Water Consumption Analysis and Prediction Using Deep Learning Approach:

A Thesis Presented by

Nigatu Agize Abebe

To

The Faculty of Informatics of

St. Mary's University

In Partial Fulfillment of the Requirements for the Degree of

Master of Science

In Computer Science

February, 2024

ACCEPTANCE

Water Consumption Analysis and Prediction Using Deep Learning Approach:

By

Nigatu Agize Abebe

Accepted by the Faculty of Informatics, St. Mary's University, in partial fulfillment of the requirements for the degree of Master of Science in Computer Science

Thesis Examination Committee:

____

Internal Examiner

Alemebante Mulu (PhD)

million

External Examiner

Million Meshesha (PhD)

____

Dean, Faculty of Informatics

Alembante Mulu (PhD)

February 2024

DECLARATION

I, the undersigned, declare that this thesis work is my original work, has not been presented for a degree in this or any other universities, and all sources of materials used for the thesis work have been duly acknowledged.

Nigatu Agize Abebe

Full Name of Student

____

Signature

Addis Ababa

Ethiopia

This thesis has been submitted for examination with my approval as advisor.

Tessfu Geteye Fantaye (PhD)

Full Name of Advisor

____

Signature

Addis Ababa

Ethiopia

February 2024

Acknowledgment

First and foremost, I would like to give thanks to God the Almighty who provided me everything to accomplish this thesis.

This thesis would not have been possible without the help, support, and patience of my principal thesis advisor, Dr. Tessfu Geteye for his guidance and encouragement. Although the deep learning project is very time-consuming and complicated, he always encouraged me to do the right thing and be professional. Not only did he never put a difficult obstacle in my way, but he also removed the obstacles in my way and helped me complete this path as well as possible. It was impossible to achieve the goal of this project without his continuous help.

I am also grateful to Addis Ababa Water and Sewerage Authority and heartfelt thanks to my close friend Ato Girma Bewketu for his unlimited support and understanding.

I am also grateful to all Ethiopian Instructors and Teachers those who have spent their entire lives learning and teaching and to St' Mary University Instructors and employee of the University who has support throughout this Master's program from start to end.

Finally, I am deeply grateful for my family, especially my sister Ms. Hana Agize and my brother Behailu Agize who has supported me throughout this Master's program from the start to the end. I would like to dedicate this Master Thesis to them as an indication of their significance in this Thesis.

Table of contents

List Acronym and Abbreviation

AAWSA	Addis Ababa Water and Sewerage Authority
ANN	Artificial Neural Network
ARIMA	Autoregressive Integrated Moving Average
Bi-GRU	Bidirectional Gated Recurrent Unit
Bi-LSTM	Bidirectional Long Short Term Memory
CNN	Convolutional Neural Network
DL	Deep Learning
DNN	Deep Neural Network
EAAD	East Addis Ababa District
GRU	Gated Recurrent Unit
KPI	Key Performance Indicator
LSTM	Long Short-Term Memory
MAE	Mean Squared Error
MSE	Mean Absolute Error
RF	Random Forest
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
SVM	Support Vector Machine

List of Figures

List of Tables

Abstract

Water is one of humanity’s most essential resources, and in order to ensure that the limited quantity of water is used effectively, water supply facilities are required. The global population is driving up both the demand for and consumption of water. Given the restricted amount of water resources on Earth, this presents a number of resource related difficulties. In order to tackle this issue, the paper uses past data to investigate and contrast seven recurrent neural network models for water use. The goal is to create a deep learning model that uses previous customer data to analyze and forecast Addis Ababa Water and Sewerage Authority's water usage. Data collection, preprocessing, feature extraction, hyperparameter tuning, model training and development, and performance evaluation are some of the rigorous experimental processes involved in our research.

With the introduction of smart water meters, it is now possible to get information on residential water usage. The data from the city of Addis Ababa (Ethiopia) was used as a case study to manage its limited resources, being water supplies. However, it is essential to acknowledge persisting challenges, including issues related to model overfitting and the critical necessity for precise hyperparameter tuning. The result of this study presents the remarkable ability of water consumption prediction through applying deep learning models, such Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), bidirectional GRU (Bi-GRU), bidirectional LSTM (Bi-LSTM) and Attention based model.

The performance of the model also evaluated using the three evaluation metrics of RMSE (Root Mean Squared Error), MSE (Mean Squared Error) and MAE (Mean Absolute Error). Thus, Bi-LSTM with Attention mechanism scores lowest loss value of prediction RMSE, MSE and MAE with values of 0.08, 0.0064 and 0.16 respectively. This implies that the model using the Attention mechanism performance better as compared to others.

Therefore, Bi-LSTM with Attention is proposed for constructing water consumption prediction model for Addis Ababa Water and Sewerage Authority. We advise future research to incorporate huge dataset sizes with a greater variety and quantity of variables.

Keywords: Water Consumption prediction, Deep Learning, Machine Learning, Attention mechanisms

CHAPTER ONE

INTRODUCTION

1.1 Background of the Study

Resource is the main component of both developing and developed countries' long-term economic prosperity. The global usage of resource is rapidly increasing from time to time. Water resource plays a key role in developing economic growth. Water resource is also one of the most important forms of energy for industrialization and the improvement of living standards. The role of resource is crucial in our day-to-day living including lighting, farming, and drinking, and food preparation. A variety of sectors, including industry, manufacturing, organizations, institutions, and household use, use water resources. The world energy projection for 2018 states that the percentage of water in global resource demand is expanding at the quickest rate. Between 2017 and 2040, the global demand for water will increase by more than 25% [1]. Ethiopia consistently uses more water. The World Development Indicators Databank [2] reports that Ethiopia's total annual household water consumption increased significantly from 54870 m3 in 2016 to 68406 m3 in 2017. Every year, the world's water consumption increases despite the limited generation capacity of water resources. Water companies must thus look into models in order to more accurately forecast and plan for water consumption. As water usage rises over time, shortages of water arise. Water prediction is therefore one way to solve this issue [3].

Ethiopia's capital city Addis Ababa receives its water services from the Addis Ababa Water and Sewage Authority (AAWSA). AAWSA was founded in 1971 with the dual goals of providing the city with drinkable water and a sanitary sewer system for the disposal of sewage. AAWSA's activities and functions were fully delineated by the sewage authority proclamation of March 16, 1972, and the preceding papers, which also granted the authority the capacity to borrow. The authority is set up as a stand-alone public body with a distinct judicial personality. The Addis Ababa regional administration owns it in its entirety. Addis Ababa is home to roughly 2.7 million inhabitants, according to the 2007 census. More current estimates from 2016 place the total population at 3.4 million. Approximately 375,000 connections are now used to provide drinking water to these people [4]. The city's wastewater and water supply systems are under the control of AAWSA. The area of the city covered by the water supply is 56% (300 km2). According to AWWSA (2020) above 20% of the population in its service area has in-house connections and use on average 80 to 100 liters per capita per day (lpcd), while the remaining population is served by yard taps. And public taps and use between 15 to 30 lpcd [4].

In scientific terms, prediction is the technique of forecasting future events. It is a technique for estimating the unknown. Strongly dependable, precise, timely, and significant predictions should be made. Making Predictions Water consumption is important when creating a plan or strategy for water use. It provides significant evidence for the effective regulation of water demand [5]. Policymakers need to take action and develop new policies in order to meet this growing demand for water. Predicting water use is one task that aids water supply businesses in reacting to certain behaviors [6]. In the past, well-known machine learning techniques for calculating water-based energy included Support Vector Machine (SVM), K-Nearest Neighbors (K-NN), Random Forest, Naive Bayes, and Artificial Neural Network (ANN) [7]. The deep learning model, a part of the machine learning model, was developed to address a complex problem containing a lot of data [8]. Text, audio, and video datasets have all been successfully processed by deep learning models. Typical deep learning models include Long Short-Term Memory, Gated Recurrent Units, and Bidirectional LSTM [9]. Such deep learning models have been used because the dataset on water usage is sequential. Deep learning techniques find widespread application in fields that handle complex problems. LSTM, bidirectional LSTM, Bi-GRU, GRU, and attention mechanism were employed in this thesis. The primary aim of this study is to analyze and predict the Water consumption of Addis Ababa Water and Sewerage Authority.

Generally, the globe is now facing a dilemma with poor water management. If this trend persist in the future, our planet's water reserves may run out, perhaps leading to a worldwide catastrophe rather than a crisis. Therefore, in an attempt to discover a solution to this enormous problem, governments employ a range of tactics and carry out research using a number of approaches and tools. The intention of this research is therefore to apply deep learning algorithms for water consumption problem in Addis Ababa Water and Sewerage Authority in order to convert the huge amount of customer dataset into useful model. This is helpful in predicting customer’s consumption and supports the efficient handling of resource of the organization.

1.2 Statement of the Problem

Water consumption is rising at an accelerating rate due to a number of causes, including the growing global population, the need to maintain positive rates of economic growth, the focus on widespread industrialization in developing nations, and continued aspirations for greater living standards. This is why accurate investment planning is necessary for energy production, generation, and distribution [10]. For this, an accurate forecasting method is required. Ethiopia's sporadically rising need for water resources is a result of the nation's expanding population, developing economy, urbanization, and technological improvements. Distributing water supplies to all nations, ethnic groups, and Ethiopian citizens is exceedingly challenging, nevertheless, due to a number of variables. Inadequate substation capacity, a deficiency of pipelines, a rise in urbanization, unequal demand for water resources among several users, and so on are some of these reasons.

The number of clients in AAWSA is periodically growing, and as a result, more water resources are needed as many enterprises may choose to launch new ventures. To meet consumer needs, precise water resource forecasting is therefore required. The Network Planning section of AAWSA is tasked with creating a feasibility study based on the annual new connection and water requests from customers who register on Excel. The department then forecasts the consumption for the upcoming year using this information. To analyze and forecast water consumption, AAWSA currently employs a manual approach. A large amount of water is consumed by the numerous residential appliances that customers may utilize. This causes water to become unavailable and could result in a large number of new customers needing water meters being added to waiting lists. There must be a scientific water consumption prediction in order to account for this water consumption trend.

For this, an accurate forecasting method is required. Ethiopia's sporadically rising need for water resources is a result of the nation's expanding population, developing economy, urbanization, and technological improvements. Distributing water supplies to all nations, ethnic groups, and Ethiopian citizens is exceedingly challenging, nevertheless, due to a number of variables. Inadequate substation capacity, a deficiency of pipelines, a rise in urbanization, unequal demand for water resources among several users, and so on are some of these reasons.

In general, some of the earlier research conducted in Ethiopia demonstrates the use of data mining; therefore, the researcher's work involved classifying customers based on their data through the application of data mining algorithms and the Waikato Environment for Knowledge Analysis tool. This study is special since the firm was introducing new data processing tools and apps to streamline and automate the billing process. These days, inadequate water management has global implications. Our world's water supplies might run out at that very time, and if this pattern is to continue, we might be faced with a global tragedy rather than a crisis. Therefore, in an attempt to discover a solution to this enormous difficulty, governments employ a number of strategies and conduct research utilizing a variety of techniques and instruments. Deep learning algorithms can be used to monitor, control, and even reduce water consumption. In order to achieve the primary goal of this thesis, we employ deep learning models such Gated Recurrent Units, Long Short-Term Memory, and Bidirectional LSTM and attention to assess and estimate water use utilizing the recently acquired data from the system.

The aim of this study is therefore to apply deep learning algorithms for developing water consumption prediction model for Addis Ababa Water and Sewerage Authority to help the organization to distribute the water with the required amount for customers. To this end, this thesis can answer the following research questions:

1. What is AAWSA's customer water consumption in terms of month and years?
2. Which variables are influencing water consumption?
3. Which types of deep learning model are efficient for analyzing water datasets and predicting water consumption?

1.3 Motivation

The minimal amount of water that each person should consume each day for drinking, washing, cooking, and keeping good hygiene is 5 liters, or 1.31 gallons, if they live in a moderate environment. Moreover, 1.6 liters of water per day for women and 2.0 liters for males is recommended, yet on average, every Ethiopian uses between 5 and 20 gallons of water per day for household usage [11]. One of the main goals of the water consumption research was to increase public awareness of the water crisis as a limited energy source. This could potentially result in the correction of faulty human water consumption patterns, as it was accomplished by employing modern technology to assess and estimate future water consumption. Utilizing sensor networks is one of the best ways to track and record the amount of water used in residences and commercial buildings in order to modify or lower water use.

1.4 Objectives

1.4.1 General Objective

The general objective of this study is to develop a model that analyze and predict the water consumption based on historical data of customers using deep learning approach.

1.4.2 Specific Objectives

To achieve the desired general objective, the following specific objectives have been set.

- ❖ To review literature done on analyzation and prediction of Water consumption.
- ❖ To collect and prepare data for data prepossessing.
- ❖ To identify the main attributes that can help to predict water consumption.
- ❖ To develop a water consumption prediction model using deep learning algorithms.
- ❖ To test the developed deep learning models.
- ❖ To evaluate the performance of predictive models with evaluation metrics.

1.5 Scope and Limitation of the Study

The analysis and forecasting of water consumption are the main goals of this study, which will help the business allocate water resources in the quantity required for a particular location. Addressing the research issues mentioned in Section 1.2 is the aim of this project. Examining the Addis Ababa Water and Sewerage Authority as a whole is difficult, though, because it is a large organization with several districts and branches spread out around the city. The study solely focuses on the AAWSA dataset as a result. Thus, the study's main focus is on postpaid clients in the AAWSA. This analysis was conducted using DL techniques; no data from prepaid consumers or other sources are included. Processing the data on a computer or other device with a high processing capability is required due to the study's sole focus on AAWSA and the volume of data. On the other hand, the research question that is addressed in Section 1.2 is the focal point of this study's response. The study used DL methods like RNN, LSTM, Bidirectional LSTM, and GRU algorithms to achieve the goal of the study.

1.6 Significance of the Study

By offering a precise prediction model, this study is crucial for AAWSA and AAWSA consumers to become friends. As a result, AAWSA is able to guarantee that water resources are allocated equitably across the company and supply water accumulation that is effective in compliance with its clients' resource requirements. One other benefit of this effort is:

- To balance supply and demand planning.
- To improve the reliability of the water grid by enabling professionals to plan and make strategic decisions on behalf of a water supplier.
- To achieve environmental and economic goals is facilitated by integrating renewable resources harmoniously and optimizing the load needed.
- To make essential decisions about the purchase of equipment, the generation of water, and the development of substations.
- It helps the organization to distribute the Water with the required amount for customers

But the research also contributes to the development of a model that predicts the trend in AAWSA water usage. Precise forecasts of future water consumption are crucial to the policy-making process since decision-makers have to decide what to do and create new plans to handle this growing demand for resources. The predictive model helps a water utility make important decisions about the construction of new infrastructure, the upgrading of existing transmission lines, the expansion of existing substations, the acquisition of machinery, and other related matters.

1.7 Research Methodology

This study employs the experimental research design, in which one or more independent factors are applied to one or more dependent variables and then adjusted to determine the impact of those variables on the dependent variables.

1.7.1 Literature Review

Comprehensive reviews of the literature are carried out in order to gain sufficient knowledge about the different methods of analysis and prediction. In particular, research on data mining methods, data analysis tools, and deep learning prediction algorithms. Thus, gaining the necessary information on a particular subject can be facilitated by reading and critiquing all of this literature as well as related publications. As a result, various books, websites, and research papers have discovered and used the appropriate techniques and resources for putting the various system components into practice.

1.7.2 Data Collection

Addis Ababa Water and Sewerage Authority water consumption data collected from the AAWSA office. Thus, we collect the Water consumption of each user under the Addis Ababa Water and Sewerage Authority East Addis Ababa District. We retrieved from the system database in Excel format by Database Administrator due to the system's many attributes and the AAWSA office's incomplete application of those qualities. The dataset that was supplied included 45,860,708 records of the city's water consumption during a 7-year period, namely from 2016 to 2022. The dataset contains Boolean, categorical, and numerical attributes with a memory capacity of 8.2 GB.

The dataset comprises the monthly water usage of hospitals, schools, museums, and residential clients' residences, according to the subject matter experts. The staff's monthly water use is included in both the active and retrieved records. The commercial dataset include historical data from the manufacturing, food and beverage, wood, mill, micro-enterprise, and metalworking industries. The unstructured data must undergo additional processing in order to serve as the input for the prediction model. Ultimately, these data are arranged and go through the preparation stage of data processing so that testing and experimentation can be done with them easily.

1.7.3 Data Processing

To ensure optimal use in model training, the gathered data is preprocessed. Meaningless data gets dropped and removed. This type of data encompasses any information that machines are unable to accurately comprehend and understand. In order to facilitate computation, data is normalized for organization and all water usage figures are converted to a range between zero and one.

1.7.4 Model Development

The development of analysis and prediction models makes use of several deep learning algorithms, such as GRU, LSTM, and Bi-LSTM. Deep learning has gained popularity recently because it performs best when trained on massive amounts of data [12]. Additionally, it mimics the neuronal network found in the human brain. Deep learning is a subset of machine learning that makes use of deep neural networks. Connected layers are used in the construction of deep learning systems. The input layer is the first layer, while the output layer is the last. The terms "Hidden Layers" refer to all of the intermediate layers. A network that has more than two neuronal layers is called "deep." Neurons are the building blocks of every Hidden layer. The neurons are all related to one another. The incoming signal from the layer above is processed by the neuron before being propagated. We tested several algorithms for our challenge by reading a lot of literature and considering the characteristics of our data review.

1.7.5 Testing and Evaluation

The suggested analysis and prediction model is evaluated, using part of the data acquired for training and the rest for testing in order to assess the model's performance. Here, the model's performance is tested and assessed using the data that have been gathered. Experts verify the data's relevance based on user and expert comments, and the test's outcome is assessed using the MSE, RMSE, and MAE.

1.7.6 Development Tools

In this study, we create the suggested model using a variety of techniques. We make use of both hardware and software tools. Hardware requirements needed to design and develop the proposed solution are listed below:

- ✓ Processor: corei7
- ✓ RAM:12GB
- ✓ HDD: 1TB

Whereas the software requirement tools that we intend to use are Pycharm and Jupyter notebook as editor. We use Python programming to develop the model.

- ✓ **Pycharm**: **-** is an Integrated Development Environment (IDE) designed specifically for Python developers, offering a large selection of vital tools. These technologies work together seamlessly to provide a pleasant environment for data science, web, and Python development that is productive.
- ✓ **Jupyter notebook: -** is a free, open-source, interactive web application that allows researchers to create documents including software code, computational results, narrative explanations, and multimedia. Although computational notebooks are not new, Jupyter in particular has been more and more well-known in recent years. An

active user-developer community and a revamped architecture that enables the notebook to speak dozens of programming languages have contributed to this quick adoption.

- ✓ **Python: -**is an object-oriented, high-level programming language with dynamic semantics that is interpreted. Its dynamic type and dynamic binding, along with its high-level built-in data structures, make it a very attractive language to employ as a scripting or glue language to join existing components. Python's easy-to-learn syntax puts readability first and lowers the cost of software maintenance. Python's support for modules and packages promotes program modularity and code reuse. The extensive standard library and the Python interpreter are freely available for usage and distribution in source or binary form on all major platforms.

1.8 Organization of the Thesis

This research thesis organized in to five chapters. A brief introduction to the research, the problem statement, the general and specific objectives, the scope and limitations of the investigation, and the study's significance are all included in the first chapter. The Second Chapter begins with a summary of the Literature Review. It then moves on to a brief explanation of the Deep Learning Algorithms Techniques and how they operate, as well as an evaluation of the Model, a review of related works, and a gap analysis. An overview of the approach is given in Chapter Three, including Data Preparation, Data Analysis, and Modeling with RNN, LSTM, Bidirectional LSTM, GRU, and Comparison of Deep Learning Models. The experiment overview and discussion of the results are presented in Chapter 4, which is followed by the proposed architecture and experiment dataset. The research's conclusion and recommendations, along with some suggestions for the future, are presented in the last chapter.

CHAPTER TWO

2. LITERATURE REVIEW

2.1 Introduction

This chapter provides an explanation of the underlying theoretical concepts needed to solve the analysis and prediction problem related to water usage. In order to better grasp the idea and look into the study problem, it includes a basic definition and reviews pertinent literature. To comprehend the use of deep learning and other machine learning techniques for water consumption analysis and prediction, important books, journal documents, articles, and research papers are further reviewed. The effort also contributes to demonstrating the significant demand for water in the community. Because of a rapid rise in water consumption, it is necessary to accurately predict how much water will be used. Scientific methods are needed to create accurate water consumption predictions, which will help address the city's and its district's water resource scarcity. Unexpected disruptions to the water supply are avoided with the help of a trustworthy water usage forecast.

