time series data. There are various types of energy comes from many energy

resources from where sufficient power is supply to the consumers by utilities [16]. SG is an electrical grid which contains a variety of operational and energy measures by Smart Meters and smart appliances. In SG energy generation, transition and consumption are possible. There are various challenges, application and the state-of-art of Big Data are discussed. The Big Data challenges and processing are defined [17].

### 1.1.2 Big Data

Big Data is a large set of data which is so vast volume that is complex to manage. Due to broadness of Big Data, it is critical task to collect, interpreted and analyse about a single system which is huge in amount. There are three types of Big Data used in Data Analytics such as structured, unstructured and substructure data. Structured data is data that has been organised into a formatted repository generally, a database so that it's elements can be made addressable for more effective processing and analysis [18]. Unstructured data is data that has been not organised in a traditional row-column database. It is the opposite of structured data which is stored in fields in a database. There is no proper model made for data implementation such as records, documents, numbers and dates. Unstructured data files often include text and multimedia contents. For example, includes e-mail messages, word processing documents, videos, photos, audio files, presentations, web pages and many other kinds of business documents [19]. Semi-structured data is mixture of structured and un structured data. For example, Common Separated Value (CSV) and extensible markup language (XML). However, due to it's complexity data are characterised in 4V such as volume, variability, velocity and variety. Volume defines how big a dataset. Velocity defines about speed of data which is processed by systems. Variability explains about data consistency. Variety explains about different types of data [20].

## 1.2 Big Data Analytics

Big data Analytics is used to interprete, process, and manage the huge data. The process of analysis of huge amount of diverse datasets which uses advanced analytics techniques is called BDA. It is easy to store, process and extract large data which are generated from various sources such as sensors, SMs, AMI and SCADA in BDA. By using BDA we can make data or information more secure, reliable, flexible and scalable. It provides good solutions of problems such as security and reliability of large data [21]. This is applied for modification, consumption and conservation for meeting the demand responses. SG employed with Big Data technologies which provides self healing, self automated, fault

tolerance, flexible and scalable system. There are four types of BDA such as Descriptive, Predictive, Diagnostic and Prescriptive Analytics. In descriptive data, users uses past data for analysis. In Diagnostic Analytics data are diagnosed for solution of problems. Predictive analysis defines data is analysed for future work. Prescriptive analysis is next step of predictive analysis. There are various types of data analytics defined below. BDA generates new discovery in the modern society in spite of many challenges are introduced in computation [22]. There are various trends of BDA which focuses on hardware, software and applications of industrial landscape [23].

### 1.2.1 Descriptive Analysis

In descriptive analysis data is analyzed. The data points are summarized in a constructive manner so that patterns might emerge. It fulfills each situations of data. In sensory evaluation the potential use of descriptive analysis is discussed [24, 25]. Here, the main ambition is to get balanced energy between the supplier and consumers. For improving the quality of energy many method such as real-time control and scheduling of distributed energy sources are needed load demand in SG. For time series data analysis historical energy consumption data are needed. The energy management is supported by time series analysis which provides better performance. The block-chain method is used for better quality of resources [26] .

### 1.2.2 Diagnostic Analysis

Diagnostic Analysis is very special technique. It is used for data interpretation and analysis perfectly to know what actually happened. It is applied in medicine or clinical domain. In this analysis, pictures of data sample defined dynamically [27].

### 1.2.3 Predictive Analytics

Predictive Analytics is used for forecasting events. It is used in various techniques from data mining, predictive modelling and machine learning so that it can predict future events. There are various types of data structures such as volumes, varieties and variabilities are defined. The traditional data processing methods are oriented [28].

### 1.2.4 Perspective Analytics

Perspective Analytics is detailed examination of events. It focuses on "what should be done". This analytics is very helpful for decision making which is beneficial for improvements of customers experience and productions. It is mainly used in the domain of busi-

ness analytics. It gives many tasks as adaptive, automated and time dependent to the organizations with adaptive, automated, and time- dependent for better performance [29].

### 1.2.5 Customer Data Analytics

In customer data analytics, behaviour of customers such as how much energy and data are used by customer. Here, demand of customer is fulfilled is checked so that actual decision of data transformation can be taken. Sometimes demand response is not fulfilled and it is most important to analyze behavior of customers. Valued added data are very effective for customer data to improve quality of the data transmission. There are various devices such as washing machine, Freeze, Woven and another electronic devices requires synchronizes and sequential demand response. By the help of customer data analytics data are efficiently stored, managed and processed. The economical balance by the customers are defined [30].

## 1.3 Demand Response of Energy

Demand response of energy is the change in the power consumption of an electric energy which is used by consumers for reducing prices. So that, the energy for demand and supply can be managed in a better way. It is a non-mandatory PJM program which allows consumers to reduce electricity bill during peak load. Demand and supply are dynamically controlled [31].

### 1.3.1 Emergency Demand Response

Emergency Demand Response is based on reliability program. It is very helpful to measure power reductions during triggered (set off). Demand Response is most important paradigm now a days. Therefore, for balancing energy Emergency Demand Response Program (EDRP) started. Here, it requires demand reduction which belongs emergency situations. This ailment is declared by New York Independent System Operator (NYISO). The telecommand is not required and interval metering is sufficient. During emergency period the energy is available as $500/MWh$ or the zonal real-time Location Based Marginal Price (LBMP) [32].

### 1.3.2 Economic Demand Response

Economic Demand Response is related to industrial energy management to reduce cost of energy. There are many facets to make programs for ancillary services such as the independent system operators (ISOs)/regional transmission organizations (RTOs). These are standard program which designed for economically sufficiency. They have paid attention towards on Demand Response [32].

### 1.3.3 Ancillary Services Demand Response

Ancillary Services Demand Response is helpful for security and stability of the grid. It provides various operations under generation and transmission of energy. Regulation services are capable to balance electricity supply and demand on small notice. Ancillary services indicates to a functions that is very beneficial for grid operators. It maintains a authentic electricity system. Ancillary services maintain the proper flow and direction of electricity, address imbalances between supply and demand. This is very helpful in the system recovery after a power system incidence. Demand Side Ancillary Service Program (DSASP) are provided to the system for regulation. Moreover, NYISO controls requirements and availability of resources [32].

### 1.3.4 Demand Side Management

Demand Side Management (DSM) is also called load management. It manages of energy generation and consumption. This is the plan for optimizing energy to reduce the price of electricity bill. DSM manages site,s consumed energy so that customers can shift load during peak time. Demand-side management (DSM) technique is helpful for the planning, implementing and monitoring activities of electric companies. These are designed to motivate the consumers to change their level and pattern of energy consumption.

## 1.4 Research Gaps

1. Big Data Analytics has a great role in meeting the various ancillary services of SG and demand response is one of the important ancillary services. Therefore, different techniques of data analytics has to be explored in context to SG.

2. Limited work has been done on Big Data Analytics in SG especially in demand response management.

3. There is a large possibility of Data Anomaly Detection Techniques as it happens in

real time data acquisition from data generation devices. There is a need to develop an algorithm which can handle the Big Data in terms of Anomaly Detection.

4. The role of data storage in the case of Big Data generated from SG has not been explored to its full potential. These storage techniques will be helpful for demand response.

## 1.5 Research Objectives

1. To study and analyze different techniques of Big Data analytics and their applications in demand response management in Smart Grid Domain.
2. To design algorithms for reducing the storage space of ambiguous data using Big Data Analytics for demand response management
3. To design algorithm using Anomaly Detection Techniques of data for efficient demand response management
4. To verify and validate the proposed algorithm

## 1.6 Thesis Contribution

1. Multi-threaded based GA-LSTM technique is used for improving the performance of the algorithm with overall execution time.
2. Filtering techniques are used to reduce ambiguous data which are generated from various resources such as Smart meters and Advanced Metering Infrastructures.
3. ARIMA and Prophet Models are applied for better performance with real data time.
4. To validate the performance of Anomaly Detection Techniques for large data, real time data of PJM has been used to validate the results with different evaluation metrics.

## 1.7 Thesis Organization

The Thesis is organized as follows.

**Chapter 1: Introduction**
The chapter begins with the background of the Smart Grid, Big Data, types of Big

Data Analytics, Demand Response Management and Demand side Management. The chapter finally comes up with thesis contribution and organization.

**Chapter 2: Literature Survey**
This chapter presents a detailed literature survey of existing demand response management in Smart Grid. Further, ARIMA and Prophet Models for time series data prediction have also been outlined. Optimization techniques such as GA and random approach have been discussed for parameter tuning. Literature related to Anomaly Detection Techniques for prediction of energy using different filtering techniques is also discussed. Machine learning techniques such as decision tree, Bays Rules, Artificial Neural Network, RNN and LSTM applications in demand response has been outlined. Finally, this chapter outlines the research gap.

**Chapter 3: Big Data Analytic for Energy Consumption Prediction**
In this chapter, a multi-layer GA-LSTM model is proposed for energy prediction. It provides a better result as compared to existing techniques. The purpose of using GA is to optimize the parameters of the LSTM. To verify the effectiveness of the proposed system, different parameters of LSTM have been used for reducing the errors. Further, Multi-threaded based GA-LSTM technique is used for improving the performance of the algorithm with overall execution time. After identifying the lower and upper bound of the LSTM parameter, GA is used to optimize the LSTM for better performance. Finally, to validate the performance of GA-LSTM approach for large data, real time data of PJM has been used to validate the results with different evaluation metrics.

**Chapter 4: Different data Filtering Techniques for Big Data**
The filtering techniques primary task is to handle non-linearity in the input dataset. Various filtering techniques reduce the redundant data for energy consumption prediction. In this chapter, five different filtering techniques such as Butterworth, Smoothing, Kalman, Frequency, and Filtfilt have been used to preprocess the five different power consumption datasets. Further, Long Short Term Memory model is used on the processed data for the power consumption prediction.Finally, results are analyzed with different performance metrics such as Mean Absolute Error (MAE), Mean Square Error (MSE), Median Absolute Error (MDAE), Correlation ($r$), Coefficient of determination ($R^2$) and Accuracy.

**Chapter 5: Demand Response Management using Prophet Model**
The ARIMA model requires expert knowledge as a prerequisite to make use of it. In addition, it is not flexible to use and is non-automatic. The Prophet model overcomes all the aforesaid limitations and is a powerful tool for prediction. It gives

precise results with non-seasonal trends and also incorporates non-linear trends with datasets. In this chapter, optimization of the parameters for the Prophet model is achieved for better performance. Secondly, preprocessing techniques are applied to clean the data. Further, the abnormal data is removed for the prediction of consumption of energy in forecasting. Finally, the ARIMA model and the Prophet model are compared and analyzed with different performance metrics.

**Chapter 6: Ensembling of Data Anomaly Detection Techniques in Energy Prediction**

It is very useful to handle the non-linearity in the input and output data through many Anomaly Detection Techniques. These Anomaly Detection Techniques helps in reducing the abnormal data for energy prediction. In this chapter, different Anomaly Detection Techniques are compared and tested as a preprocessing technique with LSTM, ARIMA and Prophet models and results are analysed with different performance metrics. Further, different Anomaly Detection Techniques such as forest, K-NN, Histogram, SVM, SOS, and OSVM have been used and compared with their preprocessing algorithm on different datasets. Finally, the novelty of the chapter lies in the preprocessing techniques on the LSTM, ARIMA and Prophet model where different Anomaly Detection Techniques have been compared.

**Chapter 7: Conclusion and Future Directions**

This chapter summarizes the conclusions drawn from the thesis along with the possible future directions.

# Chapter 2

# Literature Survey

In this section, many researchers have done works in Big Data Analytics, Smart Grid and Demand Response Management domain. Here, different types of magnificent researchers contributed in the Big data, Smart Grid and Demand Response.

## 2.1 Big Data Analytics in Smart Grid

Due to development of modern technologies consumption of energy is increased rapidly. Huge data is generated everyday from various sources which is called Big Data. These large data are created complexities in data processing. Therefore, current status and many challenges in Big Data Analytics in SG are discussed [17]. There are huge data are generated in SG by Smart Meter which is called Big Data. Smart Meter measures consumed energy at every 15 minutes or hourly. Many Big Data Analytics algorithms have been described in the smart grid literature. However, smart meter analytics are defined for better software performance [33]. There are various types of techniques such as Decision tree and SVM-based data analytics for stealing detection in SG are defined [34]. The energy load forecasting in large data is described [35]. Large Data is managed in SG and various issues such as Energy generation, consumption and production are management [36].

## 2.2 Demand Response Management in Smart Grid

Demand Side Management is discussed in SG using BDA. Here, advanced structured of Grid which controls and monitors peak demand and favourable load [37]. Application of Big Data in many challenges are discussed [38]. The ubiquitous deployment of improved sensing architectures in Cyber Physical way as like the Smart Grid which has resulted in an unmatched data explosion. The structures of Big Data such as high volumes and velocity and time-series data are described [39, 40]. In authors discussed about the use of a algorithm such as smart-direct load control (S-DLC) and load shedding. These algorithm short outs the power outages during sudden grid load changes as well as minimize the peak-to-average ratio. The algorithm is used for forecasting, load-shedding and S-DLC.

It is applicable in Internet of Things and stream analytics. The real-time load control is possible in this algorithm. It generates a daily schedule for consumers supplied with self regulating electronic devices. Everything is based on their requirements, thermal comfort, and the predicted load model [41].

## 2.3 ARIMA and Prophet Models for time series data prediction

The researchers developed Building-level occupancy data. It is based on ARIMA model which is used for prediction of energy [42]. Many authors defined about food prediction which s based on historical data and these effects food chain of the company. The historical required data was used to improve several autoregressive integrated moving average (ARIMA) models by using Box–Jenkins time series procedure [43]. The modeling and forecasting is based on time series data where requirement of food in a food company is predicted. ARIMA model is used for by using time series approach. To improve various ARIMA models by using BoxJenkins time series method [44]. The authors defined about COVID-19 pandemic disease. Here, mainly Machine Learning model is used for forecast the pattern of the disease in Indonesia where they are finding out the approx value of returning normality. The Facebook's Prophet Forecasting Model and ARIMA Forecasting Model are used to compare their performance and accuracy on the dataset [45]. Moreover, the authors used time series data which is based on air pollution forecasting. They used SARIMA and Prophet model [46]. Here, Authors defined application of Facebook's prophet model for successful sales prediction and it is based on real-world data [47]. The author also discussed about traffic prediction which is based on Prophet model and Gaussian process regression techniques [48].

## 2.4 Optimization techniques GA

Genetic Algorithm (GA) is an algorithm which is based on natural selection and method of genetics. It is population based technique. This algorithm is used for finding optimal solution of complex issues. GA is used for optimization which is based on fitness function [49]. The authors described about prediction of wind energy by using GA-LSTM with time series data [50]. Moreover, some authors highlighted aggressive behaviour prediction by using GA-LSTM optimization method [51]. Survival of the fittest is the main purpose of this algorithm which is discovered by Charles Darwin's Evolution Theory. They analysed about Evolution Theory which is based on GA [52].

## 2.5 Anomaly Detection Techniques for prediction of Energy

For prediction of Energy Anomaly Detection Techniques has great role in SG. Here, SG Data is used regression-based online Anomaly Detection Techniques [53]. Graph based Anomaly Detection Techniques is used on the SG [54]. Unsupervised anomaly detection technique which is based on the Histogram where it is based on outlier score [55]. The author also described time-pattern profiling from SM to find consumed energy [56]. Graph matching method are discussed to find in electric energy on SG [57]. Anomaly Detection Techniques in SG are described [58].

## 2.6 Machine Learning Techniques

Machine learning (ML) is a subset of artificial intelligence (AI). Now a days, there is massive use of ML in every field. Machine Learning is an approach which is broadly discussed as a machine is capable to imitate to intelligent human behaviour.

### 2.6.1 Decision Tree

Decision tree is a supervised machine learning. In this tree, leaves are final decision or outcomes. Here, we know what is input and what is the related out put of the data during training period. The data is constantly split according to alright parameters.

### 2.6.2 Bays Rule

Bayes Rule is a way for descriptions of probability of an incidence which is based on former knowledge of conditions that is related to that events. It,s application such as Bayesian probabilistic regression model are used. It is helpful for curve fitting method. The Curve fitting method is the way of building a curve, or mathematical function, which has the best fit for a series of data points, possibly subject to constraints.

### 2.6.3 Artificial Neural Networks

The Artificial Neural Network (ANN) is Biological term. The biological neural networks which develop the architecture of a human brain. As like to the human brain the neurons are interconnected to each other in many layers of the network. Nodes are nothing but these are neurons of the brain. There are more than trillions of nerve cells are found in human brain. Neural Network Algorithms are very important for our work because there

is an interaction between input and output. ANN is used in various area such as speech recognition, image analysis and biology.

## 2.7 Time Series Prediction Techniques

Deep Learning is subset of ML and AI with an algorithm which is simulated by the structure and function of the brain, which is called an Artificial Neural Network (ANN). For example, computer vision, speech recognition natural language processing.

### 2.7.1 ARIMA

Auto-Regressive Integration Moving Average (ARIMA) is new emerged technique. This technique is combination of AR, I, and MA. It is helpful for load forecasting and other professions. Here, p, d, q indicates for AR, I and MA. In this technique, knowledge about past data is necessary. Prediction of data is based on past data. ARIMA stands for Auto-Regressive Integrated Moving Average. It Performs forecasting a time series data by making a stationary series using different orders of inferencing. ARIMA(p, d, q) is non-seasonal model p: number of autoregressive terms. d: number of non-seasonal differences needed for stationary. q: number of lagged forecast errors in the prediction equation. ARIMA Model performs on the time series data which is based on historical demand data. These data are utilized for future prediction and helpful in energy consumption issues [43].

### 2.7.2 Prophet

Prophet is an open-source software released by Facebook's Core Data Science team. It is a procedure developed for forecasting time series data which is based on an Additive Model (AM). AM is non-parametric regression analysis where the predictor does not work over predetermined knowledge but build according to information which is derives from data. Here, non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. It works best with time series that have strong seasonal effects and several seasons of historical data. Bayesian curve fitting method is used for operation.

### 2.7.3 RNN

Recurrent Neural Network (RNN) is part of ANN. The Long Term dependance is drawback of RNN. It is type of algorithm which is used for sequential data. The Apple's Siri and Google's voice search are used RNN algorithm especially. Moreover, It is the first

and foremost algorithm that remembers its input. Because of an internal memory, which built it excellently matched for machine learning problems that elaborated sequential data. The Long Term Dependance is a drawback of RNN. The prediction of consumption of energy is necessary for consumers [59].

### 2.7.4 LSTM

Long Short Term Memory (LSTM) is a type of Deep Learning method. Recurrent neural networks (RNN) are the state of the art algorithm for sequential data and are used by Apple's Siri and Google's voice search. It is the first algorithm that remembers its input, due to an internal memory, which makes it perfectly suited for machine learning problems that involve sequential data. LSTM is mainly used for time series dataset for prediction of energy. It works with feedback connections and memorises previous information inside the network. It has capability of solving time series and nonlinear prediction problems. The major problem of RNN is "Long term dependency" therefore, LSTM is used to overcome this problem. The cell state is the key of the LSTM and it is like a conveyor belt. LSTM is capable of adding or removing the information and it is regulated by structures which are called gates. Gates are the mode where information are optionally chosen. These gates are working with sigmoid neural network and a point to point multiplication operation. There are mainly three types of gates such as input gate, output gate and forget gate. Tanh, sigma and Relu are the activation functions mainly used in the LSTM network. The below subsection describes about the different techniques of LSTM for handling the large datasets. Prediction of energy consumption is described [59]. Deep learning model such as LSTM are defined for energy forecasting by using Feature Selection ($FS$) and Genetic Algorithm ($GA$) which is compared with machine learning approaches [60].

# Chapter 3

# Big Data Analytics for Energy Consumption Prediction

Smart Grids (SG) have smart meters and advance metering infrastructure (AMI) which generates huge data. This data can be used for predicting energy consumption using big data analytics. A very limited work has been carried out in the literature which shows the utilization of big data in energy consumption prediction. In this chapter, the proposed method is based on Genetic Algorithm - Long Short Term Memory (GA-LSTM). LSTM memorises values over an arbitrary interval that manages time series data very effectively while GA is an evolutionary process that is used for optimization. GA combines with LSTM to process hyperparameters such as hidden layers, epochs, data intervals, batch size and activation functions. Hence, GA creates a new vector for optimum solution that provides minimum error. These methods provide the best performance when compared with existing benchmarks. Moreover, GA-LSTM is used in a multi-threaded environment which increases the speed of convergence. Here, the multi-core platform is operated for solving one dimensional GA-based inverse scattering problems. The result shows that GA-LSTM provides better convergence as compared to random approach techniques. For validating the results, Pennsylvania-New Jersey-Maryland Interconnection (PJM) energy consumption data has been used while adopting different performance evaluation metrics.

## 3.1 Introduction

Smart Grid (SG) is a technologically evolved electrical grid. It incorporates information technology into the grid and enables two-way communication between the electric utility and the end consumer. The physical infrastructure is replaced with a digital one, and conventional analog technologies are replaced with improved digital and power electronics. This technology makes the existing grid more efficient and reliable by reducing the number of outages and adding a self-healing or auto restore capability. Power is immediately rerouted when an outage occurs, and power is fixed to the affected area.

Further, it promotes using renewable energy resources, reducing the carbon footprint.

Also, SG being technologically advanced, consists of various energy measurement devices such as smart meters and advanced metering infrastructure (AMI). These appliances generate massive data, which can be termed as big data [61].

The generation of huge data also depends on other equipment such as supervisory control and data acquisition (SCADA) and phase measurement unit (PMU), which generates data in seconds [62]. Since there is a large number of measuring devices, data generation needs to be handled in a very efficient way. Therefore, big data management becomes an essential task in SG. Moreover, many other tasks can be done using this data. One of them is energy consumption prediction.

The demand for energy increases due to economic and population growth. This growth can lead to an increased supply and demand gap if not predicted well beforehand. Hence, big data analytics play a large role in the proper utilization of energy, and energy prediction can be one of the ways to reduce the demand and supply gap. Moreover, for the stability of demand and response, load forecasting has an essential role to play in the SG system [63]. Many researchers have tried to achieve reliable and efficient energy management through big data techniques. For getting such a type of energy management system, they combined data analytics and a scalable selection procedure so that the prediction of supply and consumption of energy could be stable. Big data analytics and cloud computing have been described for managing energy supply and demand in SG [64]. For managing data, researchers illustrated various big data techniques in SG.

Big data analysis is a critical challenging task and can be overcome by various smart tools and techniques such as support vector machine (SVM) and decision tree analysis (DTA). In a similar line, Wang *et al.* discussed short-term load forecasting, which is based on recurrent neural network (RNN) and long short-term memory (LSTM) [65]. RNN is rather an enhancement of an artificial neural network (ANN), which is useful for processing the output directly to the first layer. In another case, LSTM is a part of deep learning, and it overcomes the drawbacks of RNN. For energy forecasting, the LSTM technique is important in analyzing the time series data. It uses big data strategies to reduce the storage space, analyze the data for decisions on different models, and make several frameworks [66]. Similarly, Pacini *et al.* suggested encoder-predictor for short-term load forecasting as an effective energy prediction [67].

Few authors used deep neural networks (DNN) for energy forecasting. Amarasinghe *et al.* discussed demand side management using DNN [68]. The authors tried to discover intelligent energy systems management and smart load distribution focused on real-time pricing. Mohammad *et al.* defined the energy load forecasting model, which is based on DNN [69]. Furthermore, power demand forecasting using LSTM Neural Network is dis-

cussed in [70]. Here, LSTM provides better performance as compared to the current work. Few authors have analyzed the DNN & Genetic Algorithm (GA) and concluded that this combination provides better performance. In addition, various authors applied the optimal RNN - LSTM model for energy forecasting. Residual Network (ResNet) and LSTM have been used in this approach to develop the forecasting approach [71]. LSTM-RNN model is largely used in energy forecasting for small datasets. Using this approach, a few authors used LSTM-RNN-based day-ahead load forecasting [72] using smart meter data of different localities. In a similar work, Sainath *et al.* discussed short-term load forecasting based on CNN and LSTM [73]. Many authors illustrated various applications, models, and challenges in predicting energy. To overcome these challenges, different machine learning models have been used. In [74], authors described statistical-based modeling, machine learning, and deep learning-based model. Further, Diamantoulakis *et al.* suggested about prediction model for energy which is based on dynamically demand response in SG [75]. They suggested dynamic energy management so that sufficient energy can be managed and costs can be reduced.

### 3.1.1 Related Works

Energy consumption prediction plays a significant role in maintaining the demand and supply gap in SG. It provides better decisions for the power utility. Since energy prediction is a time series data, it is desirable to work on techniques where big data challenges can be handled by minimizing the error between actual and predicted value. In this context, M H Rashid uses smart meter data and developed big data analytics techniques for analyzing time series data. [76]. However, the author has taken a small dataset and compared it with other techniques which are not effectively considered.

Minimal work with respect to energy forecasting using big data analytics has been done using existing methods such as the backward propagation neural network, support vector regression (SVR), generalized radial basis function neural network, and multiple linear networks. In a different work, Khuri *et al.* described 0/1 multiple knapsack problem [77] where proposed technology works on historical data. However, the authors have not used big data analytics. Few authors work on a similar line of energy management, and they tried to improve energy consumption prediction using CNN and Bi-directional LSTM (Bi-LSTM) neural network [78]. They applied electrical energy consumption prediction using the Bi-LSTM model to improve results. In this approach, the authors used a small dataset. Other researchers described dynamic test data generation using GA in energy prediction strategy [79]. However, none of them have used large datasets.

Sulaiman *et al.* used smart meter data and solved big data analytics using an adaptive

neuro-fuzzy inference system [80]. They used this data to predict the day scheduling and verified the prediction accuracy to 84.03%. In a very close work, A D Teres used the MapReduce algorithm and developed histogram visualization for SG [81]. However, the research was not intended for energy prediction. Simhan *et al.* discussed cloud-based approach for dynamic demand response for the SG [82]. However, the authors did not focus on energy forecasting. In a different approach, Kaur *et al.* tried to elaborate on the LSTM-based regression approach to solving the energy management of smart homes [83]. They verified the results with the existing techniques, and data were taken for 112 houses to validate the results. Furthermore, Couceiro *et al.* made a stream analytics for energy prediction [84]. They used data streaming for handling large datasets for real-time applications in power systems. However, their work was not validated in a real-time data stream. A short-term load forecasting using LSTM-RNN in the SG [85]. Here, the authors validated the result for a single household to forecast the load.

In recent research, Zhang *et al.* used SVR and adaptive GA to optimize the parameters to get the best load forecasting model [86]. They performed and validated their results on a specific ratio value using tiny datasets. In a similar work, Eseye *et al.* proposed machine learning tools based on binary GA [87]. They applied the feature selection process and gaussian process regression for measuring the fitness score. A similar approach is discussed in [88], where the authors used a hybrid model of GA and LSTM. They used half-hourly data from the Australian energy market operator. However, their testing and training datasets were verified on small datasets. The authors used GA-ANN techniques for wind forecasting [89]. The authors used meteorological data and compared double-stage backpropagation-trained ANN. In a similar work, Jaidee *et al.* presented a method for finding optimal parameters of a deep learning model by GA [90]. They tested the results with many other techniques, including LSTM. However, their validation was limited to small datasets.

### 3.1.2 Motivation

Load forecasting is difficult in SG due to its complex and nonlinear relationship with different datasets. The researchers have adopted other data mining and machine learning techniques, but very few have taken large datasets to validate their proposal. The massive use of classification and regression analysis still poses a challenge when large data is considered at the implementation level. From the literature review, it has been observed that very little work has been done on big data techniques for energy prediction. It has also been observed that time series data can not be handled using conventional machine learning tools when large data is involved.

Further, load forecasting techniques involve large datasets, and to get early convergence, we need some optimization tools, along with LSTM. Few authors proposed a different algorithm to develop load forecasting with big data. Still, none have analyzed the results regarding the multi-threading approach of GA-LSTM, which increases the convergence speed. Further, energy prediction is one of the techniques to understand the proper utilization of energy resources. Therefore, we need to analyze big data and use it for load forecasting. Accurate load forecasting may reduce the supply and demand gap of electrical usage.

### 3.1.3 Contributions

In this chapter, a multi-layer GA-LSTM model is proposed for energy prediction. It provides a better result as compared to existing techniques. The purpose of using GA is to optimize the parameters of the LSTM. To verify the effectiveness of the proposed system, different parameters of LSTM have been used to reduce errors. The significant contributions of this chapter are as follows.

- Multi-threaded based GA-LSTM technique is used for improving the performance of the algorithm with overall execution time.
- After identifying the lower and upper bound of the LSTM parameter, GA is used to optimize the LSTM for better performance.
- To validate the performance of the GA-LSTM approach for extensive data, real-time data of PJM has been used to validate the results with different evaluation metrics.
- To find the interval size of the optimal data that gives minimum mean square error.

### 3.1.4 Organisation

Section II provides the methodology of the proposed work. Section III explains the dataset along with performance and evaluation parameters. Section IV outlines the results and discussions. Finally, the chapter is concluded in section V.

## 3.2 Dataset and its Description

### 3.2.1 Data Description

The dataset is a multivariate time-series data collected from Pennsylvania-New Jersey-Maryland Interconnection (PJM), which is a regional transmission organization (RTO)

Table 3.1: Hourly sample dataset of energy consumption

| SN | Date | Time(hrs) | Energy (MWh) |
|---|---|---|---|
| 1 | 01/01/2002 | 1.00 | 14107 |
| 2 | 01/01/2002 | 2.00 | 14410 |
| 3 | 01/01/2002 | 3.00 | 15174 |
| 4 | 01/01/2002 | 4.00 | 15261 |
| 5 | 01/01/2002 | 5.00 | 14774 |
| 6 | 01/01/2002 | 6.00 | 14363 |
| 7 | 01/01/2002 | 7.00 | 14045 |
| 8 | 01/01/2002 | 8.00 | 13478 |
| 9 | 01/01/2002 | 9.00 | 12892 |
| 10 | 01/01/2002 | 10.00 | 14097 |

in the United States of America [91]. PJM is a part of the Eastern Interconnection grid operating an electric transmission system serving all parts of Delaware, Illinois, New Jersey, and North Carolina. The hourly energy consumption data comes from PJM's website and are in Megawatts Hour (MWh). The dataset is daily and weekly based on time series data. The dataset is of the PJM East that consists of data from 2002-2018 for the entire eastern region where 2002 to 2015 is used for training and 2015 to 2018 is used for testing [92].

Energy consumption has unique characteristics. The regions have changed over the years, so data may only appear for specific dates per region. GA-LSTM model is applied to these large datasets. The values of variables are compared between actual and predicted values.

Since hourly-based data is very complex and unsuitable for the LSTM model, the focus was laid on daily and weekly-based data. This data is compatible with the GA-LSTM model and provides more than 90 percent of result accuracy. For validation of the proposed work, three types of datasets are used, and they are hourly, daily, and weekly whose sample data is mentioned in Table 3.1, Table 3.2 and Table 3.3 respectively.

### 3.2.2 Performance measures used in this energy forecasting

#### 3.2.2.1 Mean Absolute Error (MAE)

MAE measures errors between two variables, such as $x$ and $y$. The observations are expressed about the same event. It is described as per the following equation.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|a_i - p_i| \tag{3.1}$$

Table 3.2: Daily sample dataset of energy consumption

| SN | Date | Energy (MWh) |
|---|---|---|
| 1 | 01/01/2006 | 363822 |
| 2 | 02/01/2006 | 389012 |
| 3 | 03/01/2006 | 431551 |
| 4 | 04/01/2006 | 439618 |
| 5 | 05/01/2006 | 388212 |
| 6 | 06/01/2006 | 392685 |
| 7 | 07/01/2006 | 394595 |
| 8 | 08/01/2006 | 393980 |
| 9 | 09/01/2006 | 417416 |
| 10 | 10/01/2006 | 444514 |

Table 3.3: Weekly sample dataset of energy consumption

| SN | Year | Week | Energy (MWh) |
|---|---|---|---|
| 1 | 2006 | 1 | 2799495 |
| 2 | 2006 | 2 | 2986229 |
| 3 | 2006 | 3 | 2884968 |
| 4 | 2006 | 4 | 2644030 |
| 5 | 2006 | 5 | 2614028 |
| 6 | 2006 | 6 | 2614028 |
| 7 | 2006 | 7 | 2562487 |
| 8 | 2006 | 8 | 2562487 |
| 9 | 2006 | 9 | 2356473 |
| 10 | 2006 | 10 | 2349789 |

where, $n$ is the number of observations, $a$ is actual energy consumption, and $p$ is the predicted energy consumption.

#### 3.2.2.2 Mean Square Error (MSE)

Mean square error (MSE) is an estimator which measures the average of the **squares of errors**. Here, the average square provides the difference between the predicted and actual values. MSE is given as follows.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(a_i - p_i)^2 \tag{3.2}$$

where, $p_i$ indicates predicted value and $a_i$ indicates actual value.

#### 3.2.2.3 Median Absolute Error (MDAE)

The median absolute error is crucial due to its robust approach to outliers. Here, the loss is calculated by taking the median of all absolute differences between the actual and the predicted value. In the below equation, $p_i$ is the predicted value of the $i^{th}$ sample and $a_i$ is the corresponding true value. MDAE estimated over $n$ samples is defined as follows.

$$MDAE(a, p) = median(|a_1 - p_1|, ..., |a_n - p_n|) \tag{3.3}$$

#### 3.2.2.4 Correlation

Correlation describes the statistical relationships between actual and predicted values. It is defined as follows:

$$r = \frac{\sum_{i=1}^{n}(a_i - \bar{a})(p_i - \bar{p})}{\sqrt{\sum_{i=1}^{n}(a_i - \bar{a})^2 \sum_{i=1}^{n}(p_i - \bar{p})^2}} \tag{3.4}$$

where, $r$ is the correlation, $a$ is the actual value, $p$ is the predicted value, $\bar{a}$ is the mean of all actual values, $\bar{p}$ is the mean of all predicted values and $n$ is the number of instances. Correlation lies in the [-1, 1] interval and considered to have good correlations if its value tends towards 1 or -1. The LSTM model is trained on 70 % of the dataset, and testing is done on the remaining 30% of the dataset. The trained LSTM model generates the predicted values compared with actual values. Correlation is the best parameter to understand the relationship between actual and predicted values. The correlation values lie between -1 and +1. The correlation sign denotes the association's nature, while the value denotes the strength of the association.

#### 3.2.2.5 Coefficient of determination ($R^2$)

The coefficient of determination ($R^2$) summarizes the explanatory power of the regression model and is computed from the sums-of-squares terms and given as per the below equation.

$$R^2 = r * r \tag{3.5}$$

where, $r$ is the correlation as mentioned in Eq. (4). $R^2$ lies in the [0, 1] range and is considered good $R^2$ if its value tends towards 1.

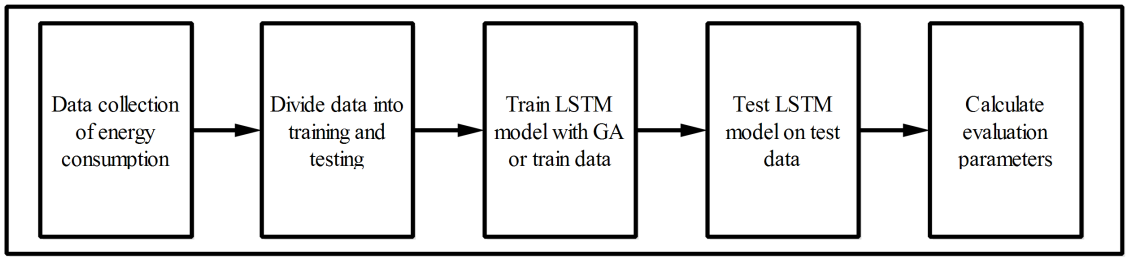



Figure 3.1: Work flow of the complete system

## 3.3 Methodology

### 3.3.1 Proposed work

The workflow of the complete system is shown in Figure 3.1. As can be seen from this figure, collected data is preprocessed and divided into training and testing sets. Once the dataset is divided, the LSTM model is trained with 70% of the dataset, and testing is done with the remaining 30% of the data. From the test data, a prediction of consumed energy is obtained. Further, to improve the model, LSTM parameters are tuned with GA for calculating the evaluation points. The below subsections presents the modeling of LSTM, GA, GA-LSTM, and multithreading in GA-LSTM.

