TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING
PULCHOWK CAMPUS

Thesis No: M-370-MSREE-2019-2023

Peak Load Forecast using Long Short Term Memory Networks.

By:
Ananta Hari Konda

A THESIS
SUBMITTED TO DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING IN THE PARTIAL FULFILLMENT TO THE REQUIREMENT FOR DEGREE OF MASTERS OF SCIENCE IN RENEWABLE ENERGY ENGINEERING

DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING
LALITPUR, NEPAL

October, 2023

COPYRIGHT

TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING
PULCHOWK CAMPUS
DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING

The undersigned certify that they have read and recommended to the Institute of Engineering for acceptance, a thesis entitled "Peak Load Forecast Using Long Short Term Memory Networks" submitted by Ananta Hari Konda in partial fulfillment of the requirements for the degree of Master of Science in Renewable Energy Engineering.

Asst. Prof. Shahabuddin Khan
Supervisor,
Department of Electrical Engineering,
Pulchowk Campus, IOE

Asst. Prof. Sanjaya Neupane
Supervisor,
Department of Aerospace and Mechanical Engineering,
Pulchowk Campus, IOE

Dr. Laxman Ghimire
External Examiner,
Director, AEPC

Asst. Prof. Dr. Sudip Bhattrai
Committee Chairperson,
Head of Department,
Department of Aerospace and Mechanical Engineering,
Pulchowk Campus, IOE

Date: October 5, 2023

ABSTRACT

Forecasting is the technique of making scientific predictions of the future based on historical data trends. With increasing smart grid technologies and penetration of intermittent renewable energy technologies, peak load forecasting has been an important task in recent years for optimizing power grid operations. In recent years, machine learning has been one of the popular methods for load forecasting. In this paper, a deep learning single Long Short Term Memory (LSTM) based model is proposed for predicting the peak load demand of Nepal and compared with other statistical models. Comparing the evaluation metrics, it is deduced that the proposed LSTM model with 32 LSTM unit and a lookback of 30 has better forecast accuracy with Mean Absolute Error (MAE) of 34.01 MW, Mean Squared Error(MSE) of 2880.75 MW, Root Mean Squared Error(RMSE) of 53.67 MW, Mean Absolute Percent Error(MAPE) of 2.950% and R^2 Score of 0.933. Among the statistical models considered, the Weighted Moving Average with a loopback of 30 days had the least forecast errors with a Mean Absolute Error(MAE) of 34.77 MW, Mean Squared Error(MSE) of 3271.68 MW, Root Mean Squared Error(RMSE) of 57.19 MW, Mean Absolute Percent Error(MAPE) of 3.005% and R^2 Score of 0.921.

ACKNOWLEDGEMENT

First of all, I want to convey my profound gratitude to my thesis supervisor Assistant Professor Shahabuddin Khan and Assistant Professor Sanjaya Neupane for their continuous supervision, consistent support and motivation throughout this work. Their guidance and advice carried me through all the stages of my research. My special thanks to Asst. Prof. Dr. Sudip Bhattrai, Head of Department, Department of Mechanical and Aerospace Engineering, Pulchowk Campus, for his assistance and advice. I would also like to express my gratitude to my committee members for making my defense an enjoyable moment and for their insightful comments and suggestions.

My special thanks to Assoc. Prof. Dr. Hari Bahadur Darlami, M.Sc. Co-ordinator, Department of Mechanical and Aerospace Engineering for his continuous encouragement during the research period. I am also grateful for his valuable support. I would also like to thank the Department of Mechanical and Aerospace Engineering, as well as the Institute of Engineering, Pulchowk Campus for their help with the thesis.

I would like to thank the Load Dispatch Center, Nepal Electricity Authority for providing the data during my research. Throughout my study, I am thankful to Er. Dinesh Ghemosu for his consistent assistance in this research work.

Finally, I would like to convey my sincere thanks to all IOE family, my friends, family and personnel for their support regarding this research from inception to end.

Ananta Hari Konda
075MSREE002
October, 2023

TABLE OF CONTENTS

List of Tables

List of Figures

List of acronyms and abbreviations

Abbreviation	**Meaning**
MAE	Mean Absolute Error
MAPE	Mean Absolute Percent Error
RSME	Root Mean Squared Error
NEA	Nepal Electricity Authority
RNN	Recurrent Neural Network
NN	Neural Network
CNN	Convolutional Neural Network
GRU	Gated Recurrent Unit
LSTM	Long Short Term Memory
NLP	Natural Language Processing
WECS	Water and Energy Commission Secretaria
GDP	Gross Domestic Product
RNN	Recurrent Neural Network
MA	Moving Average
SMA	Simple Moving Average
WMA	Weighted Moving Average
SES	Simple Exponential Smoothing
DES	Double Exponential Smoothing
HWES	Holt Winters Exponential Smoothing
COVID 19	Coronavirus
SVM	Support Vector Machine
ANN	Artificial Neural Network
MW	Megawatt
GW	Gigawatt
EV	Electric Vehicles
DER	Distributed Energy Resources
ISO	Independent System Operators
MAED	Model for Energy Demand Analysis
BAU	Business as usual
LDC	Load Dispatch Center
Adam	Adaptive Moment Estimation

CHAPTER ONE : INTRODUCTION

1.1 Background

Time series forecasting is a significant aspect of prediction, where historical data points of a particular variable are collected and analyzed to create a model that describes the inherent pattern. The model is subsequently employed to project the time series into the future. This modeling approach is particularly useful when little knowledge is available on the underlying data-generating process or when there is no satisfactory explanatory model that relates the prediction variable to other explanatory variables[1]. Time Series is simply a series of data points that has a time component involved in it. Every data point present in the dataset is associated directly with the date and time component. The Components of time series are level, trend, seasonality, cyclicity and noise.

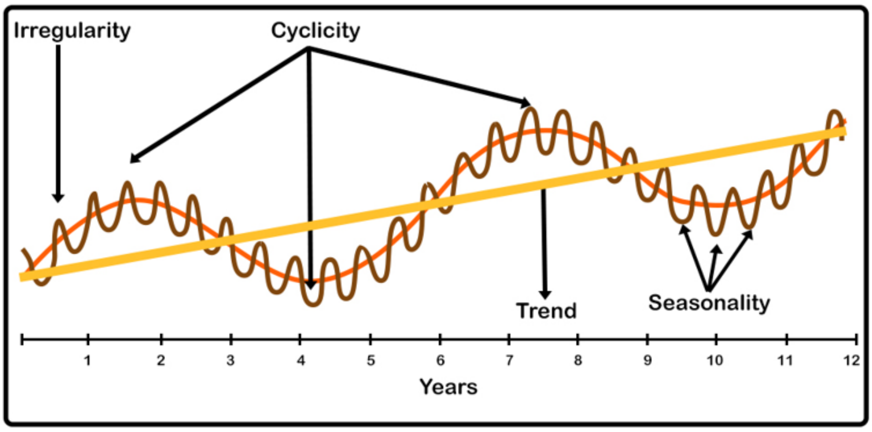


Figure 1.1: Components of time series

The level is the baseline for the entire time series. It is the average of the time series and the baseline to which different other components are added. The trend is the indication of whether the time series has moved higher or lower over the time period. Seasonality is the pattern in a time series that repeats after a fixed interval of time. Cyclicity is the pattern in the time series which repeats itself after some interval of time but the interval of time is not fixed. Noise is the random variation in the time series and does not have any Pattern. Noise is not used for forecasting purposes.

Decision-making in the energy sector has to be based on accurate forecasts of the load demand[2]. Precise models for predicting electric peak loads are crucial for operational decisions, including optimizing the scheduling of power stations, managing fuel procurement, planning energy transactions, scheduling power projects, forecasting energy prices, ensuring economic load distribution, making unit commitments and assessing power system security. It is estimated that a 5% to 15% reduction in peak load would bring substantial benefits in saving resources and decreasing real-time electricity tariffs, which calls for effective peak load management strategies[3]. Load forecast is extremely important for energy suppliers, independent system operators, financial institutions and other participants in electric energy generation, transmission and distribution[4]. Peak load forecast not only helps power plants by giving sufficient start-up time to avoid grid congestion but also enhances the stability and security of power grids. However, peak load forecast is a complicated task because consumption is influenced by many factors such as day type, anomalous days, weather conditions, vacations, economic factors, status and distinctive habits of individual customers[5]. The financial costs of forecast errors are so high that much research is focused on reducing the error even by a fraction of a percentage point[6].

1.2 Problem Statement

Electric utilities make many resource commitments that require forecasts of loads from a few weeks to one year ahead of time. Such decisions can include:

1. Commitment of generating units
2. Short run hydropower scheduling
3. Economic dispatch of committed units
4. Predictive automatic generation control
5. Spinning reserve
6. Fuel allocation
7. Energy purchases and sales
8. Real-time prices
9. Load interruption
10. Load control
11. Generator and transmission line maintenance
12. Available transmission capability

Forecast inaccuracies lead to higher expenses or "regret." For example, if actual loads end up being lower than anticipated, it could mean that units were committed unnecessarily, resulting in increased fuel costs and potentially higher maintenance expenses. Additionally, purchasing expensive power that was not required or missing a profitable opportunity to sell bulk power could occur. Quoting excessively high real-time prices might hinder sales, and unnecessary interruptions or load controls may be implemented, annoying consumers and a reduction in revenue.

Conversely, if the actual loads exceed expectations, insufficient resources might be at hand to satisfy security constraints, such as spinning reserve margins, thereby jeopardizing the reliability of the system. To address the unexpected surge in load, economically unfavorable generation or spot power purchases might become necessary. Alternatively, avoidable load interruptions or controls might be initiated if the load had been accurately forecasted. Commitments to sell power may have been agreed upon at a price lower than the utility's valuation of that power. Too low real-time prices might have been quoted, resulting in revenue falling short of the utility's cost[7].

The significance of improved forecasts lies in the extent to which their utilization can minimize the various causes of disappointment or regret. Since errors in peak load forecasts lead to less-than-optimal unit commitment decisions, peak demand predictions provide valuable guidance for decision-making in scheduling, contingency analysis, load flow analysis, mitigating imbalances in power generation and demand, and implementing load switching strategies, thereby enhancing network reliability and power quality.

Therefore, accurate peak load prediction holds substantial significance in the realm of electrical power management and distribution. It is a pivotal factor in upholding the dependability, effectiveness, and cost-efficiency of electrical grids and power generation facilities. Some key reasons highlighting the importance of peak load forecasting are as follows:

1. Resource Planning: Peak load forecasts help utilities and power generation companies plan for the necessary resources, such as power plants, transformers and transmission lines to meet the highest levels of electricity demand. Accurate forecasts ensure that there is sufficient capacity to prevent shortages during peak periods.

2. Infrastructure Investment: Predicting peak loads allows utilities to make informed decisions about infrastructure investments. They can allocate resources for maintenance, upgrades and the construction of new facilities based on forecasted demand, optimizing capital expenditure.

3. Efficient Operation: Grid operators use peak load forecasts to optimize the operation of the electrical grid. This includes scheduling power generation, managing grid stability and ensuring that electricity supply matches demand, minimizing inefficiencies.

4. Demand Response Programs: Utilities often implement demand response programs to reduce peak load by encouraging consumers to reduce their electricity usage during peak periods. Accurate forecasting is crucial for designing and implementing effective demand response strategies.

5. Energy Procurement: Accurate peak load forecasts help utilities and energy providers procure the right amount of electricity at the right time. Overestimating or underestimating peak demand can lead to financial losses and disruptions in service.

6. Grid Reliability: Predicting peak loads ensures the reliability of the electrical grid by preventing overloads and blackouts during periods of high demand. Grid stability is essential for both residential and industrial consumers.

7. Cost Savings: Accurate forecasts can lead to cost savings by optimizing the use of existing infrastructure, reducing the need for emergency measures and avoiding penalties associated

with capacity shortfalls.

8. Resource Allocation: Beyond power generation, peak load forecasting is essential for allocating resources in various industries. For example, in data centers, it helps manage cooling systems and prevent overheating during periods of high computational demand.

9. Emergency Preparedness: Peak load forecasts also have applications in emergency preparedness. They help emergency services plan for extreme weather events or other situations that may affect electrical demand and grid reliability.

10. Renewable Integration: As renewable energy sources like solar and wind become more prevalent, peak load forecasting becomes critical for integrating these intermittent sources into the grid. Forecasts help balance supply and demand and ensure grid stability.

In summary, peak load forecasting is essential for ensuring the efficient and reliable operation of electrical grids, optimizing resource allocation and minimizing costs. It also contributes to environmental sustainability by reducing the need for additional fossil fuel-based power generation during peak demand periods. Accurate forecasts are essential for making informed decisions in the complex and dynamic field of electrical power management.

1.3 Objectives

1.3.1 Main Objective

The main objective of the research is to forecast the daily peak load of Nepal using single LSTM network with past load data of ten consecutive years.

1.3.2 Specific Objective

The specific objectives of the research are listed as follows:

1. To predict the daily peak load demand of Nepal.

2. To forecast the peak load demand of Nepal using LSTM and other statistical models such as Moving Average, Weighted Moving Average, Simple Exponential Smoothing, Double Exponential Smoothing and Holt-Winters Exponential smoothing and calculate the evaluation metrics such as MAE, MSE, MAPE, RMSE and R^2 Score for each model.

3. To compare the accuracies of the model with other statistical models.

1.4 Scope and Limitation of the Work

The scope of the research is to forecast the daily peak load demand of Nepal using a single LSTM network with the past autocorrelated load dataset of ten consecutive years and compare the evaluation metrics with different hyperparameter tunning. Also, the forecast accuracy of different statistical models like Moving Average with different lookback periods, Weighted Moving Average with optimized weights and Exponential Smoothing with optimized smoothing constants are also studied in the dataset. The evaluation metrics of the statistical models are also calculated and compared.

The scope of the LSTM model has gained a lot of significance since it addresses and handles long-term dependencies and prevents the vanishing gradient problem of the time series data. Although LSTMs are designed to mitigate the vanishing gradient problem, they can still occur in deep architectures, making it challenging to capture very long-range dependencies effectively. While LSTMs excel at capturing long-term dependencies, they may have limited short-term memory compared to simpler models like Gated Recurrent Units (GRUs). This can affect their ability to capture very recent patterns in the data. Choosing the right architecture and variant of LSTM (e.g., standard LSTM, GRU, bidirectional LSTM) for a specific task can be challenging and may require experimentation. Finding the right set of hyperparameters can be a time-consuming process. Like other neural networks, LSTMs can be prone to overfitting when there is limited training data. A dataset of only ten consecutive years is considered in this study. Thus, there is some probability of overfitting. Overfit models perform well on the training data but poorly on unseen data. Regularization techniques such as dropout and L2 regularization are often needed to mitigate overfitting. Also, the time resolution and time duration of the predicted peak load demand can also be considered in the study. Considering only the peak value is the limitation of this study.

Other exogenous variables such as weather variables (minimum temperature, maximum temperature, average temperature, average relative humidity, average pressure, cloud cover, rainfall volume, average wind speed, daily solar radiation, etc.), calendar variables (Time of the day, day of the week, week of the month, month of the year, season, year number, holidays, special events, etc.), economic variables (Gross National Product(GNP), Gross Domestic Product (GDP), Population Growth Rate, Consumer Growth Rate, Tariff Structure, Electricity Price, price elasticity of electricity, etc. and other variables such as customer type (residential, commercial, industrial etc.) among others can also be incorporated for better accuracy.

CHAPTER TWO : LITERATURE REVIEW

Peak load forecast, also known as peak demand forecasting, is a crucial aspect of energy management and planning. It involves predicting the maximum electricity demand that a power system or grid is likely to experience during a specific period, typically a day, week, month or year. This forecast is essential for utilities, grid operators and energy providers to ensure they can generate or procure enough electricity to meet the highest levels of demand without disruptions or overloads. Peak load forecasting is a vital tool for ensuring the reliable and efficient operation of electrical grids and energy systems. It involves the use of historical data, statistical methods and sometimes advanced technologies to predict the highest levels of electricity demand accurately. Accurate forecasting enables utilities and grid operators to make informed decisions about capacity planning, infrastructure upgrades and demand management.

2.1 Accuracy of Peak Load Forecast

The accuracy of peak load forecasts can vary widely based on several factors, including the forecasting method used, the quality and quantity of historical data, the complexity of the underlying factors influencing peak load and the forecasting horizon (e.g., short-term, medium-term, long-term). Achieving high accuracy in peak load forecasting can be challenging, but it is essential for efficient power grid management and resource planning. Here are some factors that influence the accuracy of peak load forecasts:

1. Data Quality: Accurate and reliable historical data is crucial for making precise forecasts. Errors or inconsistencies in the data can negatively impact forecast accuracy.