As a result, the primary purpose of the research was to forecast water usage in Addis Ababa using deep learning techniques which this gives civilians more time to prepare for water limitations or restriction.

2.2 Background

The case study concentrated on Addis Ababa city because its population and economic growth differ markedly from that of other cities in the country. The city is currently 540 km2 in size, according to the city map. The city has a lovely ambiance and is located at an elevation of 2,000 to 3,000 meters above sea level. At its lowest and greatest points, the measured yearly average temperature ranges from 10 to 25 degrees Celsius. There is about 1,250 mm of rain there annually. The amount of water available in the city is not enough to fulfill the growing demand [13] resulting from human activities (such as population development and economic expansion). It is anticipated that by 2050, water demand would outnumber water supply in this city. Roughly 450,000 m3/day of surface and ground water are available overall; leakage makes up 36.5% of that amount [14] [15] [16]. Rising urbanization, increased individual water use, and the consequences of climate change are all predicted to exacerbate the problem of water shortage in cities. The Addis Ababa Water and Sewage Authority was reestablished by Decree No. 10/87, subsequent to the Proclamation

No. 68/1963. Since its establishment, the authority which was granted authority by Decree No. 10/87 has worked with partners to make water and sewage services accessible to the general population. In order to meet Addis Ababa's rapidly increasing population, the authorities utilized water sources and constructed a drainage system to supply clean water and advanced sewage services to city inhabitants. In order to safely serve the community and give it the services it needs, it has built eight branch offices.

Water is just one of the many resources the city can manage. It gives individuals a means of support and protects both humans and plants. Generally speaking, water is used for energy generation, transportation, agriculture, and recreation [17]. In sixty nations, the United Nations estimates that water scarcity affects seven million people [18]. A significant portion of the global population today struggles with water scarcity [19]. According to national statistics, 17% of Ethiopians live in households with regular access to water. Significant ecological and social changes are occurring in Ethiopia in order to maintain the equilibrium between the supply and demand for water. As a result, the government has made it its top priority to implement a policy pertaining to water usage [20] [21]. It is essential to anticipate and forecast water usage in order to improve resource management and to offer technical assistance for resource assessment and management [22]. Furthermore, by projecting water consumption, water supply companies can improve their management and service quality. It is almost impossible to envision a modern world without the use of water, given the growing worldwide need for water supplies. Moreover, trustworthy calculations of water usage can provide the requirement for year-round water management and use, guaranteeing the quality of the water supply.

This research is based on energy consumption management, with a focus on water consumption management. After that, we conducted more study to focus solely on Deep Learning's ability to estimate water consumption. Using machine learning techniques, we examined a few studies on electricity or other forms of energy consumption.

2.2.1 Approach to Water Consumption Analysis and Prediction

With the recent development of computing technology, machine learning and deep learning are often used to predict nonlinear data in the hydrology field. The nonlinear time series properties of water consumption can be reflected by machine learning and deep learning models like random forest (RF) and long short-term memory, which are used to predict water consumption. When recurrent neural network and ARIMA models were tested to anticipate potential water demand, it was found that the RNN model outperformed the ARIMA model [23]. In order to create a water consumption forecast model for every client, this study employed a Recurrent Neural Network technique based on deep learning. Deep learning models such as recurrent neural networks (RNNs) can take dependencies between successive time steps into consideration. Due to their propensity to hold onto historical data in time series, they are frequently utilized to forecast sequential jobs. On the other hand, they struggle with the vanishing/exploding gradient problem. To solve the RNN vanishing gradient problem, many models were published in the literature, such as LSTM and GRU neural networks. Below is a description of these models along with others..

2.2.1.1 Autoregressive Integrated Moving Average (ARIMA)

ARIMA is a statistical analysis technique that makes use of time series data to forecast future trends or to get a deeper understanding of the data set. In our study we used the ARIMA to forecast the future water demand of the city using the historical water consumption dataset. Based on [24] Time-series data and statistical analysis are used by the ARIMA model to understand the data and forecast future values. The ARIMA model makes predictions using linear regression and attempts to explain data by using time series data on its historical values. In this study we used the ARIMA parameters of P (number of lag observations in the model), D (number of times the raw observations are differenced) and Q (the size of the moving average window) with values (5,1,0).

2.2.1.2 Machine Learning Approach

Nowadays, a lot of industries, including industry, health, the environment, energy, and municipal utilities, use machine learning, a subset of artificial intelligence. Because it can identify patterns in historical data, machine learning is a well-known effective tool for forecasting the future. Working with multiple sorts of data, applying numerous algorithms and statistical techniques, processing large amounts of data, data analysis, and future prediction are just a few of the noteworthy aspects of machine learning. The other two are self-learned and automatic improvement through experience. A selection of the machine learning algorithms are described.

Support Vector Machine (SVM): The SVM model, which can examine the data for both regression and classification, is one supervised machine learning approach. The SVM is a linear model, but it can also be used to non-linear models. SVM does this analysis using a method known as the Kernel approach. By splitting the input dataset into two phases or dimensions, the kernel, as a mathematical function, is one of the SVM hyperparameters that seeks to identify the most effective and efficient dividing line or boundary [25].

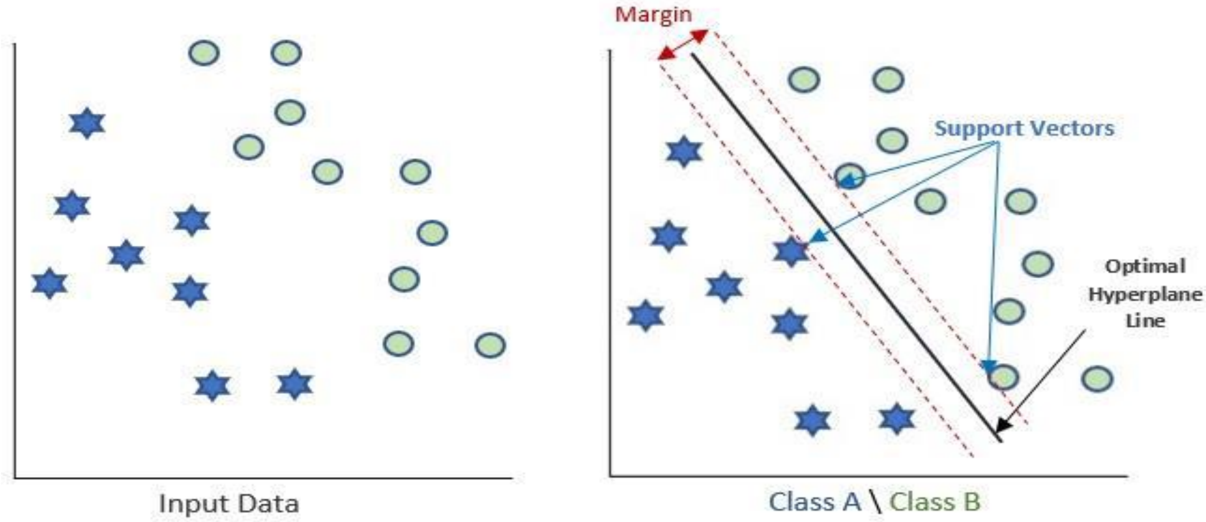

Figure 2. 1 The input data is separated by hyper plane line

Random Forest: Another popular machine learning model (Figure 2.2) is a tree-based approach with supervised learning. In the context of the technique, "random" refers to the use of numerous randomly generated decision trees, which collectively constitute a "Forest" of trees. One decision tree's training set contains a great deal of variation. When multiple decision trees are used simultaneously by the RF on a single dataset sample, the resultant low variation is obtained from all of the decision trees. In actuality, the confluence gathering and decision tree construction in each sub-branch improve the algorithm's performance. As a result, more than one decision tree is combined to provide the output or result.

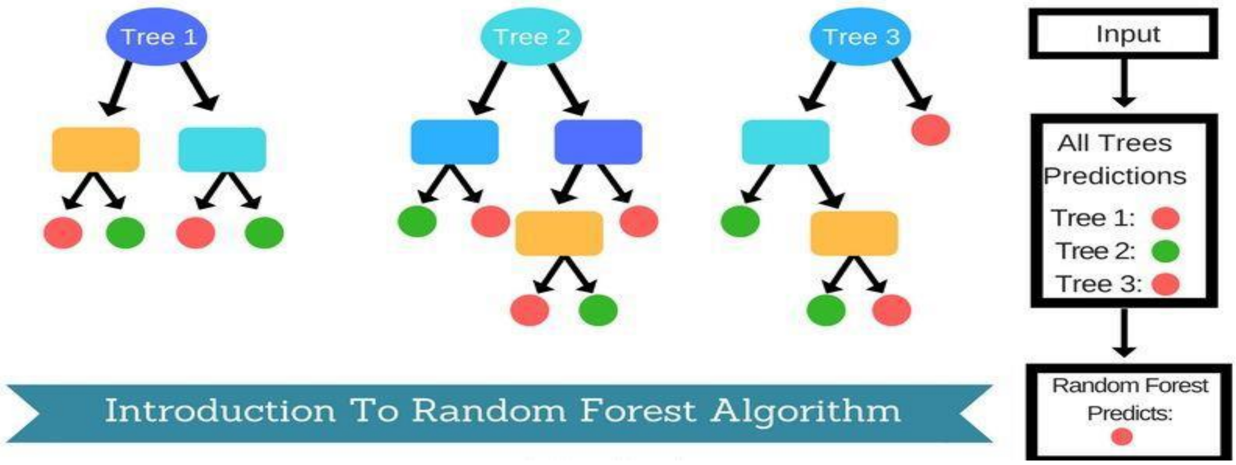

Figure 2. 2 Random Forest[26]

(Extreme Gradient Boosting) represents a further efficient tree-based method. XGBoost can be used to solve regression and classification difficulties in machine learning problems with a small or medium-sized structured dataset. The infinite decision trees, for example, lead to overfitting issues and increased model complexity. These problems can be resolved by the XGBoost approach with Ridge regression and Lasso regression [26]. This program is able to deal with missing values since it understands their trend. By "learning" from the best missing values, the XGBoost algorithm automatically develops this pattern during the "training" phase. Using XGBoost's autonomous learning capacity is another way to address the sparseness of raw data issue.

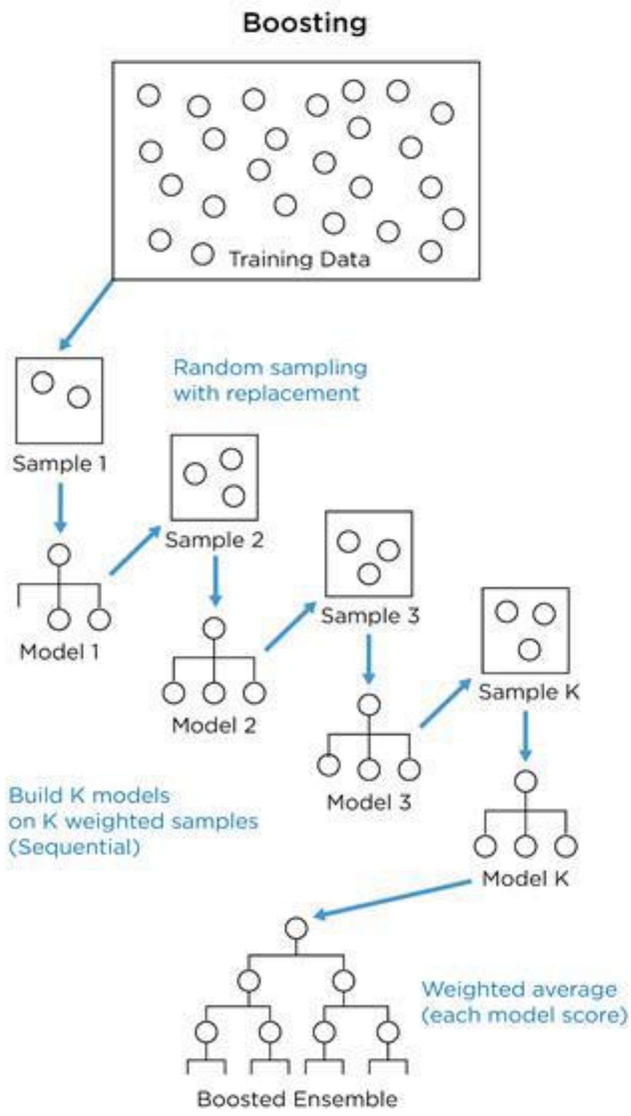

Figure 2. 3 XGBoost algorithm[26]

K- Nearest-Neighbors Regressor (KNN): Initially established by Fix et al. [27] and then revised by Cover et al. [28], the KNN approach (Figure 2.4) is one of the non-parametric strategies used for both classification and regression challenges. Depending on how effectively this approach performs, each data point is assigned a value or weight. The algorithm tries to find out how similar the newly entered data point is to the points in the training dataset, and then assigns a new value to the incoming input [29]. The new input data point, which denotes a new observation or input, and the data points in the training set are separated by the KNN. Therefore, before using this strategy, it is important to consider the size of our dataset. Because it calculates the longer distances that yield inferior outcomes on a larger scale.

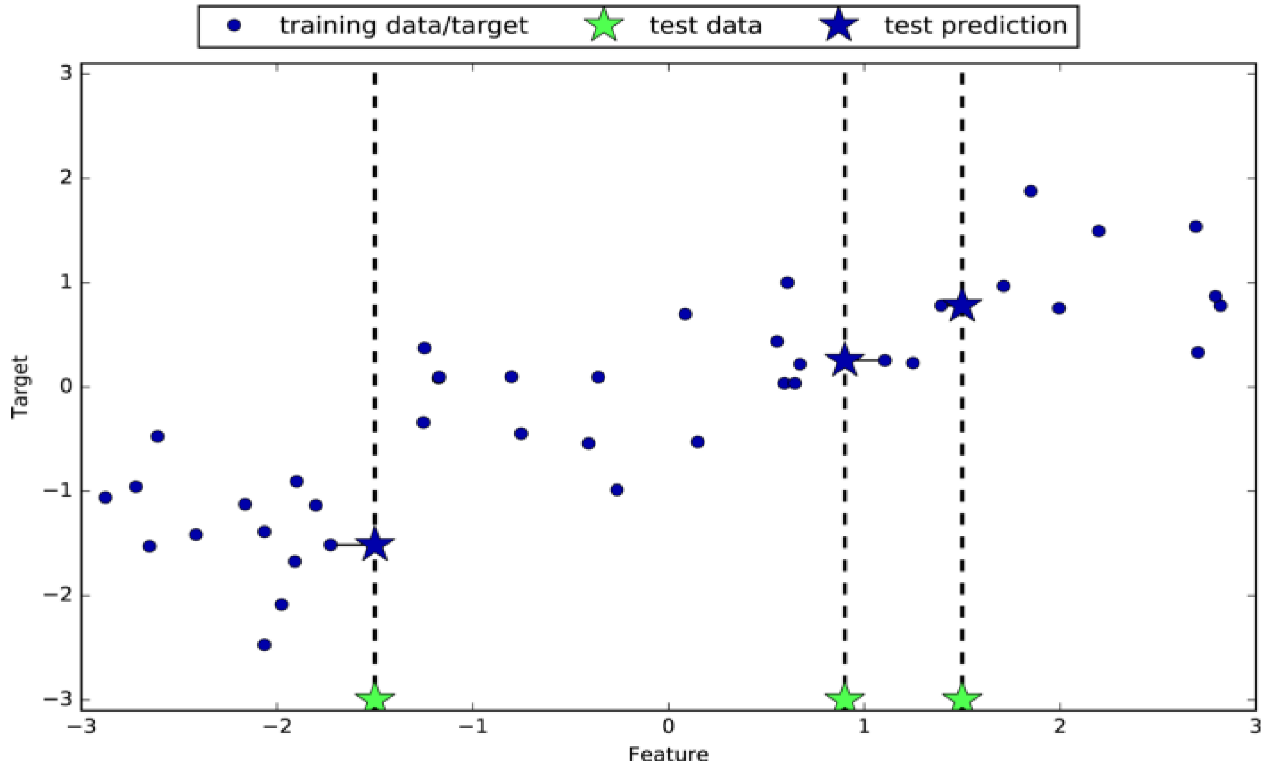

Figure 2. 4 K- Nearest-Neighbors Regressor [29]

Regression and KNN for classification vary primarily in that the former calculates the Mode while the latter computes the Mean of the closest K neighbors. With the ability to store all of the training data, the KNN can forecast numerical objective values using distance functions.

2.2.1.3 Deep Learning Approach

Recurrent Neural Networks

Alternatively, the neurons in Prediction of Water Consumption Using Machine Learning RNNs are connected to a sub network of neurons in other types of Artificial Neural Networks. Figure 2.5 illustrates a simple RNN layer architecture. Like regular ANNs, RNNs can have a large number of hidden layers or connections with complex behaviors.

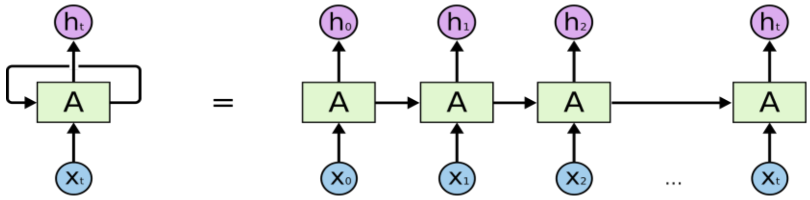

Figure 2. 5 The RNN architecture [30]

LSTM Model

Various machine learning models are applied to predict non-linear data. The LSTM model is an effective advanced neural network model when the input data is sequential data. Thus, the LSTM model was used to predict water consumption in this study.

Hochreiter and Schmidhuber created the artificial recurrent neural network architecture known as the LSTM with deep learning applications in mind. Because the LSTM model was developed to address the vanishing gradient problem while training conventional RNNs, it is widely used in studies related to time series prediction. The components of an LSTM unit are as follows: an input gate (it), an output gate (ot), a forget gate (ft), and a cell state (Ct). States communicate with one another regarding information. In order to update the information, the cell state uses forget gates, input gates, and output gates to send outdated information (Ct−1) and updated information (Ct) to the subsequent cell. After receiving the previous hidden state data (*h*t−1) and the new input data (xt), the forget gate decides what information to send to the cell state. Through the use of an activation function (tanh), the input gate establishes which information among the new information has to be updated and creates a new information value (C˜t). The information designated as the output is decided by the output gate. Equations 2.1–(2.6) can be applied to illustrate this procedure. In this instance, σ is the activation function; Wf, Wi, Wc, and Wo are the gate weights; while bf, bi, bc, and bo are the gate biases [31].

$$f_t = \sigma \ \ W_f \cdot [h_{t-1}, x_t] + b_f \ \ , \qquad (2.1)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i), \qquad (2.2)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C, \qquad (2.3)$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t, \qquad (2.4)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o), \qquad (2.5)$$

$$h_t = o_t \cdot \tanh(C_t). \qquad (2.6)$$

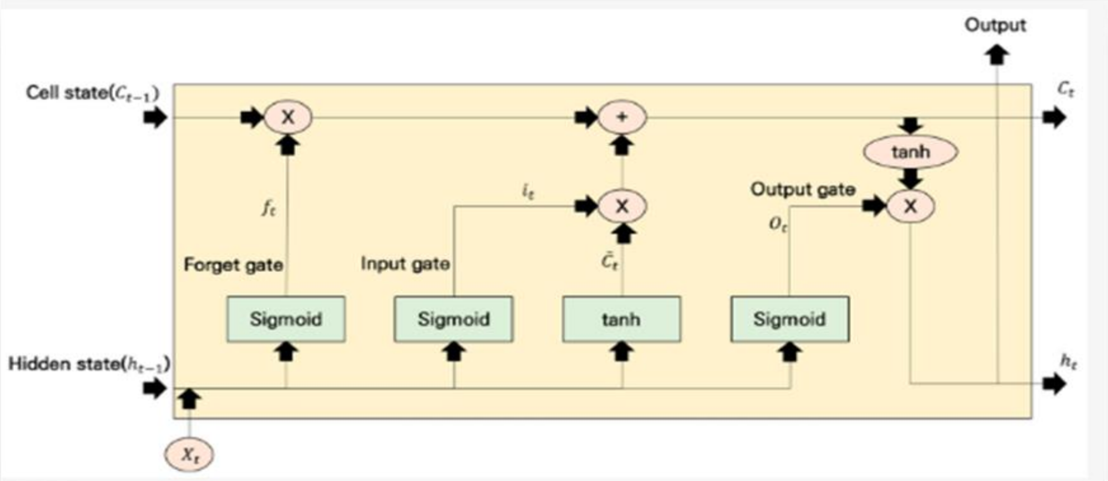

Figure 2. 6 Conceptual diagram of the LSTM [31]

Bidirectional LSTM Model

LSTM models that combine input data from the past and future of a single time step are known as bidirectional LSTMs. With Bidirectional LSTM, we can store knowledge from both the past and the future at any moment. In order to establish a foundation for weight distribution in the following stage, this Bidirectional LSTM extracts bidirectional nonlinear features from water consumption data. The long-term data of the "past" and "future" is preserved by Bi-LSTM. After that, the data is linked as this layer's output. Many feature extractions can be avoided by using the bidirectional LSTM layer. The whole Bidirectional LSTM module connects two LSTM networks with opposing timings to provide bidirectional data information for the input sequences using the same [32].