### 3.3.2 Long Short-Term Memory

LSTM is mainly used for the time series datasets to predict energy. It works with feedback connections and memorizes previous information inside the network. It has the capability of solving time series and nonlinear prediction problems. The major problem of RNN is "Long term dependency"; therefore, LSTM is used to overcome this problem. The cell state is the key to LSTM, like a conveyor belt. LSTM can add or remove information, and it is regulated by structures called gates. Gates are the mode where information is optionally chosen. These gates work with a sigmoid activation function and a point-to-point multiplication operation. There are mainly three types of gates: input gate, output gate, and forget gate. Tanh, sigma ($\sigma$), and Relu are the activation functions mainly used in the LSTM network. The below subsection describes the different techniques of LSTM for handling large datasets.

#### 3.3.2.1 Handling a very Long Sequence data with LSTM

LSTM is capable of learning and capturing previous sequences of inputs. It can work nicely with one output, having many inputs but suffers if a long input sequence exists.

It is called sequence labeling or sequence classification. There are six modes of handling very long sequence data for classification problems. The starting point is to use the long sequence data without any process. However, this may take a long time to train. Further, an attempt to back-propagate across extremely long input sequences may result in vanishing gradients and, in turn, an unlearnable model. A reasonable limit of 250-500 time steps is often used in practice with large LSTM models. Therefore, a way to handle these types of long sequence data is to truncate them. Here, removing the time steps from the beginning or at the end of input sequences is done. It may be possible to summarize the input sequence in some problem domains. For example, when input sequences are words, it may be possible to remove all words from input sequences above a specified word frequency, such as and, &, the, and many more.

#### 3.3.2.2 Process of LSTM

In this subsection, the step-by-step working of LSTM is explained. The first step in LSTM is to decide what information to select from the cell state. This decision is taken by the forget gate, which determines what information to keep and what information to discard. Information from the input and previous hidden states is passed through a sigmoid function which squishes the values between 0 and 1. Values closer to 1 are kept, and values closer to 0 are discarded.

The second step is to obtain the current cell state from the previous cell state and input. The previous hidden state and the input are passed through the input gate, which consists of the sigmoid function, which squishes the values between 0 and 1 based on their importance. Values closer to 0 are unimportant, while values closer to 1 are. The hidden state and the input are also passed through the tanh function, which creates a candidate vector between 1 and -1; this regulates the network. The output of the input gate and the candidate vector is then multiplied. Finally, the obtained value is added to the product of the previous cell state and the forget vector to get the current cell state.

The third step decides what the new hidden state will be. The input and previous hidden state are passed through a sigmoid function to obtain the output. Next, the current cell state is passed through a tanh function. The obtained value and the output are then multiplied to decide what information the next hidden state carries. The product of this multiplication is the hidden state passed to the next LSTM cell along with the current cell state. The structure is shown in Figure 3.2.

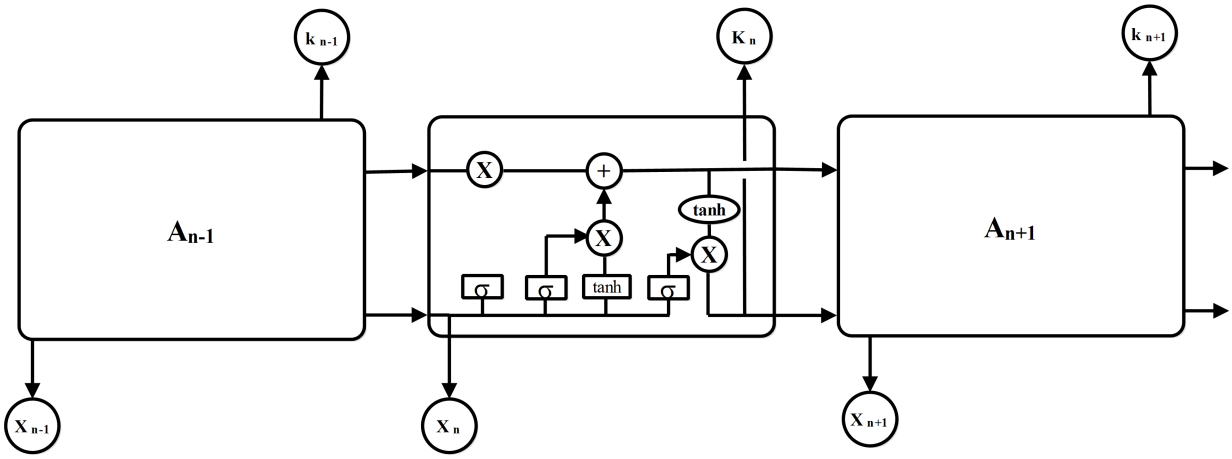



Figure 3.2: The operation of LSTM [1]

#### 3.3.2.3 Modeling of LSTM

This subsection explains the mathematical modeling of LSTM cells at every time step. LSTM cell contains several components such as forget gate $F$, which decides what information should be thrown away or kept, a candidate layer $C$ which holds all the possible values to be added to the cell state, an input gate $I$ which is used to update the cell state and output gate $O$ which decides what the next hidden state should be. Further, we represent the hidden state by $H$, and $C$ represents the cell state, and both of these are vectors. The current LSTM cell is considered as the time step $t$. In the following equations $'*'$ is an element-wise multiplication, $'+'$ is an element-wise addition.

First, the input and previous hidden state are passed through the forget gate of the LSTM cell, which has a sigmoid activation. It uses sigmoid activation because it needs to decide whether to forget information or not. The closer to 0 means to forget, and the closer to 1 means to keep.

$$F_t = \sigma(X_t * U_f + H_{t-1} * W_f) \tag{3.6}$$

here, $X_t$ is an input vector, $U_f$ and $W_f$ are the weight vectors for the forget gate and candidate gate, respectively, and $H_{t-1}$ is the previous cell output or the hidden state. The following equation represents the new state of the LSTM. We pass the hidden input and current input into tanh function to squish values between -1 and 1 which helps regulate the network.

$$C'_t = tanh(X_t * U_c + H_{t-1} * W_t) \tag{3.7}$$

where, $C'_t$ is the current cell state at time step $t$, it gets passed to the next time step. $H_{t-1}$ is the previous cell output and $X_t$ is the input vector. The input gate is represented as per the below equation. We pass current input and previous hidden state into a sigmoid function that decides which values will be updated by transforming the values between 0 and 1. 0 means not important, and 1 means important.

$$I_t = \sigma(X_t * U_i + H_{t-1} * W_i) \tag{3.8}$$

where, $I_t$ is an input gate at the time step of $t$, $U_i$ and $W_i$ are the weight vectors for the input gate and candidate gate, respectively, whereas $H_{t-1}$ is the previous cell output. The output gate is represented as follows. Here the input vector and the previous hidden state are passed through a sigmoid function.

$$O_t = \sigma(X_t * (U_o + H_{t-1}) * W_o) \tag{3.9}$$

where, $O_t$ is an output gate at the time step of $t$, $X_t$ is an input vector, $U_o$ and $W_o$ are the weight vectors for the output gate and candidate gate, respectively, whereas $H_{t-1}$ is the previous cell output. The current time step is mentioned below.

$$C_t = f_t * C_{t-1} + l_t * C'_t \tag{3.10}$$

where, $C_t$ is current cell step at time step of $t$, $f_t$ is a forget gate vector, $I_t$ is the input gate. The current cell output is mentioned below equation. This uses the output gate and cell state to give us the current hidden state.

$$H_t = O_t * tanh(C_t) \tag{3.11}$$

here, $H_t$ is the current cell output at time step of $t$ and $tanh(C_t)$ is the activation function used to find the current cell state. Now with the current memory state $C_t$, we calculate the new memory state from the input state and $C'$ layer.

$$C_t = C_t + I_t * C'_t \tag{3.12}$$

where, $C_t$ is the current cell state at time step $t$, and it gets passed to the next time step, and $C'_t$ is the new candidate gate. Now LSTM cell takes the previous memory state $C(t_1)$ and does element-wise multiplication with forget gate $F_t$ as per the equation mentioned

below.

$$C_t = C_{t-1} * F_t \tag{3.13}$$

This output will be based on our cell state $C_t$ but will be a filtered version. Therefore, we apply $tanh$ to $C_t$, and after this, we make element-wise multiplication with the output gate $O$, which will be our current hidden state $H_t$.

$$H_t = tanh(C_t) \tag{3.14}$$

Now we pass $C_t$ and $H_t$ to the next time step and repeat the same process.

### 3.3.3 Genetic Algorithm (GA)

GA is based on the survival of the fittest, which Darwin proposed. Mainly five steps are involved in GA: initial population, selection operator, fitness function, crossover, and mutation. The fitness function has a significant role in GA. Based on the requirements of LSTM, seven sets of chromosome samples are taken, and they are data interval size, number of epochs, batch size, number of hidden layers, dropout rate, and number of units in each layer. The selected dimensions are used for processing the GA-LSTM model. The results depend on the fitness score, which provides better results after comparing the predicted and actual values.

Moreover, mutation and crossover have an essential role in this algorithm. Here, chromosomes work as a potential solution to the target problem. It behaves as a binary string in a chromosome for processing the model. The chromosomes are generated randomly, and the one which provides the best performance is selected. The basic process of the flow chart of a GA is shown in Figure 3.3.

### 3.3.4 Optimization in LSTM Network with GA

The operation of the LSTM cell is shown in Figure 3.2 where three gates perform in coordination with each other. LSTM can keep or forget information according to the requirements in these operations. This proposed work is divided into two stages. The first stage is the experimental part, where appropriate network parameters of the LSTM are designed. In the LSTM design, the sequential input layer works on five hidden layers. By applying GA, an optimal number of hidden neurons is found in each layer. GA searches the optimized hidden layers in the LSTM model. In this model, the hyperbolic tangent

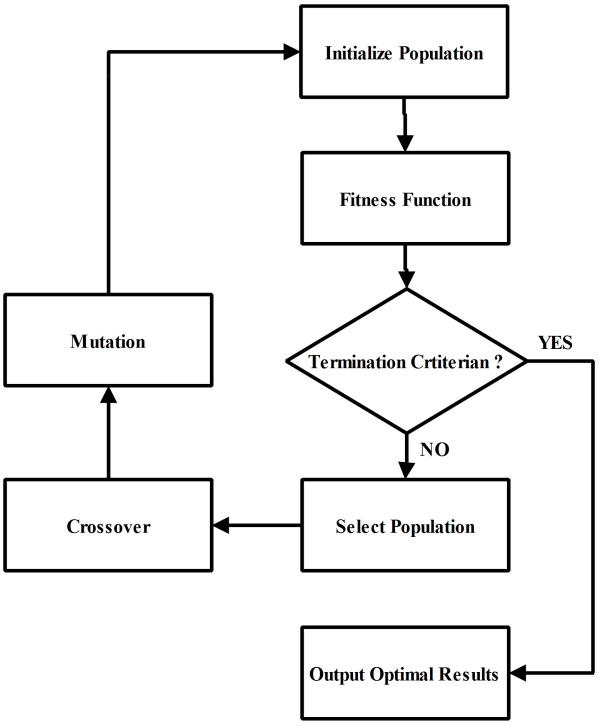



Figure 3.3: The flow chart of Genetic Algorithm

**Algorithm 3.1** Genetic Algorithm with LSTM

1. Initialize the GA parameters
   - cr = 0.9;
   - mr = 0.1;
   - iterations = 20;
   - popSize = 20
2. Initialize LSTM parameters
   - d = 7;
   - dataInterval = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100];
   - nEpochs = [50, 100, 150, 200, 250, 300, 350, 400, 450,500];
   - batchSize = [8, 16, 32, 64];
   - nHiddenLayer= [2,3,4,5];
   - dropoutRate = [0.1, 0.2, 0.3, 0.4];
   - nUnits = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100];
   - nActivationFunction = ['relu', 'sigmoid', 'tanh']
3. t = 1
4. InitPop[P(t)] ; Initializes the population
5. EvalPop[P(t)] = LSTM (chromosome); Evaluates the population

**while** stopping condition **do**
  Crossover()
  Mutation()
  MemoriseGlobalBest()
**end while**

6. Return the individual with the best fitness as the solution;

**Algorithm 3.2** Algorithm 2: Random approach with LSTM

1. Parameter initialization
   - iterations = 20;
   - bestchromosome = []
   - bestAccuracy = 0
2. Initialize LSTM parameters
   - d = 7;
   - datainterval = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100];
   - nEpochs=[50, 100, 150, 200, 250, 300, 350, 400, 450, 500];
   - batchSize = [8, 16, 32, 64];
   - nHiddenLayer = [2, 3, 4, 5]
   - dropoutRate=[0.1, 0.2, 0.3, 0.4];
   - nUnits = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100];
   - nActivationFunction = ['Relu', 'sigmoid', 'tanh'];
3. t = 1
4. While t $\leq$ iterations;

chromosome = Generate random set of LSTM parameters;
Evaluate accuracy = LSTM (chromosome)
if accuracy > bestAccuracy;
bestAccuracy = accuracy
bestChromosome = chromosome
t = t+1

5. Return the bestAccuracy and bestChromosome as solutions.

function is used for input nodes and hidden nodes. The range of *tanh* is (-1 to 1). The activation function of the output node is designed as a non-linear function that works with the regression method. The objective of this model is to predict energy consumption for the next year. The initial weight of the network sets the random values.

GA is combined with the LSTM model in the second stage, where the fitness function is the main feature. GA is the evolutionary algorithm for selecting the initial population based on the fitness function. At the initial stage, the population is generated randomly. After reproduction, the best pairs of fitness scores are selected. The experimental results depend on fitness scores. Here, seven dimensions in one chromosome sample are created. Performance is measured through benchmark and GA-LSTM. This approach has an advantage in predicting energy consumption with large datasets. The experimental result is compared with Mean Absolute Error (MAE), Mean Square Error (MSE), Median Absolute Error (MDAE), correlation, coefficient of determination, and accuracy. GA-LSTM provides the optimal solution for large-dimension data.

Here, chromosomes are represented by strings of arrays, and to obtain fitness value, the

MSE of the prediction model is used. The detailed algorithm is mentioned in Algorithm 1. This algorithm describes the use of GA to optimize the LSTM parameters. It uses crossover, mutation, and selection of the best chromosome that gives the best accuracy as fitness value. In step 1, GA parameters are initialized, and in step 2, LSTM parameters are initialized. Similarly, Algorithm 2 describes the random approach for LSTM parameters optimization. The fitness function ($F$) is defined as per the below equation.

$$F = min(MSE(LSTM(x))) \tag{3.15}$$

where, $x$ is a vector of parameter and the sample chromosomes is like $x$ = [3, 30, 200, 32, Relu, 0.1, 30] which can be verified from Table 3.4. It returns the MSE between the actual and predicted values of the testing dataset.

### 3.3.5 Multi-threading in GA-LSTM

Multithreading uses the CPU cache, translation look-aside buffer (TLB) cache, and single core or multiple cores to carry out various tasks concurrently. It is a process in which the CPU provides multiple threads simultaneously to execute a task in less time. The CPU cache reduces the average data access time from the main memory, while TLB reduces the average time for memory allocation in the main memory. In GA-LSTM, data is loaded, and the model is trained after that. These processes go step by step, and the user needs to wait for their execution. But through multithreading, these tasks can be performed in parallel by running a number of threads that get queued and operate at high speed without getting blocked.

There are many benefits of multithreading. Firstly, it eliminates the multiple processor subsystem and the hardware. Secondly, a single server can perform a number of tasks simultaneously by dispatching multiple threads at a time. This reduces the number of servers required while loading large data. Thirdly, the applications run one after the other and wait for the former to get over. The latter applications don't get blocked. Instead of that, they get queued and increase responsiveness to the operation.

Finally, the memory allocated to processes is relatively high if multithreading is not used.

## 3.4 Results

PJM dataset from 2002 to 2018 has been used to validate the results, wherein the dataset from 2002 to 2015 has been taken for training, and the dataset from 2015 to 2018 is taken

Table 3.4: LSTM hyperparameters

| SN | Parameters | Values |
|---|---|---|
| 1 | Number of hidden layers | [2,3,4,5] |
| 2 | Data interval size | [10,20,30,40,50,60,70,80,90,100] |
| 3 | Epochs | [50 to 500 with an interval of 50] |
| 4 | Batch size | [8,16,32,64] |
| 5 | Activation Function | [Tanh, Sigmoid, Relu] |
| 6 | Dropout rate | [0.1,0.2,0.3,0.4,0.5] |
| 7 | Number of units | [10,20,30,40,50,60,70,80,90,100] |

for testing. GA-LSTM model is trained and tested, and the validation of the proposed work is analyzed. In the initial stage, the number of LSTM units is formed into vectors of hidden layers, epochs, batch size, interval size, and activation functions. In the proposed work, the parameters of LSTM are optimized and verified for the effectiveness of the GA-LSTM model. The performance of the GA-LSTM network is measured using MAE, MSE, MDAE, correlation, coefficient of determination, and accuracy. A comparison of actual and predicted results was made, and it was found that error is reduced using the proposed model. For validation of the proposed work, three types of datasets have been used hourly, daily, and weekly, whose sample data is mentioned in Section III. Since the hourly-based dataset is very complex and not suitable for the GA-LSTM model, we have used daily and weekly-based datasets in our work.

### 3.4.1 Experimental setup and simulation parameters

The proposed algorithm uses Xeon Processor with 64 GB RAM (20 cores) and a 1TB SSD. To increase the speed of the simulation, a multi-threaded GA-LSTM algorithm is used. LSTM parameters have been shown in Table 3.4. A maximum of 5 hidden layers are used for better validation of the results. It is seen from this table that as the data interval size increases, an epoch is also increased.

Further, three types of activation functions, tanh, sigmoid, and Relu, are used. The purpose of using three types of activation functions is to verify the proposed methodology for large datasets. Further, these activation functions will give better choices while making crossover and mutation in GA. It can be seen in the table that the dropout rate varies between 0.1 to 0.5, and the number of units is taken between 10 to 100 at an interval of 10.

GA parameters such as crossover, mutation, population size, and the number of iterations are mentioned in Table 3.5.

Table 3.5: GA parameters

| SN | Parameters | Values |
|---|---|---|
| 1 | Crossover rate | 0.9 (Single point crossover) |
| 2 | Mutation Rate | 0.1 (Single point mutation) |
| 3 | Population size | 100 |
| 4 | Iteration | 20 |

Table 3.6: Optimized parameters of LSTM and Random for Daily and Weekly energy consumption prediction using GA

| | | LSTM | | Random | |
|---|---|---|---|---|---|
| **SN** | **Parameters** | **Daily Energy Prediction** | **Weekly Energy Prediction** | **Daily Energy Prediction** | **Weekly Energy Prediction** |
| 1 | Number of hidden layers | 4 | 3 | 3 | 2 |
| 2 | Data interval size | 60 | 60 | 50 | 40 |
| 3 | Epochs | 450 | 450 | 400 | 450 |
| 4 | Batch size | 16 | 16 | 8 | 16 |
| 5 | Activation Function | Relu | Relu | Relu | Relu |
| 6 | Dropout rate | 0.3 | 0.2 | 0.3 | 0.2 |
| 7 | Number of units | 60 | 40 | 50 | 50 |

It is seen from the table that the single-point crossover is used. Another parameter is a mutation, where the rate at a single point is 0.1. The size of the initial population is 100. Further, the maximum number of iterations is taken as 20. The selection criteria used is the roulette wheel.

After optimizing the parameters of LSTM, the best parameters are found, which are mentioned in Table 3.6.

This table provides the best parameters for daily and weekly energy prediction, and it can be observed that the Relu activation function provides the best performance.

Similarly, it can be seen that the optimized batch size is 16 for both daily and weekly energy prediction. Epochs are found to be 450 for both cases. The optimized data interval size is 60 for both cases. The details of other parameters of LSTM and Random are mentioned in Table 3.6.

Table 3.7 shows the experimental values of results with GA and Random approaches. The performance metrics have been calculated for the daily and weekly datasets with respect

Table 3.7: Performance and evaluation parameters for a daily and weekly dataset for Random and GA

| | **Daily Energy Consumption** | | **Weekly Energy Consumption** | |
|---|---|---|---|---|
| **Evaluation Parameters** | **GA** | **Random** | **GA** | **Random** |
| Mean Absolute Error | $5.34 \times 10^2$ | $1.27 \times 10^3$ | $1.35 \times 10^3$ | $2.23 \times 10^4$ |
| Mean Squared Error | $1.27 \times 10^3$ | $7.66 \times 10^4$ | $2.90 \times 10^4$ | $6.45 \times 10^6$ |
| Median Absolute Error | $7.46 \times 10^1$ | $9.80 \times 10^2$ | $4.20 \times 10^2$ | $8.31 \times 10^3$ |
| Correlation | 0.931 | 0.551 | 0.892 | 0.496 |
| $R^2$ | 0.868 | 0.304 | 0.792 | 0.246 |
| Accuracy | $82.42^*$ | $51.26^*$ | $80.27^@$ | $48.22^@$ |

***with acceptable error of $10^3$; @ with acceptable error of $10^4$.**

to MAE, MSE, MDAE, correlation, coefficient of determination ($R^2$), and accuracy. It can be seen from this table that the accuracy of GA is 82.42 as compared to the random approach, which is 51.26 for the daily dataset. Similarly, for the weekly dataset, the accuracy of GA is 80.27, and the random approach is 48.22.

Further, it can be observed that correlation and $R^2$ are better in GA as compared to the random approach. The other evaluation parameters, such as MAE, MSE, and MDAE are also shown, and it gives the best performance for the daily and weekly dataset in GA compared to the random approach. The acceptable error is mentioned as $10^3$ and $10^4$ for the daily and weekly datasets, respectively.

### 3.4.2 Energy predication on a daily dataset

GA-LSTM prediction can be used for large datasets where GA is used to optimize the parameters of the LSTM. Figure 3.4 shows the energy consumption of actual versus predicted daily energy. The daily energy curve is given in MWh since PJM covers a larger area of the USA. It is seen that the predicted energy of PJM is very close to the actual energy. This prediction will help to schedule the generating units of PJM. LSTM uses 70% of the data for training and 30% for testing. This property of LSTM reduces the testing data. With the GA approach, the daily dataset's energy consumption prediction gives very few errors.

Figure 3.5 shows the convergence of daily energy prediction by using the random approach and GA-LSTM approach. The convergence refers to the different systems moving towards performing the same task. The random sets approach is heuristic by nature; hence, it is

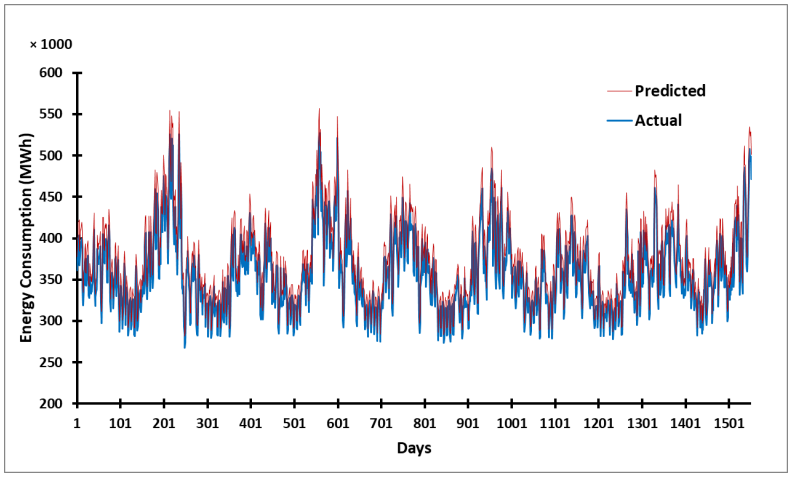



Figure 3.4: Actual v/s prediction of daily energy consumption prediction

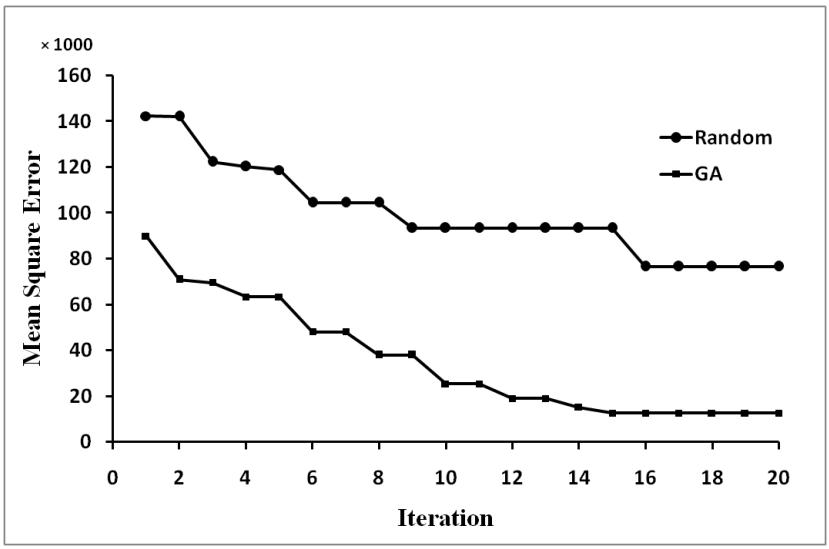



Figure 3.5: Convergence of daily energy consumption prediction

very helpful in discovering things themselves. This graph shows that random approach convergence takes larger iterations while the GA approach takes 15 iterations to converge. The convergence graph is taken with MSE and shows a lower value than the random approach. This proves that GA provides the optimized result with a fewer number of iterations, and it converges at low iterations.

Figure 3.6 shows parameter sensitivity for different data interval sizes for daily energy consumption prediction. Parameter sensitivity analysis shows uncertainty in the output of a model. It can be used to validate sources of uncertainty in the model input. Further, this is a method for finding or establishing the response of a model, which changes when parameters are varied. This graph shows MSE versus data interval size. It is observed from the figure that MSE is low for data interval of size 60. Similarly, other optimized

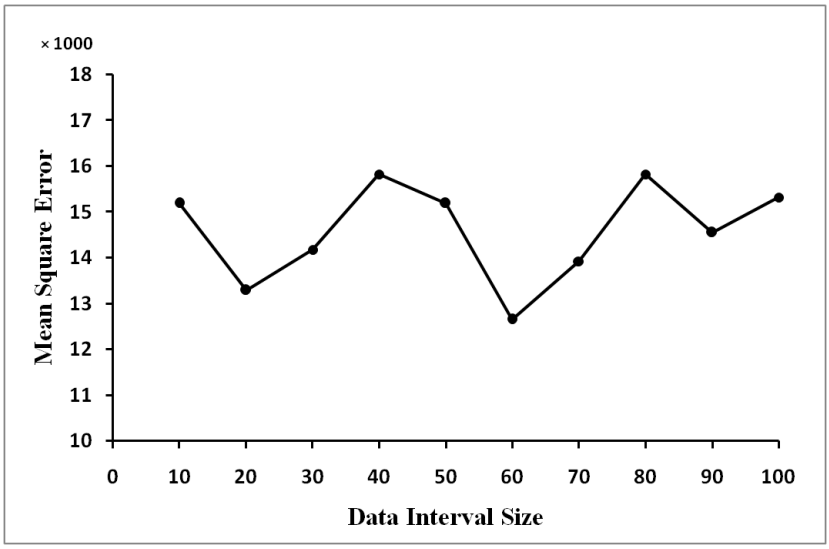



Figure 3.6: Parameter sensitivity for data interval size of daily energy consumption prediction

parameters can be seen from Table 3.6.

### 3.4.3 Energy prediction on weekly dataset

Figure 3.7 represents weekly actual energy consumption versus predicted energy consumption. The utility mainly makes weekly predictions for the week ahead scheduling of the generating units, and it is one of the SG's most widely used short-term load forecasting. Since the data is large, it can be seen from this figure that the energy is 1000 MWh. It is also observed that actual versus predicted energy is very close to each other with a tiny error. Further, this energy prediction can maintain the balance between the demand and supply of the PJM. This prediction will save a large amount of money for utility and better utilization of the generating plants.

The validation of the weekly energy forecasting is achieved through a convergence graph. Figure 3.8 represents the convergence of weekly energy prediction with GA and random approach. This convergence graph is shown till the 20th iteration and found that the random approach has a slower convergence rate than GA. The random approach converges at the 15th iteration while there a no certainty of convergence through the random approach. It is also observed that MSE has a higher value of random approach than GA.

Parameter sensitivity is a method for finding or establishing how the responses of a model change when parameters are varied. There is a great role of parameter sensitivity in the optimization problem. Figure 3.9 shows parameter sensitivity with respect to MSE versus data interval size. It has been observed from the figure that MSE has a lower value at a

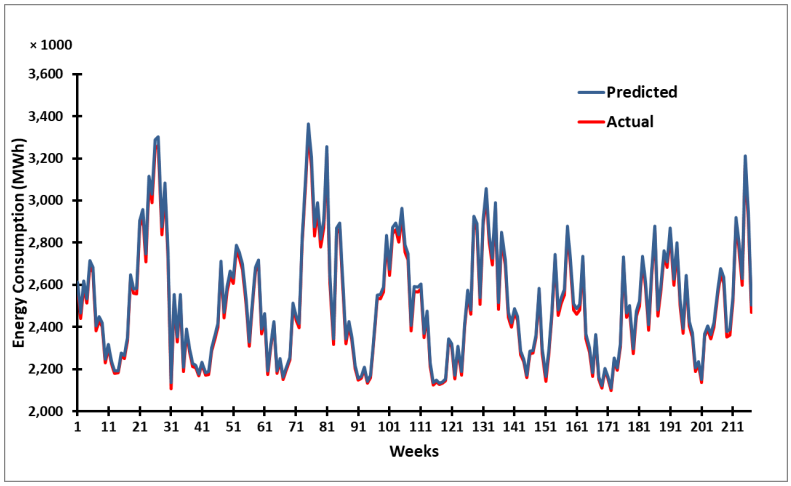



Figure 3.7: Actual v/s prediction of weekly energy consumption prediction

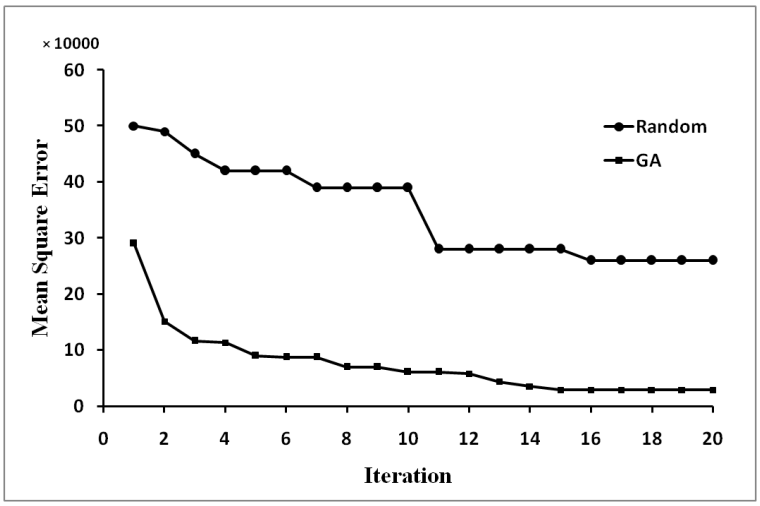



Figure 3.8: Convergence of weekly energy consumption prediction

data interval of size 60. This proves that data interval size optimization is a very accurate method that will give a better convergence at lower iterations. Similarly, other optimized parameters can be seen from Table 3.6.

### 3.4.4 Multithreading

The competitive performance of multiple threads is shown in Table 3.8. Here, the program is run on a machine having 4-cores. The different number of threads are run starting from 1 to 8. As the number of threads increases from 1 to 4, the total execution time decreases. Still, as we increase the number of threads from 5 to 8, the total execution time increases, and this is evident from Figure 3.10. Therefore, the optimal number of threads must be 4 to run the LSTM-GA program in the 4-core machine.

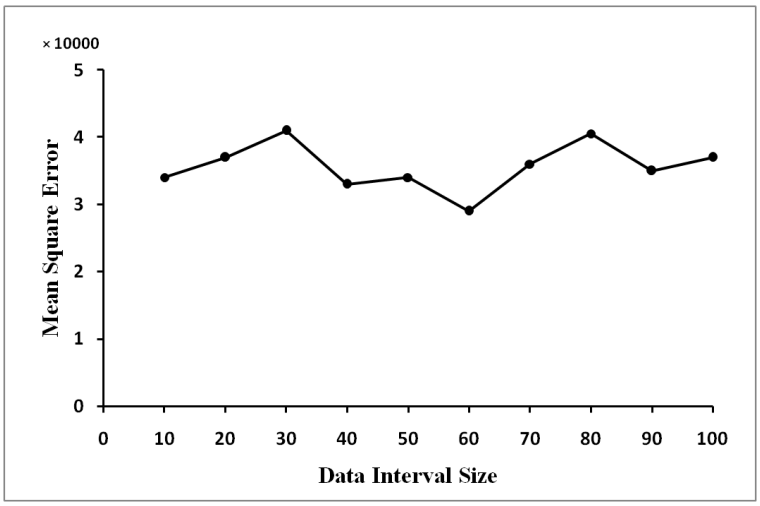



Figure 3.9: Parameter sensitivity for data interval size of weekly energy consumption prediction

Table 3.8: Time taken by GA-LSTM with different number of threads

| Number of threads | Time taken(sec) |
|---|---|
| 1 | 822 |
| 2 | 545 |
| 3 | 476 |
| 4 | 364 |
| 5 | 390 |
| 6 | 490 |
| 7 | 600 |
| 8 | 900 |

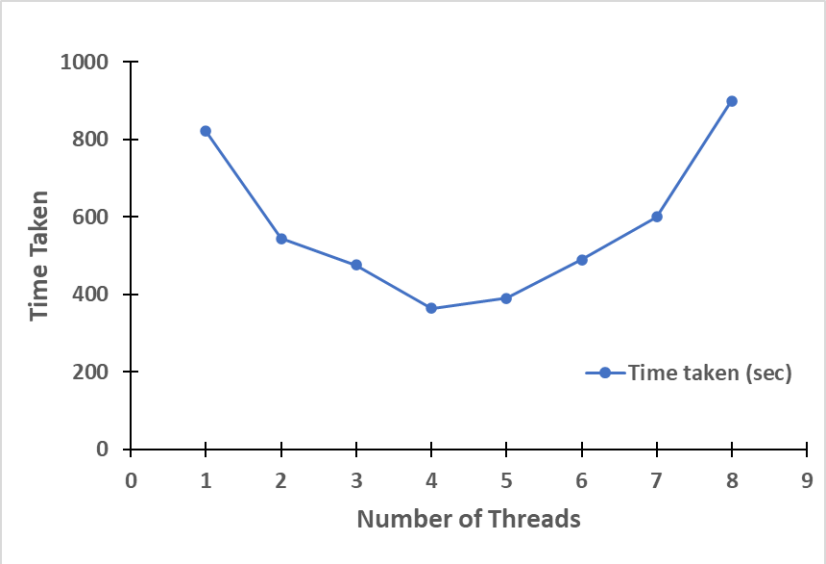



Figure 3.10: Multithreading: Number of threads v/s execution time

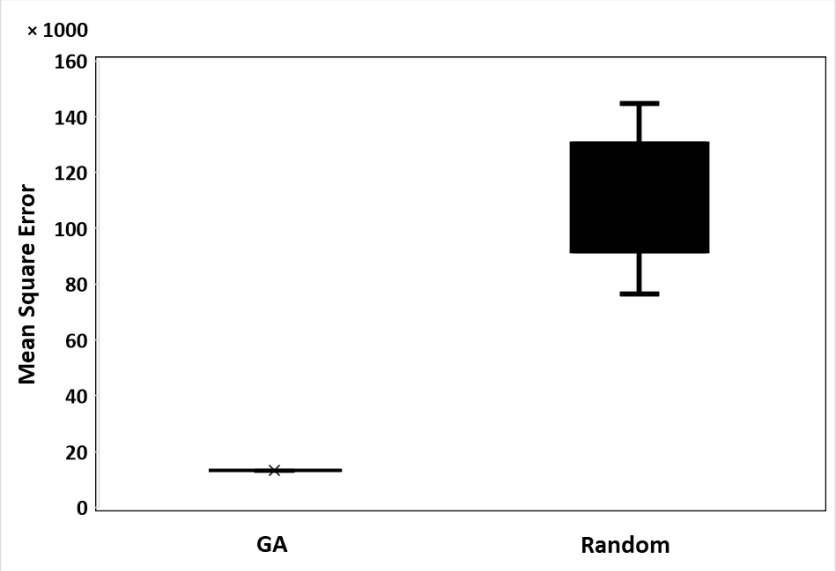



Figure 3.11: Comparison of genetic algorithm and random approach for mean absolute error - Daily basis

### 3.4.5 Variability of MSE with GA and random approach

A Box plot is shown in this subsection to validate the performance of GA and the random approach. Boxplot is a standardized way of displaying the variation of any quantity, which emphasizes the system's stability. Further, we need to have information on the variability or dispersion of the data. A boxplot is a graph that indicates how the data's values are spread out and also identifies outliers. The variation of MSE with a random approach and GA is shown to prove the efficiency of the proposed GA-LSTM algorithm. A total of 10 simulations are performed for daily and weekly energy consumption prediction. Figure 3.11 and Figure 3.12 present the box-plot for energy consumption for daily and weekly energy consumption prediction. Here, the variation of MSE in the random approach is more than GA. It is found that GA variation is significantly less than the random approach, which proves the stability of MSE. Since there is less variation of MSE for the energy prediction using GA, it outperforms the random approach.

