Østfold University College

MASTER'S THESIS

Prediction of Water Consumption Using Machine Learning:

Using machine learning techniques to predict hourly water consumption in sustainable smart city

Elahe Kalashak

Autumn 2021

Master's Degree in Applied Computer Science
Faculty of Computer Science

Abstract

Energy demand and consumption are increasing as the world's population grows. This raises numerous challenges concerning resource constraints, given that the energy resources of the earth are limited. Recent technologies such as the Internet of Things (IoT) with a system of interrelated computing devices and machine learning techniques have collected, transferred, managed, and analyzed large amounts of data in smart sustainable cities. In the IoT scenario, sensor networks have a significant role in collecting, transmitting, and sharing data. These networks with real-time information processing in the Cloud-based servers can be utilized for energy consumption monitoring, energy demand management, traffic control, and various gas emission assessment for municipalities and governments in smart sustainable cities. The analysis and management of the big data collected through IoT sensors in smart cities provide the ability to manage energy resources, such as water supplies. Hence, this study aims twofold: first, to predict hourly water consumption by machine learning approaches, second, to develop a solution in a real-world problem. The data from the city of Sarpsborg (Norway) was used as a case study to manage its limited energy resources, being water supplies. This report provides an overview of the relevant studies from the literature, consisting of practical machine learning algorithms with an accurate prediction of hourly water consumption. The result of this study presents the remarkable ability of hourly water consumption prediction through applying supervised learning models, such as tree-based algorithms, Gradient Boosting algorithms, and finally, some discussion about the inefficiency of Longest Short-Term Memory (LSTM) as an Artificial Neural Network (ANN) algorithm based on the technique we have used for training and testing phase.

Keywords: Machine Learning, Water Consumption, Big Data Management, Energy Management, Sensor Network

Acknowledgements

I would like to express my sincere appreciation to my dear supervisor, Hasan Ogul, for his encouragement, patient guidance, and expert advice. Although the machine 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.

The technical contribution of "Sarpsborg municipality" is truly appreciated because of the support, information, and opportunity they gave us to study the real-time dataset in water consumption prediction.

Special thanks to Monica Kristiansen Holone and Harald Holone, in Computer Science faculty at Østfold University College, who always provided us with the opportunity to learn and continue this journey away from stress, with generous and timely support. I would also like to thank all the Østfold computer science faculty members and distinguished professors, Cathrine Linnes, Dr Ricardo Colomo-Palacios, Susanne Koch Stigberg, who have always been patient and kind in helping me progress and learn in this field. It was a great honor to have outstanding teachers and mentors like you.

I would like to thank my family's incredible support and love for the encouragement and emotional support, my dearest Iranian friends Alireza and Babak for being by my side in all the difficult moments, and my kind Norwegian classmates, who accompanied me on the path of success.

Contents

Abstract i
Acknowledgements iii
Contents v
List of Figures vii
List of Tables ix
Chapter 1 Study Overview 1
1.1 Introduction 1
1.2 Motivation 2
1.3 Problem Statement 2
Chapter 2 Background 3
2.1 Sensor Network 3
2.2 The Internet of Things (IoT) 4
2.3 Microsoft Azure 5
2.4 What is Big Data? 6
2.5 Big Data Analytics (Characteristics and Techniques) 6
2.6 Which Type of Data Exist 6
2.7 Machine Learning 8
2.7.1 Types of Machine Learning Algorithms 8
2.7.2 Models and Algorithms 9
2.7.3 Regression Evaluation Metrics 17
2.7.4 Hyperparameters Optimize Machine Learning Models 18
2.7.5 ReactJS 19
2.8 Research Questions and Methodology 19
2.8.1 First Section: Machine Learning Research Approaches 19
2.8.2 Second Section: The ReactJs Research Approaches 23
Chapter 3 Related Work 25
3.1 Overview 25
3.2 The ReactJs Related Work 35
Chapter 4 Methodology 39
4.1 Strategy & Analytic Approach 39
4.2 Data Analysis 40
4.2.1 Data Discovery 40
4.2.2 Data Preparation and Transformation 42
4.3 Machine Learning Models 45
4.3.1 Predictive Modelling 45
4.3.2 Prediction Model 48
Chapter 5 Results and Evaluation 51
5.1 SVM Result 51
5.2 AdaBoost Regressor Result 52
5.3 Ridge Regressor Result 53
5.4 RF Result 53
5.5 KNN Regressor Result 54