2. Data Quantity: Having a large and representative dataset can improve accuracy. More data points, especially for a wide range of conditions (e.g., seasons, and weather patterns), can lead to better predictions.

3. Forecasting Method: The choice of forecasting method plays a significant role in accuracy. Some methods, such as advanced machine learning techniques (e.g., neural networks), may capture complex patterns better than traditional statistical methods.

4. Weather Data: Weather conditions, especially temperature, humidity and wind speed, have a substantial impact on electricity demand. Accurate and timely weather data is crucial for forecasting accuracy.

5. Economic and Demographic Factors: Economic indicators, population growth and industrial activities can influence electricity consumption trends. Incorporating these factors into forecasting models can enhance accuracy.

6. Technological Advances: The use of advanced forecasting models, including machine learning and deep learning approaches has the potential to improve accuracy particularly when

dealing with complex and nonlinear relationships in the data.

7. Model Calibration: Proper calibration of forecasting models and regular updates to adapt to changing conditions is essential for maintaining accuracy over time.

8. Ensemble Forecasting: Combining forecasts from multiple models (ensemble forecasting) can often yield more accurate results than relying on a single model.

9. Continuous Monitoring and Feedback: Continuous monitoring of forecast performance and feedback loops for model improvement can lead to higher accuracy over time.

10. Expert Judgment: In some cases, expert judgment and domain knowledge can complement quantitative forecasting methods and improve accuracy, especially when dealing with unusual or unforeseen circumstances.

11. Seasonal and Cyclical Patterns: Identifying and incorporating seasonal and cyclical patterns in the data can lead to more accurate forecasts, especially for medium-term and long-term predictions.

12. Demand Response Programs: Forecasts should account for the impact of demand response programs, which can significantly affect peak load by encouraging consumers to reduce consumption during peak periods.

It is important to note that while peak load forecasts aim for high accuracy, they are not always perfect due to the inherent uncertainty associated with factors like weather and human behavior. Utility companies and grid operators often use probabilistic forecasting to provide a range of potential outcomes with associated probabilities, allowing them to plan for a range of scenarios. Ultimately, achieving high accuracy in peak load forecasting is an ongoing process that involves the continuous improvement of data quality, modeling techniques, and the incorporation of relevant external factors. The goal is to minimize errors and uncertainties to ensure the reliable and efficient operation of electrical grids.

2.2 Factors Affecting Peak Load Forecasting

There are many factors that affect the maximum load patterns of any country. The selection of suitable variables for peak load forecasting is related to generalized mathematical models. The precision of the model depends on the caliber of data fed into it. Several factors need to be considered for peak load forecasting, including the time factor, economic factors, weather conditions and customer factors.

Time Factor: Time holds the utmost significance in peak load forecasting, given that it exerts the greatest influence on consumer demand. Xue and Geng (2012) classified the load influences factor into three types, namely short-term, medium-term and long-term influencing factors.[8] In short-term load forecasting, the factors that come into play often manifest within specific forecast periods and typically lack a significant time duration, such as sudden weather changes. Medium-term load forecasting involves factors that persist over several forecast periods and exhibit distinct time-related characteristics, such as seasonal climate variations. For long-term load forecasting, the influence factors sustain for a long time, usually many forecasting periods, and have notably the characteristic of time duration, for example the change of gross national product and the population[9].

Economic Factor: The load pattern is additionally influenced by economic factors like industrial development, population growth, Gross Domestic Product (GDP), and electricity costs, among others. Long-term load forecasting is significantly affected by economic factors, however, it is also important for medium-term and short-term forecasting[10]. According to the different horizons of forecasting, the different economic factors could contribute to e.g. time-of-use for short-term forecasting, purchasing power for medium-term forecasting, and GDP for long-term forecasting etc[9].

Weather Condition: Multiple meteorological factors that can be taken into account for load forecasting include temperature, humidity, wind speed, and cloud cover.

- Temperature: The majority of current activities revolve around the use of electricity. There is a correlation between load and temperature to some extent. There is a positive correlating contribution between temperature and electric load curve especially in the summer season [8]. This is due to the fact that fluctuations in temperature impact people's comfort level requirements. In the summer, as temperatures rise, the demand for cooling appliances increases, leading to a higher load consumption. Conversely, in the winter, as temperatures drop, the increased use of heating appliances raises the load consumption.

- Humidity: Humidity plays a role in short-term peak load forecasting by intensifying the perceived temperature severity, particularly in summer and rainy seasons. Consequently, load consumption tends to rise on humid days during the summer.

- Wind speed: On windy days, the human body perceives temperatures to be much lower, necessitating the use of heating appliances and, consequently, leading to an increase in load consumption.

- Cloud cover: The impact of cloud cover on electricity usage is contingent upon the timing of consumption. When cloud cover occurs, it can lead to a decrease in temperature and subsequently reduce electricity consumption. Among these weather factors, temperature and humidity are the most commonly used load predictors to minimize the operational cost[9].

Customer: The load profile can vary among different customer classes. Electric utilities typically cater to diverse customer types, including residential, commercial and industrial consumers. The key factors influencing electricity consumption for customers are primarily the quantity, type and size of their electrical equipment. As electrical equipment and installations differ from one customer to another, so do their consumption patterns. There are recognized types of customers that have similar properties[9]. Due to the various factors such as weather, ergonometric factors, and social activities among many, the load appears as a non-stationary random process in time series but most of the factors that affect load have regularity that enhances the effective prediction.

2.3 Types of Peak Load Forecasting

Peak load forecasting is a critical aspect of energy management and power grid planning. There are several types of peak load forecasting methods, each designed to address specific needs and challenges. The choice of method depends on factors such as the availability of data, the nature of the load and the desired level of accuracy. some common types of peak load forecasting are described below:

1. Short-Term Peak Load Forecasting:
   (a) Hourly/Daily Forecasting: This type of forecasting predicts peak electricity demand for the next few hours or days. It is essential for real-time grid management, scheduling power generation, and load balancing.

   (b) Weekly/Monthly Forecasting: These forecasts extend the prediction horizon to a week or a month. They are valuable for resource planning and managing energy contracts.

2. Medium-Term Peak Load Forecasting:
   (a) Seasonal Forecasting: Seasonal peak load forecasting focuses on identifying peak demand patterns that repeat during specific seasons, such as summer or winter. It helps utilities plan for seasonal variations in demand and supply.

3. Long-Term Peak Load Forecasting:
   (a) Yearly/Decadal Forecasting: Long-term forecasts project peak demand over years or decades. These forecasts are essential for infrastructure planning, capacity expansion and energy policy development.

4. Static and Dynamic Peak Load Forecasting:
   (a) Static Forecasting: Static forecasts assume that the factors influencing peak load remain relatively constant over the forecast period. They are suitable for situations with stable demand patterns.

   (b) Dynamic Forecasting: Dynamic forecasts account for changing factors, such as population growth, economic fluctuations and technology adoption, to predict peak loads accurately in evolving environments.

5. Traditional Statistical Methods:
   (a) Time-Series Analysis: Techniques like moving averages, exponential smoothing and ARIMA (AutoRegressive Integrated Moving Average) models are used for historical data analysis to capture trends, seasonality and cyclic patterns in peak load data.

(b) Regression Analysis: Regression models can incorporate external variables like weather, holidays and economic indicators to improve forecasting accuracy.

6. Machine Learning Methods:

   (a) Artificial Neural Networks (ANN): ANNs, including feedforward and recurrent neural networks like LSTM and GRU, are used for capturing complex relationships in data and handling nonlinear patterns.

   (b) Random Forest: Random Forest algorithms can be applied to forecast peak loads by considering various features and their importance in the prediction.

   (c) Support Vector Machines (SVM): SVMs can be used for peak load forecasting, especially when dealing with high-dimensional data.

   (d) XGBoost: Gradient boosting algorithms like XGBoost are effective for ensemble forecasting, combining the predictions of multiple models.

7. Hybrid Forecasting Methods:

   (a) Combination of Models: Hybrid methods combine the strengths of different forecasting approaches, such as combining statistical models with machine learning models, to improve accuracy.

   (b) Expert Judgment: Sometimes, human experts are consulted to provide qualitative insights, which are then integrated with quantitative models for forecasting.

8. Demand Response Forecasting: This type of forecasting focuses on predicting the effectiveness of demand response programs, which encourage consumers to reduce their electricity usage during peak periods. It helps utilities and grid operators plan for demand response events.

9. Probabilistic Forecasting: Rather than providing a single-point estimate, probabilistic forecasting provides a range of possible outcomes with associated probabilities. It's valuable for risk assessment and decision-making.

10. Machine Learning for Load Curve Analysis: Machine learning techniques, such as clustering and anomaly detection, can be used to analyze load curves and identify patterns and outliers that can inform peak load forecasts.

Each type of peak load forecasting method has its strengths and limitations and the choice of method depends on the specific requirements of the task and the available data. Many utilities and energy companies use a combination of these methods to improve forecasting accuracy and reliability.

2.4 Output of Peak Load Forecast

The output of a peak load forecast typically includes predictions or estimates of the following:

Peak Load Value: This is the most critical output of a peak load forecast. It represents the expected maximum electricity demand that a power system or grid is likely to experience during a specific period. It is often expressed in megawatts (MW) or gigawatts (GW) and is a key parameter for grid management and resource planning.

Timestamps: Peak load forecasts usually provide the time and date at which the peak load is expected to occur. This information is essential for grid operators to prepare for high-demand periods.

Confidence Intervals or Probabilities: Some peak load forecasts provide confidence intervals or probabilities associated with the peak load value. This helps in quantifying the uncertainty of the forecast and provides a range of potential outcomes.

Load Curve: A load curve is a graphical representation of electricity demand over a specific time period, typically a day. Peak load forecasts often include load curves that show how demand is expected to vary throughout the day, highlighting peak periods.

Historical Data Comparison: To assess the accuracy of the forecast, it is common to include a comparison with historical peak load data. This allows grid operators to see how well the forecast aligns with past peak load events.

Weather Data: For short-term forecasts, especially in regions where weather strongly influences electricity demand, weather data (e.g., temperature, humidity, wind speed) associated with the forecasted period may be included. This helps grid operators understand the factors driving the peak load.

Explanatory Variables: For more complex forecasting models, the output may include the contribution of various explanatory variables or features (e.g., economic indicators, population data) that influence peak load predictions. This information can provide insights into the drivers of peak load.

Resource and Infrastructure Planning: For longer-term forecasts, the output may guide resource and infrastructure planning decisions. It can inform investment strategies for power generation facilities, transmission lines and distribution networks to meet future peak demand.

Demand Response Considerations: Peak load forecasts may consider the expected impact of demand response programs, which encourage consumers to reduce their electricity usage during peak periods. This information helps utilities and grid operators plan for demand response events.

Scenario Analysis: In some cases, multiple scenarios may be presented, reflecting different assumptions or external factors. For example, scenarios may consider the impact of extreme weather events or changes in industrial activity on peak load.

Communication and Reporting: Peak load forecasts are typically communicated to relevant stakeholders, including utility companies, grid operators, policymakers and the public. Clear and timely reporting is essential for informed decision-making. The specific output details can vary depending on the forecasting method, the forecasting horizon (short-term, medium-term, long-term), and the requirements of the stakeholders involved.

Accurate and reliable peak load forecasts are essential for ensuring the stable and efficient operation of electrical grids, as they guide resource allocation, grid management and infrastructure planning. The main difference between peak load forecast and standard load forecast is that the output is generally one value or a pair of values (eg peak load with its occurring time/date) while standard load forecast is generally a set of values. The output of the peak load forecast can be summarized as follows:

1. Forecast the total peak consumption on a given peak day[11].
2. Forecast the load usage pattern during a given peak period[12].
3. Forecast Peak Time[13].
4. Forecast Peak Value[14].
5. Forecast Peak Value and its corresponding Time[15].
6. Forecast Peak Value and forecast its occurring time separately[16].
7. Forecast peak (or together with valley) value as an additional input to produce load profile[17].

2.5 Advantage of Peak Load Forecast

Peak load forecasting has several advantages, particularly for businesses and utilities. These advantages include:

Resource Planning: Accurate peak load forecasting allows utilities and businesses to plan their resources more efficiently. They can allocate the right amount of generation capacity, transmission infrastructure and manpower to meet peak demand without overinvesting in excess capacity.

Cost Savings: By accurately predicting peak loads, utilities can optimize their resource allocation, reducing unnecessary expenditures on excess capacity. This leads to cost savings and, in turn, potentially lower electricity rates for consumers.

Reliability and Grid Stability: When utilities can predict peak loads accurately, they can ensure that they have sufficient generation capacity to meet demand. This helps maintain grid stability, reducing the risk of blackouts or brownouts during periods of high demand.

Environmental Benefits: Peak load forecasting can help utilities better plan for and integrate renewable energy sources into their grid. By aligning peak demand with peak generation, they can reduce the need to rely on fossil fuels during high-demand periods, which can lead to lower greenhouse gas emissions.

Demand Response Programs: Utilities can use peak load forecasts to implement demand response programs. These programs encourage consumers to reduce their electricity usage during peak periods in exchange for incentives, helping to flatten the load curve and reduce the need for additional infrastructure.

Energy Efficiency: Businesses and consumers can use peak load forecasts to implement energy-efficient practices and technologies to reduce their electricity usage during peak hours. This not only lowers costs but also benefits the environment by reducing energy consumption.

Improved Customer Satisfaction: By managing peak loads effectively, utilities can minimize the occurrence of power outages and voltage fluctuations during peak demand, leading to improved customer satisfaction and a better overall customer experience.

Risk Management: Peak load forecasts help utilities and businesses anticipate potential challenges associated with meeting peak demand, allowing them to develop contingency plans and strategies to mitigate these risks.

Regulatory Compliance: Many utilities are subject to regulatory requirements related to grid reliability and environmental goals. Accurate peak load forecasting helps them meet these requirements and avoid penalties.

Long-Term Planning: By analyzing historical peak load data and using forecasting models, utilities and businesses can make informed decisions about future infrastructure investments, ensuring that they can meet the growing electricity demand efficiently and sustainably.

In summary, accurate peak load forecasting provides numerous advantages, including cost savings, grid reliability, environmental benefits and improved customer satisfaction. It also supports efficient resource planning, risk management and long-term infrastructure development.

2.6 Limitation of Peak Load Forecast

Peak load forecasting is a challenging task with several limitations and complexities. Accurate predictions of peak electricity demand are crucial for efficient grid management and resource planning. However, there are various factors that make peak load forecasting inherently uncertain and subject to limitations. Some of the key limitations of peak load forecasting are:

Complexity of Factors: Peak load is influenced by a wide range of factors, including weather conditions, economic activity, population growth, industrial processes, technological advancements and consumer behavior. Predicting how all these factors will interact and affect electricity demand can be complex.

Weather Sensitivity: Weather has a significant impact on electricity demand, especially in regions with extreme climates. Variability in temperature, humidity, wind speed and sunlight can lead to sudden and unpredictable changes in peak load.

Nonlinear Patterns: Peak load data often exhibits nonlinear patterns, making it challenging to model accurately using traditional statistical methods. Nonlinearities can arise from factors such as consumer behavior and the adoption of renewable energy sources.

Data Quality and Availability: The accuracy of peak load forecasts heavily relies on the quality and quantity of historical data. Data gaps, inaccuracies, or limited historical records can hinder forecasting accuracy.

Data Time Granularity: Peak load forecasting requires data at a high time granularity (e.g., hourly or sub-hourly). In some regions, such data may not be readily available or may require substantial processing to use effectively.

Emerging Technologies: The integration of emerging technologies, such as electric vehicles (EVs) and distributed energy resources (DERs) can introduce additional complexities into forecasting as they impact both supply and demand patterns.

Regulatory and Policy Changes: Changes in energy policies, regulations and market dynamics can have unforeseen effects on peak load. Forecasters may not always have access to timely information about these changes.

Extreme Events: Unpredictable events like natural disasters, equipment failures or cybersecurity threats can lead to sudden spikes in demand or supply disruptions making forecasting challenging.

Demand Response Programs: The effectiveness of demand response programs, which encourage consumers to reduce their electricity usage during peak periods can be uncertain. Accurately predicting the impact of these programs is challenging.

Short Lead Times: Short-term peak load forecasting, which is essential for grid management, often requires forecasts with very short lead times (e.g., hours or minutes) adding to the difficulty of accurate prediction.

Inaccurate Assumptions: Assumptions about future conditions, such as economic growth rates, energy efficiency improvements and consumer behavior, can lead to forecasting errors.