## 3.5 Conclusion

This chapter introduces the prediction of energy consumption on a large real-time dataset obtained from PJM. It uses big data analytics using machine learning to predict the energy consumption for a large dataset. Two models, the random and GA approach, are applied to achieve this. On comparing the performance of both models, it was found that GA-LSTM outperforms LSTM. Further, multi-threaded GA-LSTM is used to increase the

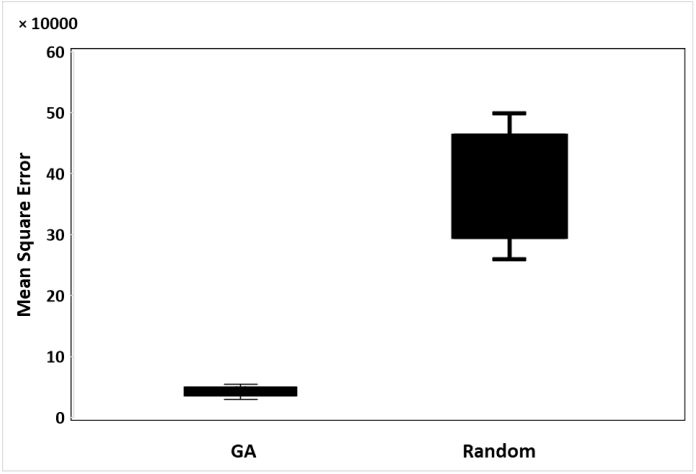



Figure 3.12: Comparison of genetic algorithm and random approach for mean absolute error - Weekly basis

speed of convergence. It has been observed that GA has higher accuracy than the random approach. The comparison is conducted experimentally for real datasets of PJM for daily and weekly energy. It has been proved that the GA-LSTM model provides optimized effective performance. The novelty lies in the multi-threaded-based GA-LSTM technique used to improve the algorithm's performance with a low execution time.

Further, after identifying the lower and upper bound of the LSTM parameter, GA is used to optimize LSTM for better performance. The results of the proposed work are verified with the variability of MSE, and found that the proposed algorithm passes all the evaluation parameter checks.

# Chapter 4

# Different data Filtering Techniques for Big Data

Smart Grid (SG) has smart instruments which can communicate using Advanced Metering Infrastructure (AMI). This will require SG a large storage space for storing the time-series data. The veracity of data increases with respect to time and becomes a challenging task for data managers. For managing large data sets, preprocessing is a fundamental part of the data management system, and often it is called Intelligent Compression (IC). IC is a method for removing redundant or repeated data from a disk or any other storage device. In this paper, various filtering techniques are used to compress and preprocess large data set of power consumption in SG. Filtering eliminates abnormal or unwanted components, signals, or features from a data set. Here, five filtering techniques have been used as Butterworth, Smoothing, Kalman, Frequency swept, and Filtfilt with Long Short Term Memory (LSTM) to predict Power Consumption in the Smart Grid. The results are compared with different evaluation metrics for five different datasets. It is found that Filtfilt filtering techniques with LSTM provide better performance and accuracy over other filtering techniques.

## 4.1 Introduction

A smart grid (SG) is an electric power grid that uses an automatic control system and information technology. The primary aim of the SG is to provide the optimal amount of information. The secondary ambition of the SG is to control energy for customers, distributors, and grid operators. It can manage demand and supply by providing efficient energy to the customers. In this system, energy generation, distribution, and consumption are possible. There are various data and energy generation resources such as sensors, smart meters, a Phaser Measurement Unit (PMU), and home appliances. The supervisory control and data acquisition (SCADA) is a high-level management resource of energy generation. These devices generate huge data, which is called big data. Big data analytics are used to improve the performance of data and energy management systems. Therefore, demand and supply can be managed in the SG. The compression method is

capable of reducing and smoothing large data. In the experimental part, various filtering techniques are used for the compression of data. By using these approaches, accuracy can be improved efficiently. Further, we can remove the redundant or ambiguous data from the system.

The data collected is extensive in scope and has a high variance. Data collected from the various nodes in the grid is primarily similarly structured. Still, when grouped with external sources such as meteorological information and geographical data, only one or two attributes are common to all the datasets. Therefore, data pre-processing and integration are necessary and integral steps in the analysis process. The life cycle and management of big data are highlighted in this paper and provide different big data challenges [93].

For managing these data,many authors highlighted their work regarding technical achievement. They explained the Long Short Term Memory (LSTM) model, which was recently used to handle time-series data. It is used in multiple applications to predict finance and speech recognition. Many authors removed the abnormal condition of previous research by using the clustering and predicting method. Further, LSTM techniques are applied to multidimensional time series where authors implemented aircraft data. Here, Spearman's rank correlation coefficient approach is used for LSTM prediction [94]. In another research paper, the authors tried to analyze the short-term cost by using Recurrent Neural Network (RNN) [95]. In a very similar work, authors tried to analyze forecasting of consumption energy using Support Vector Machine (SVM) [96]. Here, the authors tried to address various green challenges of big data [97] and its applications to energy forecasting.

Some authors explained short-term load forecasting where they have used a smoothing technique [98]. There are different smoothing techniques such as simple exponential, random walk, and moving average. In many research works, authors used smoothed moving averages where past observations are weighted similarly. In some papers, data preprocessing is applied to improve the system's performance. Gorriz *et al.* discussed the preprocessing with the time series data using Savitzky- Golay filtering technique [99].

Moreover, the Butterworth filter is a digital signal processing filter. Stephen Butterworth introduced it. In this technique, deionising is eliminated to find better signal information. The limitation of the Butterworth filter is short out by using the matched wavelet filter. It is applied to digital data points for the smoothness of the data [100]. Here, the authors tried to improve the performance of the system. It can increase the quality of the data without disturbing the signal tendency.

In another filter, the Ensemble Kalman filter (EnKF), a group of items is viewed as a

whole. The EnKF is used to consider the level of the system underlying hypotheses of the Kalman filter [101]. Moreover, the authors tried to quantify the imbalance in the initial conditions. They have also found out the magnitude of the model-error component. This study provides an idea for further development to the EnKF for the measurements.

In a similar paper, the authors represented an algorithm that is used for the series of measurements observed over a time of period [102]. It can measure noise, unwanted disturbances, and other inaccuracies in the system. It produces estimates of unknown variables. Filtfilt filtering technique is another method that runs the filter forward and backward in the time across data which produces a zero-phase response. It is usually non-linear-phase and adds some delay to the signal [103]. Johansson *et al.* highlighted the frequency swept signal in some sources where the term chirp is used. The exponential smoothing technique is another one that is used for records forecasting qualitative information in the time series [104]. It is different from other forecasting techniques. It measures maximum and minimum weights to the most recent and old information. The forecasting with the exponential smoothing technique is more reliable, providing a quick response. Therefore, this approach has a significant role in the applications.

### 4.1.1 Related work

Energy management is a necessary part of the SG system. Therefore, data preprocessing and compression methods can play a significant role in analytics. The dataset is hourly-based time-series data, which can be a large dataset. For handling big data, these compression techniques can be very effective. Many authors discussed the preprocessing of time series data with different filtering techniques. Gorrize *et al.* highlighted the Savitzky-Golay filter where two parameters are discussed about the window size, and the degree of their polynomial [99]. The window size parameter shows how many data points will be used to fit a polynomial regression function. The second parameter works over the degree of the fitted polynomial function. In every window, a new polynomial is fitted, which provides the smoothness effect in the input dataset.

Butterworth filtering approach can adjust the component values. Here, other applications can be removed by using this matched wavelet filter. Further, it can improve signal filtering performance by using a Butterworth wavelet filter. These wavelets are mathematical functions that divide the data into different frequency components. It overcomes the limitations of the Butterworth method. The Butterworth is a signal processing filter which was discovered by Stephen Butterworth. Butterworth can solve the equations for two and four-pole filters. It is an algorithm that measures the time series data. It can measure noise, unwanted disturbances, and other inaccuracies and abnormalities in the system. In

one such paper, the authors discussed the Butterworth model where a signal is created by denoising in the dataset [100].

The smoothing technique is very cost-effective because it provides risk management mitigation. Here, various methods have been discussed for forecasting. All techniques are based on time-series data. The main aim of this technique is to design a load forecasting approach. Therefore, the electricity market would get highly accurate results. Many authors are involved in developing various techniques related to energy forecasting. Patel *et al.* discussed forecasting models where they tried to make an efficient energy forecast model [98].

In a similar paper, authors highlighted energy forecasting using a simple moving average and weighted moving average method. Here, the moving average model is used to predict the storage process. It is used to forecast the value, which is used as a simple combination of an average of current actual values in the time series dataset [104]. They used this technique because of real-time application, and huge data is preprocessed.

Gustofsson *et al.* defined Filtfilt filtering, which is the most important technique and works as a forward and backward in time across the data. It produces a zero-phase response [105]. Here, the authors selected the starting stage for filters for the forwards and backward situations of the signals. The main aim of this paper is to find the same outcomes. Moreover, zero starting stage can show unexpected results. It can be lasting a very short time at the starting or ending points. There were some limitations, such as partial fraction problems. Therefore, the authors proposed to overcome these limitations in stable and unstable conditions. They tried to remove transient's problems. In a nutshell, in this paper, the authors improved accuracy by using various evaluation matrices.

Many authors have worked on Kalman Filter, a filtering technique that estimates accurate measurements. Due to redundant information added in the signal, denoising is created. Therefore, it is needed to remove this redundancy from the signal to get accurate information. The Ensemble Kalman filter is a very effective algorithm. Here, the authors proposed a powerful algorithm based on meteorology and oceanography. These algorithms provided better performance than the standard Kalman filter (KF). The outcomes are better as compared to the sigma point Kalman filters. In this state, nonlinear issues are compared. Generally, in this technique, M. Roth *et al.* elaborated high-dimensionality of the filter [101]. They defined new ideas for effective performance.

The swept-frequency signal is also a very effective filter. Here, the authors elaborated on frequency response. They worked over full-waveform inversion (FWI), which uses the high-resolution velocity models. It can minimize the differences between actual versus

predicted waveforms [106]. The Frequency swept signal approach measures the source calibration. Mainly, calibration is a documented comparison of the measurement devices compared against a traceable reference standard or device. The reference standard should be more accurate than the device to be calibrated. The frequencies are delivered as 1.5 Hz. The inversion of a mixed substances dataset provides no better results than this technology. A detailed comparison of the related research is outlined in Table 1.

Table 4.1: Literature survey for the prediction of energy consumption

| SN | Authors/Ref | Techniques | Description | Application/domain |
|---|---|---|---|---|
| 1 | Usman *et al.* [2019] [95] | Recurrent Neural Network (RNN) | RNN is used for load forecasting in SG. | Load Forecasting |
| 2 | Sultana *et al.* [2019] [96] | Support Vector Regression (SVR) | SVR is used in data analytics for load and price forecasting via enhanced support vector regression technical report for MSCS | Load and price forecasting |
| 3 | Zhang *et al.* [2019] [107] | Data Integrity Verification Scheme (DIVS) | It is used for data verification in SG. | cloud storage |
| 4 | Roopa *et al.* [2017] [108] | Savitzky-Golay | This technique is used in evaluation for denoising ST segment and smoothing data | Remove denoising |
| 5 | Wu *et al.* [2016] [97] | Greening big data | Big data meet green challenges: Greening big data | Green challenges of big data |
| 6 | Hameed *et al.* [2015] [104] | Smoothing technique | Smoothing techniques is used for Time Series Forecasting | Smoothing data |
| 7 | Patel *et al.* [2013] [98] | Short term forecasting | Short term load forecasting by using time series analysis | Smoothing dataset |
| 8 | Plessi*et al.* [2012] [106] | Wave form inversion | Full waveform inversion and distance separated simultaneous sweeping | Smoothing data |
| 9 | Patel *et al.* [2009] [109] | Nonlinear filtering | Nonlinear system identification using exponential swept-sine signal | Smoothing data |
| 10 | Gorriza *et al.* [2004] [99] | ICA and Savitzky | Preprocessing time series with ICA and Savitzky-golay | Preprocessing time series |

### 4.1.2 Motivation

Nowadays, everyone likes digital libraries and e-commerce brokers that provide excellent services. Due to the growth of large data, the consistency of data is affected by the existence of duplicates in a data repository. The abnormal data creates excess memory usage and high execution time. Therefore, many organizations want to develop some

techniques to reduce these problems. The filtering technique has a vital role in separating the redundant data from the repository system. They used these techniques for the real-time application of the preprocessing of large data.

Moreover, they reconstructed the original signal by smoothed ICs. They have filtered the matrix and found minor high-frequency variance for the old version. An ANN is a biological system that follows the natural process as a biological neural network (human brain). These all research techniques inspired us to do work in the preprocessing method. Therefore, we applied many filtering techniques with LSTM in data preprocessing to implement our work.

### 4.1.3 Contribution

The filtering techniques ' primary task is handling non-linearity in the input dataset. Various filtering techniques reduce the redundant data for energy consumption prediction. The following are the significant contribution of the work done.

- Five different filtering techniques such as Butterworth, Smoothing, Kalman, Frequency, and Filtfilt have been used to preprocess the five different power consumption datasets.
- Long Short Term Memory (LSTM) model was used on the processed data for the power consumption prediction.
- Results are analyzed with different performance metrics such as Mean Absolute Error (MAE), Mean Square Error (MSE), Median Absolute Error (MDAE), Correlation ($r$), Coefficient of determination ($R^2$) and Accuracy.

### 4.1.4 Organization

In this paper, Section II defines the detailed methodology and workflow of a comparative performance study of different filtering techniques with LSTM. Section III describes other filtering techniques along with LSTM with the performance and different evaluation metrics. Section IV describes the datasets. Section V shows the outline of the results and discussions. Finally, Section VI provides the conclusion.

## 4.2 Methodology

Figure 4.1 shows the flow of the proposed work. The methodology is explained through six steps. In the first step, raw data is collected, time-series power consumption data.

This data is hourly based, which is converted into daily based datasets. Secondly, we preprocessed the raw data by using different filtering techniques. For this purpose, other preprocessing techniques such as Normalization (Data values are measured in a specific range), handling of missing values, and removal of duplicate data are carried out. Data preprocessing is a method of the data mining process where raw data is prepared for the working model. It is one of the essential phases of a dataset that removes various abnormalities such as data duplication, missing data, outliers, and carry out feature selections, and these are explained as follows.

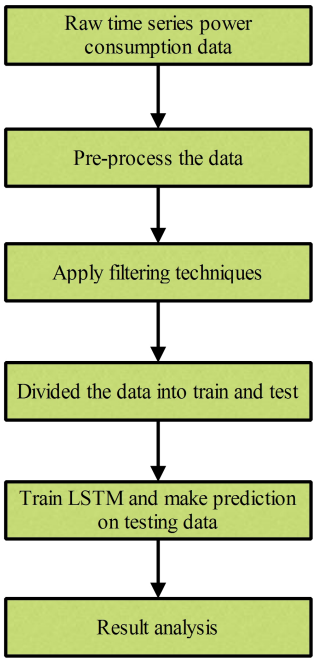



Figure 4.1: Flow of the proposed work

1. **Data Cleaning:** In this part, the irreverent and missing values from the dataset are removed. In missing data, such as ignoring the tuples and filling the missing values are cleaned.

2. **Data Transformation (DT):** Data transformation is a significant phase of data preprocessing. Here, data is transformed into appropriate forms of data. The techniques which are used for DT are explained as follows:

   (a) **Normalization**: Normalization is a technique where data values are measured in a specified range.

   (b) **Removal of duplicate data**: For solving problems, duplicate data are removed.

   (c) **Handling of missing data**: Handling of missing data is the most important

technique for the mining process. In this technique, missing data is analyzed.

Further, various filtering techniques with LSTM such as Butterworth, Smoothing, Kalman, Frequency Swept, and Filtfilt are applied. In the first phase of preprocessing, raw time-series power consumption data are used, which are hourly based. The different dataset has been used, such as American Energy Power (AEP), Commonwealth Edition (COMED), Bright and Sunny Town (DAYTON), Duke Energy Ohio (DEOK), and Dominion Virginia (DOM). After completing this process, filtering techniques are used, which shows better performance. Here, data is divided into two parts. The first is the training phase, where seventy percent of data is used, and the remaining thirty percent is used for testing. We have trained data into various datasets and made predictions on testing using multiple techniques. In the last step, we analyzed results with better accuracy by using multiple filtering techniques with LSTM.

## 4.3 Different filtering techniques

Different filtering techniques such as Butterworth, Smoothing, Kalman, Frequency Swept, and Filtfilt are explained in this section. After the collection of the data, the preprocessing process is carried out. After completing this process, we applied filtering techniques that provide better performance. This dataset is divided into two stages as the training and testing stages. Moreover, we have trained data into LSTM and made predictions on testing data. Finally, results are analyzed to get better accuracy by using various filtering techniques with LSTM. The different filtering techniques which are used are explained below.

### 4.3.1 Butterworth filter

Butterworth filter is a digital filter. Wavelet transform is a linear transform where (except the first function) other functions are measured. Further, shifted versions of one function are called mother wavelets. The discrete signal processing is carried out with the mother wavelet. It is used to overcome the limitation of the Butterworth filter. It combines complex functions and a low residual resolution of the approximated original signal. The original signal is reconstructed from the complex functions and residual approximation in this approach. The following equations are must true of the Fourier spectrum magnitudes of $l$ (low pass) and $h$ (high pass).

$$|l(w)|2 + |h(w)|2 = 1 \tag{4.1}$$

Table 4.2: Description of the filtering techniques

| SN | Filtering Techniques | Description |
|---|---|---|
| 1 | Butterworth [8] | Butterworth filter is a digital filter. It is used for discrete-time series processing. In this filter, Infinite Impulse Response (IIR) is applied for the design of low pass and high pass filters. Weblet filter is used to overcome the limitations of this technique. It helps remove denoising from the signal. |
| 2 | Smoothing [12] | It is a signal processing filter that makes data smooth. It provides a maximally flat magnitude response. |
| 3 | Filtfilt [13] | Filtfilt runs the filter forward and backward in time across the data, which produces a zero-phase response. It usually works as non-linear-phase |
| 4 | Kalman Filtering [10] | It is a type of algorithm which is used for a series of measurements over time. It can measure noise, unwanted disturbances, and other inaccuracies in the system. |
| 5 | frequency swept signal [14] | Here, full-waveform inversion approach are used. Here, authors presented four full-waveform inversion results, which are obtained without offset. They tried to deal with a descended dataset and with the blended dataset. |

For cancellation of any aliasing is guaranteed by setting is follows:

$$h_k = (-h)_k l_{(1-k)} \tag{4.2}$$

The filters, $l$ (low pass) and $h$ (high pass) are related to the mother wavelet, $psi(x)$, and the scaling function, $fi(x)$, by their 2-scale relations.

$$\psi(x) = 2\sum kgk\psi(2x-k) \tag{4.3}$$

$$\psi(x) = 2\sum khk\psi(2x-k) \tag{4.4}$$

or in the frequency domain by

$$\psi(w) = G(w/2)\psi(w/2) \tag{4.5}$$

$$\psi(w) = H(w/2)\psi(w/2) \tag{4.6}$$

where, $G(W/2)$ is detailed function and $H(W/2)$ is residual function of the fourier spectrum magnitude, $psi(w)$ is the Fourier spectrum magnitudes, $l$ is (low pass) filter & $h$ is (high pass) filter and $k$ is a constant.

### 4.3.2 Smoothing Technique

The simple moving average is represented in the following form, where it is a prediction for the next period $t + 1$. The number of periods is the actual values (demand) for the past period, such as two periods ago, three periods ago, etc. In the last step, we analyzed the prediction on testing data.

$$L_{t+1} = \frac{E_t + E_{t+1} + ... + E_{t-n+1}}{n} \tag{4.7}$$

where, $L_{t+1}$ is forecast for the next period $t+1$, $n$ is the number of periods to be averaged, $E$, $E_{t-1}$, +....+ $Et - n + 1$ are actual value for the past periods, two periods ago and so on.

### 4.3.3 Kalman filter

Kalman filter measures the previous time step and the current time step. In this technique, there are two distinct phases defined. The first one is predicted, and another is the updated phase. The prediction phase uses the state measures from the previous time step $t - 1$. It produces a state measurement at the current time step $t$. Further, at time $t$, $z(t)$ is an observation of the true state $MHP(t)$.

$$M\tilde{H}P_{SP}(t+1) = MHP(t)\frac{MHP_{clsk}(t+1)}{MHP_{clsk}(t)} \tag{4.8}$$

Where, $MHP(t)$ is the actual state.

### 4.3.4 Frequency swept signal filtering

This approach measures the source calibration terms that are solutions are gathered every iteration. Further, to avoid the drawback, some relative source scaling factors are calculated.

$$g_s = \sum \beta_t^k w_t \delta(x - x_t^k) \tag{4.9}$$

Where, $f_t$ is the estimation of source calibration, $\beta_t^k$ is the relative source scaling factors, $w_s$ is a wave equation solution gathered per modeling. Further, $x$ is the actual value of the relative source scaling factors, and $x_t^k$ is the predicted value of the relative source scaling factors. Finally, $\delta(x - x_t^k)$ is differences between actual and predicted value.

### 4.3.5 Filtfilt filtering

The bandpass filtering causes delay. Therefore, the bandpass filter is used twice in the opposite directions to prevent this. Here, the function Filtfilt performs these operations. The following equations are used for solving problems.

$$\tilde{z}(t) = \sum_{0}^{M} b_l z_0(t-1), \quad z(t) = \sum_{0}^{M} b_l \tilde{z}(t+1) \tag{4.10}$$

Where, $z(t)$ is the time series equation of phases in the frequency band. Given a time series $z_0(t-1)$, its phase in the frequency bands is measured as an input. At first, a bandpass filter with the passband is selected and applied to $z_0(t+1)$. The bandpass filtered signal $z(t)$ is the one that provides output.

### 4.3.6 Long Short Term Memory (LSTM)

In this paper, LSTM is mainly used for the time series dataset to predict energy consumption. LSTM has a memory that works with feedback connections and remembers previous information inside the network. It has the capability of solving time series and non-linear prediction problems. The major problem of RNN is "Long term dependency"; therefore, LSTM is used to overcome this problem. The cell state is an essential key of the LSTM, and it is used as a conveyor belt. It can add or remove the information, and the structure, called gates, regulate it. Gates are the mode where information is optionally chosen. These gates work with a sigmoid neural network and a point multiplication operation. There are three main types of gates: input gate, output gate, and forget gate. Tanh, $\sigma$ and Relu are the activation functions mainly used in the LSTM network. Figure 4.2 provides the basic structure of LSTM model. There are different equations used in this model and are defined as follows.

The dependent variable of the LSTM is mentioned below.

$$y_{ij}^1 = \sigma(b_j^1 + \sum_{m=1}^{M} w_{m.j}^1 x_i^0 + m - 1, j) \tag{4.11}$$

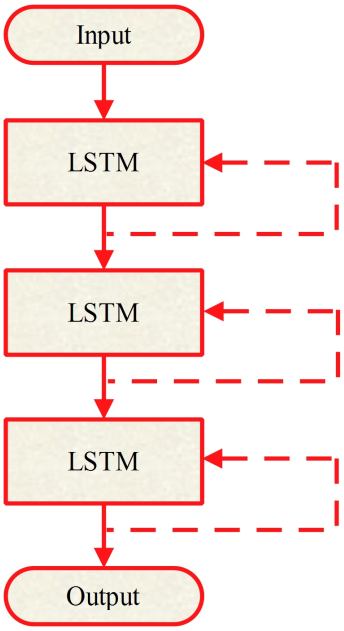



Figure 4.2: Basic structure of the LSTM

In this equation, $y_{ij}^{1}$ is a dependent variable where the weight of each layer is calculated with bias value $b_j^1$. In this equation, the result of the vector is calculated by the output vector of the previous layer. Further, $\sigma$ is an activation function, $X_i^0$ is the independent variable, $W$ is the weight of layers, and $b$ is the biased value associated with input layers, respectively. The output layer is given as follows.

$$y_{ij}^{l} = \sigma(b_j^l + \sum_{m=1}^{M} w_{m.j}^{l} x_i^0 + m - 1, j) \tag{4.12}$$

Where, $y_{ij}^{1}$ is output from the first layer where the pooling layer combines with the output of a neuron cluster in one layer into another single neuron. It is capable of reducing the number of parameters. The maximum pooling layer of LSTM is defined as follows.

$$p_{ij}^{l} = max_{r \in R} y_{ij}^{l} \tag{4.13}$$

Where, $p_{ij}^{l}$ is the maximum pooling layer of LSTM for storing time information. LSTM gives a solution by consolidating memory units to update the hidden state. The input gate is given as per the below equation.

$$i_t = \sigma(W_p iPt + W_h iP_t - 1 + W_c i * C_t - 1 + b_i) \tag{4.14}$$

where, $i_t$ is a input gate and the hidden states determine through the input gate, $i_t$ is input state at a time $t$, $C_t$ is cell state at time $t$ and $W_c i * C_t$ is weighted input. Forget state is given as per the following equations.

$$f_t = \sigma\{W_p f^p t + W_h f_t^h - 1 + W_c f * c_t - 1 + b_f\} \tag{4.15}$$

Here, $f_t$ is the operation of forget gate at time $t$, which consists of LSTM and output of each gate represented by $i$, $j$ and $o$, $W$ is the weight matrix of each gate unit respectively. $P_t$ is the critical features of electric energy consumption as the output of the pooling layer at time $t$ and used as input to the output of the memory cell. $C_t$ is cell state at time $t$, and $b$ is the benchmark of the forget gate. The operation of the output gate and cell state are given below.

$$o_t = f_t * C_t - 1 + i_t * \sigma\{W_p cPt + W_h c_t^h - 1 + b_c\} \tag{4.16}$$

$$c_t = f_t * C_t - 1 + I_t * \sigma(W_p cPt + W_h ch_t - 1 + b_c) \tag{4.17}$$

The feature vector is one of the important parameters of the LSTM. It is mentioned as per the following equation.

$$h_t = o_t * \sigma\{C_t\} \tag{4.18}$$

Where, $h_t$ is a feature vector. The last layer of the feature vector is mentioned as follows.

$$d_i^l = \sum_j w_j i^l - 1(\sigma(h_i^l - 1 + b_i^l - 1) \tag{4.19}$$

Last layer $d_i^l$ is fully connected layers. $\sigma$ is a non-linear activation. $W$ is the weight of the $i^{th}$ node for layer $(l-1)$ and the $j^{th}$ node for layer $j$, and $b_i^{l-1}$. The bias is determined by $j^{th}$ node for layer $l$ and $b_{1_i}^l$. Further, $*$ is element wise multiplication and $+$ is element wise addition.

$$F_t = \sigma(X_t * U_f + H_t - 1 * W_f) \tag{4.20}$$

Where, $F_t$ is forget gate of LSTM, $X_t$ is the input vector, $W_f$ is an input vector, $U_f$ and $W_f$ are the weight vectors for the forget gate and candidate gate, and $H_{t-1}$ is the previous

cell output.

$$\overline{C}_t = tanh(X_t * U_c + H_{t-1} * W_t) \tag{4.21}$$

where, $C_t$ is the current memory state at time step $t$. $tanh$ is an activation function of the candidate layer, $X_t$ is input vector, $U_c$ is weight vector, $H_{t-1}$ is previous output and $W_t$ is weight vector.

$$I_t = \sigma(X_t * U_i + H_{t-1} * W_i) \tag{4.22}$$

In the above equation, $I_t$ is the input gate at time step $t$, $U_i$, and $W_i$ are weight vectors for the input gate and output gate, respectively, and $H_t1$ is the previous step. The below equation provides the information of the output gate as follows.

$$O_t = \sigma(X_t * U_o + H_{t-1}) * W_O) \tag{4.23}$$

where, $O_t$ is an output gate at time step $t$, $X_t$ is an input vector, $U_o$ and $W_o$ are weight vector of output gate and candidate gate. $H_t$ is the previous cell output.

$$C_t = f_t * C_{t-1} + l_t * \overline{C}_t \tag{4.24}$$

where, $C_t$ is the current cell memory and $f_t$ is an forget gate vector. The previous cell output is given as below.

$$H_t = O_t * tanh(C_t) \tag{4.25}$$

In this equation, $H_{t-1}$ is the previous cell output, $C_{t-1}$ is the previous cell memory, $H_t$ is current cell output, $W$ and $U$ is weight vectors for forget gate $f$ respectively, Further, the candidate gate is named as $c$, input gate as $l$ and output gate as $o$. The forget gate is named as $F$, candidate as $(C^{'})$ and Input gate as $I$ and output gate as $O$. The current memory state is given as,

$$C_t = C_t + (I_t * C_t^{'}) \tag{4.26}$$

Where $C_t$ is the current memory state at time step $t$ and it gets passed to the next time step and $I_t$ is the input gate. The current cell output is mentioned below.

Table 4.3: Description of the dataset

| SN | Dataset | Description |
|---|---|---|
| 1 | AEP [110] | American Energy Power (AEP) dataset is hourly based Energy Consumption data. The consumed energy is measured in Megawatts (MW). |
| 2 | COMED [110] | Commonwealth Edition is called COMED. It is hourly based Energy Consumption data. It is largest electric utility in Illinois. The consumption of energy is measured in (MW) |
| 3 | DAYTON [110] | DAYTON is name of company of light and power. It is hourly based Energy Consumption data. It is measured consumed energy in Megawatt (MW). |
| 4 | DEOK [110] | Duke Energy Ohio/Kentucky (DEOK) is hourly based Energy Consumption data. The consumption of energy is measured in Megawatt (MW). |
| 5 | DOM [110] | Dominian Virginia Power (DOM) is hourly based Energy Consumption data. The consumption of energy is measured in Megawatt (MW). |

$$H_t = Tanh(C_t) \tag{4.27}$$

In this equation, $H_t$ is the current cell output at time step of $t$, $tanh(C_t)$ is the activation function used to find the current cell memory. We pass these two $C_t$ and $H_t$ to the next step and repeat the same process. Figure 4.2 shows the different LSTM steps.

## 4.4 Dataset and its description

### 4.4.1 Data Description

The dataset is multivariate time-series data. We have used five datasets such as American Energy Power (AEP), Commonwealth Edition (COMED), DAYTON, Duke Energy Ohio/Kentucky (DEOK), and Dominian Virginia Power (DOM). The description of the dataset is mentioned in Table 3. The detailed period of the dataset is provided in Table 4.

### 4.4.2 Model evaluation parameters

#### 4.4.2.1 Mean Absolute Error (MAE)

MAE measures error between two variables such as $b$ and $s$. The observations are explained about the same event. In the below equation, where $b$ versus $s$ are comparisons

Table 4.4: Duration of the dataset

| SN | Dataset | Duration |
|---|---|---|
| 1 | AEP[110] | 2006 to 2019 |
| 2 | COMED[110] | 2011 to 2018 |
| 3 | DAYTON[110] | 2004 to 2018 |
| 4 | DEOK [110] | 2012 to 2018 |
| 5 | DOM [110] | 2005 to 2018 |

of predicted versus the observed value of variables, it is defined as follows.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|b_i - s_i| \tag{4.28}$$

#### 4.4.2.2 Mean Square Error (MSE)

Mean square error (MSE) is a mean square deviation (MSD) estimator which measures the average of the squares of the errors. Here, the average square provides the difference between the predicted value and the actual value. MSE is given as follows.

$$MSE = \frac{1}{n}\sum_{j=1}^{n}(b_i - q_i)^2 \tag{4.29}$$

where, $q_i$ indicates predicted value and $b_i$ indicates actual value.

#### 4.4.2.3 Median Absolute Error (MDAE)

The median absolute error is robust by nature. Here, we calculate loss by taking the median of all absolute differences between the actual and the predicted value. In the below equation, $q_i$ is the predicted value of the $j^{th}$ sample, and $b_j$ is the corresponding true value. MDAE is estimated over $n$ samples is defined as follows.

$$MDAE(b, q) = median(|b_1 - q_1|, ..., |a_n - q_n|) \tag{4.30}$$

#### 4.4.2.4 Correlation (r)

Correlation describes the relationships of analysis or presentation of masses of mathematical data between actual and predicted values. It is defined as follows:

$$s = \frac{\sum_{j=1}^{n}(b_j - \bar{b})(q_j - \bar{q})}{\sqrt{\sum_{j=1}^{n}(b_j - \bar{b})^2 \sum_{j=1}^{n}(q_i - \bar{q})^2}} \tag{4.31}$$

Where, $b$ is the actual value, $q$ is the predicted value, $\bar{b}$ is the mean of all actual values, $\bar{q}$ is the mean of all predicted values, and $n$ is the number of instances. Correlation lies in [-1,1] and is considered good correlations if its value tends towards 1 or -1.

#### 4.4.2.5 Coefficient of determination ($\mathbf{R}^2$)

The coefficient of determination ($R^2$) explains the ability of the theory of the regression model. It is calculated from the sums-of-squares terms and given as per the below equation.

$$R^2 = r * r \tag{4.32}$$

$R^2$ lies in [0,1] and is considered good if its value tends towards 1.

#### 4.4.2.6 Accuracy (Acc)

The accuracy is calculated as the percentage deviation of predicted with actual values and given as per the following equations.

$$\begin{aligned} Acc &= \tfrac{100}{n} \textstyle\sum_{i=1}^{n} q_i \\ q_i &= \begin{cases} 1 & \text{if } abs(p_i - a_i) \leq 10 \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{4.33}$$

## 4.5 Results analysis and discussion

The performance of comparative results is computed on Xeon Processor with 64 GB RAM (20 core) and 1TB SSD to increase simulation speed. For better validation of the results, various filtering techniques with LSTM have been used. The comparative performance study of the LSTM model with different filtering technology on five datasets is shown in

Table 5. The purpose of using these datasets is to get more accurate results. The data is converted into seventy and thirty percent as training and testing. It is being verified that different datasets can be used with filtering techniques to test the LSTM model.

The derived results are measured by MAE, MDAE, $R^2$, $r$, and accuracy, providing minimum error by comparing different techniques. Moreover, better results are found which performs effectively in these visualizations. This study discusses the comparative performance study of the LSTM model with different filtering technology on different datasets. Table 5 represents various filtering techniques to be fed to the LSTM network. The evaluation matrix MAE provides a minimum value of 4627.94 after comparing other filtering techniques on datasets in the dataset of AEP. So, the LSTM+Filtfilt filtering technique performs minimum error.