If the water consumption at time t is taken into consideration, the input features vector is xt. The preceding moment's output is $ht-1$, while the prior buried layer's state is $ct-1$. Therefore, as indicated by formulas (2.12) and (2.13), the hidden layer's current state is ct and its output is ht.

$$c_t = f_t c_{t-1} + \mathrm{i}(\mathrm{w}^c \mathrm{X} + b^c) \quad (2.12) \qquad h_t = \mathrm{o}_t s(c_t) \quad (2.13)$$

Formula (1) indicates that the previous output was $ht-1$, and the input X for the current time is composed of the input feature vector xt, which is the water consumption data at time t. The weight is represented by wc, the offset by b, and the states of the input and output activation functions are indicated by g and s, respectively. The activation values of the input (i), forgetting gate (f), and output gate (o) at time t are represented by it, ft, and ot, respectively, as illustrated in Figure 2.10.

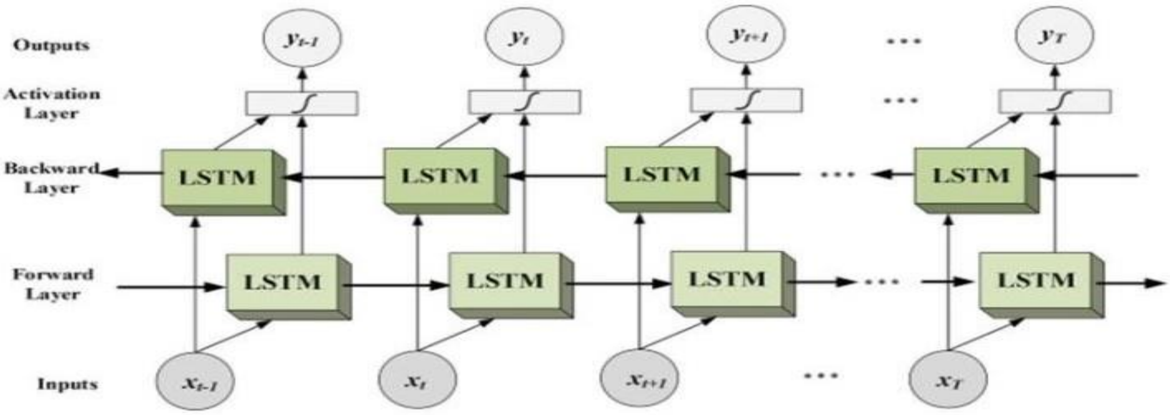

Figure 2. 7 Bidirectional LSTM Architecture [32]

GRU Model

The gated recurrent unit is a more complex type of recurrent neural network based on long short-term memory. In other respects, the GRU internal unit is similar to the LSTM internal unit in that it combines the inbound port and forgetting port into a single update port. The GRU is believed to be simpler to compute and implement than the LSTM unit that served as its model. It keeps the LSTM resistant to the problem of disappearing gradients. Internal state upgrades require fewer computations due to its simpler internal structure, which facilitates training. A reset port controls whether the current state should be combined with past data, and an update port controls how much of the state data from the previous instant is kept in the current state. The mathematical procedures that control the locking mechanism of the GRU cell are as follows: [33].

Where Wz, Wr, and W stand for the associated connected input vector's weight matrices. Bz, b, and b are bias, and Uz, Ur, and U stand for the weight matrices from the prior time step. The candidate hidden layer is represented by h̜t, the update gate by zt, the reset gate by rt, and the logistic sigmoid function by σ. [33].

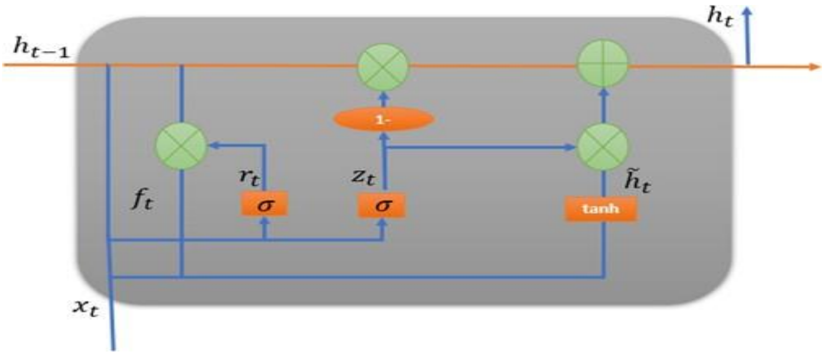

Figure 2. 8 The cell structure of a gated recurrent unit [33]

Attention based Model

Thanks to the attention mechanism important parameters can be given more weights during training which helps the model to fit the given data. The model can concentrate more on input factors with greater weights [34]. In this study we have implemented the attention mechanism from [35]. We used all algorithms to implement the attention mechanism.

2.3 Model Evaluation

Depending on the pertinent metrics, mean absolute error, mean absolute percentage error, or root mean squared error are used to evaluate the performance of the trained model. The final step is to assess each model's performance using the required hyperparameters.

Mean Squared Error (MSE): - determines the error or total square difference (actual value minus expected values).

$$RSE = \frac{1}{n}\sum_{i=1}^{n} (y_i - \tilde{y}_i)^2 \dots\dots\dots \text{(2.14)}$$

Root Mean Squared Error (RMSE): - is the variation between the actual values as determined by RSME and the values predicted by the model. Stated differently, it refers to an inaccuracy in the technique utilized to ascertain the precision and inaccuracy percentage of any machine learning model. A lower RMSE value thus indicates good model performance. The unit's exact match to the target variable is one of RMSE's primary properties.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \dots\dots\dots \text{(2.15)}$$

Where,

n Denotes the feature variables, $\boldsymbol{y}_i$ **i**[th] denotes actual values, $\hat{\boldsymbol{y}}_i$**i**[th] denotes the predicted.

Mean Absolute Error (MAE): - shows the mean absolute deviation of a dataset between its values as expected and as observed. Based on the absolute value of the inaccuracy, the mean absolute error (MAE) represents our average amount of forecast error. A minimum value denotes good model performance. The range of the MAE's values was 0 to infinity.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}| \quad (2.16)$$

Where,

$\hat{y}$ - The predicted value of y, $\bar{y}$ -Mean value of, N- Number of observation/rows

2.4 Review of Related works

2.4.1 Foreign Related Works

Although, the concept of predicting utility demand using sophisticated machine learning algorithms is currently being researched in a wide range of fields. The core of our research is the management of energy consumption, with a focus on water consumption. Note that the fundamental parameters of this research were established by first examining other publications concerning electricity or other types of energy consumption control with machine learning techniques. From then on, we focused our research on machine learning capabilities to predict the amount of water consumed in an hour.

Froehlich et al. [36] developed a low-cost, easy-to-use method for measuring a house's water usage, utilizing a non-intrusive sensor in every valve. Particularly in the kitchen sink, toilet, and shower, they measured the water pressure as each valve was opened or closed. Based on the magnitude of the pressure drop, they estimated the volume of water used in a section of water infrastructure by analyzing the pressure of water waves in valves that were sent to sensors. In this experiment, a standard water spigot or outlet was equipped with a HydroSense sensor. Employing machine learning techniques, it assessed labeled data gathered by sensors from ten houses in four cities with different plumbing systems, age ranges, and architectural designs. They obtained 97.9% accuracy by using the cross-validation technique and linear regression to examine the stream of residual trials in the test set.

Somontina et al. [37] explored the possibility of measuring a home's monthly water consumption using a single, non-intrusive point. The three main objectives of the study were to: identify fixtures and faucets, measure household water consumption in real-time using a non-intrusive sensor, and compute the monthly cost and volume of water consumed in the home. The amount of water consumed in this study was measured using radiofrequency (RF), a machine learning technique with 92.9% accuracy.

Seyedzadeh et al. [38] carried out one review using state of the art machine learning techniques to estimate building energy use. They suggested that reducing the risk of global warming could be achieved by increasing the energy efficiency of new structures. The optimal model to improve energy performance was investigated using machine learning techniques such as ANN, SVM, clustering, and Gaussian-based regressions. They found that artificial neural networks (ANNs) are useful for temperature and humidity prediction data and have been used widely in energy prediction. ANN is an effective tool for energy modeling with trustworthy building forecasts, but because it necessitates exact sample selection, accurate network architecture, and fine parameter setting, its structured energy modeling is unable to accommodate a local smallest problem. While SVM can create a model with a limited number of samples and parameters, it cannot. While clustering, as an unsupervised learning technique, can classify buildings based on their many features rather than their kind or structure, Gaussian process (GP) and Support Vector Machines (SVM) can offer sufficient performance with minimal parameters.

Another study, Sornam et al. [39] talked about how data mining tools are important for developing smarter, greener design in smart cities. In order to produce smarter buildings and a cleaner environment, this study looked at different data mining algorithms for energy usage prediction. They used a seven-phase approach to data analysis, with data mining being one of the steps to find a useful pattern. 1) Support Vector Machines (SVM) can be a useful tool for classifying both linear and non-linear data; 2) Decision Trees are a reliable algorithm for extracting rules from sensor data; 3) Neural Networks are used to enhance the performance of the trained network; and 4) Meta-algorithms, such as the ensemble method, can reduce variance and enhance prediction accuracy by combining several machine learning algorithms into a single model. 5) Data mining employing machine learning algorithms can manage big datasets because sliding windows can be a useful method for evaluating the flow of data collected by sensors.

Fernández et al. [40] examined the function of big data in smart home energy management. Economic planning may reduce building occupant energy consumption by up to 40% of total energy consumption, according to a 2011 European Commission statement. In order to find the optimum model for managing vast data that projected users' weekly energy use, the role of several machine learning algorithms was examined in this study. In order to obtain useful information for enhancing energy efficiency, their objective was to test several machine learning techniques on the unprocessed data produced by smart houses. Four modules were created out of their research, with three parts devoted to machine learning. 1) Several weighted algorithms with 74% accuracy, a supervised classifier, and clustering techniques were used for the recognition data used by each device. This energy usage pattern was predicted with 90% accuracy using machine learning algorithms.2) The ability to analyze and interpret recorded data about users' energy use helps define consumption patterns and provide recommendations to other users who are behaving similarly to change their energy consumption habits. This study's experiment showed how machine learning might be used to classify, store, and analyze data according to requests, allowing big data processes to handle enormous datasets. Furthermore, other smart settings that are comparable to the project's smart homes can benefit from the same techniques and methodologies that were researched for using and evaluating data collected in smart houses.

Study by Vafeiadis et al. [41] utilized smart meter data—such as water or electricity usage sensors—to recognize occupants inside of buildings. In order to express two states, such as presence or absence, they ran an experiment using the water and power sensors dataset to identify work status. The amount of water or power that occupants use is used as a barometer of occupancy in this experiment because consumption indicates whether or not residents are there. A few of the machine learning algorithms employed in this research include the 80.94% accurate Decision Tree (with AdaBoost), the 79.83% accurate SVM-POLY (the Polynomial), the 80.06% accurate SVM-RBF, the 80.23% accurate Random Forest, and the 80.21% accurate ANNs (the backpropagation algorithm). Using machine learning techniques and expertise, they concluded, is a solution to the occupancy recognition challenge.

Study by Li et al. [42] Energy use in Chinese homes. Their aim was to employ machine learning techniques to forecast families' annual energy consumption. Three neural network models—General Regression Neural Network (GRNN), Backpropagation Neural Network (BPNN), Radial Basis Function Neural Network (RBFN), and Support Vector Machine (SVM) model—were tested on the remaining nine residences. Both machine learning models were successful in predicting power usage, based on the research conducted in these 59 households. Although GRNN fared better than other Neural Networks models in this domain, SVM had the best prediction accuracy with an error of less than 10%. Taking this into consideration, they concluded that the most notable aspect of SVM, the structure risk minimization (SRM) principle, is used in neural networks to reduce the upper bound of the generalization error rather than the training error [43].

Study by Zhao et al. [44] To model the energy consumption of different buildings, Energy Plus software is utilized. To build a prediction model, they used machine learning techniques including the Gaussian RBF Kernel and the SVM algorithm. Their goal was to reduce energy consumption by forecasting building energy use according to complex features. Each house had a number of significant involvement criteria, such as the type of cooling, air infiltration, thermal zones, structure, duration, and certain utilities like water heaters or lights. Other parameters included people, fenestration surface, building shape, and location. In fact, this effort involved performing the SVR and the RBF for the training phase of several SVMs simultaneously. Using the Pisvm tool on a large dataset, they expedited the model training phase. Three experiments were carried out in order to assess the SVMs' ability to predict and analyze each building's energy usage. Upon examining multiple buildings with consideration to their structural features, the third stage involved examining parallel support vector machines on an extensive dataset featuring a variety of unique constructions. The results of the experiment indicated that SVMs and SVR produced the best results for this study, despite their claims that real-time energy data from different types of homes could be better to have more accurate and better results close to the actual situation and that this assessment based on simulation buildings with the historical dataset is insufficient. More data should definitely be collected for machine learning algorithms in order to increase accuracy. Subsequently, the large-scale dataset analysis requires more time, and the training phase requires a significant amount of time as we explore all the building datasets. In order to improve this difficulty, they opted to use parallel SVM execution to shorten the learning time. SVM proved to be the most effective method for analyzing large energy datasets by minimizing average training sets, whereas SVR produced good results when assessing energy usage by identifying the most important research factors.

Study by Zhao et al. [45] an examination of renewable and sustainable energy sources, and how a few variables impacted building energy consumption forecasts. The difficulty in accurately predicting energy usage due to several factors, such as occupancy, lighting, and behavior changes caused by HVAC systems, drove them to pursue this goal.

Study by Y. Zhang and Q. Li [46] to predict a solar photovoltaic station's photovoltaic power output over 1.5 head in time, a stacked LSTM network was created. The outcome forecasting model may provide accurate forecasts since the expected power output signal reacts to every variation and follows the trend. Applying the models to the test data results in an RMSE of 0.11368.

Study by G. Shen, Q. [47] this investigation examined historical power usage statistics as well as information from a mining company. When compared to utilizing a model, the accuracy achieved by the regressive CNN-SVR is higher. The results show that the only methods that can reliably predict consumption in the near future are CNN and a regular regressive SVM. The RMSE, MSE, and MAPE values—0.6687, 0.8564, and 1.975%, respectively—were used to evaluate the effectiveness of the RCN SVR model. This implies that there is little chance of prediction error in the proposed model.

Techniques, like Box and Jenkins, The UCI machine learning data repository provided the researcher with a dataset of 2075259 recordings spanning 47 months that was collected from Paris and made available to the public [48]. A method for predicting power consumption and spotting abnormalities based on LSTM neural networks was reported in a paper by [49]. By utilizing the LSTM model, the forecasting inaccuracy is reduced by 22% when compared to the ARIMA technique. Using data from the preceding several months or years, the study aimed to identify unusual monthly or annual behaviors. The anomaly and electrical theft are then located using the detection result, proving its greater use than an algorithm.

Study by A. Mosavi and A. Bahmani [50] it was assessed how accurate different sales forecasting techniques were. In general, the accuracy of the LSTM and ARIMA models differs. Because non-linear models are thought to provide better accuracy for challenging tasks, the researcher selected a non-linear LSTM model that would perform better than the linear ARIMA model in both instances. The outcomes of the one-day-ahead prediction scenario were not statistically significant, even if the results suggest that LSTM could be improved.

2.4.2 Local Related Works

We looked far and wide for studies on water consumption prediction using deep learning algorithms for Local Related Works, but we were unable to locate any in Ethiopia. Through trend analysis and prediction, Wegayehu Enbeyle's work [51] used advanced time series (ARIMA) techniques to forecast the amount of water consumed in southwest Ethiopia. The study's goal is to identify which ARIMA models best fit the existing data on water use in Tepi Town, southwest Ethiopia, so that water use may be properly predicted. The study examined monthly water use data for Tepi town from January 2016 to December 2021. By

2022 GC, the water supply is anticipated to increase in light of the conclusion of his prediction period.

The related work discussed above uses a variety of approaches, measurements, and data types for different machine learning algorithm applications. Some papers compare algorithms based on water usage statistics, while others use standard machine learning approaches to predict water consumption. Some others use data to predict stock prices. Instead, using the real AAWSA data, we apply advanced and effective deep learning techniques to estimate and analyze water consumption based on each client's monthly and annual water consumption. In addition to predicting, we thoroughly investigate the district's key customer segments' water usage habits.

Table 2. 1 Summary of literatures review

Reference	Research Title	Model used	Research Gaps	Accuracy	Dataset
[52]	Development of a Deep Learning-Based Prediction Model for Water Consumption at the Household Level.	LSTM, ARIMA	The dataset is very small	MAPE has 0.89	7 days data Korea
[53]	Real-time Water Consumption Prediction	GRU	The goal was to use only for pipe bursts alarming.	MAPE has 6%	five days Standford
[54]	Prediction of Water Consumption Using Machine Learning	SVR LSTM	The dataset covers 6 month but is very small	Accuracy, P and R46.21% ,60.40% and46.21% Respectively for MLP model.	Sixth month data of City of Sarpsborg (Norway)
[55]	LSTM and ARIMA for sales forecasting in retail	LSTM, ARIMA, ANN And SVM.	The author predicts one grocery may affect theresult	RMSE and MAE 0.1375 0.1468 for the LSTMmodel.	4 years 80 000 Records Kaggle.

[56]	A CNN-LSTM-Based Model to Stock Prices prediction	CNN-LSTM, CNN, RNN, And LSTM.	The researcher does notconsider the size of geographical locations	MAE, RMSE,and $R^2$27.564,39.688 and 0.9646 for CNN-LSTM Model	21 years/ 7127 trading days

2.5 Summary

The related studies show that, even with successful modeling by machine learning algorithms, there remains a gap. This gap arises from the inability of the models mentioned above to capture the sequential dependency between samples. As a result, RNN models, such as LSTM, were used in the most recent studies, and the results were the best. Additionally, the use of ensemble models and deep learning models, which combine many algorithm types, was found to be an efficient way to provide accurate predictions. To the best of my knowledge, only a single study conducted in Ethiopia has investigated the topic of predicting water use; the majority of other studies are based on other foreign nations. For AAWSA specifically, it is not feasible to directly adopt the majority of those research' findings. As a result, the foundation for this research is AAWSA. The researcher endeavored to examine several academic works about the forecasting of water usage by various water suppliers. The fact that most water providers follow different procedures for handling customers and services, organization structures, and strategies globally has also led to the use of different approaches in the studies that have already been conducted. These approaches include different datasets, preprocessing activities, and research methods.

It is suggested in Related Works that by incorporating various client consumption factors and datasets, a more potent model may be created, improving the results. It should be emphasized that, based on our research into water consumption management, we did not find any research in Ethiopia that examined water consumption prediction using deep learning algorithms. In general, as stated above, this research uses four times more sample size (45,860,708) datasets than all the other researchers mentioned here in the study.

CHAPTER THREE

3. RESEARCH METHODOLOGY

3.1. Introduction

In order to determine how one or more independent factors affect one or more dependent variables, this study employed an experimental research design in which one or more independent variables are changed and applied to the latter. This chapter describes the implementation of the water consumption prediction system in detail. The dataset was gathered by the researcher from the Addis Ababa Water and Sewer Authority office. The dataset is preprocessed to eliminate inaccurate values and fill in any gaps after it has been collected. After that, the preprocessed dataset is normalized such that its values fall into a range. Lastly, the suggested water consumption prediction model is trained using deep learning techniques. The suggested model's overall system architecture is depicted in Figure 3.1.