5.6 XGBoost Result 54
5.7 LSTM Result 55
5.8 Comparison of Algorithms 56
Chapter 6 Discussion 59
Chapter 7 Conclusion and Future Work 67
Bibliography 69
Appendix A The Results of all Runs 77

List of Figures

Figure 1.1. IoT and Cloud services 1
Figure 2.1.Different types of the Cloud services 4
Figure 2.2.Hub hosted in the Cloud 5
Figure 2.3. IoT process implementation of Sarpsborg municipality (Norway) 5
Figure 2.4.Machine Learning Lifecycle 8
Figure 2.5. Machine Learning tasks and Supervised Learning process 9
Figure 2.6.The input data is separated by hyperplane line 10
Figure 2.7.Data separation into two dimensions by a decision surface 10
Figure 2.8.The functionality of Random Forest. The final result is the majority of voting (red ball) [21] 11
Figure 2.9. Bootstrap and Aggregation in the Random Forest [22] 11
Figure 2.10.The process of Bagging regression model. The Mean means Aggregation [23] 12
Figure 2.11.An ensemble learning method example [28] 13
Figure 2.12.The new strong models are made by Boosting and ensemble learning techniques and the average of these models is used for regression as a final result or output [30] 14
Figure 2.13. KNN regression plot by using n_neighbors = 1 / using more closest neighbor and prediction by computing the Mean of the relevant neighbors [36] 15
Figure 2.14.The RNN architecture. The figure shows an RNN layer (left) and its unfolded schema (right) [38] 16
Figure 2.15. The LSTM structure [35] 16
Figure 2.16. Our research methodology is a combination of SLR and Snowballing 22
Figure 2.17. SLR methodology for ReactJs studies 24
Figure 4.1.Our methodology structure based on the Foundational methodology structure 39
Figure 4.2.Data Analysis Road Map 40
Figure 4.3.DEFA structure, the data collection process. (modified from [87]) 41
Figure 4.4.Data Categories and Transforms by Encoding 43
Figure 4.5.Initial Machine Learning models' evaluation on our dataset 45
Figure 4.6.Our Method to split the dataset for Training and Testing phases. The hourly water consumption is shown on the time axis. 47
Figure 4.7.Water Consumption Prediction procedure 49
Figure 5.1. Comparison of Algorithms' results 57

List of Tables

Table 2.1. Brief description of Sarpsborg Data 7
Table 2.2. The abbreviations' description of LSTM 17
Table 2.3. Inclusion and Exclusion criteria for Machine Learning studies 21
Table 2.4. The result of Research Execution 22
Table 2.5. Inclusion and Exclusion criteria for ReactJs studies 24
Table 3.1.The summary of the ReactJS studies 37
Table 4.1. The final dataset used in the Training and Testing phases 44
Table 5.1. The results of the applied SVM model with details 52
Table 5.2.The results of the applied AdaBoost model with details 52
Table 5.3. The results of the applied Ridge Regressor model with details 53
Table 5.4. The results of the applied RF model with details 53
Table 5.5. The results of the applied KNN model with details 54
Table 5.6. The results of the applied XGBoost model with details 54

Chapter 1 Study Overview

1.1 Introduction

In recent years, energy management has been one of the most critical issues that has attracted human attention. As the world's population is increasing, natural resources are also running out. Water is one of the limited resources. The United Nations Development Program (UNDP), through the human development report in 2006, compared the poor and rich countries in access to water and reported that there is extraordinary inequality in access to water. Daily water consumption in Europe is between 200 to 300 liters for each person, 575 liters in the USA and 300 liters in Norway. In contrast, in countries like Mozambique, the average consumption is less than 10 liters per year [1]. The average annual consumption of bottled water in Italy is 200 liters per habitat, the highest in the European Union. In comparison, Finland has registered the lowest annual consumption of bottled water at a rate of only 16 liters [2]. Lack of proper water management has become a global challenge. This trend is estimated to continue for the next 19 years [3], our planet's water resources will run out, and we will face not only a crisis but also a global catastrophe. Therefore, governments with different strategies and research through various techniques and tools try to find a solution for this great challenge. The use of IoT, the Cloud, and sensor network technologies associated with machine learning techniques have shown to be promising in monitoring, controlling, and minimising water consumption. The structure of IoT and Cloud services is shown in Figure 1.1.