Model Complexity: More complex forecasting models may require substantial computational resources and expertise in machine learning or data science.

Ensemble Forecasts: Combining forecasts from multiple models (ensemble forecasting) can improve accuracy, but it adds complexity and requires careful model selection and integration.

Probabilistic Forecasts: Accurate peak load forecasts should ideally provide not only point estimates but also measures of uncertainty and probabilistic forecasts. Developing reliable probabilistic forecasts can be challenging.

Given these limitations, peak load forecasting is an ongoing research area, and practitioners often use a combination of methods, real-time monitoring, and expert judgment to improve accuracy and adapt to changing conditions. Advanced forecasting techniques, such as machine learning and deep learning, are also being explored to address some of these challenges.

2.7 Recurrent Neural Network(RNN)

A popular method in Deep Learning for modeling sequential data is the Recurrent Neural Network (RNN). RNNs were the conventional choice for handling sequential data. A deep feedforward model might require specific parameters for each element in a sequence and might unable to generalize to variable-length sequences. In contrast, Recurrent Neural Networks employ the same set of weights for every element in a sequence, reducing the number of parameters and enabling the model to handle sequences of varying lengths. Due to their design, RNNs can also generalize to structured data beyond just sequential data, such as geographical or graphical data. Recurrent neural networks, like many other deep learning techniques, have been around for quite some time. They were initially developed in the 1980s, but it wasn't until recent years that we fully realized their potential. The introduction of long short-term memory (LSTM) in the 1990s, along with increased computational power and the abundance of data at our disposal, has propelled RNNs to the forefront of the field.

Recurrent Neural Networks (RNNs) are a category of neural networks designed for modeling sequential data. RNNs, constructed as extensions of feedforward networks, exhibit behavior akin to that of the human brain. Neural networks emulate the functioning of the human brain in the realms of artificial intelligence, machine learning, and deep learning, enabling computer algorithms to identify patterns and address common problems.

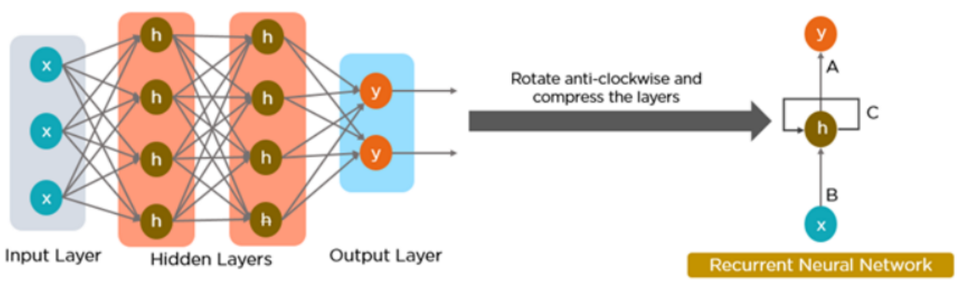
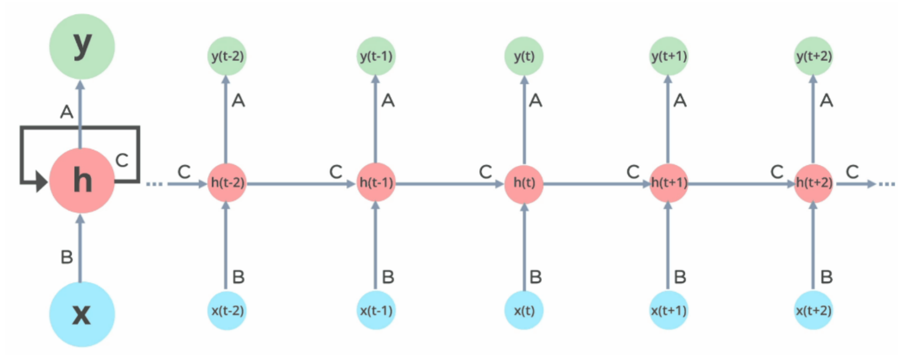


Figure 2.1: Recurrent Neural Network

In traditional neural networks, all inputs and outputs are treated as independent entities. Yet, in certain situations, like predicting the next word in a sentence, the preceding words are crucial, necessitating the retention of previous context. Consequently, Recurrent Neural Networks (RNNs) were developed, incorporating a Hidden Layer to address this issue. The central element of an RNN is the Hidden state, which retains specific information related to a sequence. RNNs feature a Memory unit that preserves all computation-related information. They use identical parameters for each input, as they perform consistent operations across all inputs and hidden layers. Information within recurrent neural networks circulates through a loop to reach the central hidden layer. The input layer, denoted as 'x,' accepts and processes the input data for the neural network before transmitting it to the middle layer.

The central layer 'h' in an RNN can include multiple hidden layers, each having its unique activation functions, weights, and biases. An RNN proves beneficial when the parameters of distinct hidden layers remain unaffected by the preceding layer, meaning there is no memory within the neural network.

The Recurrent Neural Network standardizes various activation functions, weights, and biases, ensuring that each hidden layer possesses identical characteristics. Instead of generating multiple distinct hidden layers, the RNN constructs a single layer and iterates through it as needed.



Source: Simplilearn.com

Figure 2.2: Recurrent Neural Network working principle

2.7.1 Limitation Of RNN

Recurrent Neural Networks (RNNs) are a class of neural networks designed for processing sequences of data, where each element in the sequence depends on the previous ones. While RNNs have proven to be effective in many sequence-related tasks, they also have several limitations: Vanishing and Exploding Gradients: RNNs are prone to the vanishing gradient problem, where gradients can become very small as they are backpropagated through time. This can hinder the training of long sequences and make it challenging for the network to capture long-term dependencies. Conversely, exploding gradients can also occur, leading to unstable training.

1. Short-Term Memory: Standard RNNs have limited short-term memory. They can effectively capture dependencies within a short window of previous time steps but struggle with longer-range dependencies. This limitation can make it challenging to model complex sequences.

2. Lack of Parallelism: RNNs process sequences sequentially, which makes them less suitable for parallel computation. This sequential nature can lead to slower training times compared to other architectures, like convolutional neural networks (CNNs).

3. Difficulty with Irregular Time Series: RNNs assume that sequences have a fixed length and regular time intervals. Handling irregular time series data or data with missing values can be challenging and may require additional preprocessing.

4. Fixed-Length Representations: RNNs typically produce fixed-length representations regardless of the length of the input sequence. This can be problematic when dealing with variable-length sequences, such as text of varying lengths.

5. Training Instability: Training RNNs can be challenging due to their sensitivity to hyperparameters. Careful tuning of learning rates, initialization methods, and regularization techniques is often required to prevent training instability.

6. Difficulty Capturing Long-Term Dependencies: While certain RNN variants like LSTM and GRU were designed to address the vanishing gradient problem and capture longer-term dependencies, they are not always able to capture very long-range dependencies effectively.

7. Memory and Computational Resources: Models with a large number of recurrent units can be memory-intensive and computationally expensive to train and deploy. This can limit their applicability on resource-constrained devices.

8. Lack of Global Context: RNNs have a limited field of view, meaning they focus on the current input and recent history. They may struggle to capture global context or patterns that require information from distant elements in the sequence.

9. Difficulty with Parallelism: Training RNNs in parallel across multiple sequences or time steps can be challenging, limiting their scalability and efficiency on modern hardware architectures like GPUs and TPUs.

10. Limited Interpretability: RNNs can be challenging to interpret due to their complex internal dynamics. Understanding why a particular prediction was made can be non-trivial.

Despite these limitations, RNNs have been a foundational technology in sequence modeling and have paved the way for more advanced models like Transformer-based architectures. Researchers have also developed variations of RNNs, such as LSTM (Long Short-Term Memory) and GRU (Gated Recurrent Unit), which aim to address some of these limitations by improving the network's ability to capture long-range dependencies and mitigate the vanishing gradient problem. However, for some tasks and datasets, other architectures like CNNs or Transformers may offer superior performance. The choice of architecture should depend on the specific requirements of the task at hand.

2.8 Long Short Term Memory

Long Short-Term Memory Networks, or LSTM networks, are a category of deep learning models within sequential neural networks that have the unique ability to maintain and retain information over extended sequences. They represent a specialized variant of Recurrent Neural Networks (RNNs) and have been specifically designed to address the problem of vanishing gradients, which is a challenge often encountered in traditional RNNs. In Python, LSTM can be implemented conveniently through the Keras library.

Distinguishing itself from conventional neural networks, LSTM incorporates feedback connections, enabling it to process entire sequences of data rather than just isolated data points. This attribute makes LSTM exceptionally adept at recognizing and forecasting patterns within sequential data, such as time series, textual information, and speech signals.

2.8.1 Architecture of LSTM

The architecture of a Long Short-Term Memory (LSTM) model consists of multiple LSTM layers stacked together. LSTMs are a type of recurrent neural network (RNN) designed to capture long-range dependencies in sequential data. The overview of the architecture of an LSTM model can be summarized as:

1. Input Layer:

   The input layer receives sequential data, which could be in the form of sequences, time series, or sequences of features. Each data point in the sequence is represented as a feature vector.

2. LSTM Layers:

   The core of the LSTM model consists of one or more LSTM layers. Each LSTM layer contains multiple LSTM units or cells. The number of units or cells in each layer is a hyperparameter set during model design.

   Each LSTM cell has three main components:

   (a) Cell State (C_t): The cell state is a vector that carries information over time steps. It can capture long-term dependencies by selectively adding or removing information through gates.

   (b) Hidden State (h_t) The hidden state is another vector that is used for making predictions at each time step. It is also influenced by the cell state and gates.

   (c) Gates LSTMs have three gates that control the flow of information within the cell:

      i. Forget Gate (f_t): Decides what information to discard from the cell state.

      ii. Input Gate (i_t): Determines which new information to store in the cell state.

iii. Output Gate (o_t): Regulates what information is used to generate the hidden state and make predictions.

3. Output Layer: The output layer can vary depending on the specific task. For example: For time series forecasting, the output layer may consist of a single neuron that predicts the next value in the sequence. For classification tasks, there may be multiple neurons, one for each class, with a softmax activation function for multiclass classification. For regression tasks, a single neuron with a linear activation function can be used to predict a continuous value.

4. Activation Functions: Within each LSTM cell, activation functions such as the sigmoid and hyperbolic tangent (tanh) functions are used to control the information flow through the gates and to update the cell state and hidden state.

5. Sequence Processing: LSTM models process input sequences one-time step at a time, starting from the beginning and proceeding sequentially. The output at each time step is influenced by the input at that time step and the previous hidden state.

6. Backpropagation Through Time (BPTT): Training an LSTM model involves backpropagation through time, where gradients are calculated for each time step, and model parameters are updated to minimize a loss function. BPTT helps the model learn to capture sequential dependencies.

7. Stacking LSTM Layers: To capture complex dependencies, LSTM layers are often stacked on top of each other. This creates a deep LSTM architecture, similar to deep feedforward neural networks. Stacking multiple LSTM layers can help the model learn hierarchical features.

8. Dropout and Regularization: Dropout layers or other regularization techniques may be added between LSTM layers to prevent overfitting and improve generalization.

9. Bidirectional LSTMs (Optional): In some cases, bidirectional LSTMs may be used. These models process sequences in both forward and backward directions, which can capture dependencies in both past and future contexts.

10. Initial State (Optional): Some applications may involve providing an initial state or context vector to the LSTM model. This can be useful for tasks that require maintaining memory across multiple sequences or segments.

11. Model Output: The final output of the LSTM model depends on the specific task. It may be a single prediction for each time step in a sequence, a single prediction for the entire sequence, or a sequence of predictions.

12. Loss Function: The choice of loss function depends on the task. Common loss functions include mean squared error (MSE) for regression, categorical cross-entropy for classification, and various custom loss functions for specialized tasks.

The architecture and hyperparameters of an LSTM model can be customized to suit the requirements of the specific task, and they may vary from one application to another. Training an LSTM model typically involves optimizing the model's parameters to minimize the chosen loss function using techniques such as gradient descent or its variants.

At a high level, LSTM operates in a manner quite similar to an RNN cell. The LSTM network architecture can be broken down into three distinct components, each with its own specific role, as depicted in the Figure. These components work together to execute their respective functions. The first component is responsible for making a decision regarding whether the information arriving from the previous time step should be retained in memory or is deemed irrelevant and can be discarded. The second component involves the cell's attempt to acquire new information based on the input provided to it at the current time step. In the final component, the cell transfers the updated information from the present time step to be utilized in the subsequent time step. This entire sequence of operations within LSTM is considered as single time step.

These three components within an LSTM unit are referred to as gates, and they regulate the flow of information into and out of the memory cell or LSTM cell. The first gate is the Forget gate, the second is the Input gate, and the final one is the Output gate. An LSTM unit, comprising these three gates and a memory cell or LSTM cell, can be considered as a layer of neurons in a conventional feedforward neural network. Each neuron in this layer possesses a hidden state and a current state.

Similar to a simple RNN, an LSTM also maintains a hidden state, where H(t-1) signifies the previous timestamp's hidden state, and H(t) represents the hidden state at the current timestamp. Additionally, the LSTM features a cell state denoted as C(t-1) for the previous timestamp and C(t) for the current timestamp.

Here the hidden state is known as Short-term memory, and the cell state is known as Long term memory.

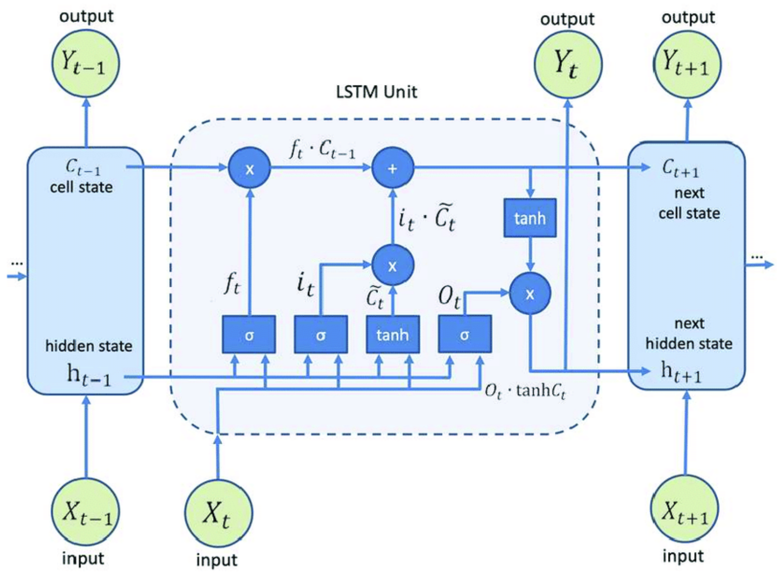


Figure 2.3: Basic Structure of LSTM Model

Where, the symbols i_t, f_t and o_t represent the input, forget, and output gates, as depicted in the figure. These gates play a crucial role in learning and retaining sequential information from previous cells. C_t and $\hat{C_t}$ denote the new current cell state and the new candidate value for the cell state, respectively. Similarly, cell state acts as a transport highway that transfers relative information way to the sequence chain. A memory to the network which carries out the information from the earlier state to the last state which helps to reduce the effect of short-term memory. The computations of LSTM cells are referred to as:

$$
\begin{aligned}
h_t &= f(h_t - 1), x_1) \\
i_t &= \sigma(w_i((h_t - 1), x_t) + b_i) \\
f_t &= \sigma(w_f((h_t - 1), x_t) + b_f) \\
o_t &= \sigma(w_o((h_t - 1), x_t) + b_o) \\
\hat{C} &= tanh(w_c(h_t - 1, x_t) + b_c) \\
C_t &= f_t.C_t - 1 + i_t.\hat{C_t} \\
h_t &= o_t.tanh(C_t)
\end{aligned}
\tag{2.1}
$$

Referring to equation 2.1, the determination to either block the signal (resulting in a 0 output) or allow it to pass (resulting in a 1 output) is contingent on the outcomes of the gates. These decisions then lead to the updating of the old cell state into the new cell state.

2.8.2 Advantages of LSTM Networks

Long Short-Term Memory (LSTM) networks are a type of recurrent neural network (RNN) architecture designed to address some of the limitations of traditional RNNs. LSTMs offer several advantages that make them particularly well-suited for various sequence modeling tasks:

1. Handling Long Sequences: LSTMs excel at capturing long-range dependencies in sequences. They are less prone to the vanishing gradient problem compared to traditional RNNs, which makes them capable of learning from and remembering information over extended time steps.

2. Sequential and Temporal Data: LSTMs are specifically designed for sequential and temporal data, making them well-suited for tasks such as natural language processing (NLP), speech recognition, time series forecasting, and more.