Table 4.5: Comparative performance study of LSTM model with different filtering technology on different datasets

| **Dataset** | | **LSTM** | **LSTM+ Butterworth** | **LSTM+ Smoothing** | **LSTM+ Kalmen** | **LSTM+ Frequency** | **LSTM+ Filtfilt** |
|---|---|---|---|---|---|---|---|
| AEP | MSE | 1101884089 | 454737246 | 216569758 | 1123812483 | 136431278 | 30272450 |
| | MAE | 26002.621 | 16697.538 | 11790.755 | 26527.782 | 9503.593 | 4627.9461 |
| | MaAE | 21373.734 | 13999.186 | 9944.215 | 9944.215 | 8528.298 | 4250.8677 |
| | $R^2$ | 0.477 | 0.070 | 0.877 | 0.483 | 0.937 | 0.9886 |
| | $r$ | 0.691 | -0.265 | 0.936 | 0.695 | 0.968 | 0.968 |
| | Accuracy | 66.688 | 69.394 | 83.227 | 70.425 | 94.716 | 99.0000 |
| COMED | MSE | 977737053 | 382216906 | 309135516 | 984191816 | 137320519 | 27968242 |
| | MAE | 24071.011 | 15158.049 | 13245.666 | 24035.648 | 8963.820 | 4042.459 |
| | MaAE | 19021.875 | 12437.810 | 10191.084 | 18417.703 | 6861.670 | 3090.790 |
| | $R^2$ | 0.373 | 0.047 | 0.726 | 0.371 | 0.889 | 0.976 |
| | $r$ | 0.611 | 0.216 | 0.852 | 0.609 | 0.943 | 0.988 |
| | Accuracy | 69.485 | 66.495 | 66.495 | 63.505 | 75.670 | 98.144 |
| DAYTON | MSE | 28449398 | 12480176 | 3279714 | 26986432 | 26986432 | 521798 |
| | MAE | 4254.614 | 2867.800 | 1446.388 | 4138.032 | 1058.411 | 550.411 |
| | MaAE | 3626.457 | 2570.551 | 1229.331 | 3577.477 | 901.683 | 447.424 |
| | $R^2$ | 0.366 | 0.112 | 0.893 | 0.365 | 0.937 | 0.981 |
| | $r$ | 0.605 | 0.334 | 0.945 | 0.604 | 0.968 | 0.991 |
| | Accuracy | 94.587 | 97.715 | 99.000 | 95.821 | 99.000 | 99.000 |
| DEOK | MSE | 67646185 | 23237652 | 32209858 | 67956104 | 11876453 | 4044295 |
| | MAE | 6627.149 | 3712.518 | 4536.105 | 6499.706 | 2704.724 | 1585.477 |
| | MaAE | 5740.930 | 3021.485 | 3789.951 | 5266.609 | 5266.609 | 1247.132 |
| | $R^2$ | 0.427 | 0.003 | 0.652 | 0.428 | 0.865 | 0.957 |
| | $r$ | 0.768 | 0.520 | 0.963 | 0.736 | 0.969 | 0.986 |
| | Accuracy | 90.055 | 97.895 | 94.754 | 85.193 | 98.779 | 99.000 |
| DOM | MSE | 836064493 | 309024164 | 144293967 | 960824703 | 160271552 | 71399558 |
| | MAE | 16812.969 | 9972.383 | 8228.668 | 17261.594 | 7146.005 | 5012.784 |
| | MaAE | 16812.969 | 9972.383 | 8228.668 | 17261.594 | 7146.005 | 5012.784 |
| | $R^2$ | 0.590 | 0.270 | 0.928 | 0.542 | 0.940 | 0.972 |
| | $r$ | 0.768 | 0.520 | 0.963 | 0.736 | 0.969 | 0.986 |
| | Accuracy | 67.114 | 73.676 | 88.231 | 61.670 | 64.802 | 80.164 |

Similarly, in the COMED dataset, the minimum value is 4042.45, which performs minimum error through LSTM+Filtfilt. The DAYTON dataset's minimum value is 550.51, which produced the minimum value through the LSTM+Filthfilth filtering technique. In the DEOK dataset, the minimum value came out to be 1585.47, which provides a minimum error. In the DOM dataset, the minimum value is 5012.78 which gave min-

imum error over all these methods. Therefore, the performance is very effective for the prediction of energy. The main goal of all these comparisons of evaluation matrices is to minimize error and increase accuracy.

Figure 4.3 to Figure 4.12represent visualization of training and testing of the dataset with different filtering techniques of the daily energy consumption. These filtering techniques are such as Butterworth, Smoothing, Kalman, Frequency Swept, and Filtfilt are fed to the LSTM network, which derives results through these techniques. The observations are shown in these figures are with AEP datasets. All the filtering techniques have been tested with AEP datasets. The derived results are measured by Mean Absolute Error (MAE), which compares the different approaches used in the proposed work. Moreover, better results are found which performs effectively in these visualizations.

The Butterworth filter graph uses wavelets that have performed data into different frequency components, which helps to remove denoising from the signal and this can be seen in Figure 4.3 and Figure 4.4 respectively with training set and testing set of Data. LSTM and Filtfilt filtering techniques provide minimum error by comparing the evaluation matrix MAE on the dataset of AEP. The visualisation of the training and testing data sets can be seen in Figure 4.11 and Figure 4.12 respectively. Therefore, various unwanted data are removed from the signal, such as denoising and discontinuities signals. Similarly, the graph shown in LSTM and Filtfilt filtering technique on datasets of COMED provides better results. In this technique, the interference is reduced from the signal, and thus it gives minimum errors.

The graph in the following filtering technique, such as LSTM and Smoothing, LSTM and Swept frequency and LSTM & Saviitzky on AEP datasets are shown in the Figure 4.5 to Figure 4.10 respectively. Here, the smoothing technique performs well and removes the unwanted component from the signal. We found minimum error by comparing with evaluation parameter MAE. The graph in LSTM on Filtfilt on dataset DEOK presented a better result. At last, the filtering technique on LSTM and Smoothing techniques represent graphs on dataset DOM, which is more clean and less interference. Due to less interference, separation among the components of the signals provides better performance.

LSTM with filtering techniques for the daily energy consumptions on actual and predicted for the best performer of LSTM is shown in Figure 13 to Figure 17. These filtering techniques are individually applied on datasets, and results are shown separately. After comparison, it is found that the Filtfilt filtering technique on AEP and COMED provide minimum error as shown in Figure 14. The smoothing filtering technique on the DAYTON dataset has the minimum error. Frequency swept filter techniques provide

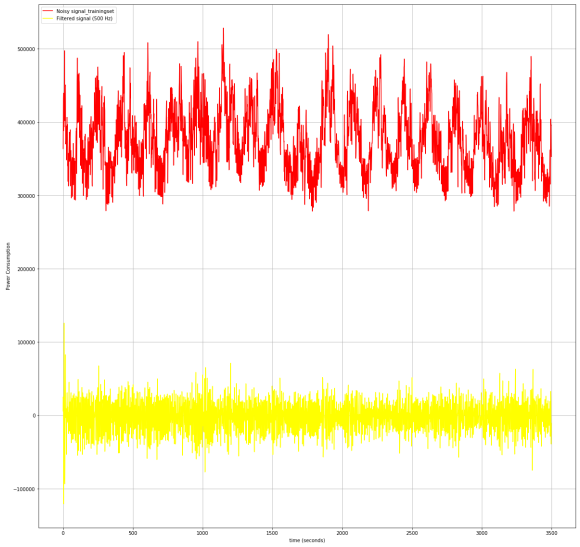



Figure 4.3: Visualization of training data set with Butterworth filtering technique of the daily energy consumption

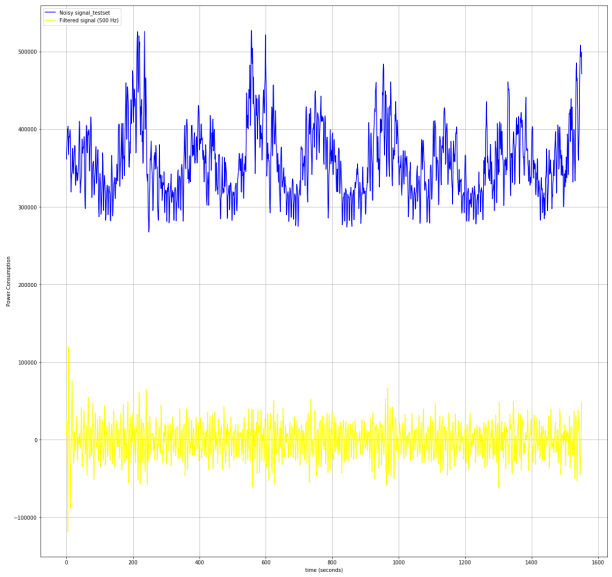



Figure 4.4: Visualization of testing data set with Butterworth filtering technique of the daily energy consumption

minimum error, and Filtfilt filtering revealed minimum error on the DAYTON dataset. In the DEOK dataset, the Filtfilt filtering technique is applied. The Smoothing approach performs minimum error. Further, it is revealed that the best performance of LSTM with filtering technique is for the daily energy consumption prediction on the MAE evaluation.

Table 6 represents the cross-validation of comparing various filtering techniques on many datasets for checking validity. Further, this is explained with Figure 18 where the blue

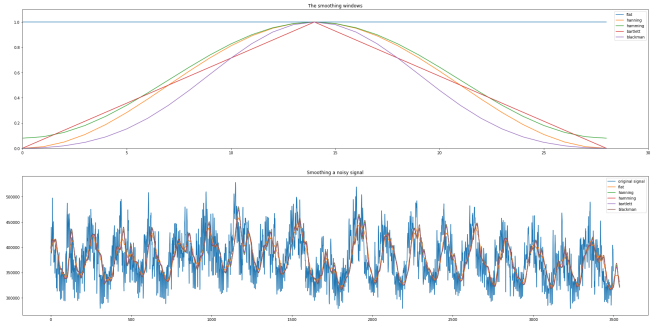


Figure 4.5: Visualization of training data set with smoothing filtering technique of the daily energy consumption

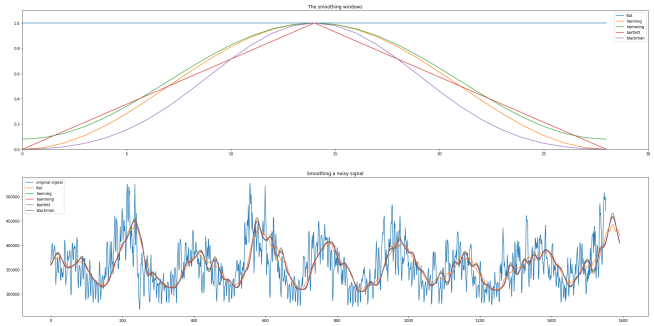


Figure 4.6: Visualization of testing data set with smoothing filtering technique of the daily energy consumption

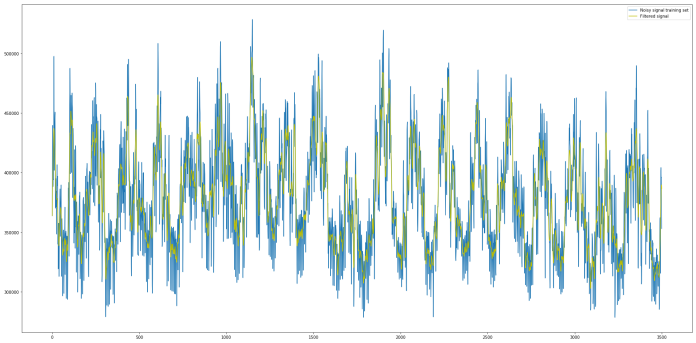


Figure 4.7: Visualization of training data set with Savitzky filtering technique of the daily energy consumption

color indicates the graph of the dataset with LSTM and Filtfilt. Orange color represents the graph of dataset LSTM and Filtfilt. Similarly, we can see the LSTM with other filtering techniques. The comparison is related to the evaluated matrix as mean absolute error. The comparison of MSE verifies the results, and the proposed approach is demonstrated with different evaluation parametric checks. This research work checks the validity with comparative performance with different datasets.

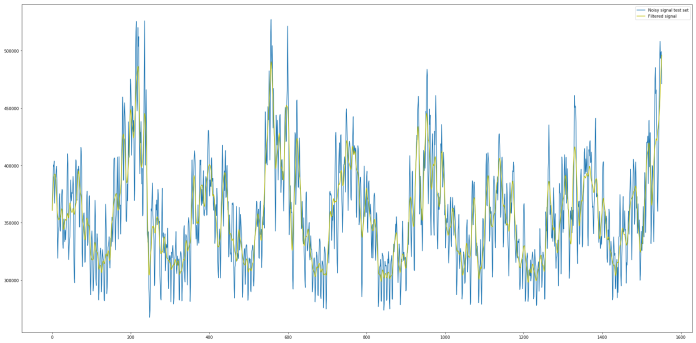


Figure 4.8: Visualization of testing data set with Savitzky filtering technique of the daily energy consumption

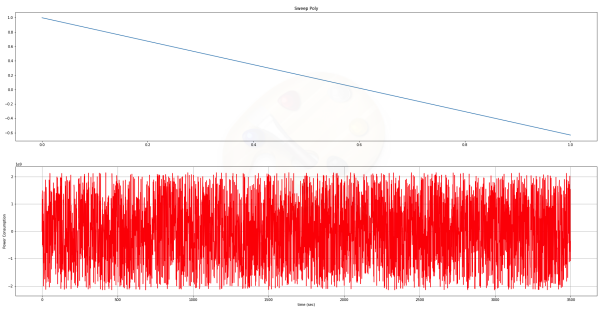


Figure 4.9: Visualization of training data set with frequency swept signal filtering technique of the daily energy consumption

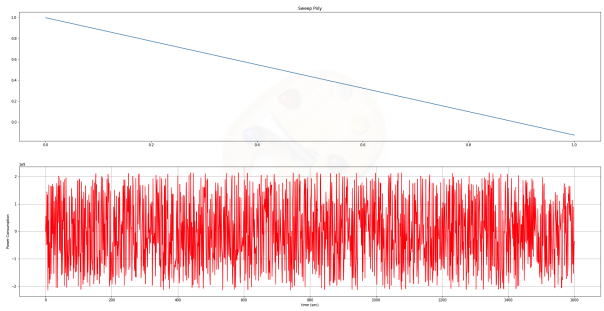


Figure 4.10: Visualization of testing data set with frequency swept signal filtering technique of the daily energy consumption

### 4.5.1 K-fold validation

Figure 18 shows the comparison between various filtering techniques with LSTM on different datasets. Here, the variation of k-fold validation of various filtering techniques with LSTM on the dataset is shown. The Filtfilt filtering technique provides K-fold cross-validation more accurately. It is used in datasets to further split into a K number of sections or folds. Here, each fold is used as a testing set at some point. This paper uses the scenario of 10-Fold cross-validation (K=10). This process is repeated until each fold of the 10 folds has been used as the testing sets. Our dataset is divided into two parts,

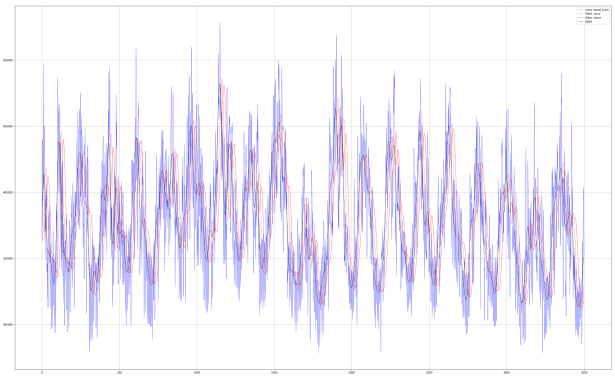


Figure 4.11: Visualization of training data set with Filtfilt filtering technique of the daily energy consumption

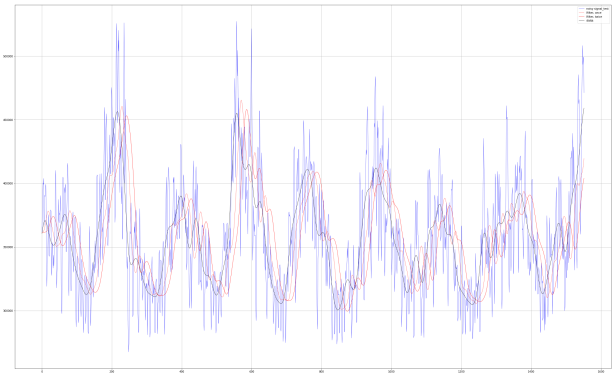


Figure 4.12: Visualization of testing data set with Filtfilt filtering technique of the daily energy consumption

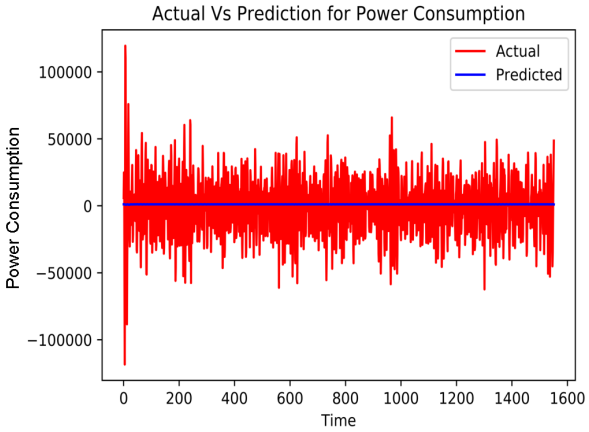



Figure 4.13: Actual vs predicted data of butterworth filter

such as 70 percent for training and the duration of data such as (2006 to 2019, 2011 to 2018, 2004 to 2018, 2012 to 2018, and 2005 to 2018). We have tested 30 percent of data from (2016 to 2019, 2015 to 2018, 2015 to 2018, 2015 to 2018, and 2015 to 2018).

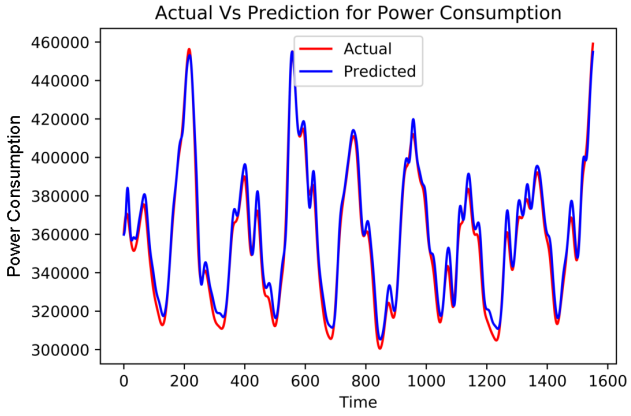



Figure 4.14: Actual vs predicted data of Filtfilt filter

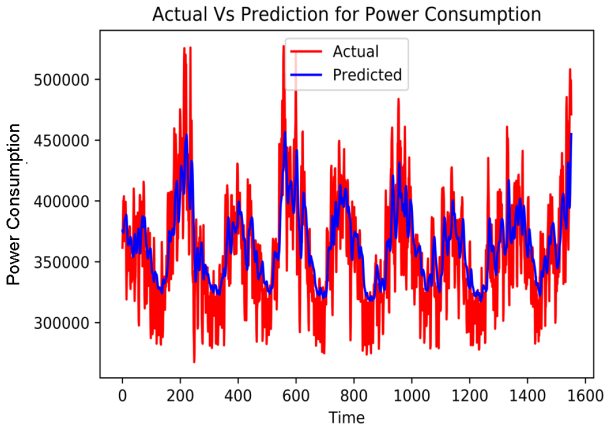



Figure 4.15: Actual vs predicted data of frequency swept signal techniques

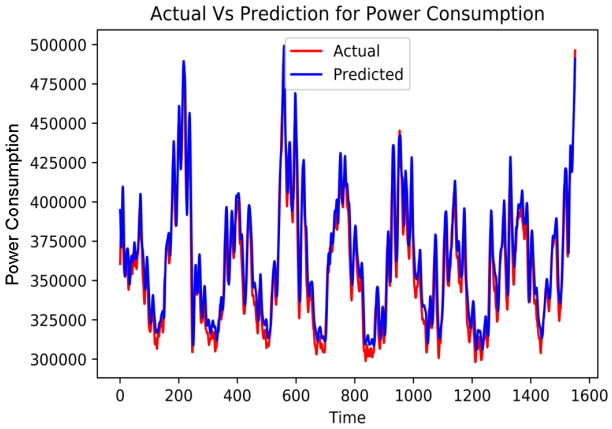



Figure 4.16: Actual vs predicted data of savitzky golay techniques

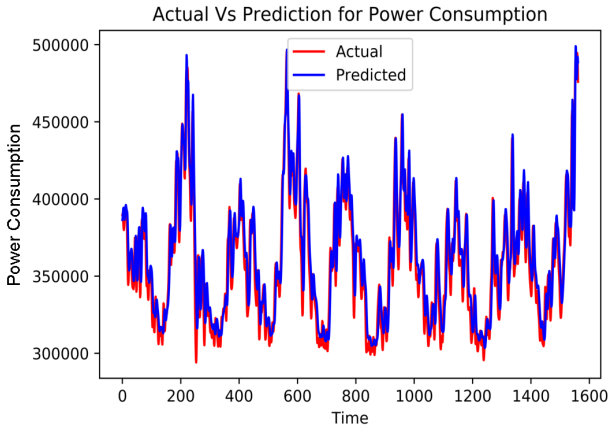



Figure 4.17: Actual vs predicted data of smoothing techniques

Table 4.6: Cross validation for MSE using different filtering techniques

| Run | DS1 | DS2 | DS3 | DS4 | DS5 |
|---|---|---|---|---|---|
| 1 | 4627 | 4042 | 1058 | 1585 | 5012 |
| 2 | 4676 | 4082 | 1098 | 1620 | 5052 |
| 3 | 4588 | 3997 | 1023 | 1545 | 4977 |
| 4 | 4663 | 4090 | 1003 | 1631 | 5042 |
| 5 | 4591 | 3996 | 1018 | 1538 | 4980 |
| 6 | 4662 | 4091 | 1093 | 1628 | 5055 |
| 7 | 4590 | 4002 | 1016 | 1542 | 4982 |
| 8 | 4676 | 4089 | 1088 | 1621 | 5054 |
| 9 | 4589 | 3996 | 1026 | 1547 | 4981 |
| 10 | 4661 | 4087 | 1022 | 1630 | 5044 |

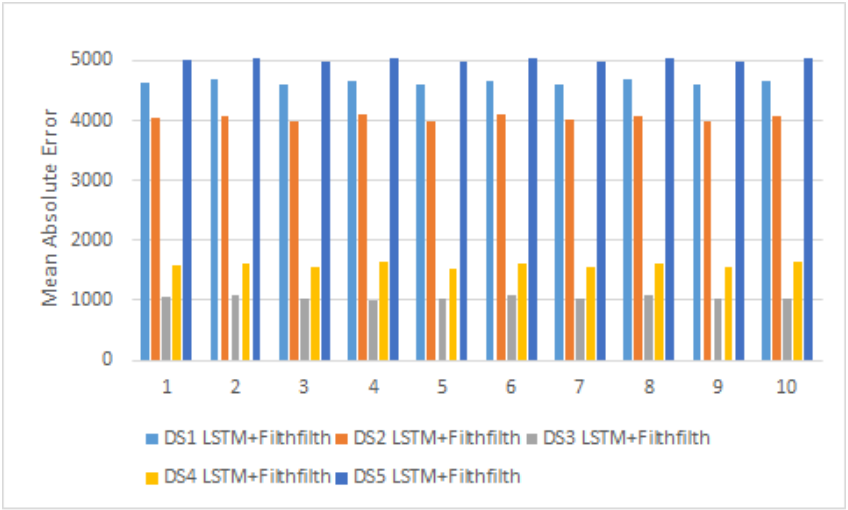



Figure 4.18: Cross validation of comparison of various filtering techniques

## 4.6 Conclusion

In this paper, we found the comparative performance analysis of many filtering techniques with LSTM to predict energy consumption. The novelty of this paper is to improve the smoothness of the large data, remove noises from the signals, and neglect non-relevant added signals from the system. Here, we improved accuracy by using various filtering techniques on data to be used in the demand and supply model. The performance analysis compared with mean square error MSE, mean absolute error MAE and median absolute error MaAE, Co-efficient of determination ($r^2$), co-relation ($r$) and accuracy. In the future, this comparative study will be upgraded by using the latest algorithm for the prediction of energy in SG.

# Chapter 5

# Demand Response Management using Prophet Model

Smart Grids (SG) generate extensive datasets regarding the system variables, viz., demand and supply. These extremely large datasets are known as Big data. Hence, preprocessing of this vast data and integration become critical steps in the load forecasting process. The precise prediction of the load is the primary concern while balancing the demand and supply in SG. Many techniques were devised for load forecasting using machine learning methods such as Deep-learning Models. However, in the case of large datasets, only a few models provide good performance, viz. Autoregressive Integrated Moving Average (ARIMA). However, this approach is complex as it takes a minimum of 50 observations for taking up an evaluation. The Prophet technique is used in the prediction of future demand response based on the past data which is in the form of a time series. This technique is valid even if a few values in the time series are not available. Furthermore, the procedure is not affected by fluctuations, trends, and abnormal variations. The automatic model fitting approach is adopted for its effective performance. Further, ARIMA and Prophet model have been used to forecast and the approach is verified using various evaluation metrics. The demand response management was achieved and being validated with two datasets. The results show the effectiveness of the Prophet model in the demand response management scheme involving large datasets.

## 5.1 Introduction

A smart grid differs from the traditional electricity system which involves the flow of energy in one way. It enables real time collection of data in the transmission and distribution network facilitating monitoring of electricity for efficient energy management. It involves technologies such as data acquisition, control, automation and communication, which work together in the grid to respond to the ever-changing demands of consumers. The flexibility can be achieved by making the customers follow demand response programs. However, several challenges are involved in implementing these programs in a real-time environment. Demand side management (DSM) in electrical power systems is

one of the solutions to these challenges by shifting the flexibility of the power system to the consumer side.

Besides load forecasting, there are two kinds of demand response (DR) in the power sector: non-dispatchable and dispatchable demand response. Dispatchable demand response deals with the customer appliances. In some cases, the utility directs the consumer regarding cutting down the air conditioner or heater load during peak demand periods, thereby reducing the cost. Hence, the consumer will be directed only when the utility can forecast and predict the peak load. The problem arises when the forecasting model shows errors. The non-dispatchable demand response or the retail price-responsive demand is when the customer has the liberty to decide whether to cut down his consumption. It is based on the retail rate design and does not remain fixed. This includes dynamic pricing programs. In this case, the issue is redundant, and in most cases, the increase in prices has no perceptible effect on the consumption pattern of consumers. A vast amount of data is derived from the SG setup, on a second-by-second basis. However, there are various challenges associated with load forecasting using this data. First, the data which is mostly accumulated is unstructured as they are gathered from a wide area containing a number of households. Second, the data collected is interlinked. Therefore, extraction of a particular data is found difficult for real-time applications. Many works were carried out on load forecasting and demand response management using big data to solve these issues.

For managing the demand-response of energy, authors have discussed the ARIMA model [111]. This model has the capability to lag its forecast errors by itself. It captures the consumed energy in the grid. The ARIMA model is used to detect abnormalities. The authors have used automated fitting methods. However, it is not suitable for electricity consumption where there is a high variation in consumption behavior. Subsequently, the occupancy levels which can improve energy prediction are highlighted. Their accumulated data is related to the premises of a residence [42]. Here, the authors collected data on the network activity and the consumption data of the consumers, on a daily basis. They used ARIMA model to forecast with accuracy. It is acknowledged that the measurements of the constructed residence have a remarkable explanatory variable. In a similar work, the authors elaborated on a short to medium term load prediction model [37]. The author highlighted the Big data approach for smart homes.

The authors have used Big data in [112] for energy prediction. In a similar work [113] data management system was used for forecasting. This model can help the customers to manage energy and reduce grid failure. They developed a model to make it economical and effective for better load distribution. In [114], authors have evaluated an hourly

demand profile, which is effectively trained. Here, they have used a hybrid model of the Bi-directional Long-Short-Term Memory (BLSTM) and the ARIMA model for prediction of energy. In another work, the ARIMA model is used in all the phases of time series data. In a different work, Wang *et al.* have dealt with short-term wind power prediction [115], where ARIMA model is used for better performance [116]. Here, the authors have described issues with a real-time price forecasting method. In [117], the authors have randomized a consumer algorithm for managing demand response [118]. Moreover, the authors have designed an optimized demand response for efficient management of supply and demand in SG.

### 5.1.1 Related work

Energy management is the most crucial part of an SG. For managing the supply and demand of energy, various big data analytics approaches were performed well in SG. Safhi *et al.* discussed load forecasting which is based on big data [119]. There are various prediction techniques discussed in the literature for managing demand and supply gaps. ARIMA model is an essential analytical method where time-series data is used. This model is trained to understand the dataset in an efficient way. ARIMA model has a great role in the prediction of the future trends of a time series. Krishna *et al.* discussed ARIMA model where it captures the process of energy consumption in the power system [111]. Moreover, it has the capability to check the validity of the consumed energy of the system.

The above model was unable to capture the experiment-based characteristics in a proper way. In order to overcome this limitation the Prophet model was proposed by some authors. In the work carried out by [120], the authors have used K-means clustering with the ARIMA model to obtain DR from an area in their university. They have used data of two academic years viz., 2014-2015. First, they obtained the predicted electricity consumption data for 2016. The electricity consumption forecast was done, and DSM variables were obtained thereafter. In [121], the authors have used two datasets viz., one dataset comprising the electricity consumption of three months (February 2013-April 2013) and the other relating to the four years (July 2009 - June 2013). On both these datasets, dynamic demand response was carried out using six different prediction models including ARIMA model and the results were finally compared.

The authors in [122] have carried out work on applications of technologies involving Big data such as online monitoring of renewable energy systems and wind turbines, based on ultra-sphere model. The third application was backup and recovery in electric power generation process. The authors believed that big data technologies are essential for buildings

and handling SG. However, their research lacked the use of the big data in SG. In author deals with the analysis of big data for renewable energy resources [123]. It further aims at presenting techniques for Demand Response Management (DRM) using big data. The technology used to carry out the DRM is a simulator. While issuing DR, the simulator studies the big data and identifies correlations. The aims is to use the technology and defined its usefulness in the future. However, the research has not been made using different Machine-learning techniques for carrying out the DR. By optimizing the programming, it would become easier for advancing the technology, in the near future.

The research in [124] was carried out to solve two main issues while handling big data. First, the authors used the expansion K-SVD sparse representation technique to extract hidden patterns of electricity consumption. Second, it can also be used to compress the data and store it efficiently. Further, they used a series of other computational techniques like SVM, PCA etc. to classify the consumers into various groups. So, basically, they worked on efficient compression of data and its extraction. However, their research lacked obtaining a DR from the extracted data. In [125], the authors have used big data technology to recognize the patterns of electricity consumption by consumers. They have further shifted the peak load by studying the consumption patterns. They used the K-means algorithm to achieve the final clusters. However, they did not predict the future energy consumption patterns of the consumers.

The work in [126] was based on obtaining DR on electricity consumption data of one month only. The authors have used the big data technologies like Apache Spark to analyse the data to obtain DR and thereby have successfully curtailed the air conditioner and heating loads. In [127], the authors have created various priority lists varying from consumer to consumer. They have further analyzed various aspects such as time, need and power consumed while creating the priority lists. However, they have not done DSM using the priority lists. Furthermore, they did not predict the future electricity consumption by the consumers.

Almajarouee *et al.* introduced techniques based on peak load where the long-term forecasting provided results accurately. They have tried to save cost and time by using this model [128]. Many authors discussed the Prophet model approach for the energy demand forecasting in SG. They tried to improve energy generation and consumption with accurate forecasting methods. The authors discussed the data cleaning algorithm [129]. Here, the authors have highlighted the errors in the models and various issues such as Benchmark-related algorithms. In a different work, the authors defined the smart energy management which can maintain quality of life for the consumer [130]. The authors paid attention in minimizing of energy by developing an efficient model.

The energy consumption was managed [131] by using operational approach techniques. This technology is designed for proper communication with SG. The Energy Management System (EMS) has a significant role in managing demand and supply [132]. However, the data sample taken to study were small datasets. The authors have highlighted the multiple homes. The problem formulation is carried out via multiple-knapsack problems [133]. The authors discussed the importance of the external variables of consumed energy. They defined the state-of-the-art energy load forecasting method. They have also defined the challenges which are involved with big data. Table 5.1 shows, at a glance, the various research work done in this area.

### 5.1.2 Motivation

ARIMA is mainly used by professionals who have prior knowledge of the intricacies of the model. If a single parameter in the equation is incorrect, the entire result will get affected. However, Prophet model uses a Bayesian curve fitting method and does not require prior knowledge of datasets. It automatically finds seasonal trends from the data. The Prophet model incorporates seasonal trends such as holidays and weekends, whereas the ARIMA model incorporates both seasonal and non-seasonal trends with time-series data. It provides great precision compared to any other method.

### 5.1.3 Contribution

The ARIMA model requires expert knowledge as a prerequisite to make use of it. In addition, it is not flexible in use and is non-automatic. The Prophet model overcomes all the aforesaid limitations and is a powerful tool for prediction. It gives precise results with non-seasonal trends and also incorporates non-linear trends with datasets. The contribution of the proposed work is as follows:

- Optimized parameters for the Prophet and ARIMA model is used for better performance.
- Applied preprocessing techniques to clean the data.
- The abnormal data is removed for the prediction of consumption of energy in forecasting.
- ARIMA and the Prophet model is compared and analyzed with different performance metrics.

Table 5.1: Works related to Energy Forecasting for Demand Response

| SN | Author | Techniques | Description | Application/ Domain |
|---|---|---|---|---|
| 1 | Newsham *et al.* [2010] | ARIMA model | To collect data related to the total occupied building, They installed wireless sensors in the building at eastern zone at Ontario | Energy forecasting |
| 2 | Krishna *et al.* [2015] | ARIMA model forecasting method | They have used automatic model fitting methods | Energy prediction |
| 3 | Asaleye *et al.* [2017] | Decision support tool | Used for renewable energy microgrids (DSTREM) | To ascertain daily energy consumption |
| 4 | Luo *et al.* [2018] | An innovative hybrid RTP forecasting model | RTP model has been used for analysing customer conducts. They got more benefit by scheduling the use of home appliances | Real-time forecasting |
| 5 | Sendric *et al.* [2019] | Data cleaning algorithm | To solve problems of ambiguous data from Big data wireless sensor networks. They used these to clean data in smart cities | Cleaning the ambiguous data |
| 6 | Gupta *et al.* [2019] | Short-term Wind Power Prediction (WPP) to improve efficiency of power systems | Hybrid ARIMA-GARCH model | Forecasting of Wind Power |
| 7 | Liu *et al.* [2020] | Big data analytics | Big data analytics in running Smart cities | To build Smart cities |
| 8 | Wang *et al.* [2020] | hybrid model based on the ARIMA model In this paper, Bidirectonal(Bi-LSTM) model and Bayesian optimization (BO) model | The ARIMA model can tackle the linear part of the time series data. Bi-LSTM can handle the nonlinear features The hybrid model provided a good prediction tool | The installation or replacement of electrolytic capacitors |
| 9 | Almazrouee *et al.* [2020] | Prophet model | Prophet model used and forecasting accuracy compared with Holt–Winter's model. | To predict Long-term peak loads |

### 5.1.4 Organization

Section 5.2 defines the detailed description of the methodology and workflow of our proposed model. Section 5.3 describes the performance of the models along with evaluation parameters. Subsequently, Section 5.4 is devoted to the discussion of the dataset and the Prophet model. Finally, the conclusion of this work is stated in Section 5.5.

## 5.2 Methodology

The methodology used for current work is described in Figure 5.1. Initially, a high volume of data is obtained from the Smart Grid (SG) and the smart meters which are connected to the electric power system. The dataset is a multivariate time-series data collected from Pennsylvania-New Jersey-Maryland Interconnection (PJM) which is a regional transmission organization (RTO) in the United States of America. PJM is a part of the Eastern Interconnection grid operating an electric transmission system serving all parts of Delaware, Illinois, New Jersey and North Carolina. The dataset is of the PJM East that consists of data from 2014–2016 for the entire eastern region where 2014 to 2015 is used for training and 2015 to 2016 is used for testing. The data consist of extraneous values and noises and hence the data needs to be filtered and the relevant data is extracted from the large datasets. Subsequent to processing, model fitting is done and then used in the Prophet model. The Prophet workflow model generates reliable forecast in the form of time series values in the demand response process.

Further, in a traditional time series model, there are certain problems, as mentioned below.