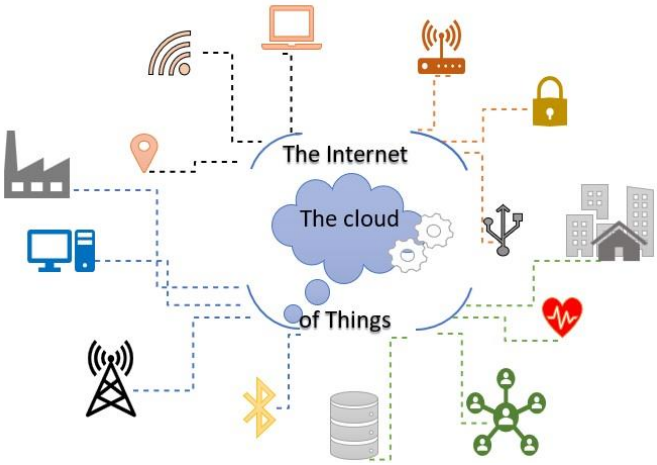


Figure 1.1. IoT and Cloud services

An essential issue about this story is how we can apply big data to improve energy consumption, traffic management, air pollution, and so forth. In this study, we considered generating data of sensor networks established in Sarpsborg city to investigate and

manage water consumption to prevent wasting water. Because of the broad scope of research, we only focused on water consumption. The importance of water, the statistics, and the reasons mentioned above have led us to utilize the ability of Information Technology (IT) in aggregating, storing, managing, and analyzing the data to take a step toward the process of improving water consumption. Perhaps this small step could be a practical starting point for further studies to conserve valuable water resources. According to this study, we were supposed to predict hourly water consumption in Sarpsborg (Norway) as a smart sustainable City. We implemented several machine learning algorithms to determine the best manner for big data management and water consumption prediction. However, there are various IoT and Cloud computing techniques for big data management; we prefered to choose machine learning algorithms based on our study on research that others have done in this case which are described in the “Related Work” section.

1.2 Motivation

The European Food Safety Authority (EFSA) recommends a daily water intake of 1.6 liters for women and 2.0 liters for men. Also, every person needs 5 liters [3] for living in a moderate climate that equals 1.31 gallons, although, in the USA, an average every American uses 100 to 175 gallons of water per day. The fundamental goal of the study on water consumption data, and the motivation behind our research, was to aware people about the case of the water crisis as one of the limited sources of energy, using new technologies to measure and predict the future amount of consumption which may lead to correct the wrong human habits of water consumption. Using a sensor network can be one of the most efficient solutions to measure and record the amount of water usage in domestic and industrial consumption to modify or decrease water consumption.

1.3 Problem Statement

Our research entails a practical aspect of water consumption forecasting and a theoretical part that examines the effect of the History Sensitivity Analysis method on time series data forecasting. In fact, the goal of this study was to use the History Sensitivity Analysis to find out the best time interval that makes sense for each algorithm for our forecast. Sensitivity Analysis refers to assessing the model results’ sensitivity to the alternation of the model's assumptions and inputs. It is a useful method to better examine the model input parameters' impact on the model behavior and show its performance [4].

An important question in machine learning modelling is how much historical data should be included to have better results and less execution time. Thus, our study used a method to investigate the sensitivity analysis of time series data for several machine learning algorithms. We wanted to apply every algorithm based on its inherent nature at different time intervals to see which time interval from the prediction point works best. At the end of this study, the reader can choose an algorithm based on the volume and size of the available time-series data that is efficient and suitable for water consumption forecasting.

Chapter 2 Background

The story begins with the management and analysis of data from the Sarpsborg municipality as a smart sustainable city, where the sensor network gathers large amounts of data from a variety of sources, such as water and electricity supplies, traffic control, air pollution, and so forth. We mentioned Sarpsborg municipality as a smart sustainable city because there is a slight difference between the two terms: smart sustainable city and smart cities. The difference between sustainable smart cities and smart cities is their focus on different sectors. Sustainable smart cities focus on transportation, energy consumption, water management, and whatever related to the built environment or natural environment. In contrast, smart cities have focused on science, information and communication technology, education, innovation, and the culture of society [5]. Norway includes 18 fylker (counties) and 422 kommuner (municipalities). Sarpsborg municipality is one of the cities of the Østfold county of Norway; its population is around 55,127, and its area is 405.61 km^2 [6].