3.2. Proposed architecture of Water Consumption Prediction

3.2.1. System Design

This section provides a detailed discussion of the design and components of the proposed water consumption prediction model. Research methodology is the systematic approach to problem solving in research. It aids in the researcher's understanding of the approaches or plans employed in the investigation. The study uses three deep learning algorithms: GRU models, Bidirectional LSTM, and LSTM. This section describes the recommended architecture for assessing and forecasting water consumption. The suggested architecture includes all of the fundamental development tasks, such as data collection, preprocessing, data splitting, and model training with recurrent neural networks, such as LSTM, bidirectional LSTM, Bi-GRU, GRU, and attention mechanism. Lastly, the model's performance was evaluated using the MSE, RMSE, and MAE. The dataset is in a time-series format covering 7 years of water consumption, monthly historical data, and the deep learning algorithm had a more robust model than conventional models for time-series prediction, which is why these three deep learning algorithms were chosen.

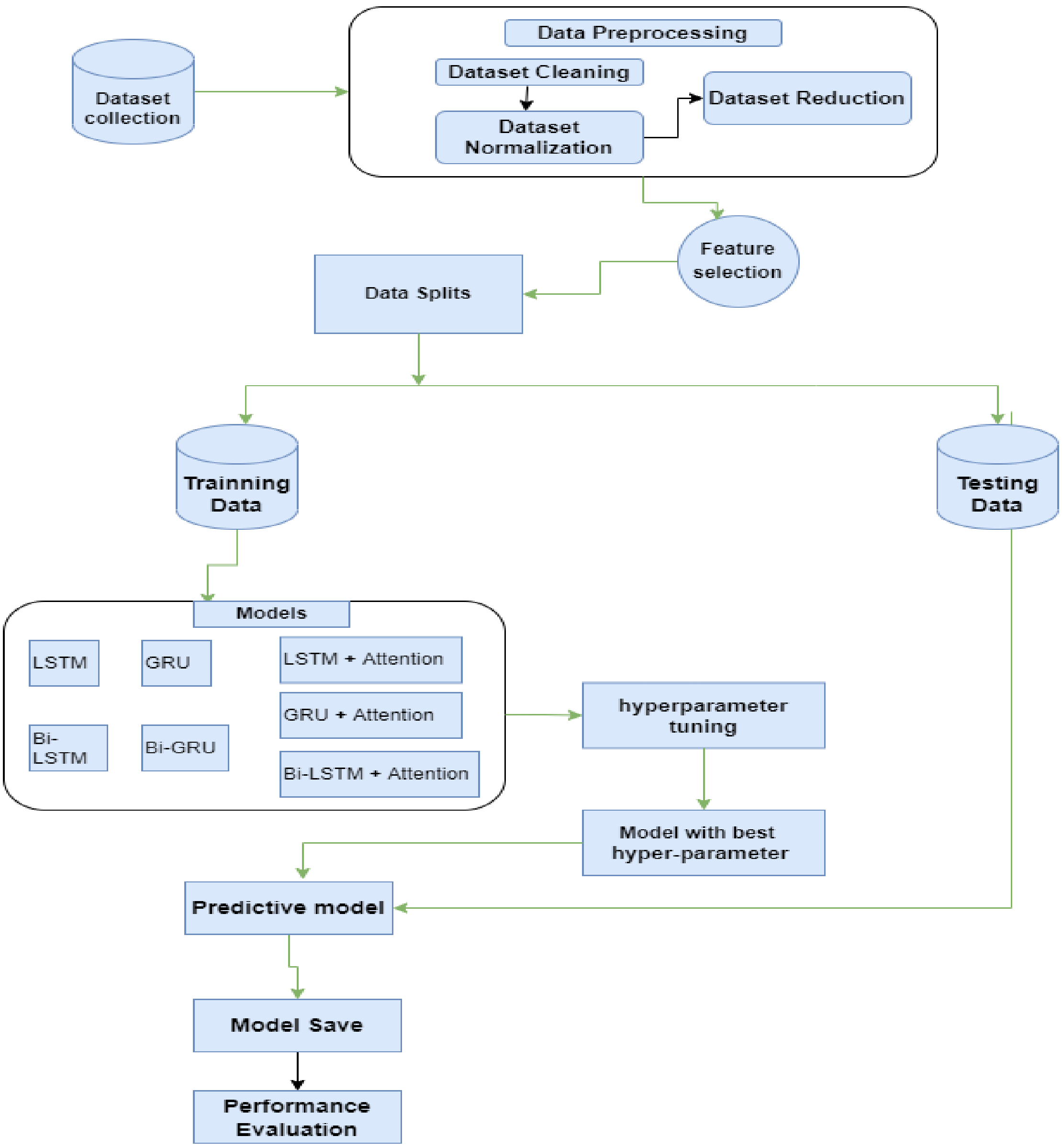

Figure 3. 1 System architecture of the proposed model

3.2.2. Dataset collection

The dataset is collected from the Addis Ababa Water and Sewage Authority office which is the water service provider of Ethiopia's capital city Addis Ababa. We went to the office and request for the water consumption dataset. The staffs finally shared the dataset via Google drive. Table 3.1 shows the shared dataset were the city's 7 years of water consumption record specifically from the year 2016 to 2022. The dataset had a memory size of 8.2 GB. The raw consumption data was provided to us as a set of CSV files containing consumption entries for each client at different billing periods. A single row consisted of a client's identifying information like customer name, customer key, contract number, reading date, bill key, branch, metro key, meter rent, previous reading, the start and end dates of their current billing period, how much water was consumed over the current billing period, and several other features pertaining to their water access. These features were numerical, categorical, or Boolean. Our goal was to transform all the raw data into a dataset that contained snapshots of the entire city's water consumption for each month.

Table 3. 1 The general statistics of the datasets per year

Year in GC	Number of record
2016	5437546
2017	5916207
2018	6385687
2019	6755830
2020	6960605
2021	7117077
2022	7287756
TOTAL	45860708

3.2.3. Feature selection

There are about 61 attributes in the gathered data. All of the variables are time-consuming to compute and have the potential to skew the algorithm's processing flow; yet, they are not necessary for our suggested prediction model. The variables that are most important to the model are then chosen. Therefore, we may shorten the computational process time and enhance the accuracy and performance of the suggested model by eliminating redundant, unnecessary, and irrelevant input variables. Since more appropriate data leads towards higher accuracy in the prediction, we used the feature selection technique to prevent high computation time and increase the quality of model performance. In this study, we considered and chose the date and time (Measurement Time) and the amount of water consumption (Value) as significant features or variables.

3.2.4. Dataset preprocessing

After data collection the next step were transforming the raw data into a format that deep learning algorithms can use. The collected data includes an unexpected range of values, missing values, incorrect combination of data, and so forth. The data preprocessing stage helps us to transform the raw data into accurate and acceptable data for the model and future predictions. In this study data pre-processing stages such as dataset cleaning, dataset reduction and dataset normalization steps are done. The Figure 3.2 shows all the preprocessing stages done in this study.

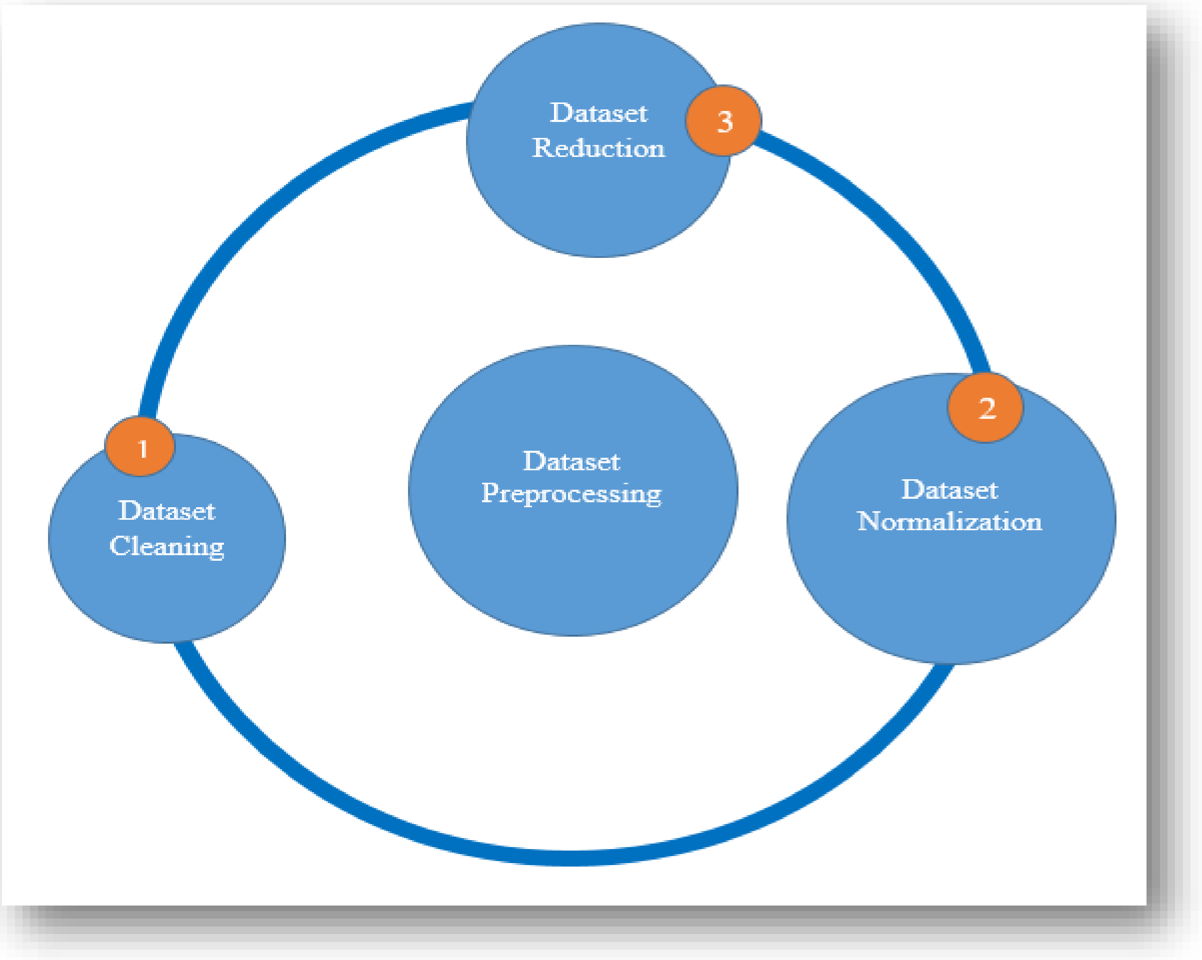

Figure 3. 2 Dataset Preprocessing

3.2.4.1. Dataset cleaning

The practice of eliminating erroneous, missing, and incomplete data from datasets and substituting them is known as data cleaning. The following is a list of several data cleansing methods. Data may have erroneous values for a number of reasons. Finding, eliminating, and attempting to close any gaps in insufficient data is crucial. It involves changing the present values or removing a single row or column of data. A few examples of data cleaning include matching the data, identifying and eliminating duplicate data entries, and replacing missing values with default values. To fill in the gaps, regular values like "Not Available" or "NA" might be used. Even if missing values can be manually filled in when a dataset is large, doing so is not recommended.

The attribute's median value can be used to replace the missing value when the data is not normally distributed, but the attribute's mean value can be used when the data is regularly distributed. Regression or decision tree approaches can be used to replace a missing variable with the most likely value. In this study we used the mean value to replace the missing value. The Figure 3.3 shows an algorithm in pseudo code to clean the dataset.

In [41]:
```
data.drop(data.columns[[0,2,3,5,6,7,8,9,10,11,12,13,14,16,17,18,19,20,21,22,23]], axis = 1, inplace=True)
data
```

Out[41]:

	Installat.	Valid fr.	Entry value
0	NaN	NaN	NaN
1	5.000000e+09	04.02.2018	359
2	5.000000e+09	07.03.2018	329.915
3	5.000000e+09	07.04.2018	50.085
4	5.000000e+09	08.05.2018	810
5	5.000000e+09	06.06.2018	391
6	5.000000e+09	04.07.2018	384
7	5.000000e+09	08.08.2018	338
8	5.000000e+09	04.09.2018	495
9	5.000000e+09	06.10.2018	366
10	5.000000e+09	06.11.2018	437
11	5.000000e+09	13.12.2018	343.558
12	5.000000e+09	09.01.2019	537.442
13	5.000000e+09	10.02.2019	393
14	5.000000e+09	14.03.2019	434
15	5.000000e+09	12.04.2019	399
16	5.000000e+09	06.05.2019	729
17	5.000000e+09	06.06.2019	727
18	5.000000e+09	04.08.2019	493
19	5.000000e+09	06.09.2019	401
20	5.000000e+09	08.10.2019	919.041
21	5.000000e+09	03.11.2019	274.959
22	5.000000e+09	22.12.2019	200
23	5.000000e+09	22.02.2020	200

Figure 3. 3 Algorithm for dataset cleaning

3.2.4.2. Dataset Reduction

This process helps in the reduction of the volume of the data, which makes the analysis easier yet produces the same or almost the same result. Storage space is also reduced as a result of this reduction. Data compression, numerosity reduction, and dimensionality reduction are a few methods of data reduction. We decreased the dimensionality of the data by applying an aggregation function to a subset of attribute values. Specifically, we used the Pandas GroupBy function to group the customer's total water use by month and year, resulting in useful information for subsequent analysis and prediction. The Figure 3.4 shows an algorithm in pseudo code to reduce the dataset in to appropriate format.

```
In [117]: B=data1.groupby(['CUSTEMER ID','MONTH','YEAR'], as_index=False) ['CONSUMPTION'].sum()
          B
```

Out[117]:

	CUSTEMER ID	MONTH	YEAR	CONSUMPTION
0	5000101808	1	2020	276.00
1	5000101808	1	2021	380.00
2	5000101808	1	2022	362.00
3	5000101808	2	2020	278.00
4	5000101808	2	2021	424.00
5	5000101808	2	2022	302.00
6	5000101808	3	2019	116.00
7	5000101808	3	2020	408.00
8	5000101808	3	2021	434.00
9	5000101808	3	2022	368.00
10	5000101808	4	2020	514.00
11	5000101808	4	2021	502.00
12	5000101808	4	2022	318.00
13	5000101808	5	2018	440.00
14	5000101808	5	2019	64.00

Figure 3. 4 Algorithm for dataset reduction

3.2.4.3. Dataset Normalization

Normalization is a useful technique for improving the quality of data. When there is a substantial difference between the values of the feature, the feature with the larger value inevitably affects the prediction output. Normalization is one of the steps in the machine learning data preparation process that we employ to scale all variables equally. This scale alteration neither changes nor eliminates the amplitude of any value. As a result, we applied the normalization technique to ensure data integrity and reduce data redundancy by placing all of our variables into the same range.

In the study, we normalize the dataset using the scikit-learn MinMaxScaler function. The function scales the data to a range in order to assist in normalizing each feature. The MinMaxScaler function, which defaults to 0 and 1, scales each feature separately so that the values have a specified minimum and maximum value. Equation 3.1 shows the formula to scale feature values to between 0 and 1.

Equation 3.1 MinMaxScaler function calculation

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

It is calculated by subtracting the minimum value from each entry and then divide the result by the range, where range is the difference between the maximum value and the minimum value. The figure 3.5 shows an algorithm to normalize the dataset.

```
from sklearn import preprocessing
from sklearn.preprocessing import MinMaxScaler
min_max_scaler = preprocessing.MinMaxScaler(feature_range=(0, 1))
dataset = min_max_scaler.fit_transform(data_df['CONSUMPTION'].values.reshape(-1, 1))
```

```
[10] dataset[0:10]

array([[0.00359979],
       [0.0033082 ],
       [0.00050227],
       [0.00812209],
       [0.00392066],
       [0.00385047],
       [0.00338922],
       [0.0049635 ],
       [0.00366998],
       [0.00438192]])
```

Figure 3. 5 Algorithm to normalize the dataset

3.2.5. Data Splitting

Data splitting is commonly used in deep learning to prevent and manage model underfitting and overfitting. In that case, a deep learning model fits training data too well and is unable to fit new data consistently. Thus, a percentage split of 80/20 was used to separate the entire dataset into training and testing datasets. Eighty percent of the dataset is included in the training set of our model, and twenty percent is included in the test set. A portion of 10% of the dataset was set aside for validation.

3.2.6. Hyper parameter Tuning

The process of choosing the best values for a machine learning model's hyper parameters is known as hyper parameter tuning, which control the model's learning process, include learning rate, neuron count, number of hidden layers, and number of epochs. The goal of hyperparameter tuning is to identify the settings that yield the best performance on a particular task. In this work, we adjust the hidden layer's hyperparameter, neuron count, and epoch count.. It is worth noting that hyperparameter tuning is an iterative process, and we performed multiple iterations to find the best combination of hyperparameter for our water consumption prediction. By performing hyperparameter tuning, we ensured that our deep learning models were optimized and capable of achieving the best possible performance for our water consumption model. We used selected deep learning model, optimizer, activation, learning rate, batch size, epochs, dropout and neurons as a parameter configuration.

3.2.7. Model Design and Development

This study uses the LSTM, Bidirectional LSTM, and GRU algorithms to forecast the amount of water consumed because of the advantages of deep learning. Because they can take into account the non-linearity of sequences, maintain the prior state, and recall past occurrences by connecting past and present neurons, recurrent networks are frequently utilized in sequential data scenarios, such as time-series problems. The RNN models are highly suitable for time-series prediction issues because of this property [57]. A few of the models employed in this research have extra characteristics to improve their functionality. As we covered in Chapter Two, by processing the input sequence both forward and backward and combining the outputs, bidirectional LSTM connections allow the model to learn from information in the past as well as the future. With three gates as opposed to two, the LSTM is more potent and efficient, while the GRU is a simpler model with two gates—the relevance and update gates—and performs computation more quickly. Attention mechanisms enable the model to focus on the most relevant parts of the input sequence by assigning different weights to different time steps and utilizing these weights to compute the output.

3.2.8. Model training

The following stage in the deep learning method to water consumption prediction is model definition. Several hyperparameters are defined during the training phase to enhance the model's performance. The number of epochs, Bach size, amount of early halting, and other parameters are examples of such hyperparameters. Fitting our data to the chosen deep learning algorithm models of LSTM, GRU, and BILSTM is essentially possible during the model's training phase. The baseline hyperparameter was examined with several hyperparameters.

3.3. Evaluation metrics

The performance of the model is assessed using the assessment metrics. The Mean Squared Error was employed in this investigation. The squared mean of the discrepancy between expected and actual data is known as the mean absolute error. The average of the error squares is calculated using the mean squared error. It does this by giving back the average of all square sums for each discrepancy between the estimated and actual values. Even in cases where the forecasts are 100% accurate, the MSE is always positive. Our model performs better when the MAE value is lower and closer to zero [58]. The estimator's bias (the degree to which the estimated values deviate from the true values) and variance (the extent to which the estimates are dispersed) are both taken into account.

CHAPTER FOUR

4. EXPERIMENTATION AND RESULTS

4.1. Introduction

This section outlines the efficacy of our suggested deep learning models for the study and prediction of water usage, including LSTM, bidirectional LSTM, Bi-GRU, GRU, and attention mechanism architecture. RNNs can retain historical data in time series and are frequently used to forecast sequential tasks since they are one of the deep learning models that can handle dependencies between subsequent time steps. However, they suffer from the vanishing/exploding gradient problem [59]. As a result, other models, including GRU neural networks [60]and LSTM [61], were put out in the literature to address the RNN vanishing gradient issue. Performance, speed, training time, and model size criteria were used to test those models against the baseline hyperparameters as well as with one another. In addition, the preparation of data and evaluation metrics among which are Mean Absolute Error, Mean Squared Error, and Root Mean Squared Error that we use to evaluate the performance of each model are covered in this part. Each test was based on a dataset that included historical water usage data collected over a seven-year period. AAWSA is the source of historical water use data. Before providing the dataset to the prediction model, we carry out feature engineering and data pretreatment operations. The following hyperparameters were used in the model's implementation and assessment: the dropout layer, activation function, optimizer, loss function, and evaluation metrics. We examine how well state-of-the-art deep learning systems estimate water usage using preprocessed historical data. Ultimately, based on the original problem statement and research questions that we establish during proposal, the model's overall result as well as the experimentation result are assessed and analyzed for future use.