- The time interval between data has to be the same throughout, while this is not a problem in Prophet model.
- Day with NA (nil data) is not allowed, while this is not an issue in the Prophet model.
- Seasonality with multiple periods is complex, while in Prophet model handles this problem by default.
- Parameter tuning by an expert is necessary for the ARIMA model, while in the Prophet model there is a default setting, from which parameters are easily interpreted. The Prophet model is extensively used in various fields for forecasting with an extensive range of data.

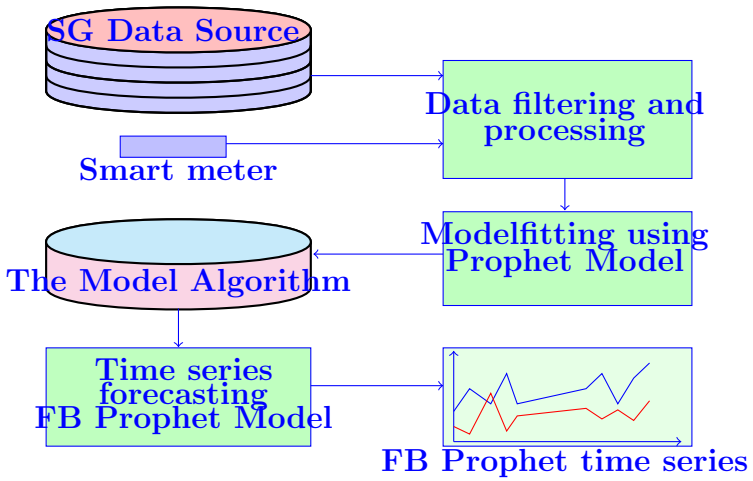



Figure 5.1: The workflow diagram of the Prophet model for the prediction of demand response

## 5.3 Mathematical Modelling of ARIMA and Prophet

### 5.3.1 Modelling of ARIMA

The ARIMA model processes data in a time series for making a prediction. The ARIMA model is used in case of both linear and multiple-regression models. Multiple regression model refers to the prediction of outcomes of dependent variables which are based on variables of independent variables. The model is generally referred to as ARIMA (p, d, q) where p, d and q are zero or positive numbers. ARIMA model makes use of a stationary time series. Using a multiple linear regression model, it can work over non-stationary time series data. The values of p, q and d can be found using auto-ARIMA. The process seeks to identify the most optimal parameters for ARIMA model settling on a single-fitted model. The process works by conducting differencing tests to determine the order of differencing 'd' and then fitting the models within the ranges of defined start p, max p, start q, and max q. The parameters p, q and d were set to (4, 1, 1). Finally, the model has trained on 2014-2015 data to obtain a prediction for 2016 consumption data.

The ADF (Augmented Dickey Fuller) test is useful in detecting the unit root in a series to understand whether the series is stationary or non-stationary. Here, the null and alternate hypotheses state that if a series has a unit root, it fails to reject the null hypothesis which says that the unit has a root. Then, the series is non-stationary. This means that the series can be linear, stationary, or difference stationary. The experimental results show that the ARMA (Auto Regressive Moving Average model) is based on real data which is a stationary time series. In the flow chart shown in Figure 5.2, three features of the stationary data have been selected. Here, the first characteristic is the constant mean,

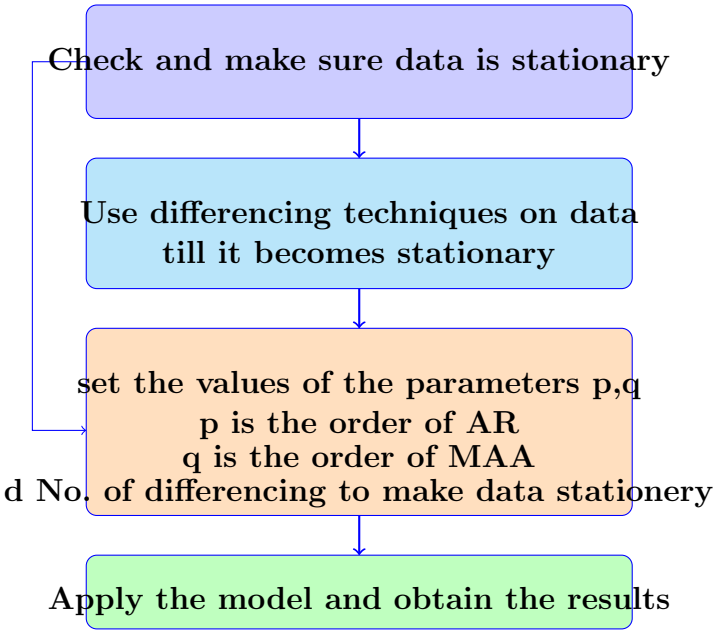



Figure 5.2: The workflow of ARIMA model

the second one is the variance which is also a constant and the third characteristic is the co-variance, where the signal of past data, at different times, is constant.

Here, the daily stationary signal does not meet the first condition, but satisfies the second and third conditions. The moving average component of ARMA is set for change mean, and therefore, the first condition is not essential for the appropriate ARMA for a given time series. Later, the process of residual checking is completed.

If conditions are satisfied, then the process is stopped or otherwise continued. In Figure 5.2, the procedure of the ARIMA model is shown. Here, the power consumption datasets were collected and then the data is preprocessed. Subsequently, any abnormal data present in the datasets is eliminated. After the selection of features, the important features are extracted by using the classifier (Support Vectors Machine) and at last the predicted energy. On the basis of lagged data, future prediction is decided. In the above model, the equations are based on an autoregressive function. It is a function where the current value is generated based on the immediately preceding value. In the second process, the current value is generated based on the last two values. An AR (0) process would imply that there are no dependencies among the terms in the equations.

The aforesaid term predicts certain errors known as moving averages. The time series is differenced to make the data stationary. There are many models such as Randomwalk model, Random-trend model, Autoregressive model and the Exponential-smoothing model. All are the special cases of ARIMA model. The time series representing the electricity

consumption of a single consumer, at time $t$, is given by the value $Y_t$. The ARIMA model is discussed as in Figure 5.2 which represents the flow chart of the model where the parameters are indicated. Here, the values of parameters are selected and residual checking is done. The residual value is differenced between observed value and the predicted value. ARIMA model aims to predict power.

$$Y_t = c + \in_t + \sum_{i=1}^{q} \alpha_i X_{t-i} + \sum_{k=1}^{r} \beta_k \in_{t-k} \tag{5.1}$$

whereas, $Y_t$ indicates the consumption of energy at time $t$, $c$ indicates obstruction of the signal at $q$. $X_{t-i}$ and $\alpha_i$ are the parameters and regressors for the AR part of the model respectively. It assumes Gaussian noise as in $\in_t$ and compounds $q$ over time periods. Further, $\in_{t-k}$ and coefficient $\beta_k$ represent the parameters and regressors of the MA part of the model respectively.

$$f(x) = x^2 + 2x + 1 \tag{5.2}$$

In Eq. (5.2), $f(x)$ is the dependent variable and it indicates the prediction of the energy consumption using time series data. The values $x^2$ and $2x$ are independent variables and define the first-order differencing for making stationary data into a non-stationary data.

The prediction of the energy consumption in a time series data is described as follows:

$$\begin{aligned} Z_{t1} = \alpha_0 - \psi_1(z_{t1} - 1) - \psi_2(z_{t1} - 2) - ... - \psi_z n_t 1 - p \\ + \epsilon - \alpha_1(\epsilon_{t1} - 1) - \alpha_2(\epsilon_{t_1} - 2) - ... - \alpha\epsilon t 1 - q \end{aligned} \tag{5.3}$$

Where, at time $t_1$, $z_{t1}$ and $\epsilon_{t1}$ are the predicted values and the random error of data $\psi(z_{t_1} - 1) \cdots, p$ indicates the model parameter, $\alpha_1 ... q$ indicates the model parameter, $p$ and $q$ are represented by the autoregressive and moving average orders. Eq. (5.3) shows some important cases of the ARIMA models, If $q_2 = 0$, then Eq. (5.3) becomes an AR model of order $p_2$, and when $p_2 = 0$, the model decreases to a MA model to work with order $q_2$. The past data is the main basis in the prediction of energy by ARIMA model.

The general forecasting equation is:

$$z_t = \mu + \phi 1(z_t - 1) + \ldots + \phi p z_t - p \\ -\theta 1(e_t - 1) - \ldots - \theta q e_t - q \tag{5.4}$$

George Box and Gwilym Jenkins have introduced the moving average parameters ($\theta$) having negative values in the equation. Hence, the actual numbers are used in the equation and there is no ambiguity, as the output was read by us at the time of use of this software. These parameters are denoted by AR(1), AR(2)...AR(N) and MA(1), MA(2)....MA(N). To recognize a suitable ARIMA model for $Z_1$, the order of differencing viz., ($d_2$) is to be decided. It is very important to make the series spatial so that the characteristics of seasonality can be removed. If the prediction of the differenced next series is constant, then we have to apply random-trend model. Here, the series is autocorrelated and the errors show the number of AR terms ($p_2 \geq 1$) and number MA ($q_2 \geq 1$). These are also needed in the equation. To determine the values of $p_2$, $d_2$ and $q_2$ is the best way for a given time series.

There are many types of non-seasonal ARIMA models that are discussed as follows: ARIMA (1, 0, 0) model is denoted as the first-order autoregressive model. If the series is stationary and autocorrelated, then it can be forecast as a multiple of its own previous value, plus a constant. The forecasting equation, in this case is as below.

$$Z_t = \mu + \phi_1 Z_{1t} - 1 \tag{5.5}$$

In Eq. (5.5), $Z_1$ is less developed data on itself by one period. This is an ARIMA (1, 0, 0) + constant model. If the mean value of $Z_1$ is zero, then the constant value will not be sufficient. If the slope coefficient $\phi$ is positive and less than 1, $Z_1$ is stationary. If the value of next time period value is predicted to be $\phi$ times it creates a great distance from the mean, as this is a time value. If $\phi_1$ is negative, it predicts a mean level with alteration of the signs. It also predicts that $Z_1$ will be below the mean of the next period if it is above the mean at that time.

In a 2nd order autoregressive model ARIMA (2, 0, 0), there is $Y(t-2)$ term on the right, as well as, on the left and depends on the signs and magnitudes of the coefficients. It describes a system where the mean level is of a sinusoidal wave pattern. It is like the motion of a mass on a spring that is subjected to random shocks. If autoregressive coefficient is equal to 1, it is a series with infinitely slow mean which returns to the

previous state. The equation for this model can be written as follows:

$$z_t - Z_1 t - 1 = \mu \tag{5.6}$$

or equivalently

$$z_t = \mu + Z_1 t + 1 \tag{5.7}$$

where, the constant term is the mean which periodically changes (i.e., the long-term drift) $Z_1$. This model could be fitted as a no-intercept regression model in which the first differencing of $y$ is the dependent variable as it includes only a nonseasonal difference and a constant term. It is "ARIMA0,1,0) with a constant." The Random-walk without drift model would be ARIMA (0,1,0) without any constant. ARIMA1,1,0) is used as differenced with first-order in autoregressive model. Here, autocorrelated means that the errors are found as in Random walk model. Then the problem can be settled by adding past data of the dependent variable to the forecast equations i.e., by regressing the first difference of $z$ on itself lagged by one period. The forecast equation is:

$$Z_t - (Z_1 t - 1) = \mu + \phi_1 (Z_1 t - 1) - (Z_1 t - 2) \tag{5.8}$$

$$yt - (Z_1 t - 1) = \mu \tag{5.9}$$

### 5.3.2 The Modeling of Prophet

Prophet has been developed by Facebook in order to overcome some problems which exist in ARIMA. Prophet works through use of an additive model whereby the non-linear trends in the series are fitted with the appropriate seasonality.

It is a time series predictive method where the aim is to predict power in SG. For this purpose, appliances are categorized as: interruptible, non-interruptible and base appliances. Power categorization of:

- Interruptible Appliances

$$F_{in} = \sum_{t-1}^{U} \sum_{ineIN} \sigma_{in} * rw_{in}(t) \tag{5.10}$$

where, $F_{in}$ is the power consumption of appliances, $ineIN$ indicates Interruptible Appliance, $\sigma_{in}$ indicates power rating, $U$ is the total time slot and $rw_{in}(t)$ is the

state of each Interruptible Appliance at time slot $t$.

$$rw_{in}(t) = \sum \begin{cases} 0 & \text{if appliances are off} \\ 1 & \text{if appliances are on} \end{cases} \tag{5.11}$$

- Non-interruptible Appliances

$$F_{in} = \sum_{t-1}^{U} \sum_{niNI} \sigma_n i * rw_{in}(t) \tag{5.12}$$

Where, $F_{in}$ is power consumption of appliances, $niNI$ indicates Non-Interruptible Appliance, $\sigma_{in}$ indicates power rating, $U$ is total time slot and $rw_{in}(t)$ is the state of each Non-Interruptible Appliances at time slot t.

$$rw_{in}(t) = \sum \begin{cases} 0 & \text{if appliances are off} \\ 1 & \text{if appliances are on} \end{cases} \tag{5.13}$$

- Base appliances are similar to fixed appliances which do not have flexibility of operation. The pattern of consumption of energy and operational period of appliances cannot be changed. It is important that these appliances must be 'ON' when user wants to switch them ON such as home appliances viz. TV, Fridge and other devices.

$$F_b = \sum_{t-1}^{U} \sum_{beB} \sigma_B * rw_B(t) \tag{5.14}$$

Where, $F_b$ represents total energy consumption, B is the base appliance at time $t$, $rw_B$ is each base appliance, $\sigma_B$ is the power rating.

Since Prophet model works over data trends, holidays and seasonal data provide complex features. Seasonality is input on the basis of day, week and year. Prophet model where consumption is represented by time series is a data method expressed, as follows,

$$Z(t) = Y(t) + S(t) + H(t) + E(t) \tag{5.15}$$

$Z(t)$ indicates the consumption, $Y(t)$ represents the data trend function, $S(t)$ indicates the seasonal data, $H(t)$ indicates the holiday-based data and $E(t)$ represents the errors.

The trend function of Prophet model $H(t)$ is highlighted by a piecewise linear growth model. It is also called a Saturation-growth model. The maximum load data does not show

a saturating growth which is a piecewise linear growth model represented, as follows:

$$h(t) = (l + a(t)^T \delta)t + (n + a(t)^T \sigma) \tag{5.16}$$

Here $l$ is the growth rate, $\delta$ indicates rate adjustment, $n$ is an offset parameter and $\sigma$ is the change point.

$$b_j(t) = \begin{cases} 1 & \text{if } t \geq U_k \\ 0 & \text{otherwise} \end{cases}$$

Where $b_j$ is the output and $U_k$ is the change point.

The seasonality function is manifested by the following equation:

$$T(t) = \sum_{n=1}^{L} (b_n cos(2 * p * in * t/q) + d_n sin(2 * p * in * t/q)) \tag{5.17}$$

In Eq. (5.17), $T(t)$ is the seasonality function. Here, the time series multiperiod seasonality method is used. The Fourier series is applied to the daily and seasonal changes. Therefore, the seasonality function is discussed as:

$$A_t = [1(t \in E_1), \cdots, 1(t \in C_m)] \tag{5.18}$$

Here, $A_t$ indicates matrix of regresses, $E$ indicates holiday's and 1 represents the holidays parameter.

$$H_t = A(t)l \tag{5.19}$$

In Eq. (5.19), $H_t$ indicates holidays and $l$ indicates a corresponding change in the forecast. It produces estimates of unknown variables.

## 5.4 Results and discussion

### 5.4.1 Description of Datasets

This section provides the description of the datasets. Table 5.3 shows the consumer preference regarding the appliance taken with the reason for selection. Table 5.4 shows the rating of each of the appliances. The dataset taken is hourly time series and forecasting was done using Facebook's Prophet model and ARIMA model. Energy consumption has

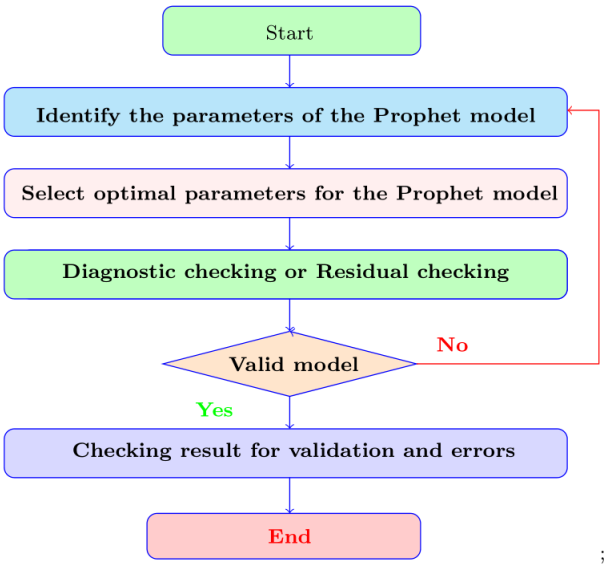



Figure 5.3: The flow chart of Prophet Model

Table 5.2: A sample of Smart Grid dataset

| Date: 01/01/2014 | |
|---|---|
| **Time** | **Energy kWh/half-hour** |
| 00:00.0 | 0.488 |
| 30:00.0 | 0.449 |
| 00:00.0 | 0.424 |
| 30:00.0 | 0.439 |
| 00:00.0 | 0.291 |
| 30:00.0 | 0.262 |
| 00:00.0 | 0.308 |
| 30:00.0 | 0.138 |
| 00:00.0 | 0.404 |

unique characteristics. First, the electricity consumption for the year 2016 was predicted using ARIMA model. However, to make use of ARIMA model, it is to be made sure that the data is stationary.

Table 5.3: Preference matrix for different devices

| SN | Time/ Reason | Priority | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 00:01-03:00 | Fridge | Air conditioner | Tubelight | Microwave oven | Sandwich Maker | Dishwasher | Hair dryer | Vacuum cleaner | Washing machine | Clothes dryer |
| | Reason | Fridge is essential for use all the 24hrs. The other items are taken according to priorities during the course of the day | | | | | | | | | |
| 2 | 3:01-6:00 | Fridge | Air conditioner | Tubelight | Dish washer | Microwave oven | Sandwich maker | Hair dryer | Vacuum cleaner | Washing machine | Clothes dryer |
| | Reason | Dishwasher is placed as the fourth priority here. Some people sleep early and wake up early. | | | | | | | | | |
| 3 | 6:01-9:00 | Fridge | Tubelight | Dish washer | Hair dryer | Sandwich maker | Microwave oven | Washing machine | Vacuum cleaner | Clothes dryer | Air conditioner |
| | Reason | It is time to go to school and office in the morning. | | | | | | | | | |
| 4 | 9:01-12:00 | Fridge | Washing machine | Vacuum cleaner | Clothes cryer | Microwave oven | Sandwich maker | Hair dryer | Dish washer | Air conditioner | Tubelight |
| | Reason | People leave home for office by 9:00am. Thereafter, the priority for people at home is to clean the house and wash clothes. | | | | | | | | | |
| 5 | 12:01-15:00 | Fridge | Clothes dryer | Air conditioner | Microwave oven | Dishwasher | Hair dryer | Vacuum cleaner | Washing machine | Tubelight | Sandwich maker |
| | Reason | It is the lunch time, so Microwave oven is used. Besides, the ambient temperature is high. So, air conditioner is also used. | | | | | | | | | |
| 6 | 15:01-18:00 | Fridge | Dish washer | Air conditioner | Tubelight | Microwave oven | Sandwich maker | Washing machine | Clothes dryer | Hair dryer | Vacuum cleaner |
| | Reason | It is time to take an afternoon nap and wake up for tea in the evening | | | | | | | | | |
| 7 | 18:01-21:00 | Fridge | Tubelight | Microwave oven | Hair dryer | Air conditioner | Sandwich maker | Dishwasher | Vacuum cleaner | Washing machine | Clothes dryer |
| | Reason | Now, time for people to return from office. take bath and use the hair dryer. After dinner they relax using Air conditioner. | | | | | | | | | |
| 8 | 21:01-24:00 | Fridge | Tubelight | Air conditioner | Microwave oven | Dishwasher | Hair dryer | Sandwich maker | Vacuum cleaner | Washing machine | Clothes dryer |
| | Reason | Some people return late from the office, and usually have their dinner at 9:00pm. | | | | | | | | | |

A stationary time series data has its mean, variance and other statistical properties constant over a given time frame. The data used here is shown in Table 5.2. It is a collection of data from the source [134]. The aforesaid data helps in the visualization and analysis of the changes. In addition, the future trend of the variables under analysis can be predicted using Machine-learning Algorithms. The entire dataset used is from smart meter energy user data from households. It had half-hour time stamps and generated energy data in kWh/half-hour. This data comprised of 1 million values, from which our final dataset of 149,999 were extracted. The dataset used for analysis with 100,000 and a test dataset with 49,999 values. A sample of the final dataset is shown in Table 5.5. A total of 24 appliances were taken into consideration for the analysis.

Table 5.4: Home appliances rating used in households

| SN | Name of the appliance | Rating (KWh) |
|---|---|---|
| 1 | Fridge | 0.2 |
| 2 | Tubelight | 0.055 |
| 3 | Air conditioner | 4 |
| 4 | Microwave | 1.7 |
| 5 | Dishwasher | 1.5 |
| 6 | Hair dryer | 1 |
| 7 | Sandwich maker | 1 |
| 8 | Vacuum cleaner | 1.4 |
| 9 | Washing machine | 0.5 |
| 10 | Clothes dryer | 2.5 |

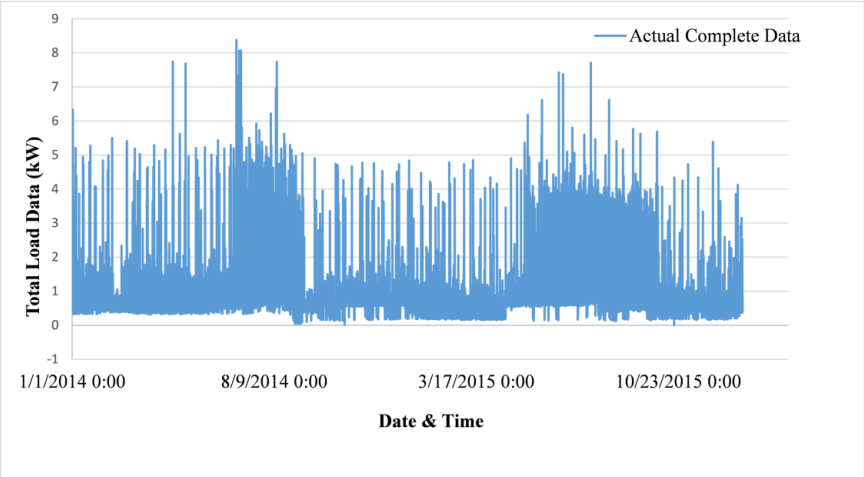



Figure 5.4: A stationary time series data has its mean, variance and other statistical properties constant over the given time frame.

### 5.4.2 Prediction using ARIMA model

The combined dataset is shown in Figure 5.4, which consists of the data from 2014 to 2015. In the beginning, it was predicted based on the electric consumption for the year 2016 using the ARIMA model. Before applying the ARIMA model, it is to be made sure that the data is stationary. With the graph plotted in Figure 5.4, it can be observed that the data is not in stationary mode.

The difference between the present and the previous period gives us the first differencing value as shown in Figure 5.6 The obtained values are then plotted to check if the statistical properties are constant. If still not constant, second differencing is obtained using the first differencing values.

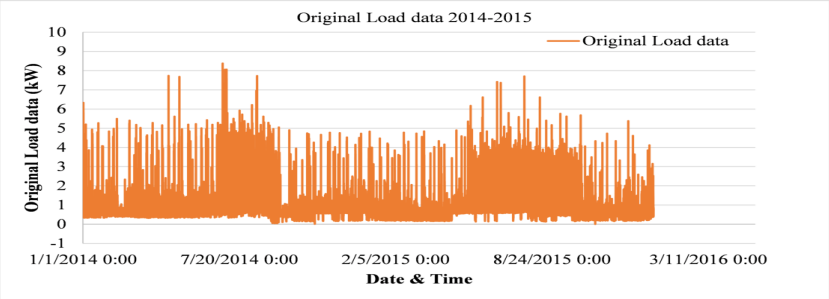



Figure 5.5: Actual Load Data for 2014-15

Figure 5.6 depicts the data undertaken and the autocorrelation present in the datasets. The method of differencing needs to be applied, repeatedly, till the data obtained becomes

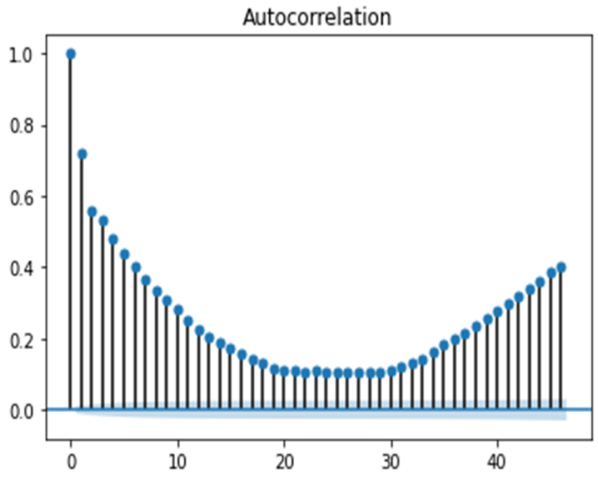



Figure 5.6: The first differencing applied to data-Autocorrelation.

stationary. The checking of the correlation between the data with its past values is called autocorrelation. For this, the autocorrelation function plot (ACF) is used. The plot shows

the correlation between various points . The correlation coefficient is plotted on the x-axis with the number of lags on the y-axis. ACF plot is mainly used to determine which one among them is to be used as data.

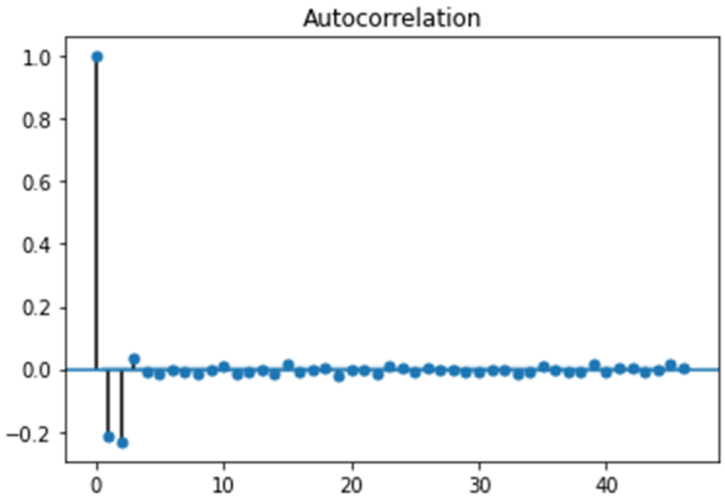



Figure 5.7: Autocorrelation present in the datasets.

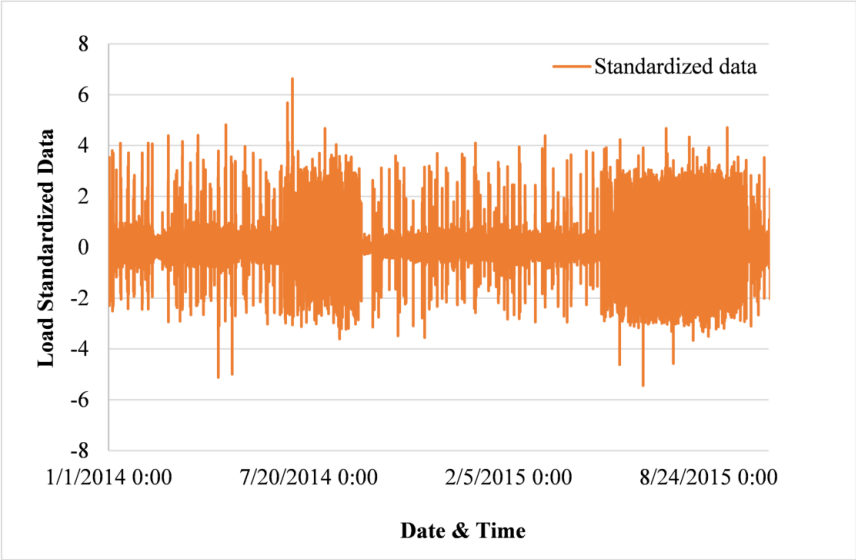



Figure 5.8: The standardized data with a prediction of day ahead

When differencing is done, the data obtained generally oscillate. This means that the subject data has achieved the constant values of statistical variables.

That is, the data is now stationary and ready to be worked upon using ARIMA model. Subsequent to making the data stationary, the parameters (p, q, d) are as follows:

- p: order of the AR term,
- q: order of the MA term, and

Table 5.5: Consumption dataset from households

| Date: 04/10/2016 | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Units of all the columns are in kW except Column 2 in Wh and Column 3 in (Wh/hh) E represents base 10 with its superscript on the right side | | | | | | | | | | |
| **Time** | **Energy** | **Wh/hh** | **Gene-ration** | **AC** | **Furnace** | **Cellar Lights** | **1st floor lights** | **Dining room** | **Micro wave** | **Total Load** |
| 30:00 | 8.00E-02 | 4.00E+01 | 5.78E-05 | 9.53E-03 | 5.34E-03 | 1.26E-04 | 1.12E-02 | 4.30E-03 | 4.73E-03 | 3.10E-01 |
| 00:00 | 1.09E-01 | 5.45E+01 | 1.53E-03 | 3.64E-01 | 5.52E-03 | 4.33E-05 | 2.35E-02 | 3.59E-03 | 4.45E-03 | 7.28E-01 |
| 30:00 | 1.13E-01 | 5.65E+01 | 1.85E-03 | 4.18E-01 | 5.50E-03 | 4.44E-05 | 2.35E-02 | 3.52E-03 | 4.40E-03 | 6.26E-01 |
| 00:00 | 4.10E-01 | 2.05E+02 | 1.74E-03 | 4.11E-01 | 5.56E-03 | 5.94E-05 | 3.40E-03 | 3.40E-03 | 4.26E-03 | 7.83E-01 |
| 30:00 | 5.40E-02 | 2.70E+01 | 3.00E-05 | 1.72E-02 | 5.30E-03 | 1.19E-04 | 2.39E-02 | 3.92E-03 | 4.41E-03 | 1.99E-01 |
| 00:00 | 4.30E-02 | 2.15E+01 | 4.42E-04 | 1.27E-01 | 5.42E-03 | 5.44E-05 | 2.38E-02 | 3.81E-03 | 4.40E-03 | 4.98E-01 |

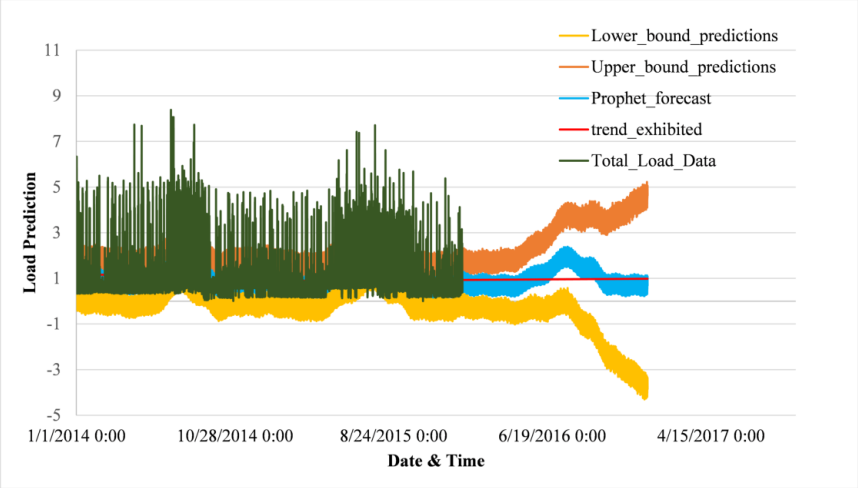



Figure 5.9: Prophet forecast analysis between actual and predicted data.

- d: number of differencing required to make the time series stationary

In Figure 5.7, the blue line represents the consumption data of predicted values for the year 2016. After extensive computation, these three values were set to be (1,0,4). Finally, the model was trained on the data relating to the years 2014-2015 to obtain a prediction for 2016. The data can be considered as stationary if it has constant amplitude and oscillates in the given time frame. In the case of signal with noise, ARIMA model acts as a filter to extract the data of the signal from the given system. The extracted signal is then worked upon to carry out future predictions. For fitting the data on the ARIMA model, it is important to make the data stationary using the differencing technique to obtain a standardised data as shown in Figure 5.8.

### 5.4.3 Prediction using Prophet model

The electricity consumption data from 2014 to 2015 was used to work out with the help of the Facebook Prophet Model. The forecast for the data of 2016 was done using the

model's inbuilt predictions. Figure 5.5 depicts the data from 2014 to 2015. The model was trained on this data to predict the electricity data for a later year. Figure 5.9 summarizes all the computations carried out in the process. The predicted values are in quite a good sync with the original data, and hence forecast was made for the year 2016. There could be certain variations in the actual data of 2016, and a margin of upper and the lower bound were created. Figure 5.9 indicates the trend of the predicted data. Figure 5.10 shows the trend exhibited by the predicted data for the electricity consumption for a year; the seasonality is considered.

Table 5.6: Performance comparison of Arima and Prophet for different Evaluation parameters

| **Model** | **MSE** | **RMSE** | **MAE** | **MAPE** |
|---|---|---|---|---|
| ARIMA | 1.06877 | 1.03381 | 0.6239 | 1.1932 |
| Prophet | 0.67546 | 0.82186 | 0.5308 | 1.0399 |

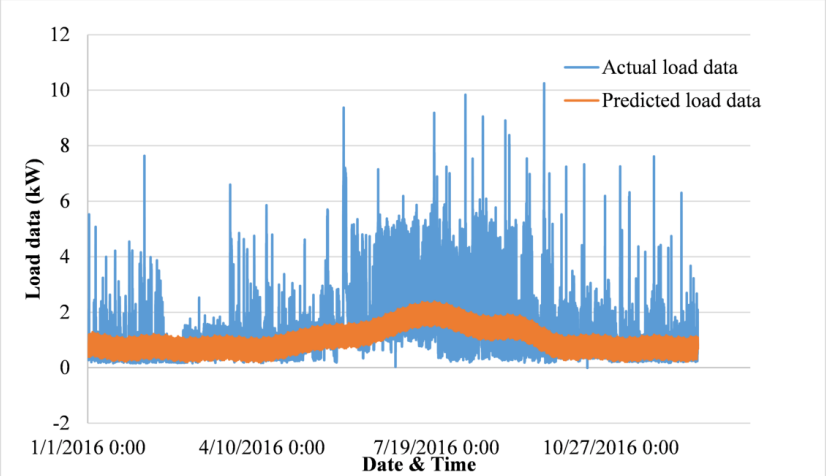



Figure 5.10: Comparison between real data and predicted data Facebook Prophet Model.

Table 5.7 shows the optimized parameters values of ARIMA and Prophet models. The parameters are optimized through a random approach in which a random set is generated, trained the model on the generated model, and make predictions. The set of parameters is selected that gives the highest prediction accuracy.