3. Variable-Length Sequences: LSTMs can handle variable-length sequences, adapting to the length of the input data. This flexibility is valuable when working with data of varying sizes, such as sentences of different lengths in NLP.

4. Gating Mechanisms: LSTMs incorporate gating mechanisms, such as the forget gate and input gate, which allow them to control the flow of information and selectively update the cell state. These gates enable LSTMs to handle and discard irrelevant information while retaining important information, enhancing their ability to capture context.

5. Parallelization: Although RNNs process sequences sequentially, LSTMs can be parallelized in some aspects, making them more computationally efficient during training and inference compared to traditional RNNs.

6. Less Susceptible to Exploding Gradients: LSTMs are designed to mitigate the issue of exploding gradients during training, which can occur in deep networks. This property facilitates the training of deep LSTM architectures.

7. Memory Cell: LSTMs have an explicit memory cell that can store and access information over time. This memory cell helps capture and remember relevant information, improving their ability to handle sequential data with long-term dependencies.

8. Stateful Processing: LSTMs can be used in stateful mode, where the hidden state from the previous sequence can be carried over to the next sequence. This is beneficial for tasks where context from one sequence is essential for understanding the next, such as speech recognition.

9. Effective for Sequence-to-Sequence Tasks: LSTMs are widely used in sequence-to-sequence tasks, including machine translation and text generation, where they can encode input sequences and generate corresponding output sequences effectively.

10. Wide Range of Applications: LSTMs have been successfully applied to a wide range of applications, including natural language understanding, sentiment analysis, speech recognition, handwriting recognition, autonomous driving, and more.

11. Interpretability: Compared to some other deep learning architectures, LSTMs can be relatively interpretable. You can visualize the activations and states of individual cells, helping to understand which parts of the input sequence are influential in making predictions.

Overall, LSTMs have become a fundamental tool in deep learning for sequence modeling tasks due to their ability to capture complex temporal dependencies, handle long sequences, and mitigate some of the issues associated with traditional RNNs. However, it's worth noting that LSTMs are not always the best choice for every sequence modeling problem, and the choice of architecture depends on the specific task and dataset characteristics.

2.9 Other Statistical Forecasting Models

2.9.1 Moving Average

The Moving Average Model utilizes previously predicted errors to estimate future values. Within this model, there is a parameter denoted as 'q,' representing the size of the moving average window used to calculate linear combinations of errors. The mathematical equation for this model is expressed as follows:

$$Y(t) = \mu + \phi(k) * \epsilon(t - k) \tag{2.2}$$

where,

μ is the mean of the series.

$\epsilon(t-k)$ is the past forecasted error
$\phi(k)$ is the weight associated with error value.

2.9.2 Weighted Moving Average

The Weighted Moving Average is a calculation method in which each data point in a dataset is multiplied by a predetermined weight, and the results are then summed together. This approach bears similarities to the Simple Moving Average (SMA), but it distinguishes itself by assigning different weights to individual data points within a specific time frame. These weights are used to provide distinct levels of importance or significance to particular data points, often based on factors like their chronological order or other relevant criteria. The primary aim of this technique is to smooth out a data series, reducing noise and facilitating the recognition of underlying data trends. It's important to note that the sum of the weights should equal 1 (or 100%).

2.9.3 Exponential Smoothing

Exponential smoothing is a time series forecasting method that assigns exponentially decreasing weights to older observations, with more recent observations receiving higher weights. These methods find extensive application in forecasting due to their simplicity and their ability to encompass various facets of time series data, such as trends and seasonality.

Simple Exponential Smoothing

Simple Exponential Smoothing is a time series forecasting method particularly useful for forecasting when there is a consistent level of data variation over time and no significant trends or seasonality present in the data. This approach is straightforward and involves calculating a weighted average of past observations, where more recent observations carry greater weight. Simple exponential smoothing focuses on capturing the underlying level of the time series data. The level at a point t is:

$$L(t) = \alpha * y(t) + (1-\alpha) * y(t-1) \tag{2.3}$$

where,
y(t) is the recent observation.
y(t-1) is the previous value.
α is the smoothing parameter whose value lies between 0 and 1. It determines the weight given to the most recent data point. A higher alpha places more weight on recent data, making the forecast more responsive to changes.

Double Exponential Smoothing

This approach is referred to as Holt's trend model or second-order exponential smoothing. Double exponential smoothing is employed in time-series forecasting when the data displays a linear trend but lacks any seasonal pattern. The fundamental concept is to introduce a component that accounts for the potential presence of a trend in the series.

In addition to the alpha (α) parameter, Double exponential smoothing incorporates another smoothing factor known as beta (β) , which regulates the decay of the influence of change in trend. This method accommodates both additive trends (smoothing with a linear trend) and multiplicative trends (smoothing with an exponential trend). The forecast equation is:

$$Y(t+1) = I(t) \ + B(t) \tag{2.4}$$

The trend component is calculated as:

$$B(t) = \beta * I(t) - I(t-1) + (1-\beta) * B(t-1) \tag{2.5}$$

Where,
I(t) is the level component.
B(t) is the trend component.

Holt-Winters or Triple Exponential Smoothing

This method represents the most advanced variation of exponential smoothing and is employed in time series forecasting when the data exhibits linear trends and seasonal patterns. The technique involves applying exponential smoothing three times, encompassing level smoothing, trend smoothing, and seasonal smoothing. It introduces a new smoothing parameter known as gamma (γ) to control the impact of the seasonal component.

The triple exponential smoothing method is alternatively known as Holt-Winters Exponential Smoothing, named after its creators, Charles Holt and Peter Winters. This approach focuses on three primary elements when making predictions: it considers the average value, the trend, and seasonality. The forecast equation is:

$$Y(t+1) = I(t) \ + B(t) + s(t+1-m) \tag{2.6}$$

The trend component is calculated as:

$$B(t) = \beta * I(t) - I(t-1) + (1-\beta) * B(t-1) \tag{2.7}$$

Where,
I(t) is the level component.
B(t) is the trend component.
m is the number of times a season repeats in time period.

2.10 Related Works

In the past, numerous efforts were made to predict the electricity demand in Nepal, taking into account various influencing factors. The load forecasting study conducted by the System Planning Department of the Nepal Electricity Authority in 2015, covering the time period from the fiscal year 2014/15 to the fiscal year 2033/34, assumes that the growth of electricity consumption can be explained by an equation that incorporates factors like income, the number of new connections, unit consumption, and the price of electricity. In this forecasting analysis, historical consumption data was examined at a sectoral level, associating it with explanatory variables such as production and income. Five consumer categories that are homogeneous with respect to demand determinants have been adopted: domestic, industrial, irrigation, commercial and others.[18].

Likewise, the Water and Energy Commission Secretariat (WECS) has released an energy demand forecast report (2015-2040) known as the Model for Energy Demand Analysis (MAED), which is a detailed, bottom-up model. Within the spectrum of energy forms, the prediction of electricity demand and, consequently, the total installed power generation capacity in the country presents a significant challenge for two primary reasons. First, it is the responsibility of any government to ensure an adequate power supply to meet the demands of the entire economic sector. Second, this field attracts substantial interest from private sector developers, both at the national and international levels, in addition to public sector investments. The complexity is further compounded by Nepal's heavy reliance on hydropower in its energy sector, which naturally involves extended development timelines and substantial upfront capital investment.

In the study, three distinct strategic economic growth scenarios have been delineated for the model:
a) A business-as-usual (BAU) economic scenario, featuring a 4.5 percent GDP growth rate.
b) A reference-level economic scenario, characterized by a 7.2 percent GDP growth rate.
c) A high-level economic scenario with an ambitious 9.2 percent GDP growth rate.
These scenarios serve as key elements in the analysis.

The demand for electricity is primarily contingent on factors such as the population and economic conditions, including GDP, of a country. Forecasting energy demand, including electricity load, is an ongoing process. Given the absence of current data in many respects, this forecasting study relies on several assumptions. Therefore, it is advisable to regularly update the electricity forecast with the inclusion of recent data and information, preferably on an annual basis, and share the results officially with the public. Energy demand projection is the basic prerequisite for the formulation of integrated energy policy, preparing plans and defining the activities for implementation. Different tools and techniques can be applied for demand projection[19].

Advanced demand forecasting methods often rely on more sophisticated approaches. One commonly employed categorization involves top-down and bottom-up models. Top-down models typically concentrate on a higher-level overview of the analysis, whereas bottom-up models identify specific, homogeneous activities or end-uses for which demand is projected. Econometric models, grounded in economic theories, come under the first category[20]. Different forecasting techniques have been attempted to forecast the load of Kathmandu Valley previously. As per [21], the best performance among the models are obtained from the deep learning model (LSTM) which is 1.56% for the dataset of Kathmandu Valley.

Very few attempts have been made considering the dataset of peak demand of the overall country. As per [22], the analysis of electricity demand time series indicates that the demand becomes stationary after a second difference, and the most suitable fitted model is ARIMA (4, 2, 0). This implies that the examined time series data exhibit an autoregressive order of 4, achieve stationarity at the 2nd difference, and have a moving average order of 0. The regression coefficients at lag 4 demonstrate a relatively rapid convergence to their mean value, suggesting that the electricity demand undergoes periodic changes every four years.

Also as per [23]-[24], the LSTM-based forecast is better than other methods and has the potential to further improve the accuracy of the forecast. The results of the comparison demonstrate that, with the right data, LSTM networks excel not only in short-term but also in long-term forecasting, surpassing traditional methods. The LSTM network, a specific type of recurrent neural network (RNN), possesses the capabilities for both short-term and long-term learning, making it well-suited for time series modeling and forecasting applications. Additionally, to enhance accuracy and capture all seasonal patterns and trends, an increasing number of variables can be incorporated into LSTM.

In this research, a deep learning model utilizing a single LSTM network is employed to predict the peak demand for the entire country. This LSTM model is then compared with other conventional statistical models. The findings reveal that the LSTM model surpasses traditional statistical methods like MA, WMA, and ES, demonstrating superior performance in forecasting autocorrelated time series data with significantly reduced errors.

CHAPTER THREE : RESEARCH METHODOLOGY

3.1 Dataset Collection

The required peak load information for a 10-year period spanning from April 13, 2012, to April 13, 2023, for the purpose of peak load forecasting, has been acquired from LDC, Nepal Electricity Authority.

Sample data including date and load can be tabulated in the table below and represented in the chart as in Fig 3.1.

Sno	Date	Load
1	4-13-2012	900.8
2	4-14-2012	889
3	4-15-2012	902.6
4	4-16-2012	683
5	4-17-2012	816.2
6	4-18-2012	859
7	4-19-2012	832.2
8	4-20-2012	846
...	...	...

Table 3.1: Peak load data set

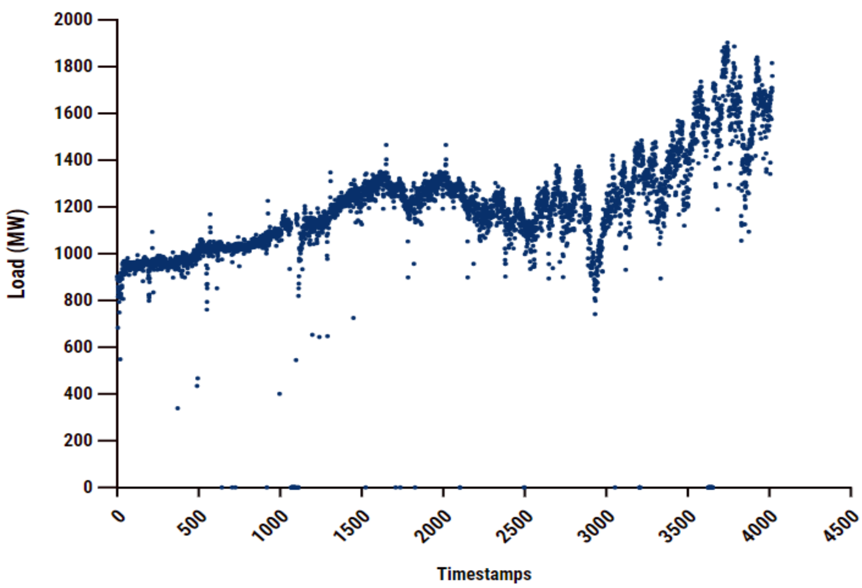


Figure 3.1: Scatterplot of system peak load dataset

3.2 Algorithm

3.2.1 Handling missing data

Any missing data in the time series data can be imputed by the following methods:
1. Mean Imputation
2. Last Observation Carried Forward
3. Linear Interpolation
4. Seasonal Interpolation

Among the four methods, the linear interpolation method is used for imputing missing values in our data since there is no distortion in the seasonality and the imputed value seems to be accurate. Missing values may be accounted for the errors in data entry. Linear interpolation is a method of interpolation that entails creating additional values using an existing dataset. It is accomplished by geometrically connecting two consecutive data points on a graph or plane with a straight line. Its simplest formula is given below.

$$y = y_1 + \frac{(x - x_1)(y_2 - y_1)}{x_2 - x_1} \tag{3.1}$$

3.2.2 Handling Outliers

Outliers are data points that deviate significantly from the rest of the observations, lying far outside the typical range. Outliers may be present because of entry errors, measurement errors or natural errors. Outliers can be visualized by boxplot. Any data that are less than 0.020% of quantile are considered to be outliers since it may be due to the failure of the grid or forced outage. Trimming outlier is one of the methods of handling outlier which help us to exclude the outlier values from our analysis. By applying this technique, our data becomes thin when more outliers are present in the dataset.

3.2.3 Data Normalization

The primary goal of data normalization is to modify our observations in such a way that they conform to a normal distribution, also known as the Gaussian distribution or bell curve. In this distribution, there is a roughly equal distribution of observations above and below the mean, with the mean and median being identical, and more observations clustered near the mean. To achieve data normalization, the min-max normalization method is employed, which scales each feature to a predefined range. This process rescales the features within the range of [0, 1]. The Min-Max transformation can be accomplished using the following formula.

$$X_(Norm) = \frac{X - X_(min)}{X_(max) - X_(min)} \tag{3.2}$$

3.2.4 Test Train Split

The process of splitting time series data for verification and validation differs from that of other machine learning algorithms. In time series data, the data points are inherently organized in a specific chronological order, typically by dates or time values. In contrast to other machine learning algorithms, where data is usually divided into training, testing, and validation sets, time series data is divided into training and test sets. Here, the validation approach involves utilizing the data from the most recent period to validate the model's performance.

In the context of this particular study, a one-step validation method is applied. This means that 3,208 data points covering the period from April 13, 2012, to February 3, 2021, are designated as the training dataset. Subsequently, 803 data points, spanning from February 4, 2021, to April 13, 2023, are employed as the test dataset to assess the model's predictive capabilities.

1. One-Step Validation
   In one-step validation, the test data is exactly after the train data

2. Multi-step Validation
   Multi-step validation is the same as one-step validation with the difference that in multi-step we do not consider the exact next point after train data points.

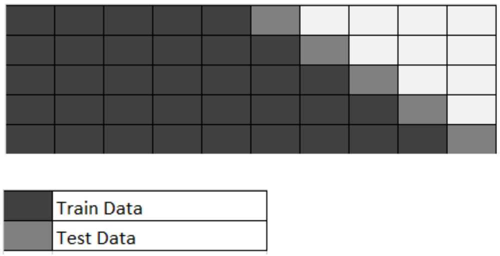


Figure 3.2: One Step Validation

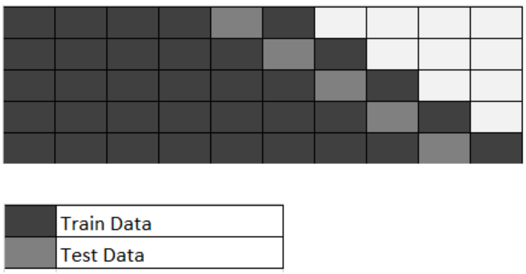


Figure 3.3: Multistep Validation

3.2.5 Building LSTM Model

In the experiment, an LSTM model is constructed with a sequence of layers, starting with a single input layer, followed by an LSTM layer, a dropout layer, and finally a dense output layer. The LSTM layers are configured with 64 and 32 LSTM units, each having a different lookback, and a dropout rate of 0.2. These LSTM layers utilize the hyperbolic tangent (tanh) activation function.

The rectified linear activation function, abbreviated as ReLU, is a piecewise linear function that directly outputs the input if it's positive; otherwise, it produces a zero. ReLU has become the default activation function for many neural networks due to its ease of training and its tendency to deliver superior performance.

For optimization, the experiment employs the Adaptive Moment Estimation (Adam) method, which computes adaptive learning rates for each parameter. Adam is an efficient stochastic optimization technique that relies on first-order gradients and consumes minimal memory. It maintains an exponentially decaying average of past gradients.