4.2. Dataset description

The dataset was gathered from the Addis Ababa Water and Sewage Authority office, which provides water services to Ethiopia's capital city Addis Ababa, as was covered in chapter three. Data on previous customers from 2016 to 2022 GC was included in the dataset. The dataset contains customer IDs, payment information for invoices paid within the designated month and year, and each client's water consumption amount. In order to develop the water consumption prediction model, 45,860,708 processed cases from the data collection were used. The dataset is divided into three subsets: testing, validation, and training. After splitting the training, validation, and testing data, the model is built using the selected deep learning prediction technique. By taking into account various hyperparameters suitable for the given water usage issue.

Figure 4.1 shows a sample of the dataset after loaded using the Pandas library. As we discussed in Section 3.2.3 the dataset collected has around 61 attributes or columns. The figure shows the first 9 attributes of the dataset. Figure 4.2 shows the water usage of Addis Ababa city in September. As shown in the figure there are users that has 0 consumption. Figure 4.3 shows the statistics of water consumption of Addis Ababa from the year 2016 to 2022.

Out[17]:

	NAME	CUST_KEY	INST_KEY	CONTRACT_NUMBER	CHARGE_GROUP	ROUTE_KEY	READING_DATE	BILL_KEY	WALK_ORDER
0	[illegible] N/A	C-000000333856	IN-000000336130	638935.0	FOUNTAIN	153AD2	NaN	BPT-0481376838	12915.0
1	[illegible] WOINESH [illegible]	7290904	7290884	140647.0	DOM	154AD2	NaN	BPT-0481032692	12950.0
2	[illegible] SHEKURE	7314662	7314642	155077.0	NON-DOM	152AD1	NaN	BPT-0481261015	64276.0
3	[illegible]	7318476	7318456	631456.0	DOM	169AD1	NaN	BPT-0481297344	58130.0
4	[illegible]	7287848	7287828	142470.0	DOM	141AD1	NaN	BPT-0481003296	24950.0

5 rows × 61 columns

Figure 4. 1 Dataset after loaded with panda

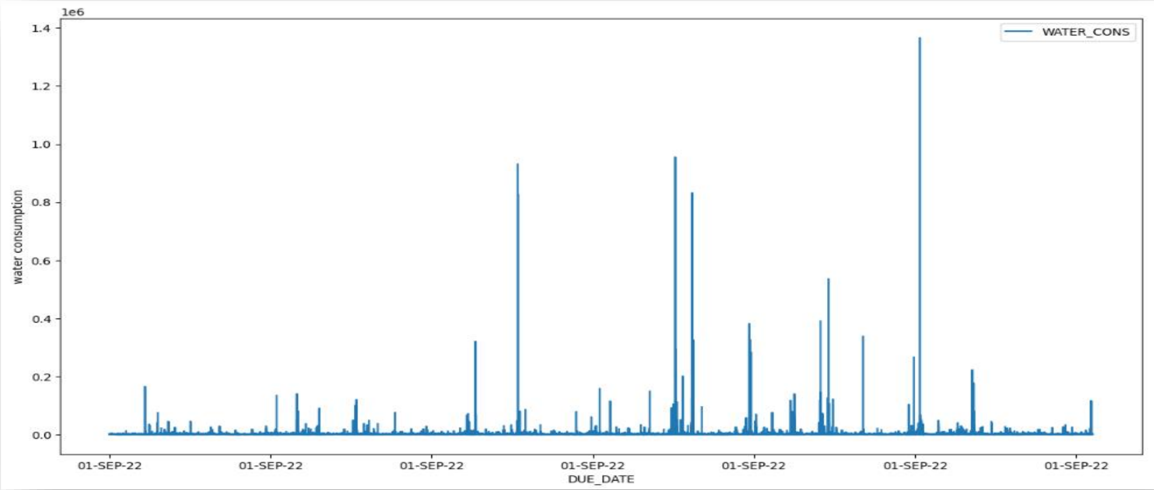

Figure 4. 2 September 2022 water usage

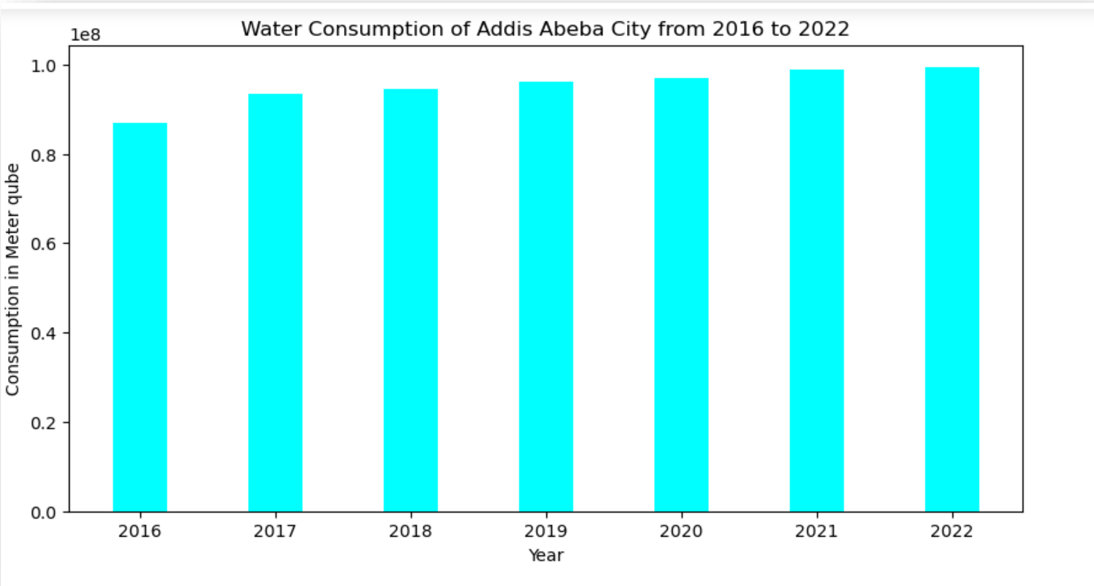

Figure 4. 3 Water consumption of Addis Ababa from 2016 to 2022

Next, we choose a few features that will be useful in developing the prediction model. An example of the dataset with certain properties is displayed in Figure 4.4 the dataset has a lot of empty values, as the graph shows. During the dataset preprocessing phase, these empty values are eliminated. The proportion of the dataset with empty values is displayed in Figure 4.5 The properties curr_cons, tot_cons, and water_cons have about 9% empty values, as the figure illustrates.

Out[19]:

	PREV_RDG	CURR_RDG	CURR_CONS	TOT_CONS	WATER_CONS	DUE_DATE
0	6024	6031	7.0	7.0	16.80	01-SEP-22
1	3380	3404	24.0	24.0	123.59	01-SEP-22
2	1924	1929	5.0	5.0	12.00	01-SEP-22
3	822	822	NaN	NaN	NaN	01-SEP-22
4	3935	3950	15.0	15.0	60.50	01-SEP-22
5	1076	1090	14.0	14.0	55.65	01-SEP-22
6	1023	1025	2.0	2.0	4.80	01-SEP-22
7	1238	1242	4.0	4.0	9.60	01-SEP-22
8	1954	1999	45.0	45.0	655.65	01-SEP-22
9	551	553	2.0	2.0	4.80	01-SEP-22

Figure 4. 4 Dataset with selected features

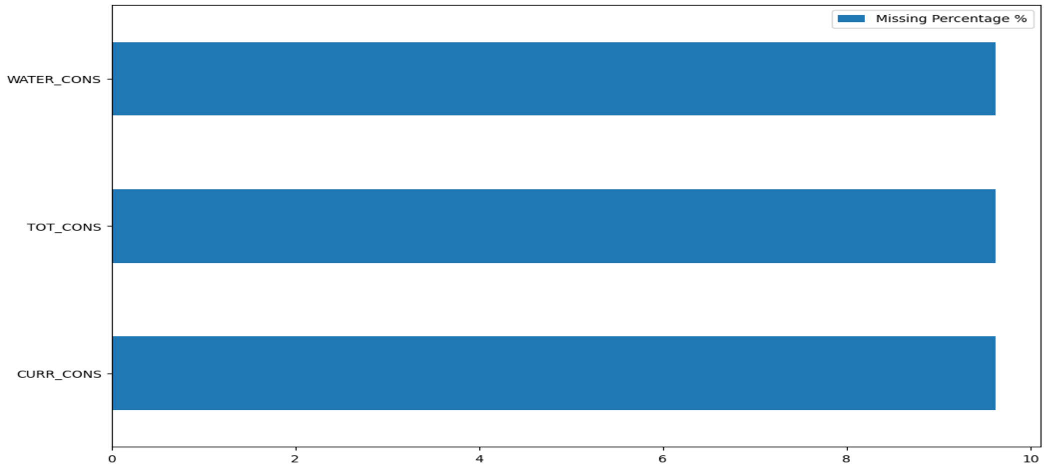

Figure 4. 5 Missing values

In Figure 4.6 the data after preprocessing phase is shown. As shown in the figure there are no empty values now. From Table 4.1, we observe that the number of user’s increases as the year increases and the water demand also increases as the number of user increases.

	PREV_RDG	CURR_RDG	CURR_CONS	TOT_CONS	WATER_CONS	DUE_DATE
0	6024	6031	7.0	7.0	16.80	01-SEP-22
1	3380	3404	24.0	24.0	123.59	01-SEP-22
2	1924	1929	5.0	5.0	12.00	01-SEP-22
4	3935	3950	15.0	15.0	60.50	01-SEP-22
5	1076	1090	14.0	14.0	55.65	01-SEP-22
6	1023	1025	2.0	2.0	4.80	01-SEP-22
7	1238	1242	4.0	4.0	9.60	01-SEP-22
8	1954	1999	45.0	45.0	655.65	01-SEP-22
9	551	553	2.0	2.0	4.80	01-SEP-22
10	614	623	9.0	9.0	31.40	01-SEP-22

Figure 4. 6 Dataset after preprocessing

Table 4. 1 Number of records and water usage in each year

Year in GC	Number of record	Water consumption in m^3
2016	5437546	86949968.0
2017	5916207	93419472.0
2018	6385687	94577534.0
2019	6755830	96130655.0
2020	6960605	96854260.0
2021	7117077	98722254.0
2022	7287756	99372966.0

4.3. Tools and programming language

Deep learning experimentations require high processor and GPU supported computing. In this study, Hardware requirements needed to design and develop the proposed model, we used a computer with a memory capacity of 12 GB RAM, 2.41 GHz processor, and 64-bit Windows 11 operating system.

Deep learning development tools for analysis and prediction come in a variety of open-source, easily customizable forms. Consequently, R and Python programming languages are distributed freely and open-source under the name Anaconda. Its popularity stems from the fact that it offers a plethora of tools for data science and deep learning in a single install, making setup quick and easy. TensorFlow, sklearn, and Keras are the tools utilized in this work. High-level neural network Keras is built in Python and can run on top of TensorFlow. It supports RNN (LSTM, GRU, and BILSTM) and can be easily and quickly prototyped.

Additionally, Keras performs smoothly on both CPUs and GPUs. Using the high-level Keras API, TensorFlow is an end-to-end open-source deep learning platform that constructs and trains deep learning models. While sklearn is employed for data preparation, Keras and TensorFlow are utilized to develop the model for training, validation, and testing.

4.4. Baseline Models

4.4.1. ARIMA Model

The ARIMA model's ability to estimate or predict Addis Ababa's water demand has been evaluated in this study. By comparing anticipated and actual values, we may assess how well the ARIMA model predicts water demand. We may evaluate the ARIMA model's suitability as a foundational model and a benchmark for evaluating the efficacy of more advanced deep learning models by means of this comparison. Throughout the experiment, we used the ARIMA (5,1,0) configuration, which consists of no moving average term (MA), 5 lags for autoregression (AR), and 1st order differencing (I). The model indicated that the forecast was biased (had a non-zero mean).

4.4.2. Machine learning algorithms

In the study, we experimented 5 different machine learning algorithms of Decision Tree, Random Forest, Multinomial Naive Bayes (multinomialNB), Logistic Regression and support vector machine (SVM). During our experiments to test theses machine learning algorithms, we used the default configurations of all algorithms. From the experiment we saw that, Logistic Regression performs better than other algorithms for this dataset

4.5. Hyper parameter Tuning

Hyperparameter tuning is the process of determining which values to use for a machine learning model's hyperparameters. Configurations known as hyper parameters, which control the model's learning process, include learning rate, neuron count, number of hidden layers, and number of epochs. The goal of hyperparameter tuning is to identify the settings that yield the best performance on a particular task. In this work, we adjust the hidden layer's hyperparameter, the number of neurons, and the number of epochs.

4.5.1. Hyper parameter tuning for LSTM Model

In this study we consider the LSTM configuration from [62] as a baseline. The study [62] considered the hyper parameters of two hidden layers, 100 epochs, 12 batch sizes and 6 neuron units as a best hyper parameter combination. In our study we tune the hyper parameters number of neuron, epoch, batch size and hidden layers.

A. Tuning number of neurons hyper parameter

In this case, we tune the number of neurons of LSTM using the selected epoch and batch size hyper parameter. We explored the effect of training this configuration for different numbers of neurons. We run our experiment for 5 repeats and we finally prints the RMSE for the train and the test sets. We used the neurons of [16, 32, 64, 100, and 128] to select the best neuron parameter of the model. The best performance of the model was scored a neuron size of 64. So that 64 is selected as best neuron hyper parameter. Table 4.2 shows the performance score of each neuron size listed.

Table 4. 2 Performance result during tuning neuron size

Hyper parameter	LSTM model performance in	
(List of Neurons)	RMSE	MAE
16	0.18	0.38
32	0.16	0.38
64	**0.15**	**0.35**
100	0.16	0.37
128	0.17	0.38

From Table 4.2 we observe that a neuron size of 64 unites is optimal hyper parameter for LSTM model. Using this optimal neuron size, we tune the hidden layer size of the model.

B. Tuning number of hidden layer hyper parameter

In this case, we tune the number of hidden layers of LSTM using the baseline epoch, batch size and our optimal number of neuron hyper parameter. We explored the effect of training this configuration for different numbers of hidden layers. We run our experiment for 5 repeats and we finally print the RMSE for the train and the test sets. We first tested our model by adding 2 hidden layers with neuron equal to the previous layer. We then add third and fourth hidden layer step wise to the model. However, our model did not improve its performance as hidden layers are added. We observed that our model configuration performs best with 2 hidden layers.

Table 4. 3 Performance result during tuning hidden layer

Hyper parameter	LSTM model performance in	
(number of hidden layers)	RMSE	MAE
2	**0.15**	**0.35**

3	0.17	0.37
4	0.18	0.38
5	0.22	0.43

From Table 4.3 we observe that 2 hidden layers is optimal hyper parameter for LSTM model. Using this optimal hidden layer size and neuron size we tune the epochs of the model.

C. Tuning epoch hyper parameter

During our experimentation to tune epoch of LSTM model, we used the optimal hyper parameters selected above. We explored the effect of training this configuration for different numbers of training epochs. We run our experiment for 5 repeats and we finally print the RMSE for the train and the test sets at the end of each of the 5 experimental repeats and select the optimal epoch having good RMSE score. We used the epoch sizes of [30, 60,100, 120 and 150] to select the best epoch parameter of the model. The best performance of the model was scored using 120 epochs so that 120 is the best for epoch parameter.

Table 4. 4 Performance result during tuning epoch

Hyper parameter	LSTM model performance in	
(number of epoch)	RMSE	MAE
30	0.16	0.36
60	0.15	0.35
100	0.14	0.35
120	**0.13**	**0.33**
150	0.15	0.35

From Table 4.4 we observe that an epoch size of 120 is optimal hyper parameter for LSTM model. Using the above optimal hyper parameters, we tune the batch size of our model.

We build different models like attention using LSTM and bidirectional LSTM.

D. Tuning batch size hyper parameter

In this case, we tune the number of batch size of LSTM using the selected epoch parameter. We explored the effect of training this configuration for different numbers of batch sizes. We run our experiment for 5 repeats and The RMSE for the train and test sets is eventually printed after five iterations of our experiment. We used the batch sizes of [12, 30, 50, 60 and 100] to select the best batch size parameter of the model. The best performance of the model was scored using the above optimal hyper parameters, and batch sizes of 30. So that 30 is selected as best batch size parameter.

Table 4. 5 Performance result during tuning batch size

Hyper parameter	LSTM model performance in	
(batch size)	RMSE	MAE
12	0.16	0.36
30	**0.13**	**0.34**
50	0.15	0.35
60	0.16	0.36
100	0.17	0.34

From Table 4.5 we observe that a batch size of 30 is optimal hyper parameter for LSTM model. Using the above optimal hyper parameters, we build different models like attention using LSTM and bidirectional LSTM.

4.5.2. Hyper parameter Tuning for GRU Model

To tune the hyper parameters of GRU we start from the configuration of the base line model [62] as LSTM model. We start from the hyper parameters of two hidden layers, 100 epochs, 12 batch sizes and 6 neuron units.

A. Tuning number of neurons hyper parameter

In this case, we tune the number of neurons of GRU using the selected epoch and batch size hyper parameter. We explored the effect of training this configuration for different numbers of neurons. After five iterations of our experiment, we finally output the RMSE for both the train and test sets. Next, we choose the neuron with the highest RMSE score. We used the neurons of [16, 32, 64, 100, and 128] to select the best neuron parameter of the model. The best performance of the model was scored a neuron size of 64. So that 64 is selected as best neuron hyper parameter. Table 4.6 shows the performance score of each neuron size listed above.

Table 4. 6 Performance result of GRU during tuning neuron size

Hyper parameter	GRU model performance in	
(List of Neurons)	RMSE	MAE
16	0.15	0.35

32	0.24	0.45
64	**0.12**	**0.31**
100	0.16	0.38
128	0.21	0.42

From Table 4.6 we observe that a neuron size of 64 unites is optimal hyper parameter for GRU model. Using this optimal neuron size we tune the hidden layer size of the model.

B. Tuning number of hidden layer hyper parameter

In this case, we tune the number of hidden layers of GRU using the baseline epoch, batch size and our optimal number of neuron hyper parameter. We explored the effect of training this configuration for different numbers of hidden layers. We observed that our model configuration performs best with 2 hidden layers.

Table 4. 7 Performance result during tuning hidden layer

Hyper parameter	LSTM model performance in	
(number of hidden layers)	RMSE	MAE
2	**0.15**	**0.34**
3	0.16	0.37
4	0.22	0.43
5	0.20	0.40

From Table 4.7 we observe that 2 hidden layers is optimal hyper parameter for GRU model. Using this optimal hidden layer size and neuron size we tune the epochs of the GRU model.

C. Tuning epoch hyper parameter

We employed the best hyperparameters for the GRU's hidden layers and neurons during our experimentation to tune the epoch. We investigated the impact of varying the number of training epochs for this setup throughout training. We ultimately publish the RMSE for the train and test sets after running our experiment five times. The next neuron selected is the one with the highest RMSE score. Next, we decide on the ideal epoch value. To determine the ideal epoch parameter for the model, we employed the epoch sizes of [30, 60, 100, 120, and 150]. Sixty epochs were used to score the GRU model's greatest performance, meaning that sixty is the ideal epoch parameter.

Table 4. 8 Performance result during tuning epoch

Hyper parameter	GRU model performance in	
(number of epoch)	RMSE	MAE
30	0.16	0.37
60	**0.14**	**0.35**
100	0.16	0.35
120	0.19	0.38
150	0.15	0.35

From Table 4.8 we observe that an epoch size of 60 is optimal hyper parameter for GRU model. Using the above optimal hyper parameters, we tune the batch size of our model.

D. Tuning batch size hyper parameter

Using the above optimal hyper parameters of GRU we tune the number of batch size. We explored the effect of training this configuration for different numbers of batch sizes. After conducting our experiment five times, we finally print the RMSE for both the train and test sets. We used the batch sizes of [12, 30, 50, 60 and 100] to select the best batch size parameter of the model. The best performance of the model was scored using the above optimal hyper parameters, and a batch size of 50. So that 50 is selected as best batch size parameter for GRU model.

Table 4. 9 Performance result during tuning batch size

Hyper parameter	LSTM model performance in	
(batch size)	RMSE	MAE
12	0.17	0.36
30	0.15	0.34
50	**0.14**	**0.33**
60	0.16	0.35
100	0.16	0.32

From Table 4.9 we observe that a batch size of 50 is optimal hyper parameter for GRU model. Using the above optimal hyper parameters, we build different models like attention using GRU and bidirectional GRU.

4.6. Model Building

From Section 4.5 we have discussed hyper parameter tuning for the LSTM and GRU algorithms. As stated in previous section we have got optimal hyper parameter values for LSTM and GRU. Table 4.10 shows optimal hyper parameter for both algorithms.