Table 5.7: Optimized parameters for the ARIMA and Prophet model

| **SN** | **Model** | **Parameters** |
|---|---|---|
| 1 | ARIMA | P(A=4), Q(Integration =1), d(differences=1) |
| 2 | Prophet | T(Trend =1.4), S(seasonality =4), H(holiday=1) |

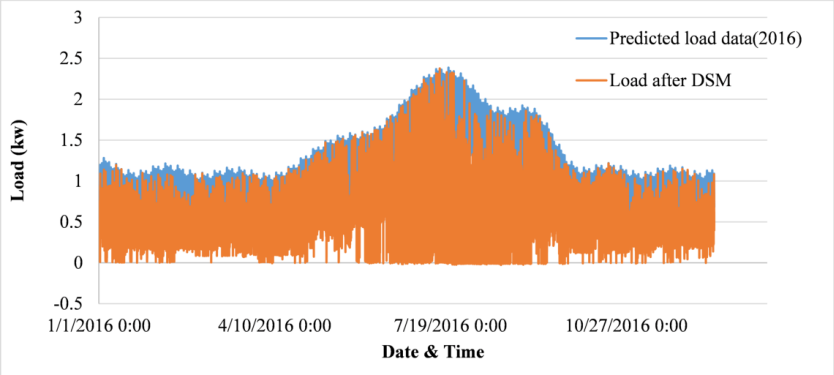



Figure 5.11: Demand Side Management on predicted data

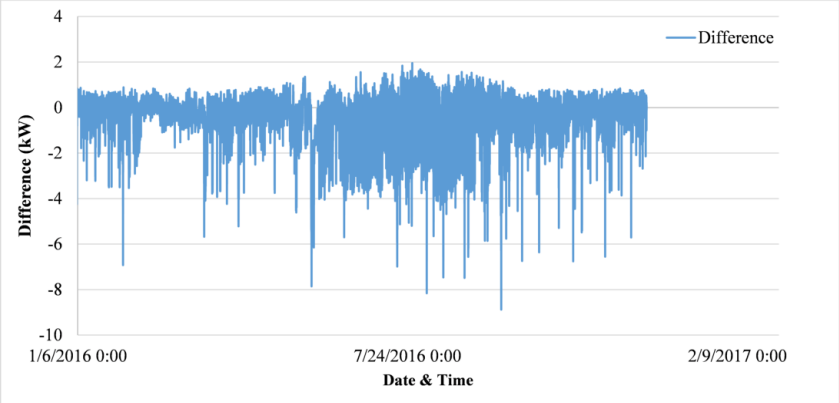



Figure 5.12: The difference between actual and predicted data

The various components of the Prophet model included were weekly, yearly and daily components and the trend exhibited by the predicted values. Figure 5.10 shows the predicted and actual values for 2016.

The values predicted, as shown in Table 5.6 with the original 2016 data, are compared for both the ARIMA and the Prophet model. In case of Prophet model, the mean square error is 0.67546 and the mean absolute error is 0.5308. The same is 1.06877 and 0.6239, respectively for ARIMA model. It is seen that the root mean square error and mean absolute percentage error of Prophet model are significantly less than those of ARIMA model. Hence, after comparing the two models, it has been found that the DR from Prophet model has better overall performance.

Facebook's Prophet Model is selected for carrying out DSM in the subject research. The data was fed to the appliances at home and the appliances switched off one-by-one on the basis of priority of usage. Figure 5.11 shows the DSM on the predicted values. The demand response was carried out between the total load and the generated electricity. For the DSM, the difference between the original and the predicted values of the year

2016 are calculated and presented is shown in Figure 5.12. The trend, over different time lengths, is shown in Figure 5.13 and can be useful for controlling the electric consumption of the appliances in households.

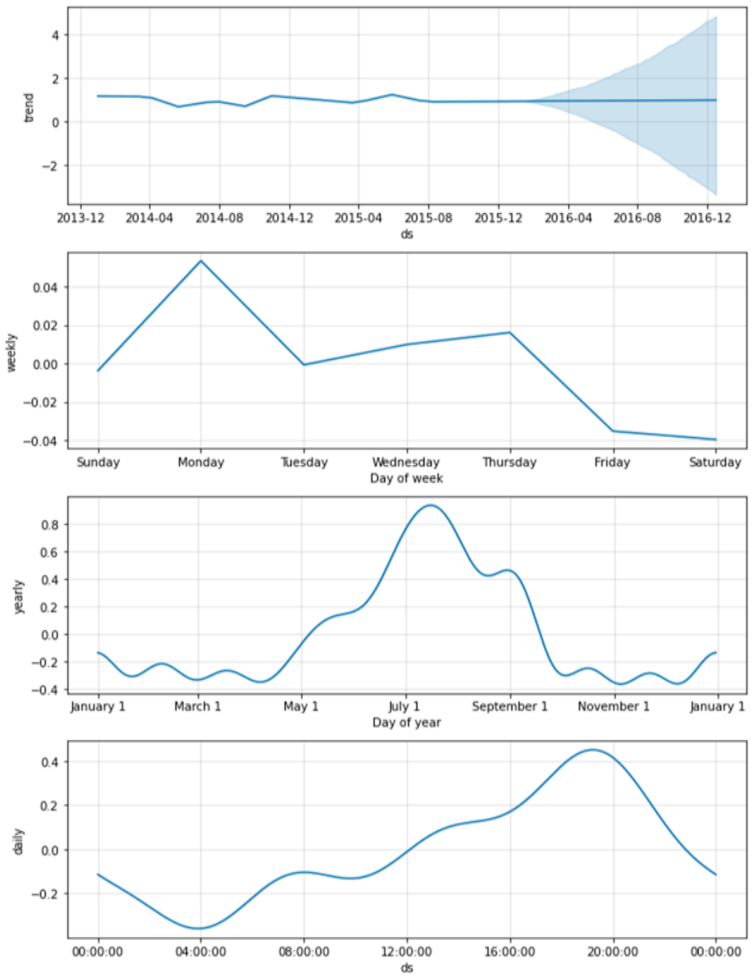



Figure 5.13: Charts (top to bottom): Trend, Weekly, Yearly and Daily-datasets used in the Prophet model

## 5.5 Conclusion and future scope

The forecasting of demand and supply of electrical energy is essential and critical in achieving efficiency and economy in the running of an SG. The time-series data available for the purpose in a real-life situation is, at times, discontinuous. The same has to be the basis for a forecast in the absence of alternatives. The ARIMA model is seen providing good forecast if data provided is complete. But the same is highly expensive as its computation requires computer systems with memory and computational speeds of a very high order. The Prophet model while offsetting all the above limitations of the

ARIMA model generates a reliable forecast even in the absence of a few values in the data. Therefore, it is felt that the Prophet model is preferable compared to ARIMA in a real time data.

Here, we used households datasets that consist of 10,00,000 records. Due to large datasets, we used ARIMA and Prophet for higher accuracy and lower error rate.

Further, due to the unavailability of the datasets, of different years, the performance of the Prophet can not be checked. However, it may give better result in next year prediction. In future works, we need to explore more time series prediction models in real time. Also, we need to explore optimization algorithms for parameters tuning of the models.

# Chapter 6

# Ensembling of Data Anomaly Detection Techniques in Energy Prediction

The rapid increase in various technologies which are interconnected with electrical energy networks generates huge data which is called Big Data (BD). These large data have created potential irregularities in energy consumption in Smart Grid (SG). Therefore, Anomaly Detection Techniques have a great role to detect abnormal data. Detecting abnormal data will help to increase large storage space for storing the time-series data. The varieties of data arise with respect to time and become a challenging work for data management. Data preprocessing is a basic part of data management in SG. In this chapter, several Anomaly Detection Techniques such as Isolation Forest (IForest), K-Nearest Neighbour (K-NN), Open Support Vector Machine (OSVM), Histogram, Local Outlier Factor (LOF), and Stochastic Outlier Selection (SOS) are used. All these techniques have been used by Long Short Term Memory (LSTM), Auto-Regressive Integrated Moving Average (ARIMA), and Prophet models. Moreover, these models are used with different Anomaly Detection Techniques which can be used for energy prediction. Further, the results are compared with different evaluation metrics for five different datasets. It is found that Anomaly Detection Techniques with LSTM, ARIMA, and Prophet models provided better performance.

## 6.1 Introduction

Due to latest improvement in industries and technologies in the SG the energy consumption has drastically increased. Therefore, there is a great need energy consumption prediction Anomaly Detection Techniques. Anomaly detection is an identification of rare experiments with extreme value that is totally different from other data points. It is the process of discovering incidence in a dataset which are dissimilar from the huge number of the data. This is used in various application province. This approach performs a heavily role in finding intrusion detection, misuse detection and behavioral analysis and traffic in the networking system. This application is used in Data Leakage prevention (DLP) system. Moreover, it has a vital role in the medical realm to detect vital functions of

patients.

We have various techniques such as IForest, K-Nearest Neighbour (KNN), Local Outlier Factor (LOF), Open Support Vector Machine (OSVM) and Stochastic Outlier Selection (SOS). Recently, anomaly detection techniques are emerging at a large level. IForest is very important Anomaly Detection Technique which is an unsupervised machine learning approach as it selects features randomly. Moreover, Khaledian *et. al* discussed IForest in realtime synchrophasor abnormality detection and classification [135]. The K-NN is supervised learning algorithm as it explains the variance in the data. OSVM is used in classification and regression method. SOS is unsupervised machine learning method as it takes input features matrix or dissimilarity matrix. The histogram is used for Anomaly Detection Technique where it shows the distribution. LOF is unsupervised Machine learning technique as it computes low density deviation. Feng et. al defined anomaly detection techniques such as KNN, SVM, Neural network and Bayesian network based on cloud network [136]. These methods work in the real-time for anomaly detection. We have enable in various anomaly detection techniques for forecasting energy consumption in the SG.

Development has made grid economically more strong and have better load distribution. Therefore, demand and supply can be manage in a proper way and hence price can be reduce. Asaleye *et. al* evaluated an hourly load profile which is effectively trained. Therefore, this model provides error-free and more accurate electric power system [114]. Here, they have used hybrid model of Bidirectional Long Short Term Memory (BI_LSTM) and ARIMA model for predicting more accurately by changing electrolytic capacitors. Wang et. al have defined ARIMA electrolytic capacitors which was not working properly. Therefore, it is used in linear and non-linear phase of time series data [115]. Gupta *et. al* explained about short term wind power prediction where ARIMA model is used for better performance [137]. Here, Luo *et. al* described issues with real-time price forecasting method as it performs better result by using various methods. J. M. Lujano *et. al* designed the optimized demand response for controlling proper demand response in SG [138]. Here, Kim *et. al* used backpropagation for controlling green house and predicted internal temperatures [139]. Ma highlighted the about review of power spatio temporal Big Data technologies, applications, and challenges [140].

### 6.1.1 Related Work

The anomaly detection techniques have great role in energy consumption prediction in maintaining the demand and supply gap in SG. It provides better solutions for power utility. Since, energy prediction is necessary part for managing energy consumption, there-

fore, it is desirable to use anomaly techniques. These techniques can handle challenges of Big Data by minimising the error between actual and predicted value.

Khaledian *et. al* explained synchroniser anomaly detection. Here, they have used classification (SyADC) tool for analyzed data where they categorize the data points. They selected the window of data points and defined combination of three unsupervised methods such as IForest, KMeans and LOOP [135]. Real-time observing and control of the SG used for the improvement of reliability and operational efficiency of power utilities is difficult. However, Mogha *et. al* evolved a real-time anomaly detection framework. These frameworks based on smart meter (SM) where data are collected at the consumes's domain as it is designed to detect the anomaly events at both lateral and customer levels. They defined hierarchical structure of the network [141]. Vinnikov et. al described about the same period the annual rain has been increased by 6 percent on the land in the 35 degree to 70 degree [142]. Hence, they designed mesh generation which is creating a mesh (subdivision of a continuous geometric space into discrete geometric and topological cells).

Barth *et. al* discussed about various types of unstructured mesh generation [143]. Moreover, they highlighted a supervised learning as well as analytical-based anomaly detection technique. Liu *et. al* analysed a Lambda system by using the in-memory distributed computing architecture in which they used spark and extension of Spark Streaming. Here, model refreshment, iterative detection are supported by the system and the evaluation is created empirically. The results are effective and the scalability of the proposed lambda [53].

Guha *et. al* discussed Title Insurance (TI), Robotic Process Automation (RPA) for used manual tasks in IT business but it has not taken any support of Artificial Intelligence (AI) and Machine Learning (ML) [144]. Since, increasing huge data the problems of detecting anomalies are increased, Deep learning is new advanced and fast technology. Therefore, Jindal et. al defined a hybrid model which is a combination of AD, using Auto encoders (AE) and a One-Class Support Vector Machine (OSVM) defined top down scheme which is based on Decision Tree (DT) and Support Vector Machine (SVM) [34]. They proposed combination of DT and SVM classifier analysis which is capable gathered data of electricity consumption. They discussed about the data mining algorithm which applied to power system and decision tree (DT). It is called classification and regression tree (CART).

Liu *et. al* explained the performance, such as computational efficiency, uncertainty manageability and interpretability [145]. They explained about the modelling trends in domestic energy consumption, which is based on density-based classifiers. They focused on

techniques which estimates the volume of outliers. The Density-Based Spatial Clustering of Applications with Noise (DBSCAN) and Ordering Points are defined. These techniques recognize the Clustering Structure (OPTICS) and Local Outlier Factor (LOF). Hurst et. al discussed about the detection of commonly occurring energy consumption within naturally creating groups with similar characteristics [56]. Moreover, they highlighted smart meter analytics from a software performance outlook. First, they designed a performance benchmark, they included common smart meter analytics tasks which included off line feature extraction. They detected anomalous points by making model and a framework for online. Secondly, they described an algorithm for generating large realistic datasets which are taken from a small data.

At last, Liu *et. al* implemented the proposed benchmark using five representative platforms [33]. Efstathopoulos *et. al* highlighted about the SG design modelling and communication [146]. Murrilo *et. al* defined the Jacobian-based Saliency Map attack [58]. Perales et. al discussed classification behaviours [147]. Here, Xu *et. al* explained diagnosing rare diseases or cancer subtypes by machine learning tools [148]. Here, they discussed about seven different outlier detection techniques. They recognized the occurrence of NTL in an Irish datasets. They have defined k-means clustering algorithm in the pre-processing step. Yeckle *et. al* defined about structured and comprehensive state-of-the art. It is based on outlier detection techniques [149].

These techniques are in the context of time series. Blazquez tried to characterize an outlier detection techniques [150]. Elmenshawy *et. al* highlighted an anomaly based Intrusion Detection System (IDS) [151]. It is designed for SG utilising operational data. This is taken from a real power plant. Here, many machine learning and deep learning models were deployed. They introduced novel parameters and feature extraction in this study. They have showed evaluation analysis for the efficacy of the proposed IDS. They defined about events, it is anomalous based on their patterns. Here, they provided to stakeholder both a visual representation of the candidate anomalies. The top-10 anomalies for a subset of Smart Meters has been represented. Osypova et. al defined Clustering Techniques for Non-technical Losses detection report [152]. They defined bloom filter which has a unique data structure and algorithms. This data structure is called the classical bloom filter. This recent technique is called probabilistic counting.

Therefore, it can efficiently generate histograms for flowing data in one pass with sublinear above up. So, this method is suitable for data processing in SG. However, where limited calculational resources are available on the selectors, they analyzed the presentation, back and forth, and capacity of this data structure. They evaluated it with the real data which is collected through the frequency disturbance recorders. It is estab-

lished using the FNET/GridEye architecture. This method can recognize the frequencies of all unique items with high accuracy. They identified low memory overhead therefore data outliers can be conveniently identified. Efficient Histogram Estimation for SG Data processing with the Loglog-Bloom-Filter. Yao et. al explained stakeholder both visual representation of the candidate anomalies. The top-10 anomalies for a subset of Smart Meters (SM) has been represented. Here, the authors defined clustering techniques for Non-technical losses detection report [153].

Aimal et. al discussed about generative model where abnormal data is detected. An efficient CNN and KNN data analytics are used load forecasting in the SG [154]. Goldstein et. al explained about Histogram-Based Outlier Score (HBOS) where unsupervised anomaly detection algorithm is used [55]. Yao et. al explained Histogram estimation for SG data processing is efficient with the Loglog-Bloom-Filter [153]. Anwar et. al highlighted an EMS's Optimal Power Flow (OPF) module can be exploited by accidental or deliberate changes in a power system model [57]. Tran et. al explained the usage of multiple methods in a systematic and ranking-based pattern which reduces in terms significant errors [155]. There are the possibilities to identify anomalies in model because of algorithm-related issues. Furthermore, Pereira et. al suggested genetic unsupervised and scalable framework for anomaly detection in time series data which is based on a variational recurrent auto encoder [156]. They defined about attention in the model about a variational self attention mechanism (VSAM). It is capable to provide better performance of the encoding-decoding process. Sharma *et. al* defined permutation and combination of four classification techniques, four feature weighting and four feature selection approaches [157]. Mookiah *et. al* highlighted about graph where data is converted into graph [54].

In Table 6.1 literature survey for anomaly detection techniques are defined. Here, authors discussed about various techniques such as Isolation Forest (IForest), K- Nearest Neighbor (K-NN), Stochastic Outlier Selection (SOS), OSVM and Histogram in their works.

### 6.1.2 Motivation

Energy prediction is a critical task in SG because its complexity and nonlinear nature with many datasets. Different anomaly detection techniques are adopted by the authors to reduce these complexities. But very limited works have been done to take huge datasets to check validity of their proposal. Massive use of classification and regression analysis is the great challenge during implementation, when large data are taken. The literature review shows that very little work has been done with respect to anomaly detection techniques for energy prediction. Anomaly detection techniques can not be handle by using conventional machine learning approach. Further, these technologies are involved

Table 6.1: Literature survey for Anomaly Detection Techniques

| SN | Authors [Ref,year] | Techniques | Description | Application/ domain |
|---|---|---|---|---|
| 1 | Newsham et. al (2010) | Isolation Forest [135] | IForest is an unsupervised learning algorithm for anomaly detection. It works on the principle of decision tree and isolating anomalies. | Anomaly detection |
| 2 | Aimal et. al (2019) | K-NN [154] | It is supervised machine learning approach. The Euclidean distance algorithm is used to calculate values to detect anomaly points. | Prediction of classification (Speech recognition) |
| 3 | Hurst et. al (2020) | LOF [56] | LOF is considered as outlier if its outlier based on local neighborhood. It will recognize an outlier which is considered the density of the neighborhood. LOF condemns well when the density of the data is not equal throughout the dataset. | Detect outliers |
| 4 | Sendric et. al (2019) | OSVM [34] | OSVM is a supervised machine learning model. It is used in classification algorithms for two-group classification issues. When SVM model is given to sets of labeled training data for each category.Then, they can categorized of new text. | Classification and Regression |
| 5 | Amelec et. al (2019) | SOS [158] | SOS takes as input either a feature matrix or alikeness matrix. The outputs for each data point is an anomalous probability. Instinctively, a data point is examined to be an outlier when the other data points have deficiency of sympathy with it. | Affinity based outlier selection |
| 6 | Yao et. al (2014) | Histogram [153, 55] | It helps in count of the number of observations in each bin which is created for visualization. We can easily observe the distribution i.e. weather it is Gaussian, skewed or exponential. Histogram helps in outliers detection. | outlier detection |

with large datasets. Some optimization tools are needed to find early convergence.

Further, energy prediction is one of the techniques to understand the proper utilisation of the energy resources and therefore, we need to analyse the Big Data (BD) and use it for load forecasting. Proper load forecasting may lead to reduce the demand and gap of electrical usage. It is regulated by structures which are called gates. Gates are the mode where information is optionally chosen. These gates are working with signed neural network and a point to point multiplication operation. There are mainly three types of gates such as input gate, output gate and forget gate. Further, load forecasting techniques involve large datasets and to get early convergence we need some optimization tools along with LSTM. Few authors have proposed different algorithm to develop load forecasting with big data but they have not analysed the results in terms of multi-threading approach of IForest-LSTM which improves the speed of the convergence. These isolation detections

lead to reduce the demand and gap of electrical usage.

### 6.1.3 Contribution

It is very useful to handle the non-linearity in the input and output data through many anomaly techniques. These anomaly techniques helps to reduce the abnormal data for energy prediction. The dataset is downloaded from the American Electric Power organization (AEPO) where PJM (Pennsylvania –New Jersey-Maryland) interconnection is introduced in 1956 [159].

The contribution of the proposed work is defined below.

1. Different Anomaly Detection Techniques are compared and tested as a preprocessing technique with LSTM, ARIMA and Prophet models and results analysed with different performance metrics

2. Different Anomaly Detection Techniques such as forest, K-NN, Histogram, SVM, SOS, and OSVM have been used and compared their preprocessing algorithm on different datasets

3. The novelty of the work lies in the preprocessing techniques on the LSTM, ARIMA and Prophet model where different Anomaly Detection Techniques have been compared

4. The novelty of the work lies in the preprocessing techniques in LSTM, ARIMA and Prophet model where different Anomaly Detection Techniques have been compared

Our major contributions are described as follows:

1. The accuracy of prediction of consumption of energy is improved

2. Reduce the execution time of algorithms

3. Enhance the classifier to increase the prediction accuracy

### 6.1.4 Organization

In this chapter, Section II describes the detail description of the methodology and workflow of our proposed model. Section III defines about the performance of the models along with evaluation parameters. After that, Section IV explains outline of the results and discussions. Finally, Section V discuss the conclusion of this work.

## 6.2 Material and Methods

### 6.2.1 Dataset and its description

In this subsection, we have used univariate time-series dataset. We acquired the datasets from Pencilum New Jersey Maryland Interconnection (PJM) [159] which is situated at United States. It resides in the Eastern Interconnection grid operating in electrical transmission system. It contributes services to the Delaware, New Jersey and North Carolina. We have used five datasets mainly North of Illions (NI), Common Wealth Edition (COMED), DAYTOM, American Energy of Power (AEP) and Dominian Virginia Power (DOM). The duration and explanation of the datasets is described in Table 6.2. Table 6.3 shows the sample of dataset of the North of Illions (NI). In this table, date, time and the amount of consumption of energy are discussed. Table 6.4 shows parameters of LSTM. In this table, various parameters such as Batch Size, Epochs, Data interval size, Activation Functions (Sigmoid, Tanh and Relu) and number of hidden layers of the LSTM model are discussed. Batch size is used in machine learning and it is the number of training. For example it is utilized in one iteration. This is the number which can be divided into the total dataset size. Activation function is a node that is put at the end or in between Neural Networks. It helps to decide if the neuron would fire or not. Table 6.5 shows parameters of Prophet Model. In this table, many parameters such as Trend, Seasonality and Holidays are defined. In Prophet Model, Trend is a pattern where values of data are in increasing or decreasing order over long time period. Time series data comprise seasonal variations and Seasonality are cycles which repeats it's data over time. Here, by default, Prophet fits weekly and yearly seasonality where the time series is long more than two cycle. All occurrences of the time series data are included during Holidays. Table 6.6 defines parameters of the ARIMA Model. In this table, many parameters such as Auto-Regressive, Integrated and Moving Average (p, d, q) are discussed. Fourier series is used to calculate values of p, d, q such as 4, 1, 1.

### 6.2.2 Anomaly Detection Techniques

Anomaly detection is a process to detect unexpected or abnormal data points in the system. It is an important approach to know fraud activities, suspicious activities, network intrusion and other abnormal occurrences. This may have more significance, but hard to detect. Therefore, various anomaly detection techniques are described below.

1. **IForest**: IForest is an unsupervised machine learning approach which is used for anomaly detection. It works on a forest of Isolation Trees (ITrees). It is based

Table 6.2: Description of the datasets

| SN | Dataset | Duration | Type | No. of rows | Description |
|---|---|---|---|---|---|
| 1 | NI [159] | 2004 to 2018 | hourly | 58451 | Northern Illinois hub (NI) measured energy consumption (hourly based) in Megawatts (MW) |
| 2 | COMED [159] | 2011 to 2018 | hourly | 66498 | Commonwealth Edition (COMED) is hourly based Energy Consumption data. Energy is estimated in (MW) |
| 3 | DAYTON [159] | 2004 to 2018 | hourly | 121276 | DAYTON is name of company of light and power (hourly based Energy Consumption) data. It is estimated consumed energy in Megawatt (MW). |
| 4 | AEP [159] | 2004 to 2018 | hourly | 121274 | American Energy of Power (AEP) is hourly based Energy Consumption data. The energy is estimated in Megawatt (MW). |
| 5 | DOM [159] | 2005 to 2018 | hourly | 116190 | Dominian Virginia Power belongs to (DOM) which is hourly based Energy Consumption data. The energy is estimated in Megawatt (MW) |

Table 6.3: Sample dataset of Northern Illinois (NI) hub

| Date | Time | Consumption of Energy (MW) |
|---|---|---|
| 2004-12-31 | 01:00:00 | 9810.0 |
| 2004-12-31 | 02:00:00 | 8509.0 |
| 2004-12-31 | 03:00:00 | 8509.0 |
| 2004-12-31 | 04:00:00 | 8278.0 |
| 2004-12-31 | 05:00:00 | 8089.0 |

Table 6.4: Parameters for LSTM model

| SN | Name Parameters | Values |
|---|---|---|
| 1 | Number of hidden layers | 3 |
| 2 | Data interval size | 30 |
| 3 | Epochs | 300 |
| 4 | Batch size | 16 |
| 5 | Activation Function | Relu |

on Decision Tree (DT) by selecting random features from the datasets which are given. Then, it chooses split value randomly between the maximum and minimum values which are the selected values. The following equation represents calculation

Table 6.5: Parameters for the Prophet model

| SN | Name Parameters | Values |
|---|---|---|
| 1 | Trend | 1.4 |
| 2 | Seasonality | 0.50 |
| 3 | Holiday | 0.05 |

Table 6.6: Parameters for the ARIMA model

| SN | Name Parameters | Values |
|---|---|---|
| 1 | p (AR) | 4 |
| 2 | q (Integrated) | 1 |
| 3 | d (differences) | 1 |

of anomaly scores for new points.

$$S(x, m) = 2^{-E(h(x))/c(m)} \tag{6.1}$$

where, x = data point m = Sample of data

$h(x)$ indicates average search hight for x from ITrees. $c(m)$ indicates average value of $h(x)$

if average value is 0, then $2^0 = 1$ which indicates anomalous point and if average value is 1 then $2^{-1} = 0.5$ which indicates regular point. This formula indicates what is the depth point and the particular point x across the trees which we have constructed, compared to the finding depth.

Steps of Training:

(a) Step 1: Building a forest of isolation trees (ITrees)

(b) Step 2: Take sample of dataset and build an ITrees until each point is isolated

(c) Step 3: Randomly select features

(d) Step 4: Randomly partition by using recursively partitioning (where number of splitting are required to separateness a sample which is equivalent to path length from the root node to the termination node)

The Figure 6.1 represents methodology of proposed work and Figure 6.2 shows the sample of forest of Isolation Trees (IT). The Figure 6.3 indicates procedure of IDT, where red colour data point shows anomalous point. Moreover, there are two

features such as x and y as well as four observations for checking anomalous point. The data point is spitted in four parts. The first condition is that the one decision signal which is normally results from an anomaly. If x is larger than 150, then the result is an outlier indicates in red colour. Green colour indicates normal points.

2. **K-NN**: The K-Nearest Neighbours (K-NN) algorithm is a supervised machine learning algorithm. It can solve both classification and regression problems. This method can be used to find the abnormal occurrences. In this method, a generative model is discussed for anomaly detection. Generative model can generate new data events. For any data point, the length to its Kth Nearest Neighbour might be viewed as the outlying score. Pod (Python package) supports three K-NN detectors: largest, mean and median, which is use as outlying score. Individually, the distance of the quiet neighbour, the average of all the K-NN and the median distance to K neighbours. It is the main concept that the same results should be close to each other in K-NN algorithm. The K-NN is used mathematically in terms of 'similarity' which could be translated as 'distance'. Here, we used the Euclidean distance algorithm to take some closest observations. We calculate the number of closest neighbours for each variable. The highest value decides the membership of the current notice. The following equation shows the calculation of the anomaly score for new points.

$$E(D) = \sqrt{(x2 - x1)^2 + (y2 - y1)^2} \tag{6.2}$$

Where E(D) is Euclidean distance and x1, x2, y1, and y2 are variables of data.

Algorithm of KNN

(a) Step 1. Input dataset

(b) Step 2. Trained data is loaded

(c) Step 3. Select from the characteristics

(d) Step 4. For each position in data: do

(e) Step 5. Discover Euclidean distance (ED)

(f) Step 6. ED is accumulated in sorted list

(g) Step 7. The initial k point is selected

(h) Step 8. Acknowledge results on the basis of majority of values

(i) Step 9. end for S(g-j + 1) - g* Where S presents Recursive Feature Elimination (RFE), j presents important calculation and g presents feature.

(j) Step 10

(k) End

3. **SOS**: Stochastic Outlier Selection (SOS) is unsupervised anomaly-selection machine learning algorithm. It takes feature matrix or dissimilar matrix as an input. This approach provides outlier probability to find output for each data point. A feature matrix is a set of features that transformed data into a dissimilarity matrix by using Euclidean distance. In SOS, a data point is considered to be an outlier when the other data points have insufficient affinity with it. Moreover, t-Distributed Stochastic Neighbor Embedding (t-SNE) method used to keep the local structure of a high-dimensional dataset [158]. It is used by SOS method to select outliers. The t-SNE is an unsupervised and non-linear method. It is used for data investigation and visualizing high-dimensional data. In another way, t-SNE arranged the data in a high-dimensional space.

4. **O**SVM: The Open Support Vector Machine is a supervised machine learning model. It has the capability to analyze data and identify designs. SVM approach is applied for both classification and regression works. In classification method the outcomes are represented in categories and in regression the outcomes presented in real numbers. Basically, the SVM algorithm is where given a set of training labeled data which belongs to one of two classes. The SVM model is dividing the training sample points into separate categories. This model predicts by recognizing points to one side of the gap to the other side. Moreover, over sampling is used to replicate the existing samples. Therefore, two-class model are created. It is not possible to forecast all the new patterns of abnormal data from limited examples. Therefore, a collection of limited examples can be costly. So, in one-class SVM, only one class data is trained which is known as "Normal class". It is very useful for anomaly detection. OSVM is a very important technique where the kernel function provides additional flexibility to the OSVM algorithm. It is called one-class classification (OCC) and known as unary classification. It is similar to unsupervised concept drift detection by using OSVM [144]. It learns a decision function for good performance of Anomaly Detection Techniques. It acquires knowledge of a decision function for nice performance. It separates the two classes by using a hyper plane with the largest possible margin. In OSVM, hyper sphere is surrounded by classes of instances, instead of hyper plane. Hyper sphere is set of points at a constant distance from a given point which is called its centre.

SVM algorithm steps:

(a) Step 1. Import the dataset

(b) Step 2. Explore the data to figure out what they look like

(c) Step 3. Pre-process the data

(d) Step 4. Split the data into attributes and labels

(e) Step 5. Divide the data into training and testing sets

(f) Step 6. Train the SVM algorithm

(g) Step 7. Make some predictions

5. **Histogram**: A histogram is a presentation of the distribution of the numerical data. In this technique, the data are binned and count for each bin. A histogram is an aggregated bar chart where several possible aggregation functions such as sum, average and count can be found. Histogram-Based outlier Score (HBOS) [55] is statistical unsupervised anomaly detection algorithm. This algorithm is far less costly as compared to nearest neighbour and clustering based outlier detection technique. HBOS works on arbitrary data which provides a standard fixed bin width histogram. An HBOS algorithm is presented, which scores recorded in linear time. It shows independent characteristics which make it much faster than multivariate approaches. It is capable to detect global outliers as reliable as by using the most recent algorithms, but it performs poorly on local outlier problems. Now, for each dimension d, an individual histogram has been calculated. Here, the height of every single bin represents a density measurement. If the maximum height is 1 then histograms is normalized. This provides surely in equal weight of every characteristic to the outlier score. Finally, the HBOS of every instance p is calculated by using the corresponding height of the bins where the instance is located. By using following equation, we calculate the value of every instance.

$$HBOS(q) = \sum_{i=1}^{d} log(1/hist_i(q)) \tag{6.3}$$

where q indicates every event and d represent dimension.

6. **LOF**: Local Outlier Factor (LOF) is an unsupervised Anomaly Detection Technique. It calculates the local density deviation in data points. The data points are measured with respect to its neighbors. If the sample which has a lower density than neighbours is considered as an anomaly or outlier. The Local outlier factor (LOF)

algorithm notices the relative density of data points and finds anomalous values by deliberating the local deviation of a data point after comparing with its neighbours [55]. In order to measure local density, LOF shares core-distance and reachability distance with DBSCAN. It can share Ordering points to identify the clustering structure (OPTICS) which is an algorithm for finding density-based clusters in spatial data. Its basic idea is similar to DBSCAN (Density Based spatial clustering of application with noise). Time-Pattern Profiling (TPF) approach performs very well to find an anomalous point from Smart Meter data in energy consumption [56]. In TPF the patterns profiler examines data values in any number of String attributes. It is capable to assign patterns as stated by the series of character types.

Algorithm: Anomaly Detection

(a) Function dataControl

(b) Pass In: datablock

(c) Function dataManagement

(d) Pass In: data block to accumulate

(e) Pass Out: dataset when finish for data

(f) Filtering and pre-processing

(g) End function

(h) Pass In: filtered data

(i) Function anomaly dectection

(j) Pass In: filtered data block

(k) Send to three classifiers

(l) FOR each classifier set anomaly threshold

(m) ENDFOR

(n) Pass Out: anomalyScores

(o) Endfunction

(p) Endfunction

(q) Function decisionComputation

(r) Pass In: Anomaly Scores

(s) Pass Out: evaluation of outliers

(t) IF > anomaly threshold THEN

(u) Log Time/User to report

(v) ELSE go to next time period

(w) ENDIF

(x) Endfunction

### 6.2.3 Prediction Models

1. **LSTM**:

Long Short-Term Memory (LSTM) is a part of deep learning model. It is a subset of machine learning which is related to artificial intelligence. This is the most prominent model for time series dataset for future prediction of consumption of energy. This model predicted energy with feedback connections in the network. LSTM has a memory which is capable to keep previous information in the system. However, this approach has the capability to solve the time series related problems. The main issue of the RNN is "Long term dependency" therefore to overcome theses problem LSTM is used. Here, the cell state is the key of the LSTM. LSTM has the capacity to add or remove the knowledge and monitor by the framework which are called gates. Gates are the way where knowledge is freely chosen. LSTM has three types of gates such as input gate, output gate and forget gate. These gates are working with activation function such as the sigmoid, Tanh and RELU. It is working with a point to point for multiplication operation. In this section, the machine learning model can help to provide better performance with assembling of different anomaly techniques. It can classify anomalies in time series data. In this subsection, the cell of LSTM contains various components such as forgetting gate F which determines the information should be discarded or kept in a candidate layer (C). It takes all the feasible values which is to be added to the cell state. An input gate, which is used to improve the cell state. The output gate O is used to know next hidden state. Further, H indicates the hidden state and C indicates the Cell state. These are vectors which used in this model. recent LSTM cell Is check out as the time step t. In the following equations $'\dot{*}'$ is an element-wise Multiplication and $'+'$ are an element-wise addition. First, the input and previous hidden state are passed through the forget gate of the LSTM cell which uses the sigmoid activation function. It uses the sigmoid activation function because it needs to decide, whether to

forget information or not. If 0 is close to forget then it is discarded, and the closer to 1 means to keep information.

There are different equations used in this model are defined below.

$$z_{jk}^{1} = \sigma(c_{k}^{1} + \sum_{n=1}^{N} x_{n.k}^{1} y_{j}^{0} + n - 1, k) \tag{6.4}$$

Here, $z_{jk}^{1}$ is dependent variable where weight of each layer is calculated with bias value $c_{k}^{1}$. In this equation, result of the vector is calculated by output vector of the previous layer. Further, $\sigma$ is an activation function, $y_{i}^{0}$ is independent variable, $x$ is weight of layers and $c$ is biased value which is associated with input layers respectively. The output layer is given as follows.

$$q_{jk}^{l} = max_{r \in S} z_{jk}^{l} \tag{6.5}$$

where, $q_{jk}^{l}$ is the maximum pooling layer of LSTM for storing time information. LSTM gives solution by consolidating memory units that can updates the hidden state. The input gate is given as per the below equation.

$$E_{i} = \sigma(W_{i} * T_{f} + G_{i-1} * V_{i}) \tag{6.6}$$

In this equation, $E_{i}$ and $W_{i}$ indicates as an input vector for forget gate. The $W_{i}$ represents as the weight vector for the forget gate. The $V_{i}$ represents input vector for candidate gate and G(i-1) is used as the earlier cell input or the invisible state. The new state of the LSTM is represented by following equation.

$$E_{o} = \sigma(W_{o} * T_{f} + G_{o-1} * V_{o}) \tag{6.7}$$

where, $E_{o}$ is an output vector at the time step of t. The $W_{o}$ is the output weight vector for forget gate. The $V_{o}$ an output vector for candidate gate and G(o-1) is used as the output or unseen state. The recent time step is mentioned as below.

$$i_{t} = \sigma(W_{p} iPt + W_{h} iP_{t} - 1 + W_{c} i * C_{t} - 1 + b_{i}) \tag{6.8}$$

where, $i_{t}$ is a input gate and the hidden states determine through the input gate, $i_{t}$ is input state at a time $t$, $C_{t}$ is cell state at time $t$ and $W_{c} i * C_{t}$ is weighted input.