As per the author of Adam, the hyperparameter 1 is typically set around 0.9, and hyperparameter 2 is commonly set to 0.99. Epsilon is usually chosen as e-7. Hyperparameters like the number of LSTM units and the lookback period were fine-tuned to ensure accurate forecasting results.

3.2.6 Other Statistical Forecasting Models

The peak load is also forecasted with other statistical forecasting models like Moving Average methods and Exponential Smoothing Methods. In Moving Average, Simple Moving Average and Weighted Moving Average is considered with different look back period of 1 day, 2 day, 3 days, 4 days, 7 days, 1 month and 2 months. In the weighted average method the corresponding weights are being optimized using the MS Excel Solver. In Exponential Smoothing Methods, Simple Exponential Smoothing, Double Exponential Smoothing and Holt-Winters Exponential Smoothing are considered and the Smoothing Constants are being optimized by the MS Excel add-ins. Also, the evaluation metrics MAE, MSE, RMSE, MAPE and R^2 of all the statistical models were compared.

3.3 Research workflow

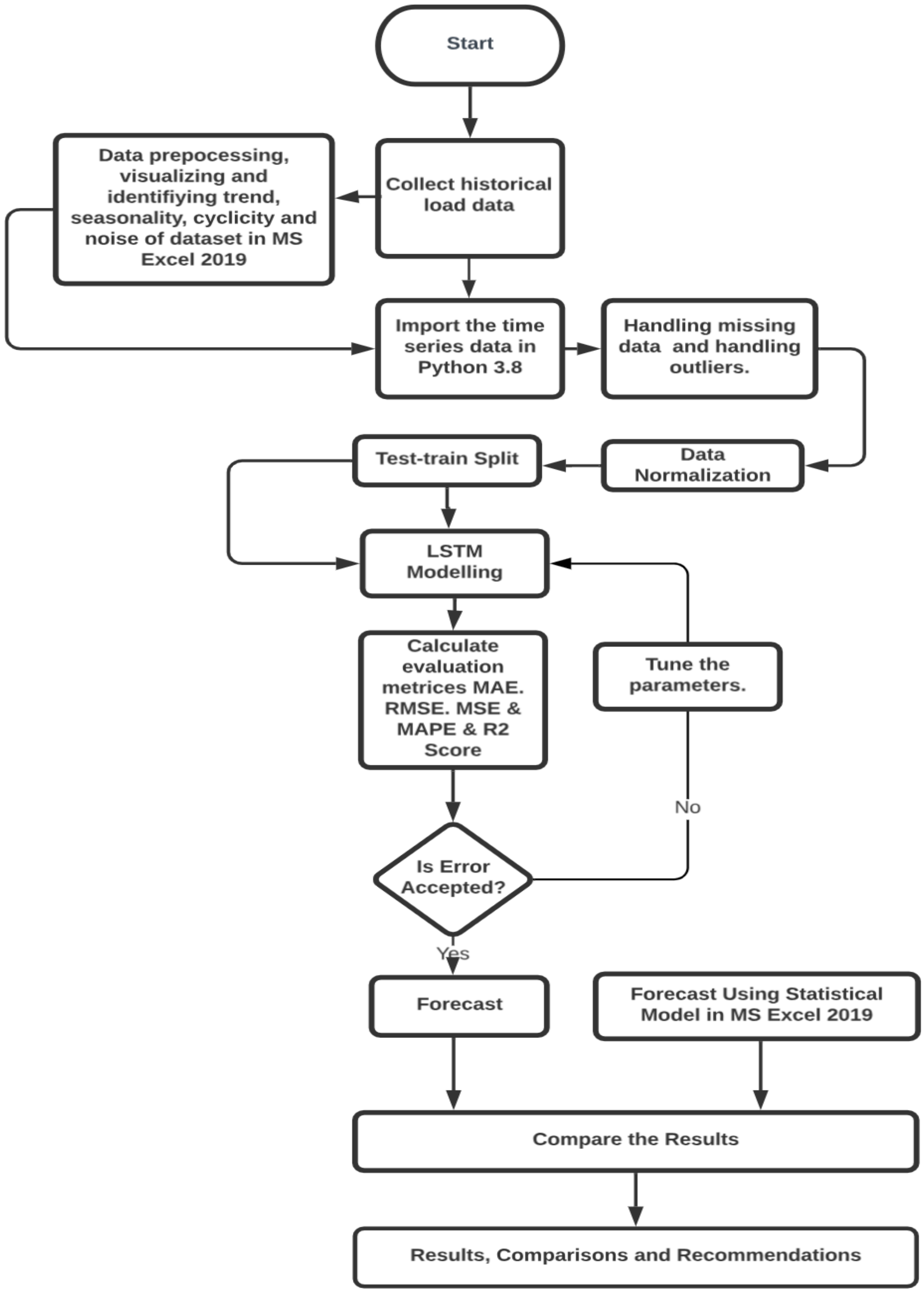


Figure 3.4: Research work flow

3.4 Tools to be used

Data preprocessing, feature engineering, visualization, trend, seasonality, cyclicity, and noise detection are conducted using Microsoft Excel. The evaluation of statistical forecasting models is performed with the XLSTAT 2023 add-ins in Microsoft Excel. Missing data and outliers are detected in Excel, and the data is subsequently exported to Python 3.8. Within Python, the datasets are processed through the LSTM model. The scripts are executed on a laptop equipped with 16GB of RAM and a core i5 processor.
The packages that are used during the experiment are as follows:

3.4.1 Python

Python, a high-level scripting language, was created by Guido van Rossum in the late 1980s at the Netherlands National Institute of Mathematics and Computer Science. Two primary Python versions currently exist: Python 2 and Python 3. Python serves as the predominant programming language for many machine learning research and development in Machine Learning. A number of factors have played a role in Python's widespread adoption and its appropriateness for machine learning applications:

- Python's ease of learning.
- The availability of numerous libraries and frameworks designed for machine learning in Python.
- Strong community and corporate support for Python.
- Python's portability and extensibility.

3.4.2 Keras

Keras is a high-level neural network API built on Python, capable of operating on TensorFlow, CNTK, or Theano frameworks. Its primary objective is to facilitate swift experimentation. This framework streamlines the prototyping process by offering user-friendly features, modularity, and expandability. Keras is also proficient in supporting convolutional networks, recurrent networks, and hybrid combinations of both.

3.4.3 Jupyter Notebook

The Jupyter Notebook is an open-source web application that empowers users to generate and distribute live code, computations, visualizations, and explanatory textual documents. It represents an innovative evolution beyond the traditional console-based approach to interactive computing. Jupyter Notebook finds extensive application in tasks such as data cleansing and transformation, numerical simulations, statistical modeling, data visualization, machine learning, and various other domains.

3.4.4 Colaboratory

Colaboratory, often referred to as "Colab," is a product developed by Google Research. Colab enables individuals to create and run Python code directly in their web browsers and is particularly well-suited for tasks involving machine learning, data analysis, and educational purposes. In technical terms, Colab is a hosted Jupyter notebook service that eliminates the need for any setup, granting free access to computing resources, including GPUs. Importantly, Colab is completely free to use.

3.5 Verification and Validation

3.5.1 Optimization Curve

An optimization curve in machine learning typically refers to a graphical representation of how a certain performance metric (often referred to as the "objective" or "loss" function) changes over the course of training an algorithm, typically through multiple iterations or epochs. It helps machine learning practitioners and researchers understand how well their model is learning and how to improve it. The components of an optimization curve are:

1. X-Axis (Horizontal Axis): The x-axis represents the training iterations or epochs. Each iteration or epoch corresponds to a complete pass through the training dataset. It is the unit of measurement for training progress.

2. Y-Axis (Vertical Axis): The y-axis represents the value of the performance metric being measured. The choice of metric depends on the machine-learning task. For example, it could be a loss function (e.g., mean squared error for regression, cross-entropy loss for classification) or an evaluation metric (e.g., accuracy, F1 score, or R-squared).

3. Training Curve: This curve represents how the performance metric changes on the training data as the model learns. It shows how well the model fits the training data. In most cases, the training loss is expected to decrease over iterations or epochs. This indicates that the model is learning from the data.

4. Validation Curve: If available, this curve shows how the performance metric changes on a separate validation dataset. It helps assess how well the model generalizes to unseen data. Typically, you would want to see the validation metric decrease as training progresses. If the validation metric starts to increase or stagnate, it may indicate overfitting.

In an ideal scenario, both the training and validation curves decrease and eventually plateau. This indicates that the model has learned the underlying patterns in the data without overfitting. The point at which the validation curve plateaus is often the "point of best generalization." If the training curve continues to decrease while the validation curve increases, it suggests overfitting, where the model is fitting the training data too closely and losing its ability to generalize while if both the training and validation curves remain high and relatively unchanged, it may indicate underfitting, where the model is too simple to capture the underlying patterns in the data. Optimization curves are used to guide hyperparameter tuning. Practitioners analyze these curves to decide when to stop training, adjust learning rates, batch sizes, or other hyperparameters, and to select the best-performing model. Early stopping is a technique where training is halted when the validation curve starts to degrade, indicating that the model has reached its optimal point and further training might lead to overfitting. After training and hyperparameter tuning, the model with the best performance metric on the validation data is typically selected as the final model for deployment or further evaluation. Optimization curves are valuable tools for understanding and improving the performance of machine learning models.

3.5.2 Evaluation Metrices

Evaluation metrics are employed to assess the quality of a statistical or machine learning model, providing insights into its performance. A key attribute of evaluation metrics is their ability to distinguish between different model outcomes. It's essential to verify the model's accuracy before generating predictions. In this research, five primary performance criteria are employed to gauge the models' accuracy. These criteria include Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and R-squared Score(R^2)

Mean Absolute Error(MAE)

MAE, or Mean Absolute Error, represents the average of the absolute discrepancies between predicted values and observed values. MAE is a linear scoring method, meaning that each individual difference holds equal weight in the calculation of the average. In essence, MAE is akin to Mean Square Error (MSE), but instead of considering the squared errors, it focuses on the sum of the absolute errors. It quantifies the average magnitude of discrepancies between actual values and their corresponding forecasts.

$$MAE = \frac{1}{N} \sum_{i=1}^{i=N} |y\mathrm{i} - \widehat{yi}| \tag{3.3}$$

Root Mean Squared Error(RMSE)

Root Mean Square Error (RMSE) is derived from the square root of Mean Square Error (MSE). MSE is computed as the sum of squared prediction errors, where each prediction error is the actual output subtracted from the predicted output, and then this sum is divided by the number of data points. RMSE provides an absolute measure of how much the predicted values deviate from the actual values. It is more commonly used than MSE because MSE values can be relatively large and challenging to compare, as MSE involves squaring the errors. The square root operation in RMSE brings the error measure back to the same scale as the prediction error, facilitating easier interpretation.

$$MSE = \frac{1}{N} \sum_{i=1}^{i=N} (|y\mathrm{i} - \widehat{yi}|)^2 \tag{3.4}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{i=N}(|yi - \widehat{yi}|)^2} \tag{3.5}$$

Mean Absolute Percent Error(MAPE)

The Mean Absolute Percentage Error (MAPE) serves as a metric for assessing the accuracy of a forecasting system. It quantifies this accuracy as a percentage and is determined by computing the average of the absolute percentage errors for each time period, calculated as the absolute difference between predicted values and actual values divided by the actual values. MAPE is the prevalent method for evaluating forecasting errors and is most effective when dealing with data that doesn't have extreme values or zeros.

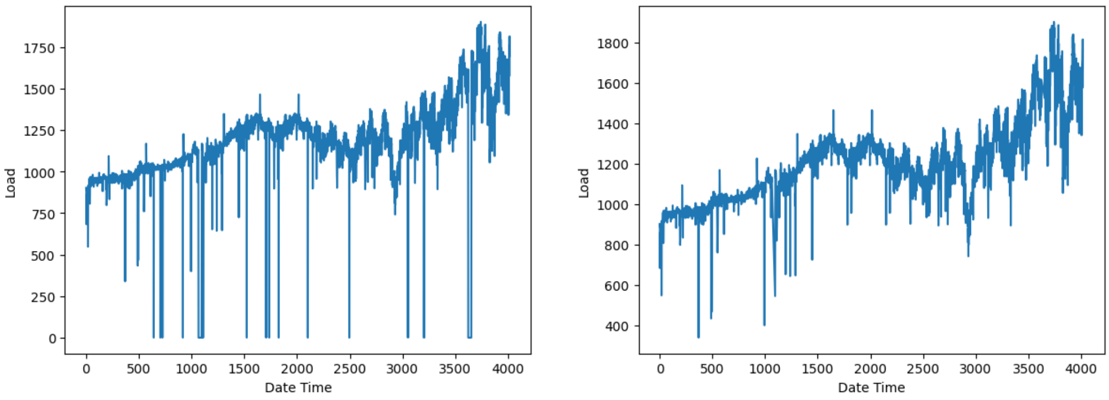
$$MAPE = \frac{1}{N}\sum_{i=1}^{i=N}|\frac{(yi - \widehat{yi})}{yi}| \tag{3.6}$$

Coefficient of determination (R^2)

R-squared (R^2) is a measure of how effectively a model predicts the outcome of the dependent variable, where the dependent variable represents the model's outcome. The possible range for (R^2) values is from 0 (minimum) to 1 (maximum). An (R^2) value of 0 signifies that the model doesn't explain or predict any of the relationships between the dependent and independent variables. Conversely, a value of 1 indicates that the model predicts the entire relationship, 100%. Intermediate values like 0.5 suggest the model predicts 50% of the relationship, and so forth. In essence, (R^2) quantifies how well the model forecasts the outcome of the dependent variable. The formula below is mostly used to find the value of (R^2):

$$R^2 = \frac{n\sum(xy) - \sum(x)\sum(y)}{\sqrt{[n\sum(x)^2 - (\sum(x)^2][n\sum(y)^2 - (\sum(y)^2]}} \tag{3.7}$$

CHAPTER FOUR : RESULTS AND DISCUSSION

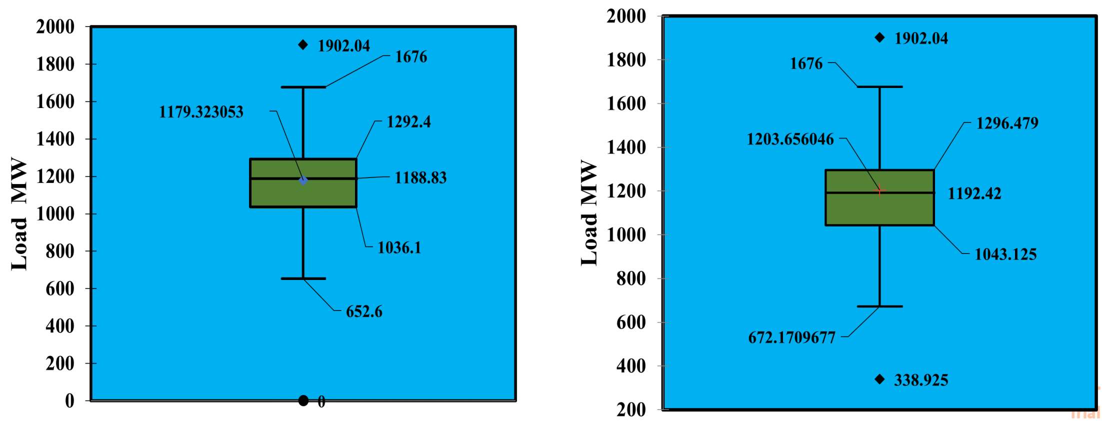


(a) Load vs timestamp of raw peak load data set

(b) Load vs timestamp of peak load data set after linear interpolation

Figure 4.1: Comparison of line plot of the peak load data set

In Exploratory Data Analysis(EDA), the load data can be analyzed in different granularity of the dataset. Out of 4018 dataset, 82 datasets (2.04% of total dataset) had missing values of 0MW. The missing data of load demand of 0MW may be due to the data entry error. Linear interpolation is used to fit the curve of the missing dataset. The box plot is used to show the dispersion of the dataset.