Table 4. 10 Optimal hyper parameters

Optimal hyper parameter	Model	
	LSTM	GRU
Hidden layers	2	2
Number of neuron	64	64
Batch size	30	50
epoch	120	60

Using the optimal hyper parameters of LSTM stated in Table 4.9, we have trained Bi-Directional LSTM, Attention with LSTM. As well using the optimal hyper parameters of GRU stated in Table 4.9, we have training Bi-Directional GRU, Attention with GRU.

From our experiments we observed that Bi-LSTM with attention performs better as compared to others. The Figure 4.7 and Figure 4.8 shows the learning curve of Bi-LSTM with attention. Figure 4.8 shows the training and validation accuracy of attention using Bi-LSTM model.

As shown in the figure the model achieves a training and validation accuracy of 69% and 64% respectively. On the other hand, Figure 4.8 illustrates the validation and training loss. The graph indicates that both the training and validation losses steadily decrease during the training process, indicating that the models are effectively gaining knowledge from the training set. This pattern suggests that the models are improving their comprehension of the underlying patterns and their ability to adjust to the given dataset. The model attains a nearly same training and validation loss of approximately 13 percent. We employed early stopping to keep an eye on the training process and stop the model from overfitting. In order to ensure model generalization and avoid overfitting, early stopping is employed. It prevents the models from becoming overly specialized to the training set by terminating the training process when the validation loss no longer improves. We can enhance the performance of the models on new, untested data by using early stopping. Training was discontinued after the validation loss was being monitored by the training controller and showed no signs of improvement. In figure 4.9 the performance of attention mechanism using Bi-LSTM to predict the test set. As shown in the Figure 4.9 there is a small bias between the actual and predicted output of the model.

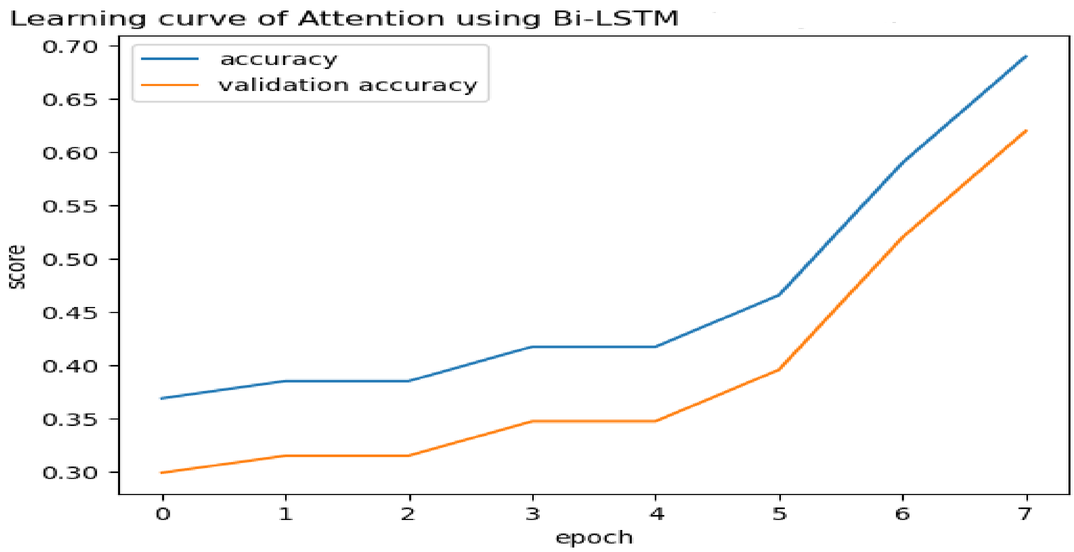

Figure 4. 7 Training and validation accuracy of attention using Bi-LSTM

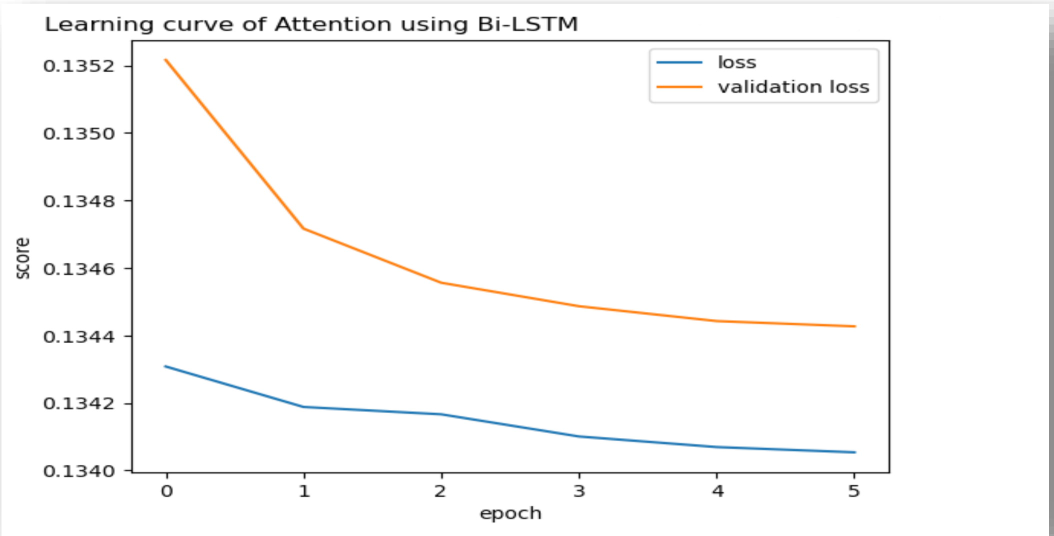

Figure 4. 8 Training and validation loss of attention using Bi-LSTM

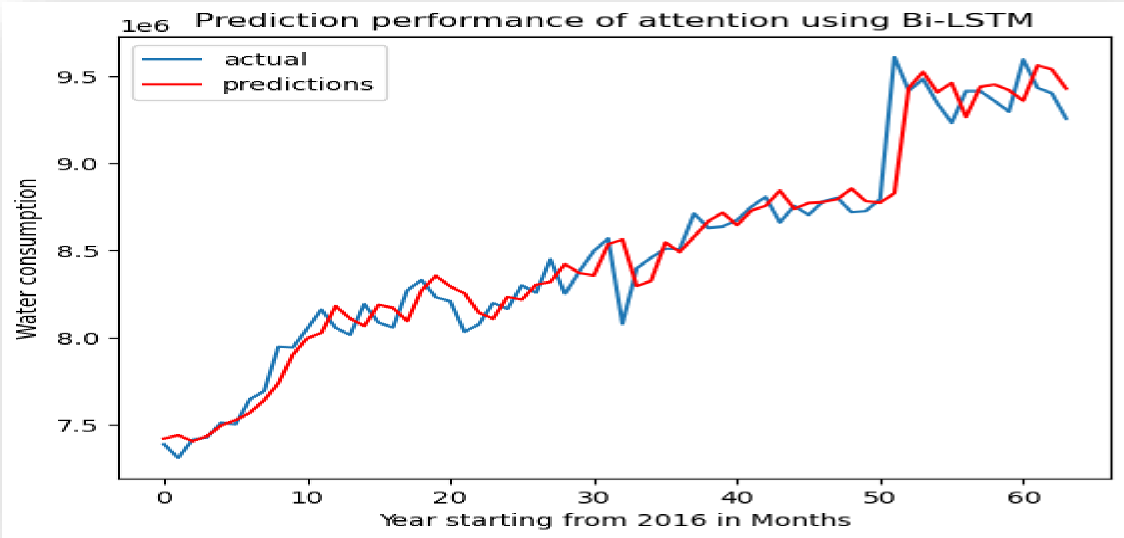

Figure 4. 9 Prediction performance of Attention using Bi-LSTM mechanism

Figures 4.10 and 4.11 illustrate the Bi-LSTM algorithm's learning curve. The Bi-LSTM model's accuracy throughout training and validation is displayed in Figure 4.10. The model achieves training and validation accuracy of 45% and 30%, respectively, as the image illustrates. The validation accuracy attempted to decrease after epoch 7, as depicted in the figure, but the training halted because of the early stopping controller. Conversely, Figure 4.12 illustrates the validation and training loss. The graph indicates that until epoch 5, both the training and validation losses consistently decrease during the training process, indicating that the models are effectively learning from the training data. Nevertheless, following epoch 5, the training controller was no longer used, and both validation and training loss continued to increase. The model attains a nearly same training and validation loss of approximately 15%.

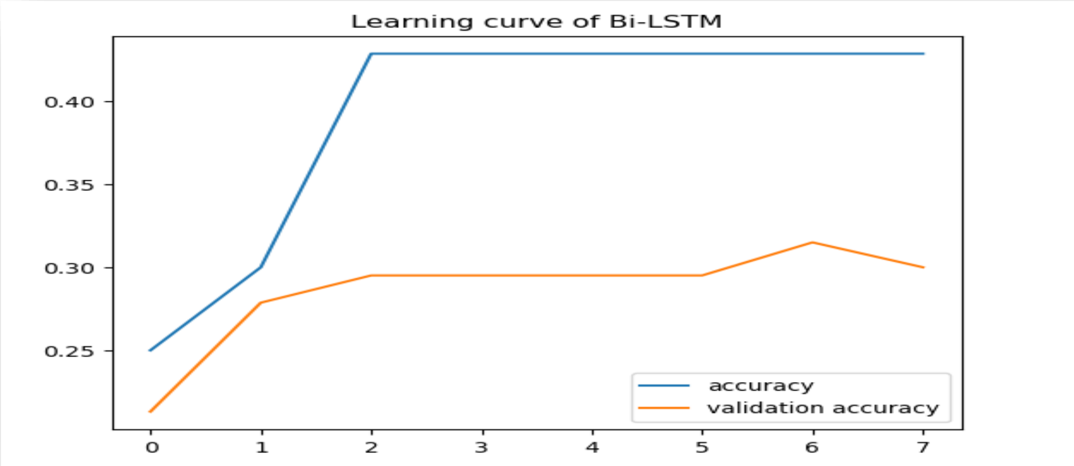

Figure 4. 10 Training and validation accuracy of Bi-LSTM

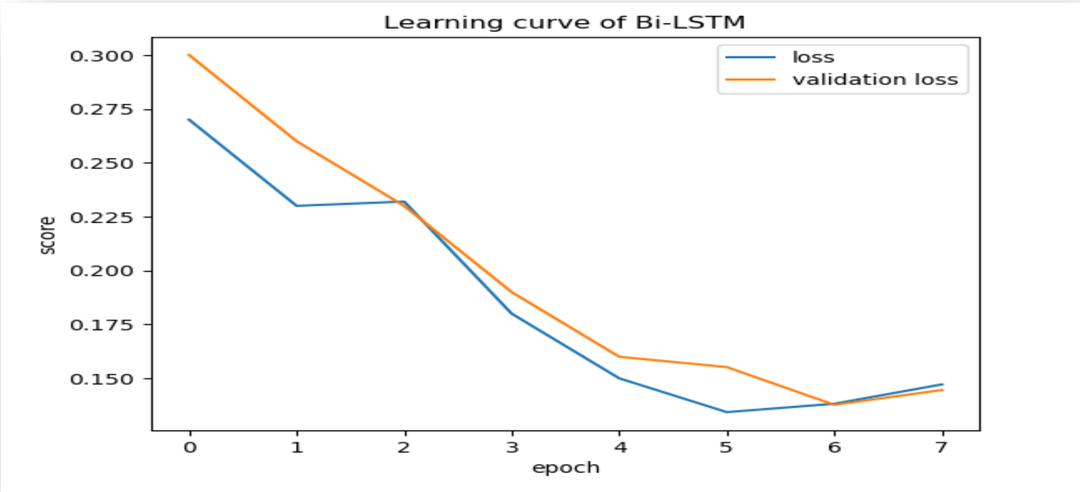

Figure 4. 11 Training and validation loss of Bi-LSTM

The GRU algorithm is used to illustrate the attention learning curve in Figures 4.12 and 4.13. The training and validation accuracy of the attention using the GRU model is displayed in Figure 4.12. The model achieves training and validation accuracy of 55% and 45%, respectively, as the figure illustrates. The graphic shows the steady growth of both training and validation accuracy. But after epoch 10, the validation loss attempted to increase, and the training using the training controller was stopped. The graph shows that training and validation losses go down gradually until epoch 8, which suggests that the models are successfully picking up new information from the training set. However, after epoch 8, the training controller was not used anymore, and the training loss kept rising along with the validation. The model achieves almost identical training and validation losses of roughly 16 percent.

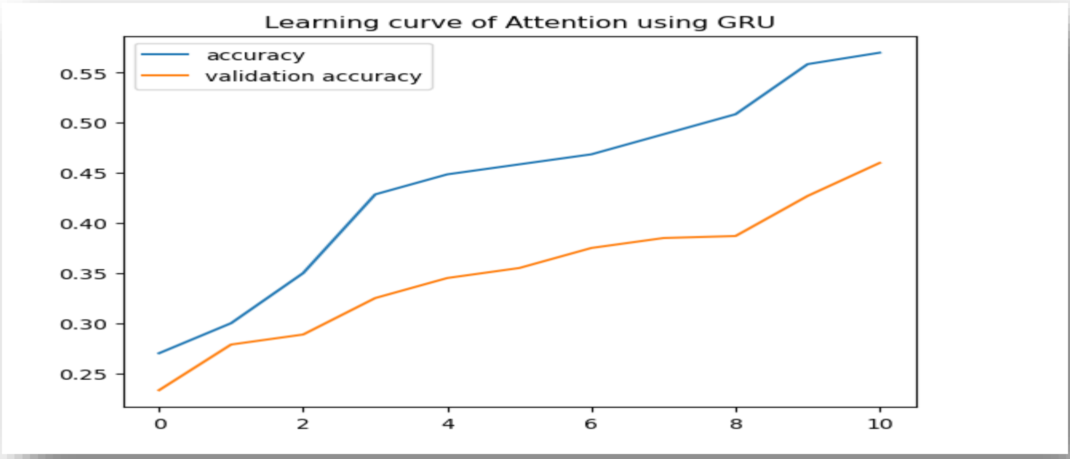

Figure 4. 12 Training and validation accuracy of attention + GRU

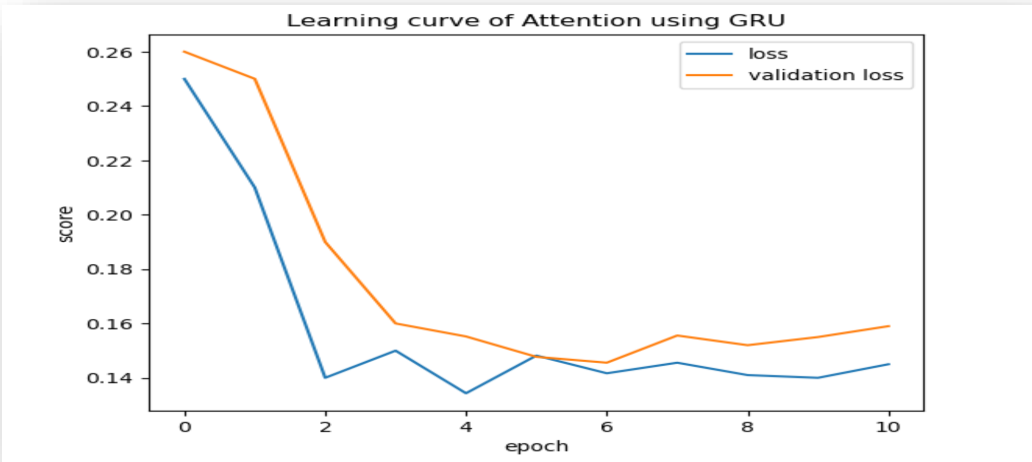

Figure 4. 13 Training and validation loss of attention + GRU

4.7. Performance Evaluation

In this phase, the effectiveness of the various models constructed in the earlier phases is assessed. Three widely used assessment metrics Mean Absolute Error, Mean Squared Error, and Root Mean Squared Error were employed to assess the performance of the model. We forecasted the testing data and compared the actual values with the anticipated values to determine each model's performance. Every model's performance is specified in Table 4.11.

Table 4. 11 Performance of deep learning models

Performance of Deep Learning models			
Model	Performance on the test set		
	RMSE	MSE	MAE
LSTM	0.13	0.016	0.32
Bi-LSTM	**0.10**	**0.01**	**0.29**
LSTM + Attention	0.13	0.016	0.18
Bi-LSTM + Attention	**0.08**	**0.0064**	**0.16**
GRU	0.14	0.0196	0.33
Bi-GRU	0.11	0.0121	0.26
GRU + Attention	**0.10**	**0.01**	**0.25**

From the analytical results shown in Table 4.11 we observe that the GRU model achieved an RMSE score of 0.14, indicating a considerably higher prediction error in comparison to the other models. This finding implies that the dataset's more complex patterns may be difficult for the GRU architecture to recognize, which could result in higher prediction errors. The LSTM model scored RMSE score of 0.13 which is slightly better than the GRU model. Even though it shows a slightly improved performance, it also showed a similar prediction error as the GRU model. The Bi-LSTM model demonstrated an enhanced performance when compared to the LSTM and GRU models, with an RMSE score of 0.10. The bi-directionality of the Bi-LSTM architecture allowed the model to record temporal relationships in both forward and backward directions, which aided in the identification of important patterns in the dataset. The GRU model with attention fared better than the LSTM, GRU, Bi-GRU, LSTM with attention, and Bi-LSTM models, with an RMSE score of 0.10. This model's capacity to concentrate on significant characteristics and interactions within the dataset was improved by the addition of attention mechanisms.

The Bi-LSTM with attention model outperformed all the other models assessed, as evidenced by its lowest RMSE score value of 0.08. The most accurate predictions were produced by this model, which successfully captured the complex patterns and dependencies found in the dataset by fusing attention mechanisms with the bidirectional design. The attention mechanism was essential in directing the model's attention onto relevant patterns within the input sequence, hence facilitating accurate forecasting. Figure 4.14 shows a bar graph for the

3 best models. From the graph we observe that the attention using Bi-LSTM algorithm scores lowest RMSE and MAE making it the best model.

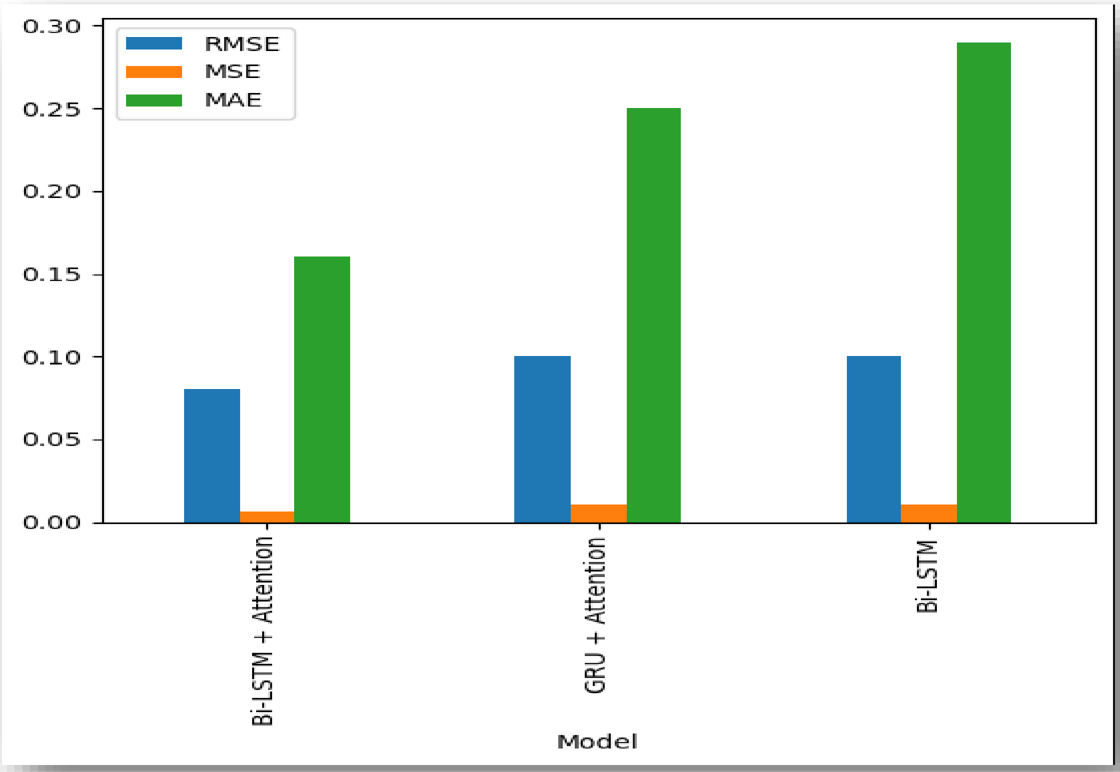

Figure 4. 14 Models with best performance

4.8. Comparison of the developed model with baseline model

In this stage we have compared the proposed model with the baseline model which is ARIMA and a model from previous study by [62]. Table 4.12 provides a comparison of the top three models of this study and baseline models. The performance of the models is assessed using Root Mean Squared Error. The study [62] used a deep learning-based long short-term memory (LSTM) approach to develop a water consumption prediction model.