Forget state is given as per the following equations .

$$f_t = \sigma\{W_p f^p t + W_h f_t^h - 1 + W_c f * c_t - 1 + b_f\} \tag{6.9}$$

Here, $f_t$ is the operation of forget gate at time $t$ which contains of LSTM and the output of each gate shows by $i$, $j$ and $o$, $W$ are the weight matrix of every gate unit corresponding. The $P_t$ is the crucial features of electric energy consumption. The output of the pooling layer at time $t$ and which is used as input of the memory cell. The $C_t$ is cell state at time $t$ and $b$ is benchmark of the forget gate. The operation of the output gate and cell state are given below.

$$p_t = g_t * d_t - 1 + j_t * \sigma\{X_q dqt + x_h c_t^h - 1 + b_d\} \tag{6.10}$$

$$d_t = g_t * d_t - 1 + j_t * \sigma(x_q dqt + x_i di_t - 1 + b_d) \tag{6.11}$$

The feature vector is one of the important parameters of the LSTM. It is mention as per the following equation.

$$i_t = p_t * \sigma\{d_t\} \tag{6.12}$$

Here, $i_t$ is a feature vectors. The last layer of the feature vector is mentioned as follows.

$$e_j^m = \sum_k x_k j^m - 1(\sigma(i_j^m - 1 + c_j^m - 1) \tag{6.13}$$

Last layer $e_j^m$ is fully connected layers. Here, we use LSTM to predict energy consumption in 60 minutes. $\sigma$ is an non-linear activation. $x$ is weight of the $j^{th}$ node for layer $(m-1)$ and the $k^{th}$ node for layer $k$, and $c_i^{m-1}$. The bias is determined by $k^{th}$ node for layer $m$ and $c_{1_i}^l$.

[*] = Element -wise multiplication

[+] = Element-wise addition

$$G_t = \sigma(Y_t * V_f + I_t - 1 * X_f) \tag{6.14}$$

Here, $G_t$ is forget gate of LSTM, $Y_t$ is input vector and $X_f$ is input vector. The $V_f$ and $X_f$ are represented the weight vectors for the forget gate and candidate gate. The $I_t - 1$ is the former cell output.

$$\overline{d}_t = tanh(Y_t * V_c + I_{t-1} * X_t) \tag{6.15}$$

Where, $d_t$ is the current memory state at time step t. Tanh is an activation function of the candidate layer, $Y_t$ is input vector, $V_c$ is weight vector, $I_t - 1$ is previous output and $X_t$ is weight vector.

$$J_t = \sigma(Y_t * V_i + I_{t-1} * X_i) \tag{6.16}$$

Here, $J_t$ is input gate at time step t, $V_i$ and $X_i$ are weight vectors for input gate and output gate.

$$p_t = \sigma(Y_t * V_o + I_{t-1}) * X_O) \tag{6.17}$$

Where $p_t$ is an output gate at time step t, $Y_t$ is input vector, $V_o$ and $X_o$ are weight vector of output gate and candidate gate.

$$d_t = g_t * d_{t-1} + l_t * \overline{d}_t \tag{6.18}$$

Where $d_t$ = Current Cell Memory, $g_t$ is an forget gate vector,

$$G_t = p_t * tanh(d_t) \tag{6.19}$$

Here, $G_{t-1}$ is Previous Cell Output, $d_{t-1}$ is Previous Cell Memory $p_t$ is Current Cell Output.

2. **ARIMA Model**:

It is a generalised form of Auto-Regressive Integrated Moving Average which is used to predict points of time series data. These two approaches are fitted in time series data and make data understandable. It works on lagged observations for forecasting current results. Moreover, previous knowledge is necessary for future prediction of energy. It makes series stationary by using different orders of different process. The ARIMA model defines data by using time series data for future prediction. This model is used for both linear regression as well as multiple regression. Multiple

regression model describes forecasting the outcome of dependent variables which are based on variables of independent variables. The main aim of the ARIMA model is for forecasting future values of the Time Series data. The model is discussed about ARIMA (q, e, r) where q, e and r are non-negative (where number is zero or positive number) mathematical values. Here, q indicates order of the AR term e indicates order of MA term and r indicates difference which is required to build time series stationary from non- stationary. We seek the values of q, e and r using auto-ARIMA. The auto-ARIMA method gets to know the most quality parameters for an ARIMA model resolving on a sign.

$$A_t = b + \in_t + \sum_{i=1}^{r} \alpha_i X_{t-i} + \sum_{k=1}^{r} \beta_k \in (t-k) \tag{6.20}$$

where $A_t$ indicates the consumption of energy at time t, b indicates obstruction of the signal at r

previous point $z_t - i$ with linear co-efficient $\alpha_i$. The ARIMA handles daily stationary signals that has no meaning of constant. It assume Gaussian noise such as $\in_t$ compounds over r time periods. Here, It is contributed linearly to the signal t-k with co-efficient $\beta_k|$.

$$f(W) = W^2 + 2W + 1 \tag{6.21}$$

In this equation, $f(W)$ is dependent variable and it indicates the prediction of the energy consumption in time series data. Here, $W^2$, $2W$ are independent variables and defines first order differentiation for making stationary to non-stationary data.

In another equation the prediction of the energy consumption in time series data is described following:

$$V_t 1 = \alpha_0 - \psi_1 v_t 1 - 1 - \psi_2 v_t 1 - 2 - ... - \psi_v n_t 1 - p + \epsilon$$
$$-\alpha 1 \epsilon (t1 - 1) - \alpha 2 \epsilon_( t1 - 2) - ... - \alpha \epsilon t1 - q$$

Where at time t1, $v_t 1$ and $\epsilon$ are predicted value and random error of data $\psi(v_t 1 - 1)...p$ indicates model parameter, $\alpha_1 ... r$ indicates model parameter q and r are represented autoregressive and moving average orders equation (3) where it shows some important cases of the ARIMA models, If r2 = 0, then equation (3) becomes

an AR model of order q2, When r2 = 0, the model is decreases to an MA model works with order r2, Auto-Regressive Integrated Moving Average (ARIMA) is a model where prediction of energy is based on past data.

3. **Prophet model**:

The main ambition of the Facebook's Prophet package is to give a simple, automated technique to predict huge number of various time series data. It is an Additive Model (AD) which is a non parametric regression model. This is developed by the Facebook for future prediction of data. This model predicts non-linear trends which are fitted with yearly, weekly, daily seasonality and holiday effects. The main objective of this technique is time Series analysis for prediction of power in Smart Grid. There are three devices used for categorization of power, mainly Interruptible devices, Non-interruptible devices and Base devices [160]. These following equations indicate consumption of energy with time series data method.

$$G_j b = \sum_{t-1}^{v} \sum_{ineIN} \sigma_j o * rw_n(t) \tag{6.22}$$

$G_j b$ represent total energy consumption

ineIN presents Interruptible Appliance

$\sigma_j(n)$ indicates Power rating U is total time slot $rw_j(t)$ is the state of each Interruptible Appliances at time slot

$rw_o$ is each base appliance by $\sigma_n$ is power rating.

$$A(t) = X(t) + T(t) + I(t) + F(t) \tag{6.23}$$

$A(t)$ indicates consumption of time series data method $X(t)$ represents the data trend function $T(t)$ indicates the seasonal based data $I(t10)$ indicates the holidays based data $F(t10)$ represents the error data

Prophet trend function, i(t11) highlighted by a piecewise linear growth model. It is called a saturating growth model. The maximum load data do not show a saturating growth which is a piecewise linear growth model which is represented as following:

$$i_t = (m + a(t)^T \delta)t + (o + a(t)^U \sigma)t \tag{6.24}$$

Here, m is growth rate $\delta$ indicates adjustment rate o is offset parameter $\sigma$ is change-points

$$b_j t = \begin{cases} 1 \, if \; t \geq U_k \; 0 \; otherwise \end{cases} \tag{6.25}$$

$$rw_n i_t = \sum 0 \, if \; appliances \; is \; off \; 1 \, if \; appliances \; is \; on \tag{6.26}$$

Base appliances is like fixed appliances which are unable to manage, the pattern of consumption of energy and all operations period of appliances is unable to changed. It is important for appliances that which must be on when user wants to start ON such home appliances TV, Freeze and another devices.

The seasonality function is manifasted by following equations:

$$U(t) = \sum_{m=1}^{M} (c_n cos(2/pin/q) + d_n sin(/pin/q) \tag{6.27}$$

In this equation, $U(t)$ is or the seasonality function. Here, the time series multi-period seasonality method is used. The Fourier series is applied for the daily, seasonality. Therefore, the seasonality function is discussed as:

$$B_t = [1(t \in F1), ..., 1(t \in D_m)] \tag{6.28}$$

Here, $B_U$ indicates matrix of regressors F indicates holiday K shows the holiday parameter

$$I_t = B(t)m \tag{6.29}$$

In this equation, $I_t$ indicates holiday m indicates list of holiday.

## 6.3 Methodology Used

In this section, the brief discussion of the methodology is discussed and it is represented in Figure 6.1. In this work, the energy consumption dataset is calculated from PJM [154]. A brief description of the dataset is explained in Section 6.2.1. To remove abnormal data, we applied different Anomaly Detection Techniques such as forest, K-NN, Histogram, SOS, OSVM and PCA. All these techniques are discussed in Section 6.2.2. Here, to improve the

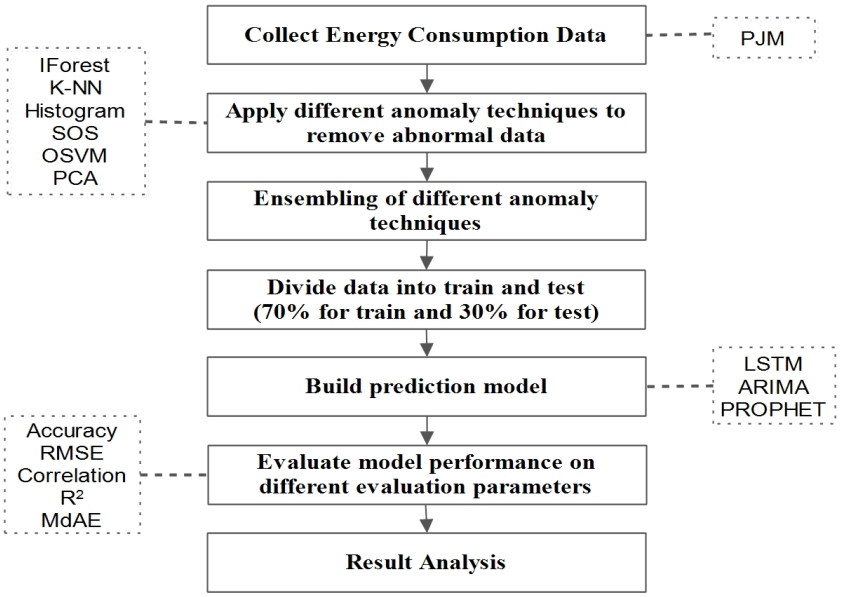



Figure 6.1: Methodology of proposed work

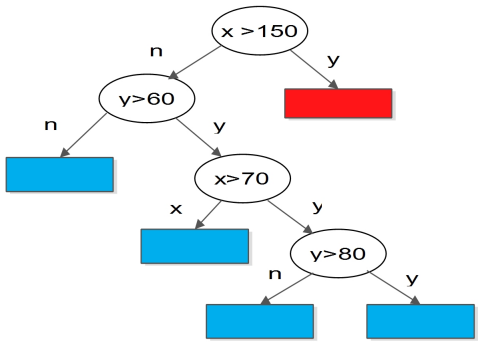



Figure 6.2: Sample of the Forest of Isolation Tree (IT)

performance of the prediction model, we used an anomaly detection techniques. Further, we divided the data into training (70%) and testing (30%). For prediction of consumption of energy we used LSTM, Prophet and ARIMA models. Afterwards, we evaluate the performance on different parameters such as, accuracy, Mean Square Errors (MSE), MaAE, correlation $r$ and correlation of coefficient $R^2$.

The Figure 6.4 shows prediction of weekly energy consumption of NI dataset. The Figure 6.5 shows prediction of weekly energy consumption of COMED dataset. The Figure 6.6 shows prediction of weekly energy of DAYTON dataset. The Figure 6.8 shows prediction of weekly energy of DOM dataset. The Figure 6.7 shows prediction of weekly energy of DEOK dataset.

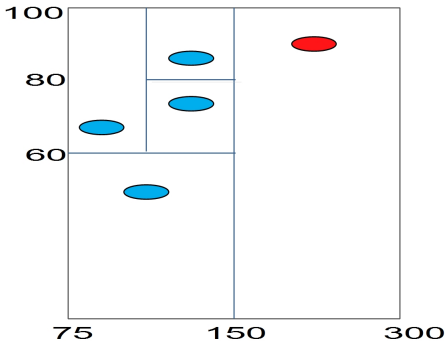



Figure 6.3: Procedure of the Forest of the Isolation Tree (ITree)

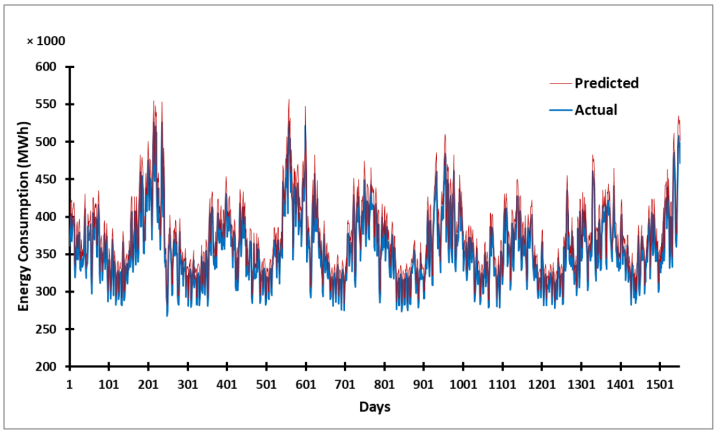



Figure 6.4: Actual v/s prediction of weekly energy consumption

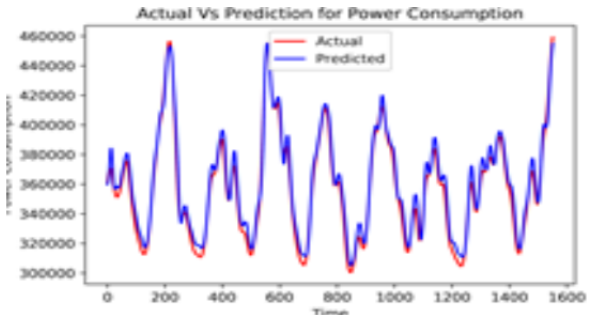



Figure 6.5: Actual v/s prediction of weekly energy consumption

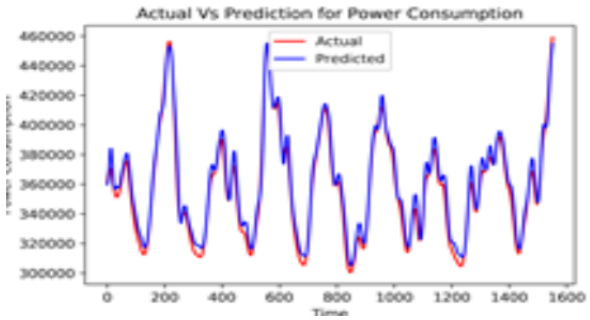



Figure 6.6: Actual v/s prediction of weekly energy consumption

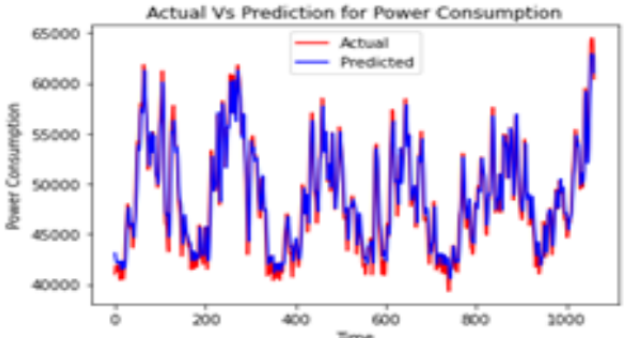



Figure 6.7: Actual v/s prediction of weekly energy consumption

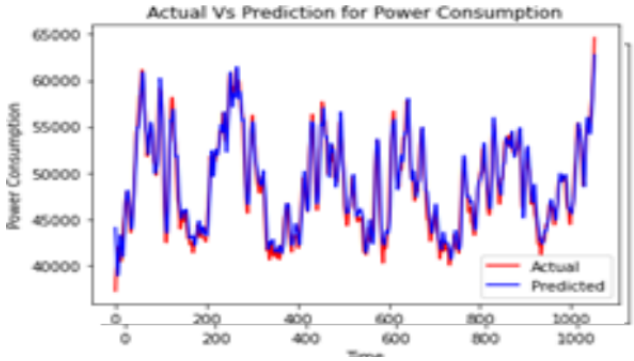



Figure 6.8: Actual v/s prediction of weekly energy consumption

## 6.4 Results and discussion

### 6.4.1 Evaluation Parameters

1. **MSE**: Mean Square Error (MSE) is an error estimation where the average of the squares is measured errors. Here, average square provides the difference between the predicted value and the actual value. RMSE is given as follows.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}|b_i - s_i| \tag{6.30}$$

where $b_i$ is predicted value and $s_i$ actual value.

2. **MaAE**: The Median Absolute Error is very important due to its robust nature to tackle outliers. Here, the loss of data is calculated by taking the median of all absolute differences between the actual and the predicted value.

$$MDAE(b, q) = median(|b_1 - q_1|, ..., |a_n - q_n|) \tag{6.31}$$

Where $b_1$ is the actual value and $q_1$ is predicted value.

3. $R^2$: The coefficient of determination $R^2$ represents the explanatory power of the regression model and calculated from the sums-of-squares terms which is following defined.

$$R^2 = r \cdot r \tag{6.32}$$

Where $R^2$ lies in [0, 1] range and defined to be good $R^2$, if value influences towards 1 the below equation.

4. **Co-relation (r)** : Co-relation describes the relationship between masses of variables. It measures errors between actual and predicted values. It is defined as follows:

$$s = \frac{\sum_{j=1}^{n}(b_j - \bar{b})(q_j - \bar{q})}{\sqrt{\sum_{j=1}^{n}(b_j - \bar{b})^2 \sum_{j=1}^{n}(q_i - \bar{q})^2}} \tag{6.33}$$

where, $b$ is the actual value, $q$ is the predicted value, $\bar{b}$ is the mean of all actual values, $\bar{q}$ is the mean of all predicted values and $n$ is the number of instances. Correlation lies in [-1,1] and considered to be good correlations if its value tends towards 1 or -1.

5. **Mean Absolute Error (MAE)** :- MAE is relationship between two variables such as $S$ and $T$ which measures the errors. The results is explained about the same incident. Moreover, where $T$ versus $S$ comparisons of predicted versus actual value of variables, it is discussed as per the following equation.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|S_i - T_i| \tag{6.34}$$

Where $T_i$ is actual and $S_i$ is predicted values.

### 6.4.2 Result Summary

We found the duration of the dataset in NI from 2004 to 2018. In COMED dataset, the duration is 2011 to 2018. In, DAYTON Dataset, the duration is 2004 to 2018. In DEOK dataset, the Duration is 2004 to 2018. In DOM dataset, the duration is 2005 to 2018. Here, all datasets are time series hourly based Power consumption dataset. These data are converted into daily Based dataset. The performance of comparative results is used

Xeon Processor with 64 GB RAM (20 core) and 1TB SSD for increase the speed of the simulation. For better validation of the results we have used various anomaly detection techniques with LSTM, ARIMA and Prophet. The main aim of using these datasets is to find more accurate results. The data is converted into seventy and thirty percent for training and testing respectively. It is to verify the compared datasets by using various anomaly detection techniques for the large dataset. Moreover, we found more accurate results in AEP, COMED, DAYTON, DEOK and NI dataset with IForest approach.

In this section, comparative performance of LSTM, Prophet and ARIMA model with different data anomaly detection techniques are discussed. LSTM-IForest, ARIMA-IForest and Prophet-IForest provide optimized results. Figure 6.4 shows the energy consumption of actual versus the predicted consumption of energy. The energy consumption curve is given in MWh because the PJM encloses larger space of the USA. It shows that predicted energy of the PJM which is very near to the actual energy. This forecasting will help to schedule the generating units of PJM.

### 6.4.3 K-fold validation

Figure 6.4 shows the comparison between various anomaly detection techniques with LSTM, ARIMA, and Prophet on datasets. Here, the variation of K-fold validation of various data anomaly techniques with LSTM, ARIMA and Prophet models are shown. Isolated Forest (IForest) provides more better performance. Therefore, these Anomaly Detection Techniques are applied with LSTM, ARIMA and Prophet's model. It provides K-Fold cross validation more accurately. This is used in datasets which are split into a K number of sections/folds. Here, each fold is used as a testing site at some point. The scenario of 10-Fold cross validation (K=10) is used. This process is repeated until each fold of the 10 folds have been used as the testing sets. Our dataset is divided into two parts such as 70 percent for training and the duration of data such as (2006 to 2019, 2011 to 2018, 2004 to 2018, 2012 to 2018 and 2005 to 2018). We have tested 30 percent data from (2016 to 2019, 2015 to 2018, 2015 to 2018, 2015 to 2018 and 2015 to 2018).

## 6.5 Conclusion

From now on, we proposed the comparative performance of analysis of various anomaly detection techniques with LSTM, ARIMA and Prophet for forecast of the energy consumption. Here, the novelty is to eliminate abnormal data from the signals. Here, we develop accuracy by using several techniques for data. By using these approach demands and supply can be managed. The performance analysis compared with various evaluation

Table 6.7: Comparative performance study of LSTM model with different anomaly detection techniques on different datasets

| **Dataset** | | **LSTM** | **LSTM+ IForest** | **LSTM+ KNN** | **LSTM+ SOS** | **LSTM+ Histogram** | **LSTM+ OSVM** |
|---|---|---|---|---|---|---|---|
| NI | MSE | 11676.895 | 11978.564 | 25346.534 | 34278.95 | 23478.657 | 23457.89 |
| | MAE | 67845.453 | 23456.784 | 23456.784 | 34567.897 | 45672.832 | 56397.846 |
| | MaAE | 56789.945 | 89743.652 | 74322.907 | 67890.543 | 56789.054 | 56732.89 |
| | $R^2$ | 0.894 | 0.456 | 0.84 | 0.853 | 0.763 | 0.785 |
| | $r$ | 0.893 | 0.789 | 0.564 | 0.845 | 0.675 | 0.86 |
| | Accuracy | 78.911 | 99.89 | 78.922 | 56.543 | 97.789 | 56.879 |
| COMED | MSE | 97778.191 | 11123.333 | 30238.543 | 56457.789 | 13567.435 | 12345.347 |
| | MAE | 23567.562 | 2567.907 | 98567.89 | 78625.906 | 85463.75 | 34582.89 |
| | MaAE | 57832.67 | 23453.672 | 567239.456 | 34567.98 | 23577.896 | 34568.706 |
| | $R^2$ 0.634 | 0.989 | 0.763 | 0.875 | 0.789 | 0.894 | 0.563 |
| | $r$ | 0.253 | 0.354 | 0.243 | 0.232 | 0.345 | 0.225 |
| | Accuracy | 80.634 | 98.222 | 87.324 | 78.965 | 75.453 | 77.899 |
| DAYTON | MSE | 11567.84 | 11678.895 | 11967.908 | 116532.786 | 17856.789 | 18567.876 |
| | MAE | 55463.865 | 45689.621 | 5698.789 | 34567.874 | 6678.456 | 67867.99 |
| | MaAE | 16945.943 | 18765.896 | 5678.678 | 38845.67 | 45678.784 | 87435.832 |
| | $R^2$ | 0.896 | 0.975 | 0.678 | 0.934 | 0.878 | 0.654 |
| | $r$ | 0.286 | 0.375 | 0.286 | 0.386 | 0.756 | 0.343 |
| | Accuracy | 56.98 | 97.89 | 68.96 | 89.907 | 76.453 | 67.89 |
| AEP | MSE | 69646.402 | 24237.472 | 34209.683 | 678956.309 | 11976.15 | 45442.673 |
| | MAE | 66276.149 | 69564.234 | 45368.105 | 64996.706 | 27047.724 | 15856.477 |
| | MaAE | 57409.93 | 30216.485 | 37898.951 | 52667.609 | 52669.609 | 12477.132 |
| | $R^2$ | 0.777 | 0.716 | 0.896 | 0.874 | 0.938 | 0.982 |
| | $r$ | 0.209 | 0.245 | 0.368 | 0.285 | 0.378 | 0.347 |
| | Accuracy | 86.678 | 97.654 | 79.345 | 88.523 | 95.345 | 77.789 |
| DOM | MSE | 83609.307 | 39978.407 | 17929.712 | 97087.297 | 16927.156 | 81399.759 |
| | MAE | 26719.569 | 97628.586 | 87688.668 | 45261.594 | 56769.005 | 60224.784 |
| | MaAE | 16812.969 | 98729.383 | 92285.668 | 18261.594 | 81468.005 | 50128.784 |
| | $R^2$ | 0.79 | 0.87 | 0.928 | 0.842 | 0.94 | 0.872 |
| | $r$ | 0.268 | 0.32 | 0.363 | 0.236 | 0.369 | 0.286 |
| | Accuracy | 68.114 | 98.676 | 88.231 | 69.67 | 66.802 | 85.164 |

parameters such as Mean Square Error MSE, Mean Absolute Error MAE and Median Absolute Error MaAE, Co-efficient of determination ($R^2$), co-relation ($r$) and accuracy. In future, these comparative studies will improve by using the latest algorithm for forecast of energy in SG.

Table 6.8: Comparative performance study of ARIMA model with different anomaly detection techniques on different datasets

| **Dataset** | | **ARIMA** | **ARIMA+ IForest** | **ARIMA+ KNN** | **ARIMA+ SOS** | **ARIMA+ Histogram** | **ARIMA+ OSVM** |
|---|---|---|---|---|---|---|---|
| NI | MSE | 12018.972 | 30272.046 | 21756.946 | 11838.412 | 13643.194 | 30272.046 |
| | MAE | 26002.621 | 47275.946 | 11790.755 | 26527.782 | 9503.593 | 4627.9461 |
| | MaAE | 21370.734 | 13999.186 | 46268.864 | 99443.215 | 85286.298 | 42504.8677 |
| | $R^2$ | 0.477 | 0.87 | 0.987 | 0.883 | 0.937 | 0.9886 |
| | $r$ | 0.291 | 0.365 | 0.336 | 0.295 | 0.368 | 0.344 |
| | Accuracy | 66.688 | 98.394 | 78.324 | 70.425 | 79.716 | 77.345 |
| COMED | MSE | 97778.191 | 58.333 | 66.381 | 76.492 | 56.922 | 27968.751 |
| | MAE | 24071.011 | 15158.049 | 13245.666 | 24035.648 | 89637.82 | 40426.459 |
| | MaAE | 19021.875 | 12437.81 | 10191.084 | 18417.703 | 68618.67 | 30906.79 |
| | $R^2$ | 0.973 | 0.447 | 0.826 | 0.871 | 0.989 | 0.976 |
| | $r$ | 0.311 | 0.216 | 0.352 | 0.209 | 0.343 | 0.388 |
| | Accuracy | 69.485 | 98.495 | 66.495 | 63.505 | 75.67 | 69.144 |
| DAYTON | MSE | 28445.255 | 12480.747 | 32796.482 | 26986.188 | 26986.188 | 52179.112 |
| | MAE | 42545.614 | 28676.85 | 14463.388 | 41385.032 | 10583.411 | 5506.411 |
| | MaAE | 36267.457 | 25705.551 | 12297.331 | 35773.477 | 90145.683 | 44745.424 |
| | $R^2$ | 0.966 | 0.112 | 0.993 | 0.865 | 0.837 | 0.781 |
| | $r$ | 0.305 | 0.334 | 0.315 | 0.204 | 0.368 | 0.391 |
| | Accuracy | 84.587 | 97.715 | 78.243 | 75.821 | 99.122 | 89.245 |
| DEOK | MSE | 67646.402 | 23237.472 | 32209.683 | 67956.309 | 11876.15 | 40442.673 |
| | MAE | 66273.149 | 37124.518 | 45366.105 | 64994.706 | 27045.724 | 15856.477 |
| | MaAE | 57404.93 | 30213.485 | 37891.951 | 52663.609 | 52665.609 | 12475.132 |
| | $R^2$ | 0.427 | 0.903 | 0.852 | 0.428 | 0.965 | 0.957 |
| | $r$ | 0.268 | 0.32 | 0.363 | 0.236 | 0.369 | 0.386 |
| | Accuracy | 80.455 | 97.895 | 94.754 | 85.193 | 78.779 | 89.632 |
| DOM | MSE | 83606.307 | 30902.407 | 14429.712 | 96082.297 | 16026.156 | 71995.759 |
| | MAE | 18813.969 | 97722.386 | 72286.768 | 19261.494 | 75466.115 | 59124.683 |
| | MaAE | 16812.969 | 99726.383 | 82287.668 | 172614.594 | 71464.005 | 50122.784 |
| | $R^2$ | 0.99 | 0.87 | 0.928 | 0.542 | 0.94 | 0.972 |
| | $r$ | 0.337 | 0.42 | 0.363 | 0.236 | 0.369 | 0.386 |
| | Accuracy | 67.114 | 97.676 | 88.231 | 61.67 | 34.345 | 82.456 |

Table 6.9: Comparative performance study of Prophet model with different anomaly detection techniques on different datasets

| **Datasets** | | **Prophet** | **Prophet+ IForest** | **Prophet+ KNN** | **Prophet+ SOS** | **Prophet+ Histogram** | **Prophet+ OSVM** |
|---|---|---|---|---|---|---|---|
| NI | MSE | 22018.972 | 55473.108 | 31656.946 | 22438.412 | 23643.194 | 50272.046 |
| | MAE | 36002.621 | 17696.538 | 12790.755 | 27527.782 | 85035.593 | 46279.9461 |
| | MaAE | 21373.734 | 14999.186 | 88556.215 | 99443.215 | 85287.298 | 42500.867 |
| | $R^2$ | 0.987 | 0.803 | 0.887 | 0.583 | 0.877 | 0.778 |
| | $r$ | 0.391 | 0.465 | 0.336 | 0.395 | 0.368 | 0.368 |
| | Accuracy | 77.688 | 98.394 | 83.227 | 70.425 | 84.716 | 67.334 |
| COMED | MSE | 95773.191 | 40221.333 | 50913.381 | 88419.492 | 15732.922 | 37968.751 |
| | MAE | 24072.011 | 16158.049 | 13245.666 | 24035.648 | 98635.820 | 50423.459 |
| | MaAE | 26021.875 | 15437.810 | 15246.084 | 19417.703 | 79617.67 | 40906.79 |
| | $R^2$ | 0.973 | 0.872 | 0.926 | 0.871 | 0.889 | 0.976 |
| | $r$ | 0.311 | 0.316 | 0.352 | 0.209 | 0.343 | 0.988 |
| | Accuracy | 69.485 | 98.495 | 66.495 | 63.505 | 75.67 | 64.144 |
| DAYTON | MSE | 28446.266 | 13480.747 | 42756.482 | 28986.188 | 29986.188 | 62179.112 |
| | MAE | 22584.614 | 28675.800 | 14466.388 | 41387.032 | 10583.411 | 55043.411 |
| | MaAE | 36267.457 | 25700.551 | 12293.331 | 35776.477 | 90156.683 | 44765.424 |
| | $R^2$ | 0.966 | 0.112 | 0.993 | 0.865 | 0.937 | 0.981 |
| | $r$ | 0.305 | 0.434 | 0.345 | 0.204 | 0.968 | 0.391 |
| | Accuracy | 84.587 | 97.715 | 87.224 | 85.821 | 78.233 | 88.123 |
| DEOK | MSE | 77646.402 | 24237.472 | 42209.683 | 77956.309 | 22856.15 | 50442.673 |
| | MAE | 66276.149 | 37121.518 | 45367.105 | 64998.706 | 28046.724 | 15857.477 |
| | MaAE | 57407.93 | 30219.485 | 37896.951 | 52667.609 | 52668.609 | 12478.132 |
| | $R^2$ | 0.627 | 0.678 | 0.652 | 0.428 | 0.865 | 0.957 |
| | $r$ | 0.268 | 0.32 | 0.263 | 0.236 | 0.369 | 0.386 |
| | Accuracy | 80.055 | 98.895 | 79.754 | 78.193 | 67.779 | 76.442 |
| DOM | MSE | 73606.307 | 40902.407 | 15529.712 | 97082.297 | 17027.156 | 81399.759 |
| | MAE | 15812.969 | 99721.383 | 72286.668 | 142616.594 | 61466.005 | 60120.784 |
| | MaAE | 84353.232 | 29354.666 | 15578.897 | 55557.898 | 43267.893 | 56488.89 |
| | $R^2$ | 0.456 | 0.78 | 0.93 | 0.63 | 0.78 | 0.65 |
| | $r$ | 0.265 | 0.36 | 0.367 | 0.267 | 0.786 | 0.267 |
| | Accuracy | 79.234 | 98.165 | 78.789 | 67.785 | 78.87 | 78.567 |

# Chapter 7

# Conclusion and Future Directions

## 7.1 Conclusion

These former studies have manifested current advances in the evolution of Big Data Analytic (BDA) where introduction of optimization in demand response management application is fulfilled. The prediction of energy consumption is based on real data which is found from PJM. There are two approaches such as random and GA are used for improvement of accuracy of the results. For increasing speed of convergence multi-threaded $GA - LSTM$ is used. After comparing it is found that $GA - LSTM$ provides better results as compared to random approach.

Many filtering techniques are comparing with LSTM model and Filtfilt filtering provides better performance. The main ambition is to removes noises from the signals and non-relevant added signals from the devices for managing demand response in SG. The smoothness of large data is main purpose. Various evaluation parameters such as MSE, MAE, MaAE, Co-efficient of determination, co-relation and accuracy are used for better performance. Moreover, output from various filtering techniques can be fed to the LSTM network. It is seen that better results are found which performs effectively. This study defines the comparative performance study of the LSTM model with many filtering technology on various datasets. The evaluation matrix MAE provides a minimum value of 4627.94 after comparing other filtering techniques on datasets in the dataset of AEP. So, the LSTM + Filtfilt filtering technique leads to minimum error.

Many time series models such as ARIMA and Prophet models are used for prediction of consumption of energy. These models creates confidence interval which is the range of measurement of unknown parameters. It describes interval with lower and upper levels. The Prophet model provides more accurate results than ARIMA model. It is more reliable than ARIMA model. The many factors of the Prophet model included with weekly, yearly and daily factors and the tendency exhibited by the predicted values. The values are predicted with the original 2016 data. These are compared for both the ARIMA and the Prophet model. In the mean square error of Prophet model is 0.67546 and the mean absolute error is 0.5308. Similarly, the values are 1.06877 and 0.6239 respectively

in ARIMA model. The root mean square error and mean absolute percentage error of Prophet model are less than those of ARIMA model. Therefore, after comparing the two models it is seen that the DR from Prophet model provides better performance.

Ensemble of data Anomaly Detection Techniques such as K-NN, OSVM,LOF, IForest and Histogram are used for the Prediction of energy consumption in SG. These techniques are used by LSTM, ARIMA and Prophet models. The results are compared with various evaluation parameters with five many datasets. Here, LSTM model provides better result with COMED dataset. ARIMA model shows better performance with DAYTON model. LSTM model performs better result with AEP dataset. The Prophet model performs better result with DOM dataset.

## 7.2 Scope for future work

Since, the proposed methods showed good performance as compared to the latest techniques, but it is always needed a scope for development. This part completed for the future supervision with respect to the proposed work. In the future, we would enlarge the proposed work regarding various characteristics as described as follows. For example, the plan proposed for prediction of consumption of energy using random and GA-LSTM algorithms which can be upgrade by using recent algorithm. In addition for the economic growth multi-threaded GA will be used to upgrade the speed of convergence in future in SG. Further, various filtering techniques can be optimized for increasing storage space. Many schemes such as ARIMA and Prophet model can be used to create better efficiencies and faster convergence. In future, the work ARIMA forecasting approach over personalities can be explored. There are very important potential for classification of the product portfolio are remaining in various computational matters. The next essential step is to refined anomaly detection techniques concerning better optimal results in SG.