(a) Box-plot of raw peak load data set

(b) Box-plot of peak load data set

Figure 4.2: Comparison of box plot of the peak load data set

The box plot of the quantitative data set depicts that the peak load dataset having 4018 number of observation have a minimum value of 338.925 MW, a maximum value of 1902.040 MW, a mean value of 1203.656 MW and a median value of 1192.420 MW. The lower quartile (Q1) is 1043.125MW and the upper quartile (Q3) is 1296.479 MW which indicates that 25% of the data falls below 1043.125 MW and 75% of the data falls below 1296.479 MW. The interquartile range is 253.354 MW. The median value or the second quartile (Q2) of 1192.420 MW represents that half

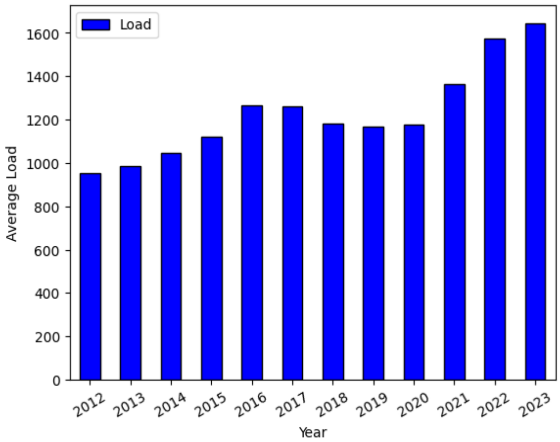


Figure 4.3: Average load of each year

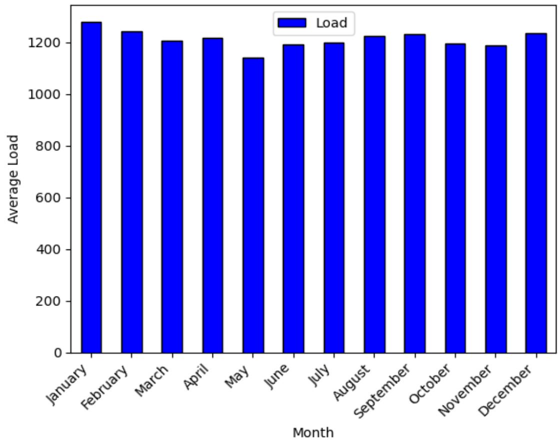


Figure 4.4: Average load of each month

of the dataset is greater than and equal to 1192.420 MW and half is lesser than 1192.420 MW. The upper and lower whiskers represent score outside the middle 50% of the dataset. On calculating mode which is equal to 3 times median minus 2 times mean we get mode equal to 1169.95 MW. Since mean is greater than the median which is greater than mode and skewness of 0.634 indicates that the data is slightly positively skewed with a longer or fatter tail on the right. The Kurtosis value of 0.761 indicates the curve is platykurtic which has flat topped.

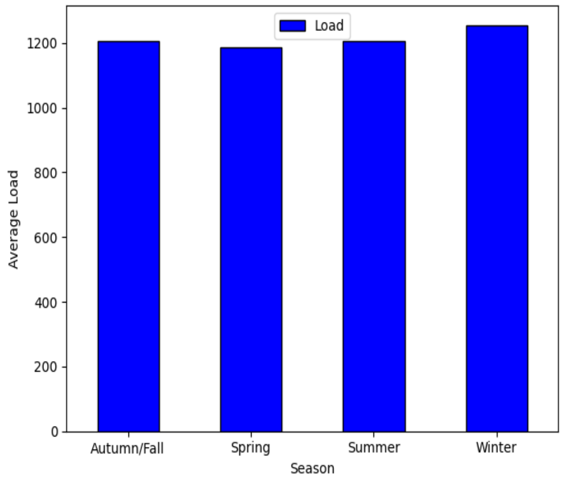


Figure 4.5: Bar-chart of average season-wise load.

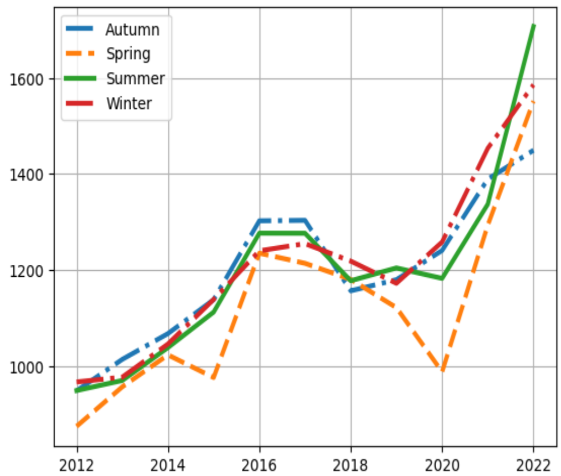


Figure 4.6: Line plot of average season-wise load.

The average load of all year depicts that the minimum average load was of the year 2012 as 952.14MW which increases upto the year 2015 to 1121.42MW. During the years 2016 and 2017, the peak leveled off. During the year 2018, 2019 and 2020 there was lesser peak load demand comparatively. It may be due to the onset of the COVID-19 epidemic. After the year 2020, the peak load demand has gradually risen to the maximum average value of 1644.18MW in 2023 upto the month of April. The average load of each month of the different year indicates that January has the highest average peak load demand followed by February. Month May has the lowest average peak load demand.

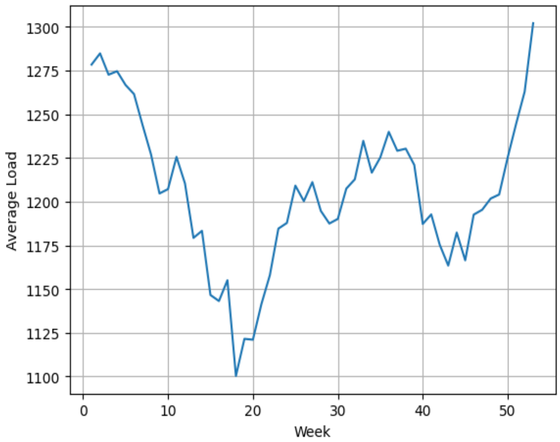


Figure 4.7: Week-wise average load.

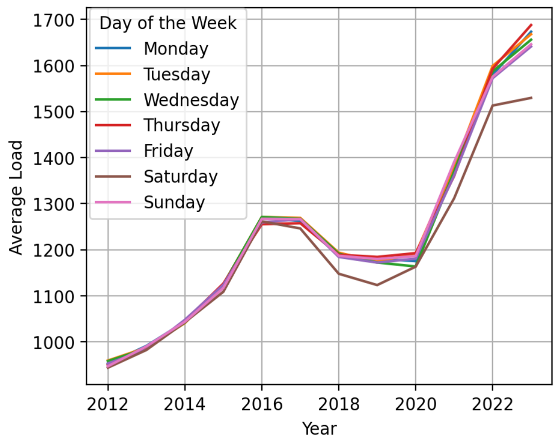


Figure 4.8: Day-wise average load.

The average season-wise load of each year indicates that the winter season has the highest average demand of 1251.59 MW while spring has the lowest average demand of 1162.56 MW. Spring and autumn season have nearly equal average demand. The average day-wise load for each year indicated that day Saturday has lower demand than other days. The week-wise average load indicates that the 20^{th} week had the minimum average demand of 1125.24 MW while the 53^{rd} week had the maximum average demand of 1302.10 MW.

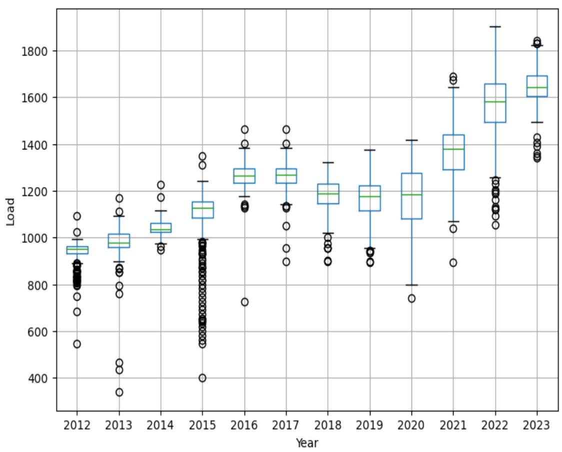


Figure 4.9: Box plot values by year

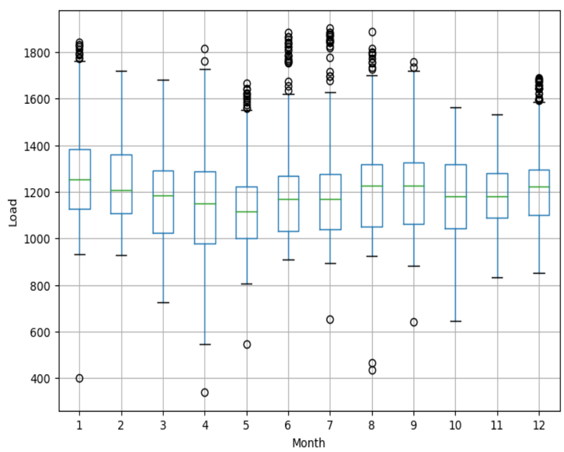


Figure 4.10: Box plot values by month

The boxplot of load values and year indicates that the year 2015 had the maximum number of lower outliers. The outliers are the data points that lie far from whiskers that are within 1.5 times the interquartile range from each box end. The dataset is less dispersed in the year 2014 while dataset is most dispersed in the year 2020. The boxplot of load values by month indicates more upper outliers are present in the monthly dataset.

4.1 LSTM Model

After the exploratory data analysis, a single LSTM model was built tunning the parameters. The influence of the lookback period, epoches, the percent of test and train data split and number of LSTM units were studied on the dataset. The dataset is split to train and test the dataset. The entire dataset was divided into a training set (80%) and a test set (20%). The training data set was normalized between the values of 0 and 1 using the min-max scaler from Python. Then, the test set was normalized according to the training normalization parameters. In order to train the model, the number of loopbacks for predicting the peak demand had to be defined. Therefore, the training set was transformed into a set of input and output sequence vectors. The model was compared for different hyperparameters such as epochs, LSTM Units and lookback periods. The accuracy of every forecasting model is examined based on the difference in errors between the actual value and the forecasted value. The most commonly used evaluation metrics are Root Mean Squared Error(RMSE), followed by Mean Absolute Percentage Error(MAPE), Mean Squared Error(MSE), and Mean Absolute Error(MAE). In this study, all four evaluation metrics are incorporated to determine the accuracy forecast of the models.

The models were optimized using the Adam optimizer, employing a default learning rate of 0.001. During training, the models utilized the Mean Squared Error (MSE) loss function. The first 3208 rows are used for training the dataset while the rest are used for testing and validation. The hyperparameters like lookback, epoch size, LSTM units were tuned for better accuracy results. The training loss assesses how effectively the model aligns with the training data, while the validation loss gauges its performance with new data. By examining the learning curves computed based on the optimization metric, such as loss or Mean Squared Error, it's evident that common learning issues like underfitting, overfitting, or inadequacy have been mitigated. The comparision table of

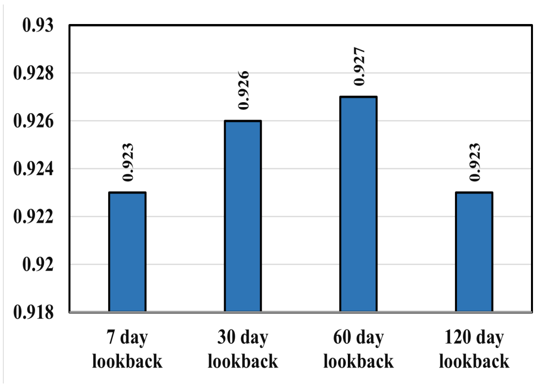
Table 4.1: Comparison of LSTM Model of 64 unit with different lookback

Sno	Model No	Lookback	Unit	Epoch	MAE	MSE	RMSE	MAPE	R^2
1	model 1	7	64	150	36.92	3225.29	56.79	3.18	0.923
2	model 2	30	64	150	37.43	3140.64	56.04	3.20	0.926
3	model 3	60	64	150	34.10	2995.67	54.73	2.97	0.927
4	model 4	120	64	150	35.79	3081.87	55.51	2.99	0.923

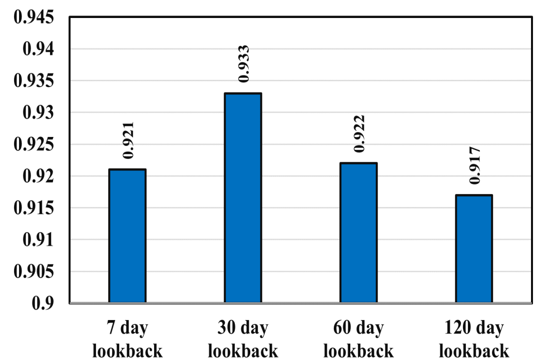
LSTM model with 64 units of different lookback indicates that model 3 with a lookback of 60 has the least forecasting error with MAE of 34.10 MW, MSE of 2995.67MW, RMSE of 54.73MW, MAPE of 2.97% and R^2 score of 0.927. The comparision table of LSTM model with 32 units of different lookback indicates that model 6 with a lookback of 30 has the least forecasting error with MAE of 34.01 MW, MSE of 2880.75 MW, RMSE of 53.67 MW, MAPE of 2.95% and R^2 score of 0.933. Thus it can be deduced that the LSTM model with 32 LSTM units and a lookback of 30 outperforms the model with 64 LSTM units and a lookback of 60. Based on the learning curves of both LSTM models, it can be observed that both the training loss and validation loss exhibit a trend of decreasing and eventually stabilizing at a specific point, with a minimal gap between their

Table 4.2: Comparison of LSTM Model of 32 unit with different lookback

Sno	Model No	Lookback	Unit	Epoch	MAE	MSE	RMSE	MAPE	R^2
1	model 5	7	32	150	36.39	3313.85	57.56	3.20	0.921
2	model 6	30	32	150	34.01	2880.75	53.67	2.95	0.933
3	model 7	60	32	150	34.18	3149.69	56.12	2.99	0.922
4	model 8	120	32	150	37.26	3317.15	57.59	3.13	0.917

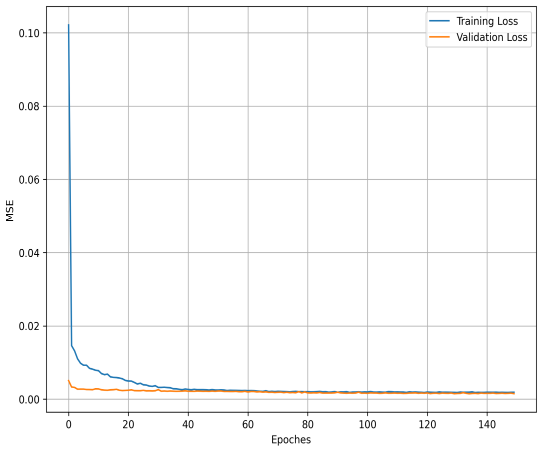


(a) R2 score comparison of LSTM model with 64 LSTM unit of different lookback.

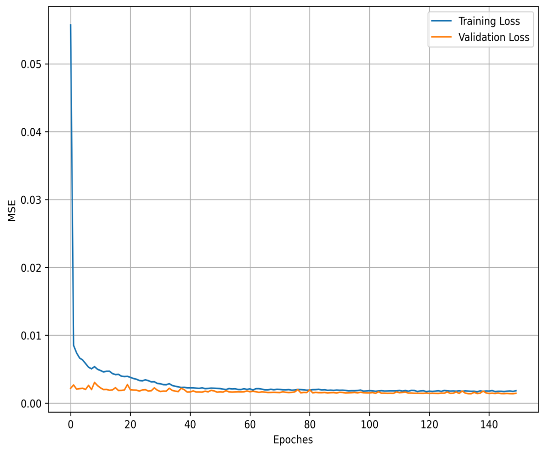


(b) R2 score comparison of LSTM model with 32 LSTM unit of different lookback.

Figure 4.11: R2 score comparison of LSTM model with different LSTM unit of different lookback.



(a) Training Vs Validation loss of LSTM model with 32 LSTM unit and 30 lookback.



(b) Training Vs Validation loss of LSTM model with 64 LSTM unit and 60 lookback.

Figure 4.12: Comparison of Training Vs Validation loss of LSTM model with different loopback and LSTM unit

final loss values. At the outset of training, the training loss is typically relatively high because the model's parameters (weights and biases) are initially set to random values, resulting in predictions that deviate significantly from the actual target values. However, as the model processes more training examples and iteratively updates its parameters, the training loss generally decreases over time. This reduction signifies that the model is enhancing its performance and becoming more aligned with the training data.

The validation loss serves as a metric for evaluating the disparity between the model's predictions and the actual target values in the validation dataset. It is instrumental in assessing the model's ability to generalize to new data. When the validation loss remains relatively constant, it suggests that the model has reached a point of performance stability or limit.