Table 4. 12 Performance of different models

Model	**Performance of ARIMA**		
	RMSE	MSE	MAE
ARIMA	0.59	0.34	0.43
Performance of Machine Learning models			
Model	Performance on the test set		
	RMSE	MSE	MAE

Decision Tree	1.54	2.39	1.42
Random Forest	1.56	2.43	1.57
Multinomial Naive Bayes	1.56	2.43	1.477
Logistic Regression	**1.52**	**2.33**	**1.37**
Support vector machine (SVM)	1.54	2.39	1.45
Performance of Deep Learning models			
Model	Performance on the test set		
	RMSE	MSE	MAE
LSTM	0.13	0.016	0.32
Bi-LSTM	**0.10**	**0.01**	**0.29**
LSTM + Attention	0.13	0.016	0.18
Bi-LSTM + Attention	**0.08**	**0.0064**	**0.16**
GRU	0.14	0.0196	0.33
Bi-GRU	0.11	0.0121	0.26
GRU + Attention	**0.10**	**0.01**	**0.25**

Table 4. 13 Comparison of baseline and previous works

Evaluation metrics	Bi-LSTM	GRU + Attention	**Bi-LSTM + Attention**	ARIMA	Machine Learning (Logistic Regression)	Machine Learning (SVM) [63]	LSTM [62]
RMSE	0.10	0.10	0.08	0.207	1.52	2.590	0.19

The evaluation results, which are shown in Table 4.13, offer important new information about how well the proposed models perform in relation to the baseline and earlier studies. We may evaluate the precision and efficiency of our models in water demand prediction by looking closely at the RMSE values. The table's analysis effectively illustrates how accurate our model is in identifying and forecasting patterns in water demand. Our model outperforms not just the baseline (ARIMA) model but also the one from the previous study. This clear domination of our methodology in this field demonstrates its benefits over rival tactics and supports its efficacy. The deep learning models' outstanding performance, as shown in Table 4.13, provides strong proof of our model's accuracy in predicting water usage. As a result, we can state with confidence that our model outperforms the baseline, demonstrating its efficacy and fostering confidence in its predictive capabilities.

Generally, the studies showed that while the RMSE of 0.207 indicates that the ARIMA model outperforms machine learning models, it underperforms deep learning methods. Testing different models is crucial, and our table highlights how well the bidirectional LSTM with attention model performed it had an incredibly low RMSE of 0.08. This model performs noticeably better than the other models evaluated in the study, offering compelling proof of its efficacy in terms of water use.

CHAPTER FIVE

5. CONCLUSION AND RECOMMENDATIONS

5.1. Introduction

This chapter discusses the research findings and recommendations for future researchers interested in working on water consumption forecasting or prediction. In this study the researcher has developed a water consumption prediction model for Addis Ababa City. The researcher used a historical water consumption dataset of the city and built a model using different deep learning algorithms.

5.2. Conclusion

The sustainable use of water resources is essential to economic progress. Moreover, one of the most vital energy sources for industrialization and rising living standards is water. Water resources are employed in many different areas, including manufacturing, households, organizations, and institutions. Ethiopia is using more and more water over time. Ethiopia's total annual household water usage increased considerably between 2016 and 2017, according to the World Development Indicators Databank. The world's water demand is growing, yet the ability of water resources to generate energy is finite. Water companies must therefore look into models in order to more accurately forecast and plan for water consumption. Water shortages develop as a result of rising water usage. Consequently, one way to solve this issue is through water prediction.

Experiments were carried out on a desktop PC with an 8 GB RAM and 2.41 GHz processor to validate the suggested techniques. The operating system used on the workstations is Windows 10 Pro (64-bit). Deep learning libraries like TensorFlow, sklearn, and Keras were used to implement the suggested methods. Running on the TensorFlow framework, Keras is a high-level neural network built in Python that supports RNN (LSTM, GRU, and BILSTM), allows for quick and simple prototyping, and functions seamlessly on both CPU and GPU. The high-level Keras API is used by TensorFlow, an open source end-to-end deep learning platform, to create and train deep learning models. Sklearn is used for data preprocessing, and Keras and TensorFlow are utilized to build the model for training, validation, and testing.

The analysis of water consumption through the year of 2016 to 2022 GC is conducted and the amount of water consumption throughout each month is analyzed. The researcher investigated about water consumption considers the different behavior of people during holidays and regular days (type of the day variable) as the most effective factor on the water consumption and water demand. In addition, we included calendar information as an essential factor in analyzing water demand because holidays have a different water usage pattern than regular weekdays. On the other hand, we explored the role of external variables on water consumption. We analyzed the relationship between weather variables and climate variables like temperature and rainy season to water consumption. We found out that there were meaningful correlations between these variables and water consumption. Thus, water consumption in AAWSA is increased from year to year with an increase in the number of customers. The usage of water also varies from month to moth throughout the year. From 2016 to 2022 GC, there is high water consumption in September and January and less water in July and August when it compared to other months. Though, AAWSA need to manage the different reasons that makes the water consumption of customers in each month and those months that have high water consumption and less water consumption needs special attention to avoid the shortage of water or the manage continuity of water usage.

In this study, the researcher developed a water consumption prediction model for Addis Ababa City using the historical water consumption dataset of the city. The study used 7 years of water consumption dataset. The dataset contains around 61 attributes and the values of the data were both string and numeric data. In this study the researcher used the date in month and total water consumption in meter cube attributes to train the prediction model. The dataset is preprocessed using different preprocessing techniques discussed in chapter 3. Using the dataset we have experimented different deep learning algorithms of LSTM, GRU, Bi-LSTM and Attention mechanism. During model training we used the MSE, MAE and RMSE loss functions.

According to the findings of some earlier research, the prediction efficiency dropped when more historical data was included. Two models were used in the research by Herrera et al. [64]. In one of these tests, they took into account the last eight weeks' worth of data, whereas in the other, all of the data was used as input into the model. The outcome demonstrated that employing too old data can negatively impact predicting models and that utilizing less data worsened performance. The significance of the input data's length was also investigated by Bejarano et al. [65]. They found that using a day's worth of historical data was adequate to identify trends in water usage, and adding more data did not enhance the model's capacity for prediction. Using the historical dataset from the study, we attempted to identify the time points that coincidentally had a recorded water consumption amount in both eras in order to determine which kinds of algorithms perform best and most efficiently when applying which historical data (distant or near prediction). Using the dataset, we have experimented different deep learning algorithms of LSTM, GRU, Bi-GRU, Bi-LSTM and Attention mechanism. During model training we used the MSE, MAE and RMSE loss functions. Different hyperparameter were compared with the baseline hyperparameter. Based on the experiments done, Bi-LSTM scores lowest loss value of prediction RMSE, MSE and MAE with values of 0.10, 0.01 and 0.29 respectively. GRU with Attention scores lowest loss value of prediction RMSE, MSE and MAE with values of 0.10, 0.01 and 0.25 respectively. Bidirectional LSTM with Attention mechanism scores lowest loss value of prediction RMSE, MSE and MAE with values of 0.08, 0.0064 and 0.16 respectively. Thus, among the models tested, the bidirectional LSTM with attention model is the top performer. This implies that the model using the Attention mechanism performance better as compared to others.

We conclude that, when it comes to water consumption prediction, the bidirectional LSTM with attention model outperforms the other models. However, the other models show comparable performance, providing fair replacements for this work. Our findings verify that deep learning methods are effective in accurately forecasting water usage, using a gathered dataset. This study demonstrates the deep learning models' ability to capture the intricacy of water usage patterns and provides a more thorough knowledge of the benefits of each model.

Nonetheless, given the quality and size of the dataset utilized, along with the deep learning tools and techniques which are crucial for the modeling performance an expanded dataset encompassing even all of Ethiopia's towns and an increase in the quantity and diversity of attributes could have resulted in an improved modeling. Consequently, the research might have been able to address other areas of the nation's problems by utilizing larger data sets with more attributes than those employed in this study. The researcher believes that if additional methodologies were applied and if other town's datasets in Ethiopia with large data sizes and a variety of product users were included in the model building experiment, which was based on AWASA data acquired, the model would have been enhanced.

5.3. Contribution

After the end of this study, the researcher has contributed the following things for other researchers and anyone who is interested on water consumption prediction models.

- We have collected previous 7 years of water consumption dataset.

- The researcher showed that a Deep learning approach performs best as compared to Persistence model, ARIMA model and machine learning models for water demand prediction.
- The researcher showed that hyper parameter tuning of deep learning models improves their performance.
- The researcher showed that Attention mechanism outperforms other approaches for water consumption prediction in the experiments.
- The researcher has built a water consumption prediction model for Addis Ababa City.

5.4. Recommendations

The Addis Ababa Water and Sewerage Authority and other academics with an interest in a related field will benefit greatly from the project, even though its primary goal is academic. While the study's findings are encouraging, more research is necessary in certain areas to improve the inclusive model and take it to a practical level in subsequent efforts. In light of this, the researcher suggests the following topics for further investigation based on this work.

The researcher would like to propose the following proposals for future research directions in energy prediction strategies based on the study's findings.

- This model was trained using a small dataset; in order to create better deep learning models, future research should concentrate on expanding the study using a larger and more varied dataset from different domains.
- The researcher uses client-related records from the dataset for this investigation, including water use in cubic meters and months and years. Future researchers ought to build forecasting models that consider tariff classifications, components (Groups), customer support centers (Werdas), population, temperature, and pressure.
- To increase prediction accuracy, considering adding extra input characteristics like weather data, holiday schedules, or economic indicators external factors that can affect water consumption.
- Prediction results may be enhanced in the future by investigating alternative strategies to enhance the model's performance, such as CNN-LSTM, LSTM-RNN, and CNN-Bidirectional LSTM. This could facilitate better research outcomes.

References

[1] Daniel Truneh, Nebiyu Getachew, "Improved access to water and sanitation service," Addis Ababa Water and Sewage Authority (AAWSA), Ethiopia, 2004.

[2] UNCHS, "an Overview of Urban Poverty in Addis Ababa.Capacity building for sustainable urban development in Ethiopia," Addis Ababa, Ethiopia, 2000.

[3] W. Lu, J. Li, Y. Li, A. Sun, and J. Wang, “A CNN-LSTM-Based Model to Forecast Stock Prices,” vol. 2020, 2020.

[4] Alvisi, S.; Franchini, M.; Marinelli, A. A short-term, pattern-based model for water-demand forecasting. J. Hydroinform. 2007, 9, 39–50.

[5] “NMWE – Natural Mineral & Spring Waters.” https://naturalmineralwaterseurope.org/ (accessed Jun. 24, 2021).

[6] R. F. Berriel, T. Lopes, A. Rodrigues, and T. Oliveira-Santos, “Monthly Energy Consumption Forecast : A Deep Learning Approach,” 2017.

[7] Gagliardi, F.; Alvisi, S.; Franchini, M.; Guidorzi, M. A comparison between pattern-based and neural network short-term water demand forecasting models. Water Sci. Technol. Water Supply 2017, 17, 1426–1435.

[8] Boudhaouia, A.; Wira, P. A Real-Time Data Analysis Platform for Short-Term Water Consumption Forecasting with Machine Learning. Forecasting 2021, 3, 682–694.

[9] Salloom, T.; Kaynak, O.; He, W. A novel deep neural network architecture for real-time water demand forecasting. J. Hydrol. 2021, 599, 126353.

[10] V. Bianco, O. Manca, and S. Nardini, “Electricity consumption forecasting in Italy using linear regression models,” *Energy*, vol. 34, no. 9, pp. 1413–1421, 2009, doi: 10.1016/j.energy.2009.06.034

[11] “DEPARTMENT OF COMPUTER SCIENCE St . Mary s University School of Graduate Studies Faculty of Informatics Department of Computer Science,” 2019.

[12] A. J. Real, F. Dorado, and J. Dur, “Energy Demand Forecasting Using Deep Learning : Applications for the French Grid,” 2020.

[13] Ethiopia Ministry of Transport 2011 Transport Policy of Addis Ababa. Available from:

https://chilot.me/wp-content/ uploads/2011/08/ (accessed 19/6/2018).

[14] World Bank 2015a Enhancing Urban Resilience in Addis Ababa: Resilient Cities Program; Document Number 100980. World Bank Group, Addis Ababa, Ethiopia.

[15] World Bank 2015b Enhancing Urban Resilience. GFDRR, Addis Ababa.

[16] World Bank 2016c Ethiopia Urbanization Review: Urban Institutions for a Middle-Income Ethiopia. Available from: http://www.cs.waikato.ac.nz/ml/weka/.

[17] D. Kemp, C. J. Bond, D. M. Franks, and C. Cote, "Mining, water and human rights: making the connection," Journal of Cleaner Production, vol. 18, no. 15, pp. 1553–1562, 2010.

[18] J. Chenoweth, "Changing ownership structures in the water supply and sanitation sector," Water International, vol. 29, no. 2, pp. 138–147, 2004.

[19] L. V. Ryan and R. J. Hunter, "Multicultural and colorblind ideology, stereotypes, and ethnocentrism among black and white Americans," European Journal of Social Sciences, vol. 5, no. 2, 2007.

[20] B. Mosello, R. Calow, J. Tucker, T. Alamirew, S. Kebede, and T. Alemseged, Building adaptive water resources management in Ethiopia, Overseas Development Institute, London, 2015.

[21] Usaid, Ethiopia water and sanitation profile, pp. 1–4, 2015.

[22] T. Finance, "Water supply and sanitation in Ethiopia," An AMCOW Ctry. Status Overv, 2015.

[23] H. V. P. Singh and Q. H. Mahmoud, "Evaluation of ARIMA Models for Human–Machine Interface State Sequence Prediction," *Mach. Learn. Knowl. Extr.*, vol. 1, no. 1, pp. 287–311, 2019, doi: 10.3390/make1010018.

[23] C. Team, "Autoregressive Integrated Moving Average (ARIMA)." [Online]. Available: https://corporatefinanceinstitute.com/resources/data-science/autoregressive-integrated-moving-average-arima/

[24] S. E. Bibri, "The IoT for smart sustainable cities of the future: An analytical framework for sensor-based big data applications for environmental sustainability," Sustainable Cities and Society, vol. 38, pp. 230–253, Apr. 2018, doi:

10.1016/j.scs.2017.12.034.

[25] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785. 26

[26] E. Fix and J. L. Hodges, “Discriminatory Analysis, Nonparametric Discrimination: Consistency Properties.,” Technical Report 4, USAF School of Aviation Medicine, Randolph Field, 1951, doi: 10.2307/1403797.

[27] T. Cover and P. Hart, “Nearest neighbor pattern classification,” IEEE Transactions on Information Theory, vol. 13, no. 1, pp. 21–27, Jan. 1967, doi: 10.1109/TIT.1967.1053964.

[28] S. Raschka and V. Mirjalili, Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow, 2nd Edition, 2nd ed. Packt Publishing, 2017.

[29] “Understanding LSTM Networks -- colah’s blog.” http://colah.github.io/posts/2015-08-Understanding-LSTMs/ (accessed Jun. 24, 2021).

[30] J. Kim, H. Lee, M. Lee, H. Han, D. Kim, and H. S. Kim, “Development of a Deep Learning-Based Prediction Model for Water Consumption at the Household Level,” *Water (Switzerland)*, vol. 14, no. 9, 2022, doi: 10.3390/w14091512.

[31] P. Hu, J. Tong, J. Wang, Y. Yang, and L. De Oliveira Turci, “A hybrid model based on CNN and Bi-LSTM for urban water demand prediction,” *2019 IEEE Congr. Evol. Comput. CEC 2019 - Proc.*, no. 2017, pp. 1088–1094, 2019, doi: 10.1109/CEC.2019.8790060.

[32] B. C. Mateus, M. Mendes, J. T. Farinha, R. Assis, and A. M. Cardoso, “Comparing LSTM and GRU models to predict the condition of a pulp paper press,” *Energies*, vol. 14, no. 21, pp. 1–21, 2021, doi: 10.3390/en14216958.

[33] Q. Kang, E. J. Chen, Z. C. Li, H. Bin Luo, and Y. Liu, “Attention-based LSTM predictive model for the attitude and position of shield machine in tunneling,” *Undergr. Sp.*, vol. 13, pp. 335–350, 2023, doi: 10.1016/j.undsp.2023.05.006.

[34] P. Hu, J. Tong, J. Wang, Y. Yang, and L. De Oliveira Turci, “A hybrid model based on CNN and Bi-LSTM for urban water demand prediction,” *2019 IEEE Congr. Evol.*

Comput. CEC 2019 - Proc., no. 2017, pp. 1088–1094, 2019, doi: 10.1109/CEC.2019.8790060.

[35] B. C. Mateus, M. Mendes, J. T. Farinha, R. Assis, and A. M. Cardoso, "Comparing LSTM and GRU models to predict the condition of a pulp paper press," *Energies*, vol. 14, no. 21, pp. 1–21, 2021, doi: 10.3390/en14216958.

[36] D. Bahdanau, K. H. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–15, 2015.

[37] J. E. Froehlich, E. Larson, T. Campbell, C. Haggerty, J. Fogarty, and S. N. Patel, "HydroSense: infrastructure-mediated single-point sensing of whole-home water activity," in Proceedings of the 11th international conference on Ubiquitous computing, New York, NY, USA, Sep. 2009, pp. 235–244. doi: 10.1145/1620545.1620581.

[38] J. A. B. Somontina, F. Carlo C. Garcia, and E. Q. B. Macabebe, "Water Consump-tion Monitoring with Fixture Recognition Using Random Forest," in TENCON 2018 - 2018 IEEE Region 10 Conference, Oct. 2018, pp. 0663–0667. doi: 10.1109/TENCON.2018.8650112.

[39] S. Seyedzadeh, F. P. Rahimian, I. Glesk, and M. Roper, "Machine learning for estimation of building energy consumption and performance: a review," Visualiza-tion in Engineering, vol. 6, no. 1, p. 5, Oct. 2018, doi: 10.1186/s40327-018-0064-7.

[40] M. Sornam and M. Meharunnisa, "Role of Data Mining Techniques in Building Smarter and Greener Environment - A Study," in 2018 International Conference on Computer, Communication, and Signal Processing (ICCCSP), Feb. 2018, pp. 1–5. doi: 10.1109/ICCCSP.2018.8452839.

[41] M. Rodríguez Fernández, A. Cortés García, I. González Alonso, and E. Zalama Casanova, "Using the Big Data generated by the Smart Home to improve energy efficiency management," Energy Efficiency, vol. 9, no. 1, pp. 249–260, Jan. 2016, doi: 10.1007/s12053-015-9361-3.

[42] T. Vafeiadis et al., "Machine Learning Based Occupancy Detection via the Use of Smart Meters," in 2017 International Symposium on Computer Science and Intelligent Controls (ISCSIC), Oct. 2017, pp. 6–12. doi: 10.1109/ISCSIC.2017.15.

[43] Q. Li, Q. Meng, J. Cai, H. Yoshino, and A. Mochida, “Applying support vector machine to predict hourly cooling load in the building,” Applied Energy, vol. 86, no. 10, pp. 2249–2256, Oct. 2009, doi: 10.1016/j.apenergy.2008.11.035.

[44] Q. Li, Q. Meng, J. Cai, H. Yoshino, and A. Mochida, “Applying support vector machine to predict hourly cooling load in the building,” Applied Energy, vol. 86, no. 10, pp. 2249–2256, Oct. 2009, doi: 10.1016/j.apenergy.2008.11.035.

[45] H. X. Zhao and F. Magoulès, “Parallel Support Vector Machines Applied to the Prediction of Multiple Buildings Energy Consumption,” Journal of Algorithms & Computational Technology, vol. 4, no. 2, pp. 231–249, Jun. 2010, doi: 10.1260/1748-3018.4.2.231.