Finally, the ensembling of data Anomaly Detection Techniques can be improve the various models for energy prediction in SG. The comparative performance of different anomaly detection techniques by using LSTM, ARIMA and Prophet can be explore in future for prediction of energy from the SG sensors. In future, reduction of the load on the storage and enhance the performance of the model can be elaborate by using different anomaly detection techniques.

# References


[1] Colah Tutorial on LSTM. https://colah.github.io/posts/2015-08-understanding-lstms/. *A LSTM Tutorial*, 2017.

[2] Anubha Jain, Manoj Mishra, Sateesh Kumar Peddoju, and Nitin Jain. Energy efficient computing-green cloud computing. In *2013 international conference on energy efficient technologies for sustainability*, pages 978–982. IEEE, 2013.

[3] Mohammad Harun Rashid. Ami smart meter big data analytics for time series of electricity consumption. In *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pages 1771–1776. IEEE, 2018.

[4] Ruchir Gupta and Yatindra Nath Singh. Reputation aggregation in peer-to-peer network using differential gossip algorithm. *IEEE Transactions on Knowledge and Data Engineering*, 27(10):2812–2823, 2015.

[5] Sushant Kumar Pandey, Ravi Bhushan Mishra, and Anil Kumar Tripathi. Machine learning based methods for software fault prediction: A survey. *Expert Systems with Applications*, 172:114595, 2021.

[6] Amit Kumar Vishwakarma, Haiwang Zhong, and Yatindra Nath Singh. Consensus mechanism for peer-to-peer energy trading. In *Recent Trends in Electronics and Communication*, pages 355–364. Springer, 2022.

[7] Ajey Kumar, Anil K Sarje, and Manoj Misra. Prioritised predicted region based cache replacement policy for location dependent data in mobile environment. *International Journal of Ad Hoc and Ubiquitous Computing*, 5(1):56–67, 2010.

[8] Albert Molderink, Vincent Bakker, Maurice GC Bosman, Johann L Hurink, and Gerard JM Smit. Management and control of domestic smart grid technology. *IEEE transactions on Smart Grid*, 1(2):109–119, 2010.

[9] Sushant Kumar Pandey and Anil Kumar Tripathi. Class imbalance issue in software defect prediction models by various machine learning techniques: An empirical study. In *2021 8th International Conference on Smart Computing and Communications (ICSCC)*, pages 58–63. IEEE, 2021.

[10] Muneeba Nasir, Abdul Rehman Javed, Muhammad Adnan Tariq, Muhammad Asim, and Thar Baker. Feature engineering and deep learning-based intrusion detection framework for securing edge iot. *The Journal of Supercomputing*, 78(6):8852–8866, 2022.

[11] Vinit Kumar Singh, Ashu Verma, and TS Bhatti. Interconnection of rural microgrids and its control. *IETE Journal of Research*, pages 1–10, 2021.

[12] Muhammad Mansoor Ashraf, Muhammad Waqas, Ghulam Abbas, Thar Baker, Ziaul Haq Abbas, and Hisham Alasmary. Feddp: A privacy-protecting theft detection scheme in smart grids using federated learning. *Energies*, 15(17):6241, 2022.

[13] Arnab Bhattacharjee, Ashu Verma, Sukumar Mishra, and Tapan K Saha. Estimating state of charge for xev batteries using 1d convolutional neural networks and transfer learning. *IEEE Transactions on Vehicular Technology*, 70(4):3123–3135, 2021.

[14] Sukhpal Singh Gill, Minxian Xu, Carlo Ottaviani, Panos Patros, Rami Bahsoon, Arash Shaghaghi, Muhammed Golec, Vlado Stankovski, Huaming Wu, Ajith Abraham, et al. Ai for next generation computing: Emerging trends and future directions. *Internet of Things*, 19:100514, 2022.

[15] Bharath Sudharsan, John G Breslin, Mehreen Tahir, Muhammad Intizar Ali, Omer Rana, Schahram Dustdar, and Rajiv Ranjan. Ota-tinyml: over the air deployment of tinyml models and execution on iot devices. *IEEE Internet Computing*, 26(3):69–78, 2022.

[16] Varshini K Sukanya and Uthra R Annie. Extraction of meaningful information from unstructured clinical notes using web scraping. *Journal of Circuits, Systems and Computers*, 2022.

[17] Yaqi Song, Guoliang Zhou, and Yongli Zhu. Present status and challenges of big data processing in smart grid. *Power System Technology*, 37(4):927–935, 2013.

[18] Bo Yang, Xuelin Cao, Chongwen Huang, Yong Liang Guan, Chau Yuen, Marco Di Renzo, Dusit Niyato, Mérouane Debbah, and Lajos Hanzo. Spectrum-learning-aided reconfigurable intelligent surfaces for "green" 6g networks. *IEEE Network*, 35(6):20–26, 2021.

[19] Ke Ma, Zhaocheng Wang, Wenqiang Tian, Sheng Chen, and Lajos Hanzo. Deep learning for mmwave beam-management: State-of-the-art, opportunities and challenges. *IEEE Wireless Communications*, 2022.

[20] Alexandros Labrinidis and Hosagrahar V Jagadish. Challenges and opportunities with big data. *Proceedings of the VLDB Endowment*, 5(12):2032–2033, 2012.

[21] Thien Van Luong, Nir Shlezinger, Chao Xu, Tiep M Hoang, Yonina C Eldar, and Lajos Hanzo. Deep learning based successive interference cancellation for the non-orthogonal downlink. *IEEE Transactions on Vehicular Technology*, 2022.

[22] Jianqing Fan, Fang Han, and Han Liu. Challenges of big data analysis. *National science review*, 1(2):293–314, 2014.

[23] Karthik Kambatla, Giorgos Kollias, Vipin Kumar, and Ananth Grama. Trends in

big data analytics. *Journal of parallel and distributed computing*, 74(7):2561–2573, 2014.

[24] Harry T Lawless and Hildegarde Heymann. Descriptive analysis. In *Sensory evaluation of food*, pages 227–257. Springer, 2010.

[25] Bart Baesens, Veronique Van Vlasselaer, and Wouter Verbeke. *Fraud analytics using descriptive, predictive, and social network techniques: a guide to data science for fraud detection*. John Wiley & Sons, 2015.

[26] Faisal Jamil, Naeem Iqbal, Shabir Ahmad, Dohyeun Kim, et al. Peer-to-peer energy trading mechanism based on blockchain and machine learning for sustainable electrical power supply in smart grid. *IEEE Access*, 9:39193–39217, 2021.

[27] Ziad Obermeyer and Ezekiel J Emanuel. Predicting the future—big data, machine learning, and clinical medicine. *The New England journal of medicine*, 375(13):1216, 2016.

[28] Priyanka P Shinde, Kavita S Oza, and RK Kamat. Big data predictive analysis: Using r analytical tool. In *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, pages 839–842. IEEE, 2017.

[29] Reza Soltanpoor and Timos Sellis. Prescriptive analytics for big data. In *Australasian database conference*, pages 245–256. Springer, 2016.

[30] Wooyoung Jeon, Jung Youn Mo, and Timothy D Mount. Developing a smart grid that customers can afford: The impact of deferrable demand. *The Energy Journal*, 36(4), 2015.

[31] Fabiano Pallonetto, Mattia De Rosa, Federico Milano, and Donal P Finn. Demand response algorithms for smart-grid ready residential buildings using machine learning models. *Applied energy*, 239:1265–1282, 2019.

[32] Farrokh Rahimi and Ali Ipakchi. Overview of demand response under the smart grid and market paradigms. In *2010 Innovative Smart Grid Technologies (ISGT)*, pages 1–7. IEEE, 2010.

[33] Xiufeng Liu, Lukasz Golab, Wojciech Golab, Ihab F Ilyas, and Shichao Jin. Smart meter data analytics: systems, algorithms, and benchmarking. *ACM Transactions on Database Systems (TODS)*, 42(1):1–39, 2016.

[34] Anish Jindal, Amit Dua, Kuljeet Kaur, Mukesh Singh, Neeraj Kumar, and Sukumar Mishra. Decision tree and svm-based data analytics for theft detection in smart grid. *IEEE Transactions on Industrial Informatics*, 12(3):1005–1016, 2016.

[35] Hicham Moad Safhi, Bouchra Frikh, and Brahim Ouhbi. Energy load forecasting in big data context. In *2020 5th International Conference on Renewable Energies for Developing Countries (REDEC)*, pages 1–6. IEEE, 2020.

[36] Houda Daki, Asmaa El Hannani, Abdelhak Aqqal, Abdelfattah Haidine, and Aziz

Dahbi. Big data management in smart grid: concepts, requirements and implementation. *Journal of Big Data*, 4(1):1–19, 2017.

[37] Soumesh Chatterjee, Suparna Dey, and Monalisa Dasgupta. A solution to demand response with soft-computing techniques. In *2020 Fourth International Conference on Inventive Systems and Control (ICISC)*, pages 122–126, 2020.

[38] Chun Sing Lai and Loi Lei Lai. Application of big data in smart grid. In *2015 IEEE International Conference on Systems, Man, and Cybernetics*, pages 665–670. IEEE, 2015.

[39] Charalampos Chelmis, Jahanvi Kolte, and Viktor K Prasanna. Big data analytics for demand response: Clustering over space and time. In *2015 IEEE International Conference on Big Data (Big Data)*, pages 2223–2232. IEEE, 2015.

[40] Nicholas Good, Keith A Ellis, and Pierluigi Mancarella. Review and classification of barriers and enablers of demand response in the smart grid. *Renewable and Sustainable Energy Reviews*, 72:57–72, 2017.

[41] Hamed Mortaji, Siew Hock Ow, Mahmoud Moghavvemi, and Haider Abbas F Almurib. Load shedding and smart-direct load control using internet of things in smart grid demand response management. *IEEE Transactions on Industry Applications*, 53(6):5155–5163, 2017.

[42] Guy Newsham and Benjamin Birt. Building-level occupancy data to improve arima-based electricity use forecasts. *BuildSys'10 - Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*, 01 2010.

[43] Jamal Fattah, Latifa Ezzine, Zineb Aman, Haj El Moussami, and Abdeslam Lachhab. Forecasting of demand using arima model. *International Journal of Engineering Business Management*, 10:1847979018808673, 2018.

[44] Saumyadip Ghosh. Forecasting of demand using arima model. *American Journal of Applied Mathematics and Computing*, 1(2):11–18, 2020.

[45] Christophorus Beneditto Aditya Satrio, William Darmawan, Bellatasya Unrica Nadia, and Novita Hanafiah. Time series analysis and forecasting of coronavirus disease in indonesia using arima model and prophet. *Procedia Computer Science*, 179:524–532, 2021.

[46] K Krishna Rani Samal, Korra Sathya Babu, Santosh Kumar Das, and Abhirup Acharaya. Time series based air pollution forecasting using sarima and prophet model. In *proceedings of the 2019 international conference on information technology and computer communications*, pages 80–85, 2019.

[47] Emir Zunic, Kemal Korjenic, Kerim Hodzic, and Dzenana Donko. Application of facebook's prophet algorithm for successful sales forecasting based on real-world data. *arXiv preprint arXiv:2005.07575*, 2020.

[48] Yu Li, Ziang Ma, Zhiwen Pan, Nan Liu, and Xiaohu You. Prophet model and gaussian process regression based user traffic prediction in wireless networks. *Science China Information Sciences*, 63(4):1–8, 2020.

[49] Sanju Kumari, Neeraj Kumar, and Prashant Singh Rana. Big data analytics for energy consumption prediction in smart grid using genetic algorithm and long short term memory. *Computing and Informatics*, 40(1):29–56, 2021.

[50] Sowmya Vaitheeswaran and Varun Ventrapragada. Wind power pattern prediction in time series measuremnt data for wind energy prediction modelling using lstm-ga networks. In *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–5. IEEE, 2019.

[51] D Hema and K Ashok Kumar. An optimized intelligent driver's aggressive behaviour prediction model using ga-lstm. *International Journal of Performability Engineering*, 17(10), 2021.

[52] Michael Ruse. Charles darwin's theory of evolution: an analysis. *Journal of the History of Biology*, pages 219–241, 1975.

[53] Xiufeng Liu and Per Sieverts Nielsen. Regression-based online anomaly detection for smart grid data. *arXiv preprint arXiv:1606.05781*, 2016.

[54] Lenin Mookiah, Chris Dean, and William Eberle. Graph-based anomaly detection on smart grid data. In *The Thirtieth International Flairs Conference*, 2017.

[55] Markus Goldstein and Andreas Dengel. Histogram-based outlier score (hbos): A fast unsupervised anomaly detection algorithm. *KI-2012: Poster and Demo Track*, pages 59–63, 2012.

[56] William Hurst, Casimiro A Curbelo Montanez, and Nathan Shone. Time-pattern profiling from smart meter data to detect outliers in energy consumption. *IoT*, 1(1):92–108, 2020.

[57] Adnan Anwar and Abdun Naser Mahmood. Anomaly detection in electric network database of smart grid: Graph matching approach. *Electric Power Systems Research*, 133:51–62, 2016.

[58] Andres F Murillo. Review of anomalies detection schemes in smart grids. *Grupo de Teleinformatica e Automacao Symantec*, 2013.

[59] Anupiya Nugaliyadde, Upeka Somaratne, and Kok Wai Wong. Predicting electricity consumption using deep recurrent neural networks. *arXiv preprint arXiv:1909.08182*, 2019.

[60] Salah Bouktif, Ali Fiaz, Ali Ouni, and Mohamed Serhani. Optimal deep learning lstm model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches. *Energies*, 11(7):1636, 2018.

[61] S Sagiroglu, R Terzi, Y Canbay, and I Colak. Big data issues in smart grid systems.

In *IEEE International Conference on Renewable Energy Research and Applications(ICRERA)*, pages 1007–1012. IEEE, 2016.

[62] J. Hu and A. Vasilakos. Energy big data analytics and security: Challenges and opportunities. *IEEE Transactions on Smart Grid*, 7(5):2423–2436, 2016.

[63] Yi Wang, Qixin Chen, Chongqing Kang, Mingming Zhang, Ke Wang, and Yun Zhao. Load profiling and its application to demand response: A review. *Tsinghua Science and Technology*, 20(2):117–129, 2015.

[64] P Diamantoulakis, V Kapinas, and G Karagiannidis. Big data analytics for dynamic energy management in smart grids. *Big Data Research*, 2(3):94–101, 2015.

[65] L. Wang, S. Mao, and B. Wilamowski. Short-term load forecasting with LSTM based ensemble learning. In *International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pages 793–800, 2019.

[66] C Stimmel. *Big data analytics strategies for the smart grid*. CRC Press, 2014.

[67] Kevin Pasini, Mostepha Khouadjia, Allou Same, Fabrice Ganansia, and Latifa Oukhellou. LSTM encoder-predictor for short-term train load forecasting. In *ECML/PKDD - The European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 2019.

[68] K Amarasinghe, D Marino, and M Manic. Deep neural networks for energy load forecasting. In *IEEE 26th International Symposium on Industrial Electronics (ISIE)*, pages 1483–1488. IEEE, 2017.

[69] Faisal Mohammad and Y Kim. Energy load forecasting model based on deep neural networks for smart grids. *International Journal of System Assurance Engineering and Management*, pages 1–11, 2019.

[70] Yao Cheng, Chang Xu, Daisuke Mashima, Vrizlynn Thing, and Yongdong Wu. PowerLSTM: Power demand forecasting using long short-term memory neural network. In *International Conference on Advanced Data Mining and Applications*, pages 727–740. Springer, 2017.

[71] Hyungeun Choi, Seunghyoung Ryu, and Hongseok Kim. Short-term load forecasting based on ResNet and LSTM. In *IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, pages 1–6. IEEE, 2018.

[72] S Sulaiman, P Jeyanthy, and D Devaraj. Artificial neural network based day ahead load forecasting using smart meter data. In *Biennial International Conference on Power and Energy Systems: Towards Sustainable Energy (PESTSE)*, pages 1–6. IEEE, 2016.

[73] T Sainath, O Vinyals, A Senior, and H Sak. Convolutional, long short-term memory, fully connected deep neural networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4580–4584. IEEE, 2015.

[74] Y Bengio, I Goodfellow, and A Courville. *Deep learning*, volume 1. MIT press, 2017.

[75] Saima Aman, Marc Frincu, Charalampos Chelmis, Muhammad Noor, Yogesh Simmhan, and Viktor K Prasanna. Prediction models for dynamic demand response: Requirements, challenges, and insights. In *IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 338–343. IEEE, 2015.

[76] M. Rashid. AMI smart meter big data analytics for time series of electricity consumption. In *17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pages 1771–1776, 2018.

[77] S Khuri, B Thomas, and O Heitk. The zero/one multiple knapsack problem and genetic algorithms. In *Proceedings of the ACM symposium on Applied computing*, pages 188–193, 1994.

[78] Tuong Le, M Vo, B Vo, E Hwang, R Seungmin, and B Wook. Improving electric energy consumption prediction using CNN and Bi-LSTM. *Applied Sciences*, 9(20):4237, 2019.

[79] C Michael, G McGraw, M Schatz, and C Walton. Genetic algorithms for dynamic test data generation. In *Proceedings 12th IEEE International Conference Automated Software Engineering*, pages 307–308. IEEE, 1997.

[80] S. Sulaiman, P. Jeyanthy, and D. Devaraj. Big data analytics of smart meter data using adaptive neuro fuzzy inference system (ANFIS). In *International Conference on Emerging Technological Trends (ICETT)*, pages 1–5, 2016.

[81] A. Teres. Histogram visualization of smart grid data using MapReduce algorithm. In *2nd International Conference on Power and Embedded Drive Control (ICPEDC)*, pages 307–312, 2019.

[82] Y. Simmhan, S. Aman, A. Kumbhare, R. Liu, S. Stevens, Q. Zhou, and V. Prasanna. Cloud-based software platform for big data analytics in smart grids. *Computing in Science Engineering*, 15(4):38–47, 2013.

[83] D. Kaur, R. Kumar, N. Kumar, and M. Guizani. Smart grid energy management using RNN-LSTM: A deep learning-based approach. In *IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, 2019.

[84] M. Couceiro, R. Ferrando, D. Manzano, and L. Lafuente. Stream analytics for utilities. predicting power supply and demand in a smart grid. In *3rd International Workshop on Cognitive Information Processing (CIP)*, pages 1–6, 2012.

[85] W. Kong, Z. Dong, Y. Jia, D. Hill, Y. Xu, and Y. Zhang. Short-term residential load forecasting based on LSTM recurrent neural network. *IEEE Transactions on Smart Grid*, 10(1):841–851, 2019.

[86] G. Zhang and J. Guo. A novel method for hourly electricity demand forecasting. *IEEE Transactions on Power Systems*, 35(2):1351–1363, 2020.

[87] A. Eseye, M. Lehtonen, T. Tukia, S. Uimonen, and R. John. Machine learning based integrated feature selection approach for improved electricity demand forecasting in decentralized energy systems. *IEEE Access*, 7:91463–91475, 2019.

[88] A. Mamun, M. Hoq, E. Hossain, and R. Bayindir. A hybrid deep learning model with evolutionary algorithm for short-term load forecasting. In *8th International Conference on Renewable Energy Research and Applications (ICRERA)*, pages 886–891, 2019.

[89] A. Eseye, J. Zhang, D. Zheng, H. Ma, and G. Jingfu. Short-term wind power forecasting using a double-stage hierarchical hybrid GA-ANN approach. In *IEEE 2nd International Conference on Big Data Analysis (ICBDA)*, pages 552–556, 2017.

[90] S. Jaidee and W. Pora. Very short-term solar power forecasting using genetic algorithm based deep neural network. In *4th International Conference on Information Technology (InCIT)*, pages 184–189, 2019.

[91] PJM. Pennsylvania New Jersey Maryland Interconnection. *https://www.pjm.com*, 2020.

[92] PJM dataset. PJM Time Series Analysis and Forecasting Data. *https://www.kaggle.com/brahimmebrek/pjm-east-eda-and-forecasting*, 2020.

[93] Min Chen, Shiwen Mao, and Yunhao Liu. Big data: A survey. *Mobile networks and applications*, 19(2):171–209, 2014.

[94] Hanlin Zhu, Yongxin Zhu, Di Wu, Hui Wang, Li Tian, Wei Mao, Can Feng, Xiaowen Zha, Guobao Deng, Jiayi Chen, et al. Correlation coefficient based cluster data preprocessing and lstm prediction model for time series data in large aircraft test flights. In *International Conference on Smart Computing and Communication*, pages 376–385. Springer, 2018.

[95] Muhammad Usman, Zahoor Ali Khan, Inam Ullah Khan, Sakeena Javaid, and Nadeem Javaid. Data analytics for short term price and load forecasting in smart grids using enhanced recurrent neural network. In *2019 Sixth HCT Information Technology Trends (ITT)*, pages 84–88. IEEE, 2019.

[96] Tanzeela Sultana and Nadeem Javaid. Data analytics for load and price forecasting via enhanced support vector regression technical report for mscs.

[97] Jinsong Wu, Song Guo, Jie Li, and Deze Zeng. Big data meet green challenges: Greening big data. *IEEE Systems Journal*, 10(3):873–887, 2016.

[98] D Patel, Amit Vajpayee, and J Dangra. Short term load forecasting by using time series analysis through smoothing techniques. *International Journal of Engineering Research & Technology (IJERT)*, 2(9), 2013.

[99] JM Górriz, Carlos García Puntonet, and Moisés Salmerón. Preprocessing time series with ica and savitzky-golay filtering. In *Neural Networks and Computational Intelligence*, pages 48–53, 2004.

[100] Mohit Bansal, Ritu Sharma, and Parul Grover. Performance evaluation of butterworth filter for signal denoising. *IJECT*, 2010.

[101] M. Roth, C. Fritsche, G. Hendeby, and F. Gustafison. The ensemble kalman filter and its relations to other nonlinear filters. In *2015 23rd European Signal Processing Conference (EUSIPCO)*, pages 1236–1240, 2015.

[102] Peter L Houtekamer and Herschel L Mitchell. Ensemble kalman filtering. *Quarterly Journal of the Royal Meteorological Society: A journal of the atmospheric sciences, applied meteorology and physical oceanography*, 131(613):3269–3289, 2005.

[103] Xiaogang Chen, Yijun Wang, Shangkai Gao, Tzyy-Ping Jung, and Xiaorong Gao. Filter bank canonical correlation analysis for implementing a high-speed ssvep-based brain–computer interface. *Journal of neural engineering*, 12(4):046008, 2015.

[104] Haifaa Hussein Hameed. Smoothing techniques for time series forecasting. Master's thesis, Eastern Mediterranean University (EMU)-Doğu Akdeniz Üniversitesi (DAÜ), 2015.

[105] Fredrik Gustafsson. Determining the initial states in forward-backward filtering. *IEEE Transactions on signal processing*, 44(4):988–992, 1996.

[106] René-Édouard Plessix, Guido Baeten, Jan Willem de Maag, Fons ten Kroode, and Zhang Rujie. Full waveform inversion and distance separated simultaneous sweeping: a study with a land seismic data set. *Geophysical Prospecting*, 60(4-Simultaneous Source Methods for Seismic Data):733–747, 2012.

[107] Shaomin Zhang, Kang Huang, Baoyi Wang, et al. A data integrity verification scheme with secure deduplication in smart grid cloud storage. *Advances in Computer, Signals and Systems*, 3(1):1–7, 2019.

[108] CK Roopa and BS Harish. An empirical evaluation of savitzky-golay (sg) filter for denoising st segment. In *International Conference on Cognitive Computing and Information Processing*, pages 18–28. Springer, 2017.

[109] Antonin Novak, Laurent Simon, František Kadlec, and Pierrick Lotton. Nonlinear system identification using exponential swept-sine signal. *IEEE Transactions on Instrumentation and Measurement*, 59(8):2220–2229, 2009.

[110] PJM Hourly Energy Consumption Data. Pjm interconnection llc. https://www.kaggle.com/robikscube/hourly-energy-consumption/data. *Neu-*

*ral computation*, 29(2):733–747, 2018.

[111] Varun Badrinath Krishna, Ravishankar Iyer, and William Sanders. Arima-based modeling and validation of consumption readings in power grids. pages 199–210, 05 2016.

[112] Hui Liu. *Smart Cities: Big Data Prediction Methods and Applications*. Springer Singapore, 2020.

[113] Amelec Viloria, Ronald Prieto Pulido, Jesus Guiliany, Jairo Ventura, Hugo Palma, Jose Torres, Osman Bilbao, and Omar Lezama. *Analyzing and Predicting Power Consumption Profiles Using Big Data*, pages 392–401. 11 2019.

[114] Damilola Asaleye, Michael Breen, and Michael Murphy. A decision support tool for building integrated renewable energy microgrids connected to a smart grid. *Energies*, 10:1765, 11 2017.

[115] Zeping Wang, Jianfeng Qu, Xiaoyu Fang, Hao Li, Ting Zhong, and Hao Ren. Prediction of early stabilization time of electrolytic capacitor based on arima-bi_lstm hybrid model. *Neurocomputing*, 403, 04 2020.

[116] Archee Gupta, Kailash Sharma, Archita Vijayvargia, and Rohit Bhakar. Very short term wind power prediction using hybrid univariate arima-garch model. pages 1–6, 12 2019.

[117] Xing Luo, Xu Zhu, and Eng Lim. A hybrid model for short term real-time electricity price forecasting in smart grid. *Big Data Analytics*, 3, 10 2018.

[118] Ranjan Pal, Charalampos Chelmis, Marc Frincu, and Viktor Prasanna. Towards dynamic demand response on efficient consumer grouping algorithmics. *IEEE Transactions on Sustainable Computing*, 1(1):20–34, 2016.

[119] Hicham Moad Safhi, Bouchra Frikh, and Brahim Ouhbi. Energy load forecasting in big data context. In *2020 5th International Conference on Renewable Energies for Developing Countries (REDEC)*, pages 1–6, 2020.

[120] Aya Yokoe Bishnu Nepal, Motoi Yamaha. Electricity load forecasting using clustering and arima model for energy management in buildings. In *Wiley Online Library*, pages 1–6, 2019.

[121] Charalampos Chelmis Saima Aman, Marc Frincu. Prediction models for dynamic demand response-requirements, challenges, and insights. In *IEEE conference*, pages 1–6, 2019.

[122] Jie Zhan, Jinxin Huang, Lin Niu, Xiaosheng Peng, Diyuan Deng, and Shijie Cheng. Study of the key technologies of electric power big data and its application prospects in smart grid. In *2014 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, pages 1–4, 2014.

[123] Tai Yeon Ku, Wan Ki Park, and Choi. Demand response operation method on

energy big data platform. In *2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN)*, pages 823–825, 2018.

[124] Yi Wang, Qixin Chen, Chongqing Kang, Qing Xia, and Min Luo. Sparse and redundant representation-based smart meter data compression and pattern extraction. *IEEE Transactions on Power Systems*, 32(3):2142–2151, 2017.

[125] Xin Zhang, Donghua Li, Ming Cheng, and Pei Zhang. Electricity consumption pattern recognition based on the big data technology to support the peak shifting potential analysis. In *2014 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, pages 1–5, 2014.

[126] Vivek Abhilash Hanumantha Vajjala. A novel solution to use big data technologies and improve demand response program in aggregated residential houses. In *2016 IEEE Conference on Technologies for Sustainability (SusTech)*, pages 251–256, 2016.

[127] Shailendra Singh and Abdulsalam Yassine. Mining energy consumption behavior patterns for households in smart grid. *IEEE Transactions on Emerging Topics in Computing*, 7(3):404–419, 2019.

[128] Abdulla Almazrouee, Abdullah Almeshal, Abdulrahman Almutairi, Mohammad Alenezi, and Saleh Alhajeri. Long-term forecasting of electrical loads in kuwait using prophet and holt-winters models. *Applied Sciences*, 10:5627, 08 2020.

[129] Bruno Sandrić, Mateo Marcelić, Iva Topolovac, and Marko Jurčevic. Survey of data cleaning algorithms in wireless sensor networks. In *2019 2nd International Colloquium on Smart Grid Metrology (SMAGRIMET)*, pages 1–4, 2019.

[130] Bahrudin Hrnjica and Ali Danandeh Mehr. *Energy Demand Forecasting Using Deep Learning*, pages 71–104. Springer International Publishing, Cham, 2020.

[131] Gheorghe Grigoraş, Gheorghe Cârţină, and Elena-Crenguţa Bobric. Trends and directions for energy saving in electric networks. In Hongyi Sun, editor, *Management of Technological Innovation in Developing and Developed Countries*, chapter 1. IntechOpen, Rijeka, 2012.

[132] Jr. Joseph W. Forbes. System, method, and apparatus for electric power grid and network management of grid elements. In *US2014/0039699A1*, pages 13/563,535, 2014.

[133] Shahrear Iqbal, Md. Faizul Bari, and Mohammad Rahman. Solving the multi-dimensional multi-choice knapsack problem with the help of ants. volume 6234, pages 312–323, 09 2010.

[134] online https://data.london.gov.uk/dataset/smartmeter-energy-use-data-in-london households.

[135] E Khaledian, S Pandey, P Kundu, and AK Srivastava. Real-time synchrophasor

data anomaly detection and classification using isolation forest, kmeans and loop. *IEEE Transactions on Smart Grid*, 2020.

[136] Longji Feng, Shu Xu, Linghao Zhang, Jing Wu, Jidong Zhang, Chengbo Chu, Zhenyu Wang, and Haoyang Shi. Anomaly detection for electricity consumption in cloud computing: framework, methods, applications, and challenges. *EURASIP Journal on Wireless Communications and Networking*, 2020(1):1–12, 2020.

[137] Archee Gupta, Kailash Chand Sharma, Archita Vijayvargia, and Rohit Bhakar. Very short term wind power prediction using hybrid univariate arima-garch model. In *2019 8th International Conference on Power Systems (ICPS)*, pages 1–6. IEEE, 2019.

[138] Juan M Lujano-Rojas, Claudio Monteiro, Rodolfo Dufo-Lopez, and Jose L Bernal-Agustn. Optimum residential load management strategy for real time pricing (rtp) demand response programs. *Energy policy*, 45:671–679, 2012.

[139] Sang Yeob Kim, Sang Min Lee, Kyoung Sub Park, and Keun Ho Ryu. Prediction model of internal temperature using backpropagation algorithm for climate control in greenhouse. *원예과학기술지*, 36(5):713–729, 2018.

[140] Ying Ma, Chao Huang, Yu Sun, Guang Zhao, and Yunjie Lei. Review of power spatio-temporal big data technologies, applications, and challenges. In *International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage*, pages 197–206. Springer, 2019.

[141] Ramin Moghaddass and Jianhui Wang. A hierarchical framework for smart grid anomaly detection using large-scale smart meter data. *IEEE Transactions on Smart Grid*, 9(6):5820–5830, 2017.

[142] K Ya Vinnikov, P Ya Groisman, and KM Lugina. Empirical data on contemporary global climate changes (temperature and precipitation). *Journal of Climate*, 3(6):662–677, 1990.

[143] Timothy Barth and Dennis Jespersen. The design and application of upwind schemes on unstructured meshes. In *27th Aerospace sciences meeting*, page 366, 1989.

[144] Abhijit Guha and Debabrata Samanta. Hybrid approach to document anomaly detection: An application to facilitate rpa in title insurance. *International Journal of Automation and Computing*, 18(1):55–72, 2021.

[145] Chengxi Liu, Zakir Hussain Rather, Zhe Chen, and Claus Leth Bak. An overview of decision tree applied to power systems. *International Journal of Smart Grid and Clean Energy*, 2(3):413–419, 2013.

[146] Georgios Efstathopoulos, Panagiotis Radoglou Grammatikis, Panagiotis Sarigiannidis, Vasilis Argyriou, Antonios Sarigiannidis, Konstantinos Stamatakis, Michail K

Angelopoulos, and Solon K Athanasopoulos. Operational data based intrusion detection system for smart grid. In *2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, pages 1–6. IEEE, 2019.

[147] Angel Luis Perales Gomez. Cyberattacks detection in industrial scenarios using machine learning and deep learning techniques. *Proyecto de investigacion:*, 2021.

[148] Haifeng Xu. An evaluation of one class classifier on gene expression data. Master's thesis, 2019.

[149] Jaime Yeckle and Bo Tang. Detection of electricity theft in customer consumption using outlier detection algorithms. In *2018 1st International Conference on Data Intelligence and Security (ICDIS)*, pages 135–140. IEEE, 2018.

[150] Ane Blazquez-Garcia, Angel Conde, Usue Mori, and Jose A Lozano. A review on outlier/anomaly detection in time series data. *arXiv:2002.04236*, 2020.

[151] Dina ElMenshawy and Waleed Helmy. Detection techniques of data anomalies in iot: a literature survey. *Technology*, 9(12):794–807, 2018.

[152] Sofia Osypova. Consumption pattern detection through the use of machine learning: Clustering techniques for non-technical losses detection rerort. Master's thesis, Universitat Politecnica de Catalunya, 2020.

[153] Yanjun Yao, Sisi Xiong, Hairong Qi, Yilu Liu, Leon M Tolbert, and Qing Cao. Efficient histogram estimation for smart grid data processing with the loglog-bloom-filter. *IEEE Transactions on Smart Grid*, 6(1):199–208, 2014.

[154] Syeda Aimal, Nadeem Javaid, Tahir Islam, Wazir Zada Khan, Mohammed Y Aalsalem, and Hassan Sajjad. An efficient cnn and knn data analytics for electricity load forecasting in the smart grid. In *Workshops of the International Conference on Advanced Information Networking and Applications*, pages 592–603. Springer, 2019.

[155] Thanh TX Tran and Ekin Ozer. Synergistic bridge modal analysis using frequency domain decomposition, observer kalman filter identification, stochastic subspace identification, system realization using information matrix, and autoregressive exogenous model. *Mechanical Systems and Signal Processing*, 160:107818, 2021.

[156] Joao Pereira and Margarida Silveira. Unsupervised anomaly detection in energy time series data using variational recurrent autoencoders with attention. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1275–1282. IEEE, 2018.

[157] Gaurav Sharma, Prashant Singh Rana, and Seema Bawa. Hybrid machine learning models for predicting types of human t-cell lymphotropic virus. *IEEE/ACM transactions on computational biology and bioinformatics*, 2019.

[158] Erich Schubert and Michael Gertz. Intrinsic t-stochastic neighbor embedding for visualization and outlier detection. In *International Conference on Similarity Search and Applications*, pages 188–203. Springer, 2017.

[159] Robert McCullough, Michael Weisdorf, Jean-Carl Ende, and Aiman Absar. Exactly how inefficient is the pjm capacity market? *The Electricity Journal*, 33(8):106819, 2020.

[160] Gengqiang Huang, Junjie Yang, and Chunjuan Wei. Cost-effective and comfort-aware electricity scheduling for home energy management system. In *2016 IEEE International Conferences on Big Data and Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom)(BDCloud-SocialCom-SustainCom)*, pages 453–460. IEEE, 2016.

# List of Publications

1. Sanju Kumari, Neeraj Kumar and Prashant Singh Rana, *"Big Data Analytics for Energy Consumption Prediction in Smart Grid using Genetic Algorithm and Long Short Term Memory"*, Computing and Informatics, Slovak Academy of Sciences, 40(1):29-56, 2021. [SCIE, IF 0.87]

2. Sanju Kumari, Neeraj Kumar and Prashant Singh Rana, *"A Big data Approach for Demand Response Management in Smart Grid using Prophet Model"*, Electronics, MDPI, 11(14), 2179-98, 2022. [SCIE, IF 2.69]

3. Sanju Kumari, Neeraj Kumar and Prashant Singh Rana, *"Comparative Performance Study of Different Filtering Techniques with LSTM for the Prediction of Power Consumption in Smart Grid"*, IETE Journal of Research, Taylor & Francis, 2022. [SCI, Under Minor Revision, IF 1.89]

4. Sanju Kumari, Neeraj Kumar and Prashant Singh Rana, *"Ensembling of Data Anomaly Techniques for the Prediction of Energy Consumption in Smart Grid"*, Journal of Engineering Research, Kuwait University, 2022. [SCIE, Under Review, IF 1.49]