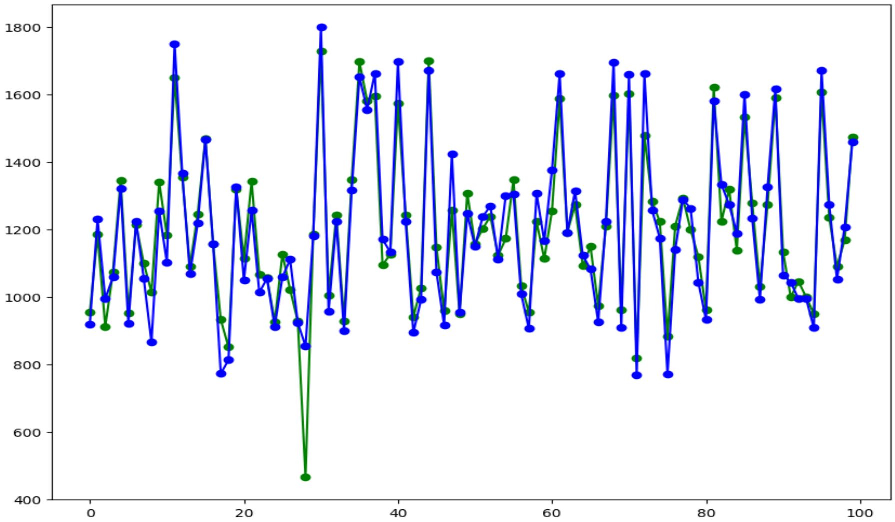


Figure 4.13: Actual Vs predicted for LSTM Model.

4.2 Other Statistical Models

Also, the load data is predicted using statistical models such as the Moving Average Method, Weighted Moving Average Method and Exponential Smoothing Methods.

4.2.1 Moving Average

Simple Moving Average

From the study, it is derived that the among moving average with different lag orders moving average with the lag value of 2 days gives the least forecasting error with an MAE of 37.42 MW, MSE of 3827.06 MW, RMSE value of 61.86 MW, MAPE of 3.2281% and R2 score of 0.910. The forecasting metric MAPE of value 3.2281% implies that the average deviation between the forecasted value and the actual value was only 3.2281% and R2 score of 0.910 indicates that 91% variance of the predicted values is predictable from the actual values.

4.2.2 Weighted Moving Average

In the weighted average method, the corresponding weights were calculated using MS Excel Solver and found that a weighted average with a lag order of 30 results in minimum forecasting error with MAE of 34.77 MW, MSE of 3271.68 MW, RMSE of 57.19, MAPE of 3.005% and R2 score of 0.921. The MAE indicates that the average absolute difference between the actual values and the predicted values is 34.77 MW. The weighted average method can signify 92.1% of the variance of

Table 4.3: Performance measures of moving average

. Sno	Look back	MAE	MSE	RMSE	MAPE	R^2 Score
1	1 day lookback	37.59	4246.29	65.16	3.23	0.90
2	2 day lookback	37.42	3827.06	61.86	3.22	0.91
3	3 day lookback	38.10	3858.84	62.11	3.28	0.90
4	4 day lookback	38.62	3926.59	62.66	3.32	0.90
5	7 day lookback	39.97	4268.92	65.33	3.45	0.89
6	1 month lookback	49.29	6001.56	77.46	4.22	0.85
7	1 year lookback	86.42	13599.14	116.61	7.13	0.67

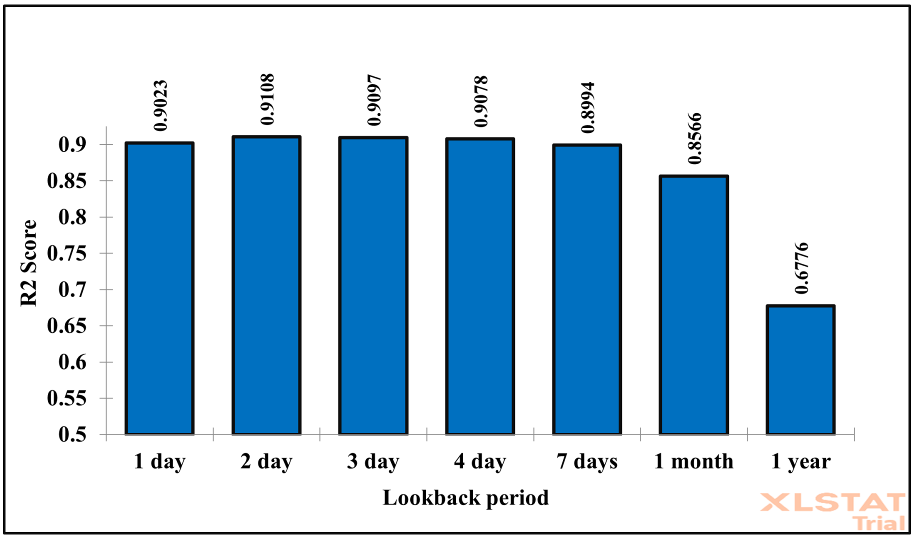


Figure 4.14: R2 score comparison of different loopback periods of moving average

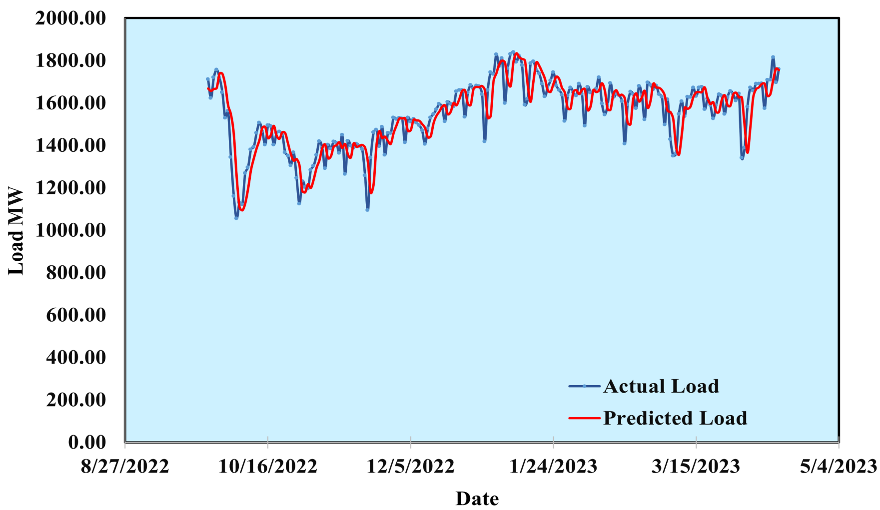


Figure 4.15: Moving average with lag order of 2 days.

the dependent variable from the independent variable.

Table 4.4: Actual and predicted data of Moving Average

Sno	Actual Data	Predicted Data 1 day lookback	Predicted Data 2 day Lookback	Predicted Data 3 day Lookback	Predicted Data 4 day Lookback	Predicted Data 7 day lookback	Predicted Data 1 month Lookback	Predicted Data 1 year lookback
1	1692.00	1690.00	1690.00	1679.33	1677.50	1574.57	1587.97	1588.20
2	1575.00	1692.00	1691.00	1690.67	1682.50	1624.71	1599.00	1588.65
3	1708.00	1575.00	1633.50	1652.33	1661.75	1651.29	1605.87	1588.73
4	1709.00	1708.00	1641.50	1658.33	1666.25	1609.29	1611.03	1589.17
5	1815.00	1709.00	1708.50	1664	1671.00	1674.57	1614.29	1589.63
6	1696.00	1815.00	1762.00	1744.00	1701.75	1697.00	1623.26	1590.38
7	1760.00	1696.00	1755.00	1740.00	1732.00	1697.86	1625.45	1590.82

Table 4.5: Performance Measures Of Weighted Moving Average with optimized weights

. Sno	Look back	MAE	MSE	RMSE	MAPE	R^2 Score
1	1 day lookback	37.60	4247.31	65.17	3.23	0.902
2	2 day lookback	36.64	3730.20	61.07	3.16	0.913
3	3 day lookback	36.23	3603.52	60.02	3.13	0.915
4	4 day lookback	35.92	3518.93	59.32	3.10	0.917
5	7 day lookback	35.43	3407.11	58.37	3.06	0.919
6	1 month lookback	34.77	3271.68	57.19	3.00	0.920
7	2 months lookback	34.82	3279.96	57.27	3.00	0.920

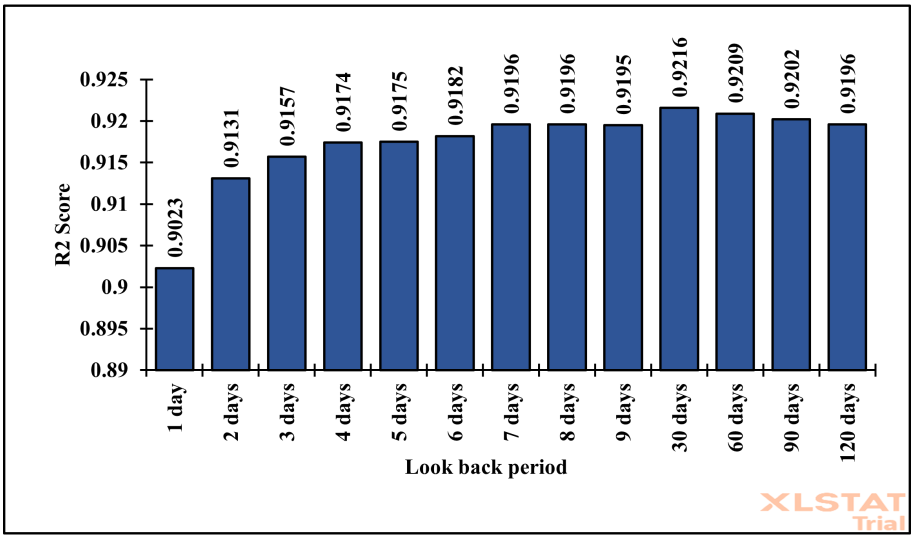


Figure 4.16: R2 Score comparison of different lag order of weighted moving average.

4.2.3 Exponential Smoothing Methods

The peak load demand is also forecasted using simple Exponential Smoothing, Double Exponential Smoothing, and Holt-Winters Exponential Method. Simple Exponential Smoothing with smooth-

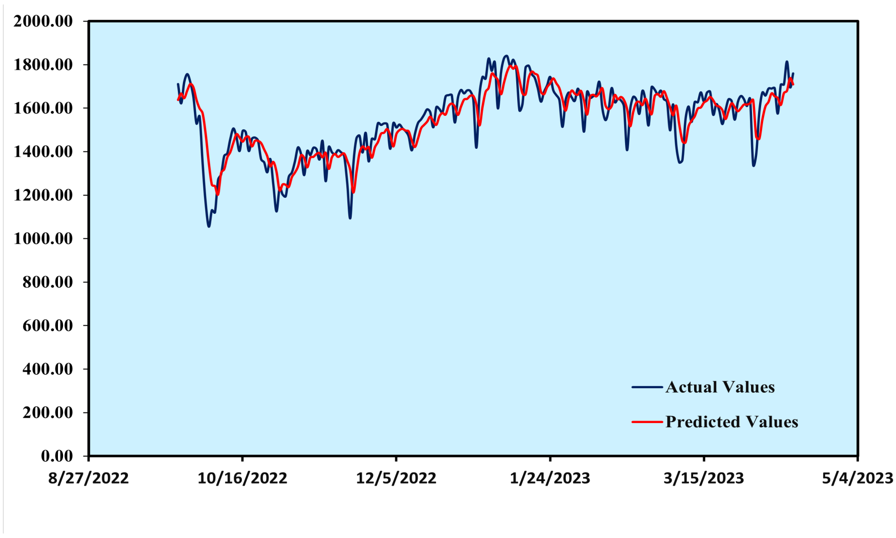


Figure 4.17: Weighted moving average with lookback of 30 days.

.

Table 4.6: Actual and predicted data of Weighted Moving Average

Sno	Actual Data	Predicted Data 1 day lookback	Predicted Data 2 day lookback	Predicted Data 3 day lookback	Predicted Data 4 day Lookback	Predicted Data 7 day lookback	Predicted Data 1 month lookback	Predicted Data 2 months lookback
1	1692.00	1690.00	1690.00	1684.10	1684.22	1634.48	1654.55	1654.22
2	1575.00	1692.00	1691.30	1691.17	1686.55	1645.47	1646.59	1645.57
3	1708.00	1575.00	1615.8	1622.95	1625.92	1613.96	1614.52	1613.85
4	1709.00	1708.00	1661.60	1674.65	1677.74	1676.37	1669.04	1667.68
5	1815.00	1709.00	1708.50	1684.05	1692.94	1691.45	1680.13	1676.28
6	1696.00	1815.00	1778.00	1771.04	1749.16	1752.97	1737.92	1734.45
7	1760.00	1696.00	1737.50	1725.59	1722.35	1718.03	1709.35	1706.80

ing constant ($\alpha = 1$) also known as Naives Forecast yields forecasting results with an MAE of 37.59 MW, MSE of 4246.29 MW, RMSE value of 65.16 MW, and MAPE of 3.23% and R^2 score of 0.902.

Simple Exponential Smoothing

Using MS Excel add-ins to optimize the value of the smoothing constant(α), we get (α) equal to 0.481. In Simple Exponential Smoothing, a forecast error of MAE of 36.01 MW, MSE of 3512.83 MW, RMSE value of 59.26 MW, MAPE of 3.10% and R^2 Score of 0.917 is obtained which means that 91.7% of the dependent variable values are explained by the observed value.

Double Exponential Smoothing

In Double Exponential Smoothing or Holt Linear or Holt's Method, optimizing α and β using Excel solver, value of α and β is obtained as 0.490 and 0.010 respectively and got forecasting errors with MAE of 36.14 MW, MSE of 3547.08 MW, RMSE of 59.57 MW, MAPE of 3.12% and

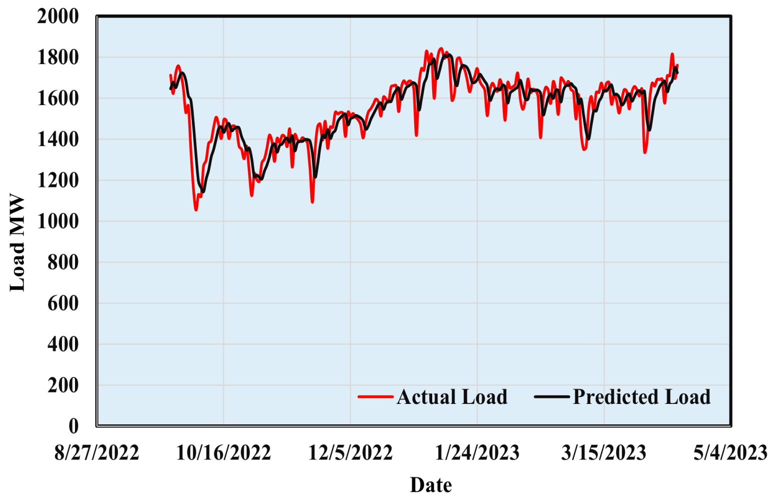


Figure 4.18: Simple Exponential Smoothing.

R^2 Score of 0.916.

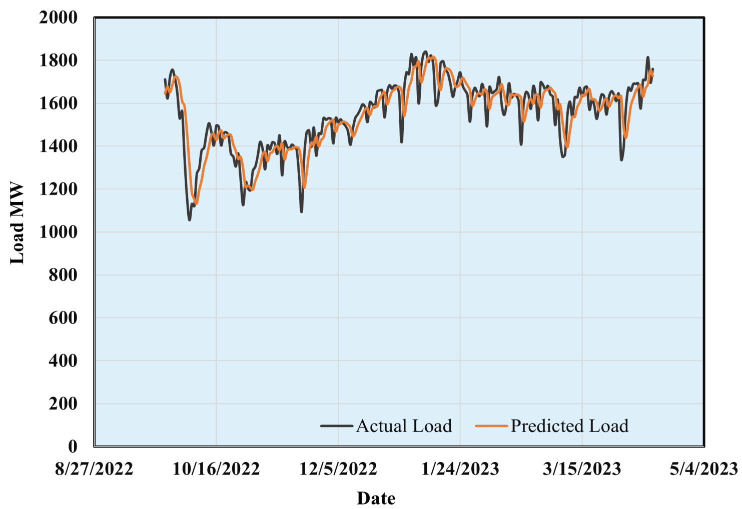


Figure 4.19: Double Exponential Smoothing.

Holt Winters Exponential Smoothing

In Holt-Winters Exponential Smoothing Method, optimizing α, β and γ using Excel solver, value of α , β and γ is obtained as 0.603, 0.131 and 0.124 respectively with seasonal period of 7 days we got forecasting errors with MAE of 37.68 MW, MSE of 3790.47 MW, RMSE of 61.56 MW, MAPE of 3.31% and R^2 Score of 0.910.