[46] H. Zhao and F. Magoulès, “A review on the prediction of building energy consumption,” Renewable and Sustainable Energy Reviews, vol. 16, no. 6, pp. 3586–3592, Aug. 2012, doi: 10.1016/j.rser.2012.02.049.

[47] Y. Zhang and Q. Li, “A regressive convolution neural network and support vector regression model for electricity consumption forecasting,” Lect. Notes Networks Syst., vol. 70, pp. 33–45, 2020, doi: 10.1007/978-3-030-12385-7_4.

[48] G. Shen, Q. Tan, H. Zhang, P. Zeng, and J. Xu, “ScienceDirect ScienceDirect Deep Learning with Gated Recurrent Unit Networks for Financial Deep Learning with Gated Recurrent Unit Networks for Financial Sequence Predictions Sequence Predictions,” Procedia Comput. Sci., vol. 131, pp. 895–903, 2018, doi: 10.1016/j.procs.2018.04.298.

[49] S. A. S. A.- Farttoosi and B. Mansouri, “Predicting Electricity Consumption in Misan Province of Iraq Using Univariate Time Series Analysis.”

[50] R. Banik, P. Das, S. Ray, and A. Biswas, “Prediction of electrical energy consumption based on machine learning technique,” Electr. Eng., vol. 103, no. 2, pp. 909–920, 2021, doi: 10.1007/s00202-020-01126-z.

[51] A. Mosavi and A. Bahmani, “Energy consumption prediction using machine learning; a review,” Energies, no. March, pp. 1–63, 2019, doi: 10.20944/preprints201903.0131.v1.

[52] W. E. A. A. H. A. S. Al-Obeidi, "Trend Analysis and Prediction on Water Consumption

in," *Hindawi*, vol. I, p. 7, 2022.

[53] El Khantach, A., Hamlich, M., Belbounaguia, N.E., 2019. Short-term load forecastingusing machine learning and periodicity decomposition. AIMS Energy 7 (3), 382–394.

[54] Abdelkarim El khantach, Mohamed Hamlich, Nour eddine Belbounaguia. Short-term load forecasting using machine learning and periodicity decomposition[J]. AIMS Energy, 2019, 7(3): 382-394. doi:

[55] SON, Hyojoo; KIM, Changwan. A deep learning approach to forecasting monthly demand for residential–sector electricity. Sustainability, 2020.

[56] Wenjie Lu, Jiazheng Li, Yifan Li, Aijun Sun, Jingyang Wang, "A CNN-LSTM-Based Model to Forecast Stock Prices", Complexity, vol. 2020, Article ID 6622927, 10 pages, 2020. https://doi.org/10.1155/2020/6622927 [doi.org]

[57] Mpawenimana, A. Pegatoquet, V. Roy, L. Rodriguez and C. Belleudy, "A comparative study of LSTM and ARIMA for energy load prediction with enhanced data preprocessing," 2020 IEEE Sensors Applications Symposium (SAS), Kuala Lumpur, Malaysia, 2020, pp. 1-6, doi: 10.1109/SAS48726.2020.9220021.

[58] C. J. Willmott and K. Matsuura, "Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance," Climate Research, vol. 30, no. 1, pp. 79–82, 2005.

[59] Rumelhart D, Hinton G, Williams R. Long short-term memory. Nature 1986;323:533e6.

[60] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9:1735e80.

[61] Cho K, Merrienboer BV, Bahdanau D, et al. On the properties of neural machine translation:

[62] J. Kim, H. Lee, M. Lee, H. Han, D. Kim, and H. S. Kim, "Development of a Deep Learning-Based Prediction Model for Water Consumption at the Household Level," *Water (Switzerland)*, vol. 14, no. 9, 2022, doi: 10.3390/w14091512.

[63] T. Ibrahim, Y. Omar, and F. A. Maghraby, “Water Demand Forecasting Using Machine Learning and Time Series Algorithms,” *2020 Int. Conf. Emerg. Smart Comput.*

[64] M. Herrera, L. Torgo, J. Izquierdo, and R. Pérez-García, “Predictive models for forecasting hourly urban water demand,” Journal of Hydrology, vol. 387, no. 1, pp. 141–150, Jun. 2010, doi: 10.1016/j.jhydrol.2010.04.005.

[65] G. Bejarano, A. Kulkarni, R. Raushan, A. Seetharam, and A. Ramesh, “SWaP: Probabilistic Graphical and Deep Learning Models for Water Consumption Prediction,” in Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, New York, NY, USA, Nov. 2019, pp. 233–242. doi: 10.1145/3360322.3360846.

Appendixes

Appendix A: Row dataset collected from AAWSA

CUST_KEY	INST_KEY	CONTRACT_NUMBE	PERIOD	PREV_RDG	CURR_RDG	CURR_CONS	TOT_CONS
7143480	7143460	706491	Mar-22	466	467	1	1
C-0000004	IN-000000416	933680	Mar-22	497	501	4	4
C-0000004	IN-000000420	932894	Mar-22	624	636	12	12
C-0000000	IN-000000090	922206	Mar-22	2504	2543	39	39
C-0000014	IN-000001456	983394	Mar-22	451	463	12	12
C-0000005	IN-000000553	938440	Mar-22	398	401	3	3
7124958	7124938	712505	Mar-22	1229	1244	15	15
7115503	7115483	766188	Mar-22	150	174	24	24
C-0000009	IN-000000957	955833	Mar-22	414	419	5	5
C-0000003	IN-000000385	931485	Mar-22	259	261	2	2
7127838	7127818	161558	Mar-22	5981	6003	22	22
7440630	7440610	930246	Mar-22	991	1017	26	26
7131977	7131957	121867	Mar-22	6384	6406	22	22
C-0000003	IN-000000330	932134	Mar-22	180	180	0	0
7095764	7095744	773604	Mar-22	1790	1797	7	7
C-0000004	IN-000000455	932828	Mar-22	551	557	6	6
C-0000005	IN-000000506	935549	Mar-22	404	413	9	9
7097499	7097479	168045	Mar-22	520	540	20	20
7102911	7102891	177070	Mar-22	1566	1569	3	3
7096478	7096458	770364	Mar-22	918	934	16	16
7144046	7144026	183207	Mar-22	1410	1417	7	7
7099741	7099721	172372	Mar-22	1148	1155	7	7
7143333	7143313	709531	Mar-22	1182	1189	7	7
7143843	7143823	187995	Mar-22	703	712	9	9
7103079	7103059	172408	Mar-22	3948	3962	14	14

Appendix B: Processed dataset

Out[19]:

	PREV_RDG	CURR_RDG	CURR_CONS	TOT_CONS	WATER_CONS	DUE_DATE
0	6024	6031	7.0	7.0	16.80	01-SEP-22
1	3380	3404	24.0	24.0	123.59	01-SEP-22
2	1924	1929	5.0	5.0	12.00	01-SEP-22
3	822	822	NaN	NaN	NaN	01-SEP-22
4	3935	3950	15.0	15.0	60.50	01-SEP-22
5	1076	1090	14.0	14.0	55.65	01-SEP-22
6	1023	1025	2.0	2.0	4.80	01-SEP-22
7	1238	1242	4.0	4.0	9.60	01-SEP-22
8	1954	1999	45.0	45.0	655.65	01-SEP-22
9	551	553	2.0	2.0	4.80	01-SEP-22

Appendix C Hyper Parameter Tunning of the epoch

C:\Users\Negatu\AppData\Local\Temp\ipykernel_6208\453909407.py:118: FutureWarning: The squeeze argument has been deprecated and is removed in a future version. Append .squeeze("columns") to the call to squeeze.

series = read_csv('dataset.csv', header=0, parse_dates=[0], index_col=0, squeeze=True, date_parser=parser)

16/16 [==============================] - 1s 4ms/step
3/3 [==============================] - 0s 8ms/step
1) Test RMSE: 312928.308
1) Test MSE: 97924126151.608
1) Test MAE: 204736.866
16/16 [==============================] - 1s 2ms/step
3/3 [==============================] - 0s 10ms/step
2) Test RMSE: 305257.234
2) Test MSE: 93181978681.125
2) Test MAE: 199372.339
16/16 [==============================] - 1s 3ms/step
3/3 [==============================] - 0s 9ms/step
3) Test RMSE: 309517.257
3) Test MSE: 95800932094.883
3) Test MAE: 202202.718
16/16 [==============================] - 1s 4ms/step
3/3 [==============================] - 0s 10ms/step
4) Test RMSE: 302873.751
4) Test MSE: 91732508784.415
4) Test MAE: 197282.425
16/16 [==============================] - 1s 5ms/step
3/3 [==============================] - 0s 13ms/step
5) Test RMSE: 313875.283
5) Test MSE: 98517693515.110
5) Test MAE: 205262.390
16/16 [==============================] - 1s 3ms/step
3/3 [==============================] - 0s 5ms/step

1) Test RMSE: 220551.045
1) Test MSE: 48642763417.653
1) Test MAE: 158508.056
16/16 [==============================] - 1s 3ms/step
3/3 [==============================] - 0s 5ms/step
2) Test RMSE: 220620.767
2) Test MSE: 48673522961.040
2) Test MAE: 158528.260
16/16 [==============================] - 1s 4ms/step
3/3 [==============================] - 0s 11ms/step
3) Test RMSE: 220569.432
3) Test MSE: 48650874370.523
3) Test MAE: 158518.656
16/16 [==============================] - 1s 6ms/step
3/3 [==============================] - 0s 7ms/step
4) Test RMSE: 220507.971
4) Test MSE: 48623765392.076
4) Test MAE: 158497.021
16/16 [==============================] - 1s 6ms/step
3/3 [==============================] - 0s 6ms/step
5) Test RMSE: 220548.447
5) Test MSE: 48641617330.320
5) Test MAE: 158503.963
16/16 [==============================] - 1s 4ms/step
3/3 [==============================] - 0s 8ms/step
1) Test RMSE: 221567.264
1) Test MSE: 49092052373.853
1) Test MAE: 160135.405
16/16 [==============================] - 1s 6ms/step
3/3 [==============================] - 0s 4ms/step
2) Test RMSE: 221518.132
2) Test MSE: 49070282961.413
2) Test MAE: 160081.136
16/16 [==============================] - 1s 3ms/step
3/3 [==============================] - 0s 4ms/step
3) Test RMSE: 221466.288
3) Test MSE: 49047316523.876
3) Test MAE: 160094.746
16/16 [==============================] - 1s 3ms/step
3/3 [==============================] - 0s 5ms/step
4) Test RMSE: 221620.466
4) Test MSE: 49115630886.083
4) Test MAE: 160139.975
16/16 [==============================] - 4s 3ms/step
3/3 [==============================] - 0s 4ms/step
5) Test RMSE: 221536.243

5) Test MSE: 49078306802.641
5) Test MAE: 160142.077
16/16 [==============================] - 12s 6ms/step
3/3 [==============================] - 0s 8ms/step
1) Test RMSE: 218211.402
1) Test MSE: 47616216082.946
1) Test MAE: 158955.094
16/16 [==============================] - 5s 7ms/step
3/3 [==============================] - 0s 0s/step
2) Test RMSE: 219271.262
2) Test MSE: 48079886404.080
2) Test MAE: 159447.634
16/16 [==============================] - 3s 2ms/step
3/3 [==============================] - 0s 8ms/step
3) Test RMSE: 218722.865
3) Test MSE: 47839691484.684
3) Test MAE: 159184.964
16/16 [==============================] - 2s 5ms/step
3/3 [==============================] - 0s 8ms/step
4) Test RMSE: 219023.035
4) Test MSE: 47971090000.435
4) Test MAE: 159343.742
16/16 [==============================] - 9s 14ms/step
3/3 [==============================] - 0s 31ms/step
5) Test RMSE: 219010.347
5) Test MSE: 47965532305.959
5) Test MAE: 159371.871
16/16 [==============================] - 2s 5ms/step
3/3 [==============================] - 0s 0s/step
1) Test RMSE: 307277.049
1) Test MSE: 94419184848.059
1) Test MAE: 254005.154
16/16 [==============================] - 3s 3ms/step
3/3 [==============================] - 0s 0s/step
2) Test RMSE: 322424.835
2) Test MSE: 103957774268.900
2) Test MAE: 268605.541
16/16 [==============================] - 2s 6ms/step
3/3 [==============================] - 0s 0s/step
3) Test RMSE: 321829.129
3) Test MSE: 103573988015.857
3) Test MAE: 268412.850
16/16 [==============================] - 24s 26ms/step
3/3 [==============================] - 0s 0s/step
4) Test RMSE: 324145.753
4) Test MSE: 105070469175.127

4) Test MAE: 270927.304

16/16 [==============================] - 1s 3ms/step

3/3 [==============================] - 0s 16ms/step

5) Test RMSE: 324029.852

5) Test MSE: 104995345266.935

5) Test MAE: 271024.335

16/16 [==============================] - 2s 5ms/step

3/3 [==============================] - 0s 8ms/step

1) Test RMSE: 289018.057

1) Test MSE: 83531437210.010

1) Test MAE: 248278.544

16/16 [==============================] - 3s 5ms/step

3/3 [==============================] - 0s 8ms/step

2) Test RMSE: 293668.397

2) Test MSE: 86241127472.232

2) Test MAE: 251022.557

16/16 [==============================] - 2s 3ms/step

3/3 [==============================] - 0s 8ms/step

3) Test RMSE: 272143.141

3) Test MSE: 74061889415.099

3) Test MAE: 229152.204

16/16 [==============================] - 1s 3ms/step

3/3 [==============================] - 0s 0s/step

4) Test RMSE: 290352.031

4) Test MSE: 84304301975.049

4) Test MAE: 248122.431

16/16 [==============================] - 3s 4ms/step

3/3 [==============================] - 0s 0s/step

5) Test RMSE: 290886.135

5) Test MSE: 84614743730.121

5) Test MAE: 244506.741

```
                  1             10             30             60  \
count       5.000000       5.000000       5.000000       5.000000
mean   308890.366487  220559.532449  221541.678435  319941.323618
std      4768.298525      40.930419      57.292708    7150.328354
min    302873.750570  220507.971267  221466.287556  307277.049010
25%    305257.233626  220548.446674  221518.132354  321829.128601
50%    309517.256538  220551.044925  221536.242639  322424.835068
75%    312928.308326  220569.432086  221567.263768  324029.852432
max    313875.283377  220620.767293  221620.465856  324145.752980

                120            150
count       5.000000       5.000000
mean   218847.782359  287213.552377
std       405.296255    8593.259837
min    218211.402275  272143.141407
```

25%   218722.864568   289018.056893
50%   219010.347486   290352.031119
75%   219023.035319   290886.135335
max   221620.465856 293668.397129
<Figure size 640x480 with 0 Axes>

Appendix D Training History

Epoch 1/30
1/1 [==============================] - 9s 9s/step - loss: 0.1182 - accuracy: 0.2444 - val_loss: 0.2498 - val_accuracy: 0.2686
Epoch 2/30
1/1 [==============================] - 0s 96ms/step - loss: 0.1181 - accuracy: 0.2889 - val_loss: 0.2505 - val_accuracy: 0.2689
Epoch 3/30
1/1 [==============================] - 0s 87ms/step - loss: 0.1181 - accuracy: 0.2889 - val_loss: 0.2508 - val_accuracy: 0.2576
Epoch 4/30
1/1 [==============================] - 0s 80ms/step - loss: 0.1181 - accuracy: 0.2889 - val_loss: 0.2507 - val_accuracy: 0.2756
Epoch 5/30
1/1 [==============================] - 0s 83ms/step - loss: 0.1181 - accuracy: 0.2889 - val_loss: 0.2505 - val_accuracy: 0.2586
Epoch 6/30
1/1 [==============================] - 0s 84ms/step - loss: 0.1181 - accuracy: 0.2889 - val_loss: 0.2503 - val_accuracy: 0.2586
Epoch 7/30
1/1 [==============================] - 0s 89ms/step - loss: 0.1181 - accuracy: 0.2889 - val_loss: 0.2500 - val_accuracy: 0.2586
Epoch 8/30
1/1 [==============================] - 0s 85ms/step - loss: 0.1181 - accuracy: 0.2889 - val_loss: 0.2499 - val_accuracy: 0.2597
Epoch 9/30
1/1 [==============================] - 0s 88ms/step - loss: 0.1181 - accuracy: 0.2889 - val_loss: 0.2498 - val_accuracy: 0.2500
Epoch 10/30
1/1 [==============================] - 0s 75ms/step - loss: 0.1181 - accuracy: 0.2889 - val_loss: 0.2498 - val_accuracy: 0.2523
Epoch 11/30
1/1 [==============================] - 0s 84ms/step - loss: 0.1181 - accuracy: 0.2889 - val_loss: 0.2498 - val_accuracy: 0.2523
Epoch 12/30

1/1 [==============================] - 0s 84ms/step - loss: 0.1181 - accuracy: 0.2889 - val_loss: 0.2499 - val_accuracy: 0.2523
Epoch 13/30
1/1 [==============================] - 0s 94ms/step - loss: 0.1181 - accuracy: 0.2889 - val_loss: 0.2501 - val_accuracy: 0.2523
Epoch 14/30
1/1 [==============================] - 0s 85ms/step - loss: 0.1180 - accuracy: 0.2889 - val_loss: 0.2502 - val_accuracy: 0.2523
Epoch 15/30
1/1 [==============================] - 0s 81ms/step - loss: 0.1180 - accuracy: 0.2889 - val_loss: 0.2504 - val_accuracy: 0.2503
Epoch 16/30
1/1 [==============================] - 0s 85ms/step - loss: 0.1180 - accuracy: 0.2889 - val_loss: 0.2504 - val_accuracy: 0.2523
Epoch 17/30
1/1 [==============================] - 0s 80ms/step - loss: 0.1180 - accuracy: 0.2889 - val_loss: 0.2504 - val_accuracy: 0.2523
Epoch 18/30
1/1 [==============================] - 0s 80ms/step - loss: 0.1180 - accuracy: 0.2889 - val_loss: 0.2504 - val_accuracy: 0.2593
Epoch 19/30
1/1 [==============================] - 0s 81ms/step - loss: 0.1180 - accuracy: 0.2889 - val_loss: 0.2503 - val_accuracy: 0.2598
Epoch 20/30
1/1 [==============================] - 0s 80ms/step - loss: 0.1180 - accuracy: 0.2889 - val_loss: 0.2502 - val_accuracy: 0.2593
Epoch 21/30
1/1 [==============================] - 0s 79ms/step - loss: 0.1180 - accuracy: 0.2889 - val_loss: 0.2502 - val_accuracy: 0.2593
Epoch 22/30
1/1 [==============================] - 0s 77ms/step - loss: 0.1179 - accuracy: 0.2889 - val_loss: 0.2501 - val_accuracy: 0.2593
Epoch 23/30
1/1 [==============================] - 0s 88ms/step - loss: 0.1180 - accuracy: 0.2889 - val_loss: 0.2501 - val_accuracy: 0.2593
Epoch 24/30
1/1 [==============================] - 0s 94ms/step - loss: 0.1180 - accuracy: 0.2889 - val_loss: 0.2501 - val_accuracy: 0.2597
Epoch 25/30
1/1 [==============================] - 0s 85ms/step - loss: 0.1180 - accuracy: 0.2667 - val_loss: 0.2502 - val_accuracy: 0.2597
Epoch 26/30
1/1 [==============================] - 0s 102ms/step - loss: 0.1179 - accuracy: 0.2667 - val_loss: 0.2502 - val_accuracy: 0.2597
Epoch 27/30

1/1 [==============================] - 0s 79ms/step - loss: 0.1179 - accuracy: 0.2889 - val_loss: 0.2503 - val_accuracy: 0.2597
Epoch 28/30
1/1 [==============================] - 0s 80ms/step - loss: 0.1180 - accuracy: 0.2889 - val_loss: 0.2503 - val_accuracy: 0.2597
Epoch 29/30
1/1 [==============================] - 0s 79ms/step - loss: 0.1179 - accuracy: 0.2667 - val_loss: 0.2503 - val_accuracy: 0.2597
Epoch 30/30
1/1 [==============================] - 0s 79ms/step - loss: 0.1179 - accuracy: 0.3333 - val_loss: 0.2503 - val_accuracy: 0.2597
C:\Users\Negatu\AppData\Roaming\Python\Python39\site-packages\keras\src\engine\training.py:3079: UserWarning: You are saving your model as an HDF5 file via `model.save()`. This file format is considered legacy. We recommend using instead the native Keras format, e.g. `model.save('my_model.keras')`.

/