Comparing the evaluation metrics among the exponential smoothing methods, the simple exponential smoothing method results in the least forecast error with a Mean Absolute Error of 36.01 MW, Mean Squared Error of 3512.83 MW, Root Mean Squared Error of 59.26 MW, Mean Absolute Percent Error of 3.115% and R2 Score of 0.917. It is followed by Double Exponential Smoothing

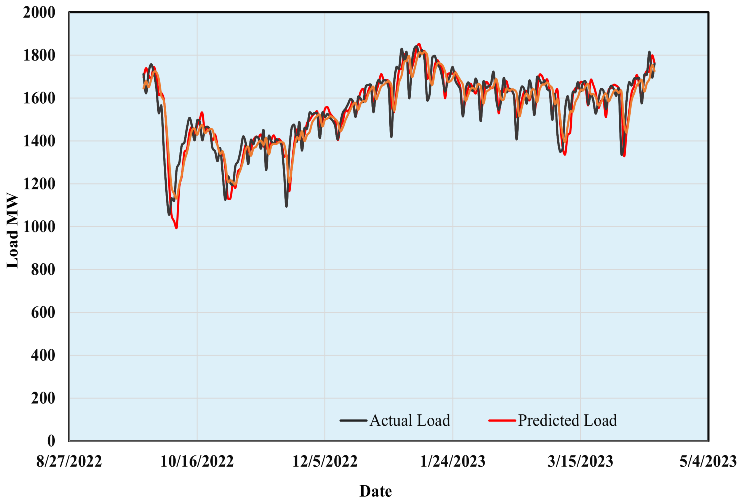


Figure 4.20: Holt Winters Exponential Smoothing.

Table 4.7: Evaluation Metrics of Exponential Smoothing Methods

Sno	Model	MAE	MSE	RMSE	MAPE	R^2 Score
1	Simple Exponential	36.01	3512.83	59.26	3.11	0.917
2	Double Exponential	36.14	3547.08	59.55	3.12	0.916
3	Holt-Winters Exponential	37.68	3790.47	61.56	3.31	0.910

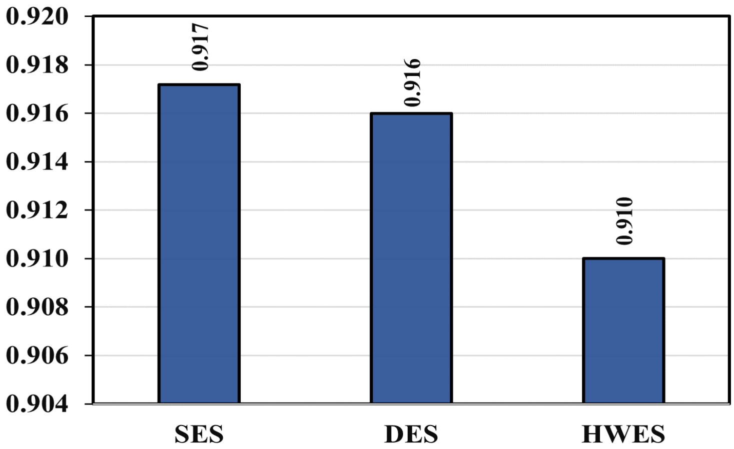


Figure 4.21: R2 Score comparison of different exponential smoothing methods.

with a Mean Absolute Error of 36.14 MW, Mean Squared Error of 3547.08 MW, Root Mean Squared Error of 59.55 MW, Mean Absolute Percent Error of 3.12% and R2 score of 0.916.

On comparing the evaluation metrics of all the studied models, it can be deduced that the proposed LSTM model 6 with 150 epochs, 32 LSTM units and loopback of 30 has better forecast accuracy with Mean Absolute Error(MAE) of 34.01 MW, Mean Squared Error(MSE) of 2880.75 MW, Root Mean Squared Error(RMSE) of 53.67 MW, Mean Absolute Percent Error(MAPE) of 2.95% and

Table 4.8: Actual and predicted data of Exponential Smoothing Methods

Sno	Actual Data	Predicted Data SES	Predicted Data DES	Predicted Data HWES
1	1692.00	1671.59	1673.77	1663.05
2	1575.00	1681.41	1683.66	1598.16
3	1708.00	1630.19	1630.84	1698.37
4	1709.00	1667.64	1669.45	1727.03
5	1815.00	1687.54	1689.82	1722.56
6	1696.00	1748.89	1752.77	1797.73
7	1760.00	1723.43	1726.28	1762.83

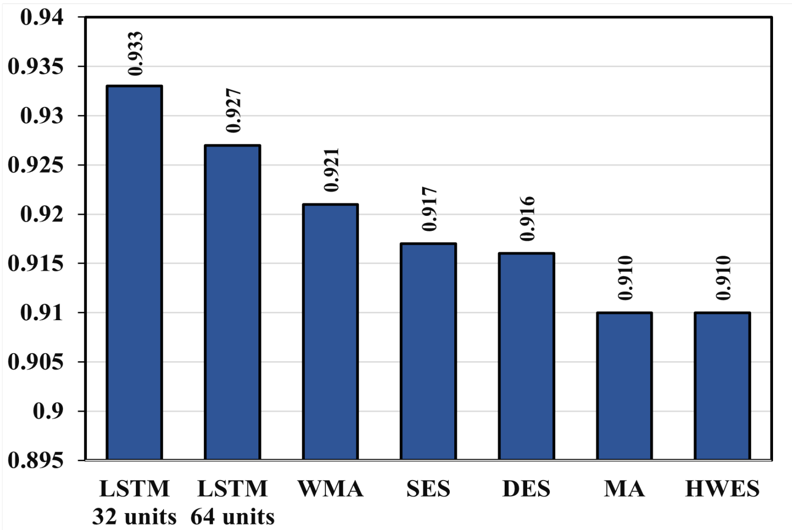


Figure 4.22: R2 Score comparison all models.

Table 4.9: Evaluation Metrics of all models

. Sno	Model	MAE	MSE	RMSE	MAPE	R^2 Score
1	LSTM model 6	34.01	2880.75	53.67	2.95	0.933
2	LSTM model 3	34.10	2995.67	54.73	2.97	0.927
3	WMA	34.77	3271.68	57.19	3.00	0.921
4	SES	36.01	3512.83	59.26	3.11	0.917
5	DES	36.14	3547.08	59.57	3.12	0.916
6	MA	37.42	3827.06	61.86	3.22	0.910
7	HWES	37.68	3790.47	61.56	3.31	0.910

R^2 score of 0.933. It is followed by LSTM model 3 with 150 epochs, 64 LSTM units and loopback of 60 with Mean Absolute Error(MAE) of 34.10 MW, Mean Squared Error(MSE) of 2995.67 MW, Root Mean Squared Error(RMSE) of 54.73 MW, Mean Absolute Percent Error(MAPE) of 2.97% and R^2 score of 0.927.

Among the statistical models considered, the weighted moving average with a loopback of 30 days with a Mean Absolute Error of 34.77 MW, Mean Squared Error of 3271.68 MW, Root Mean Squared Error of 57.19 MW, Mean Absolute Percent Error of 3.005% and R2 Score of 0.921 had least forecast error which is followed by Simple Exponential Smoothing with a Mean Absolute Error of 36.01 MW, Mean Squared Error of 3512.83 MW, Root Mean Squared Error of 59.26 MW, Mean Absolute Percent Error of 3.115% and R2 Score of 0.917. The Holt-Winters exponential smoothing with seasonal period of 7 days had the least forecasting accuracy with a Mean Absolute Error of 37.68 MW, Mean Squared Error of 3790.47 MW, Root Mean Squared Error of 61.56 MW, Mean Absolute Percent Error of 3.31% and R^2 score of 0.910.

CHAPTER FIVE: CONCLUSIONS AND RECOMMENDATIONS

5.1 Conclusions

The exploratory data analysis of the quantitative data set depicts that the peak load dataset having 4018 number of observation have a minimum value of 338.925 MW, a maximum value of 1902.040 MW, a mean value of 1203.656 MW and a median value of 1192.420 MW. The lower quartile (Q1) is 1043.125MW and the upper quartile (Q3) is 1296.479 MW which indicates that 25% of the data falls below 1043.125 MW and 75% of the data falls below 1296.479 MW. The median value or the second quartile (Q2) of 1192.420 MW represents that half of the dataset are greater than and equal to 1192.420 MW and half is lesser than 1192.420 MW.
The conclusions deduced from this study can be highlighted as:

1. The daily peak load of Nepal was forecasted using single LSTM networks with a past load of ten consecutive years and the evaluation metrics MAE, MSE, MAPE, RMSE and R^2 Score for the model were calculated.

2. The peak load demand of Nepal was also forecasted using other statistical models such as Moving Average, Weighted Moving Average, Simple Exponential Smoothing, Double Exponential Smoothing and Holt-Winters Exponential smoothing and the evaluation metrics for each model were calculated and compared.

3. It can be deduced from the comparison of the evaluation metrics that the proposed LSTM model with 150 epochs, 32 LSTM units and a loopback of 30 has better forecast accuracy with Mean Absolute Error(MAE) of 34.01 MW, Mean Squared Error(MSE) of 2880.75 MW, Root Mean Squared Error(RMSE) of 53.67 MW, Mean Absolute Percent Error(MAPE) of 2.95% and R^2 score of 0.933. It is followed by LSTM model with 150 epochs, 64 LSTM units and a loopback of 60 with a Mean Absolute Error(MAE) of 34.10 MW, Mean Squared Error(MSE) of 2995.67 MW, Root Mean Squared Error(RMSE) of 54.73 MW, Mean Absolute Percent Error(MAPE) of 2.97% and R^2 score of 0.927. Among the statistical models considered, the Weighted Moving Average with a loopback of 30 days with a Mean Absolute Error of 34.77 MW, Mean Squared Error of 3271.68 MW, Root Squared Mean Error of 57.19 MW, Mean Absolute Percent Error of 3.005% and R2 Score of 0.921 has least forecast errors. The Holt-Winters exponential smoothing with seasonal period of 7 days had the least forecasting accuracy with a Mean Absolute Error of 37.68 MW, Mean Squared Error of 3790.47 MW, Root Mean Squared Error of 61.56 MW, Mean Absolute Percent Error of 3.31% and R^2 score of 0.910.

5.2 Recommendations

Future research direction includes incorporating exogenous variables, hybrid or improved models and forecasting the time widths of the peak demand for further improving forecasting accuracies. Since the accuracy of the forecast depends on the volume of the dataset, the model can be trained for more datasets. So, there can be made some enhancement for further studies like:

1. The experiment was carried out on a standard computing device equipped with 12 GB of RAM and a core i5 CPU. For more demanding tasks, an environment with a GPU with ample memory, and a powerful processor can be employed.

2. The hyperparameters, such as the learning rate, activation functions, model architecture, and batch size, were initially configured with default values from Keras, and no optimization techniques were applied. To attain the best possible LSTM performance, it's typically necessary to perform meticulous tuning of hyperparameters. This tuning involves adjustments to factors such as the number of layers, the quantity of neurons, learning rates, and batch sizes, ultimately leading to more precise and accurate results.

3. Multiple hidden layers can be added.

4. Other Autoregressive models can be incorporated for better comparisons of the forecast.

5. To enhance the accuracy of results, various exogenous variables can be integrated into the analysis. These include weather-related variables (e.g., minimum temperature, maximum temperature, average temperature, average relative humidity, average pressure, cloud cover, rainfall volume, average wind speed, daily solar radiation, etc.), calendar-related variables (e.g., time of day, day of the week, week of the month, month of the year, season, year number, holidays, special events, etc.), economic variables (e.g., Gross National Product (GNP), Gross Domestic Product (GDP), Population Growth Rate, Consumer Growth Rate, Tariff Structure, Electricity Price, price elasticity of electricity, etc.), and other factors such as customer type (residential, commercial, industrial, etc.). Incorporating these variables can lead to improved accuracy in forecasting.

6. Time resolution and peak time of the predicted peak load demand can also be considered in further study.

REFERENCES

[1] G. P. Zhang, "Time series forecasting using a hybrid arima and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, 2003.

[2] B. Bhandari, S. R. Shakya, and A. K. Jha, "Short term electric load forecasting of kathmandu valley of nepal using artificial neural network," *Kathford Journal of Engineering and Management*, vol. 1, no. 1, pp. 43–48, 2018.

[3] S. Dai, F. Meng, H. Dai, Q. Wang, and X. Chen, "Electrical peak demand forecasting-a review," *arXiv preprint arXiv:2108.01393*, 2021.

[4] G. Adepoju, S. Ogunjuyigbe, and K. Alawode, "Application of neural network to load forecasting in nigerian electrical power system," *The Pacific Journal of Science and Technology*, vol. 8, no. 1, pp. 68–72, 2007.

[5] N. Kandil, R. Wamkeue, M. Saad, and S. Georges, "An efficient approach for short term load forecasting using artificial neural networks," *International Journal of Electrical Power & Energy Systems*, vol. 28, no. 8, pp. 525–530, 2006.

[6] R. Weron, A. Misiorek, *et al.*, "Modeling and forecasting electricity loads: A comparison," in *Proceedings of the*, Citeseer, 2004.

[7] B. Hobbs, S. Jitprapaikulsarn, S. Konda, V. Chankong, K. Loparo, and D. Maratukulam, "Analysis of the value for unit commitment of improved load forecasts," *IEEE Transactions on Power Systems*, vol. 14, no. 4, pp. 1342–1348, 1999. DOI: 10.1109/59.801894.

[8] B. Xue and J. Geng, "Dynamic transverse correction method of middle and long term energy forecasting based on statistic of forecasting errors," in *2012 10th International Power & Energy Conference (IPEC)*, IEEE, 2012, pp. 253–256.

[9] N Phuangpornpitak and W Prommee, "A study of load demand forecasting models in electric power system operation and planning," *GMSARN Int. J*, vol. 10, pp. 19–24, 2016.

[10] S. Khatoon, A. K. Singh, *et al.*, "Effects of various factors on electric load forecasting: An overview," in *2014 6th IEEE Power India International Conference (PIICON)*, IEEE, 2014, pp. 1–5.

[11] S. Nuchprayoon, "Forecasting of daily load curve on monthly peak day using load research data and harmonics model," in *2016 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, IEEE, 2016, pp. 338–342.

[12] J. Zhengyuan, "Electricity consumption forecasting in peak load month based on variable weight combination forecasting model," in *2008 IEEE International Conference on Automation and Logistics*, IEEE, 2008, pp. 1265–1269.

[13] J. Liu and L. E. Brown, "Prediction of hour of coincident daily peak load," in *2019 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, IEEE, 2019, pp. 1–5.

[14] A. Goia, C. May, and G. Fusai, "Functional clustering and linear regression for peak load forecasting," *International Journal of Forecasting*, vol. 26, no. 4, pp. 700–711, 2010.

[15] J. Xue, Z. Xu, and J. Watada, "Building an integrated hybrid model for short-term and mid-term load forecasting with genetic optimization," *International Journal of Innovative Computing, Information and Control*, vol. 8, no. 10, pp. 7381–7391, 2012.

[16] H.-T. Yang, T.-C. Liang, K.-R. Shih, and C.-L. Huang, "Power system yearly peak load forecasting: A grey system modeling approach," in *Proceedings 1995 International Conference on Energy Management and Power Delivery EMPD'95*, IEEE, vol. 1, 1995, pp. 261–266.

[17] A. AlFuhaid, M. El-Sayed, and M. Mahmoud, "Cascaded artificial neural networks for short-term load forecasting," *IEEE Transactions on Power Systems*, vol. 12, no. 4, pp. 1524–1529, 1997.

[18] S. P. D. NEA, *Load forecast report - ministry of energy.* 2017, July.

[19] WECS, *Electricity demand forecast report - ministry of energy.* 2017, January.

[20] S. C. Bhattacharyya, *Energy economics: concepts, issues, markets and governance*. Springer Science & Business Media, 2011.

[21] K. Chapagain, S. Acharya, H. Bhusal, *et al.*, "Short-term electricity demand forecasting for kathmandu valley, nepal," *Kathmandu University Journal of Science Engineering and Technology*, vol. 15, no. 3, 2021.

[22] C. Subedi and R. Khatiwada, "A time series analysis on demand of electricity in nepal," pp. 21–29, Jan. 2017.

[23] S. Muzaffar and A. Afshari, "Short-term load forecasts using lstm networks," *Energy Procedia*, vol. 158, pp. 2922–2927, 2019.

[24] A. Parajuli, S. Shah, P. Shrestha, K. Chapagain, and N. Dhulikhel, "Impact analysis of short-term electrical demand factors using multivariate forecasting approach: A case of kathmandu valley," 2022.