Given that energy consumption has become a crucial issue worldwide and legislative powers strive to find a solution, we decided to improve our knowledge in energy consumption management. To achieve this goal, in the first step, we were required to cooperate with the Sarpsborg municipality for using a dataset related to energy consumption. Then water dataset was chosen as a case study for data management and consumption prediction in this research. Applying machine learning for data analysis and prediction of water consumption was the next step.

2.1 Sensor Network

The main reason for increasing sensors' use in various aspects is easy deployment and low cost [7]. Various IoT sensors are applied based on our requirements for different goals in IoT, such as moisture IoT sensors, noise and acoustic IoT sensors, temperature IoT sensors, water level IoT sensors, light IoT sensors, image IoT sensors, chemical IoT sensors, and gyroscope IoT sensors. The type of integrated sensors in our study for water consumption investigation was LoRa (Long Range) for Sarpsborg's IoT network and LoRaWAN protocol. The LoRa has structured as a physical layer based on LoRaWAN protocol and can transfer a huge volume of data or information over a high range of a geographic area. Indeed, low power can send data over long distances using radio frequencies, making it a remarkable and efficient technology [8]. LoRa Technology includes outstanding characteristics such as low cost, long-range, low power, and open standard. It means it has the capability to decrease the cost of operating and infrastructure investments, it penetrates deeply in the dense urban structure and can cover sensors in long distances which are more than 30 miles far away in the rural areas, increase the lifetime of a battery up to 20 years through the use of LoRaWAN protocol which is perfect for low power, and with the help of LoRaWAN protocol [9] provides some form of

collaboration among telecom operators, applications, and IoT solution providers to expedite the adoption and deployment process.

2.2 The Internet of Things (IoT)

IoT, through some software, has access to the Cloud as a platform and generated data by a sensor network is transferred to the Cloud. The Cloud as a computing platform increases the computing efficiency and data storage, which is done with a high level of performance, almost a hundred percent reliability, and extensive scalability [1]. There are several Cloud services (Figure 2.1) [7]. In our study, Sarpsborg municipality has applied Microsoft Azure.

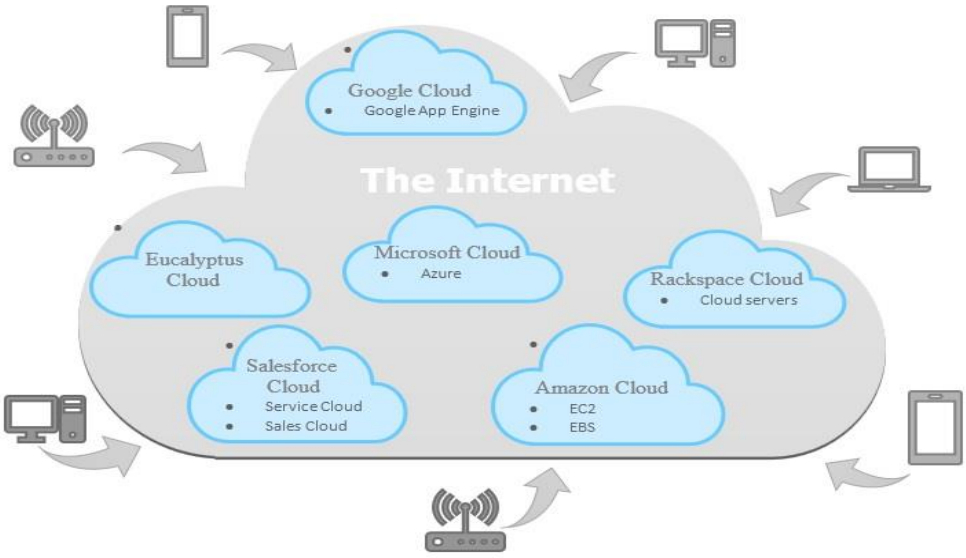


Figure 2.1.Different types of the Cloud services

The sensor network is built by sensor devices to collect a massive volume of information from different resources. The connections between devices and IoT applications are carried out through Hub hosted in the Cloud (Figure 2.2), creating a bi-directional connection between devices and the Cloud [10]. IoT Hub as a managed service is a center for sending messages and supports sending information from IoT devices to the Cloud and vice versa [11]. The process and storage of data start as soon as the data arrives at the Cloud, which has the ability in real-time response, so the Cloud can decide to begin automatic adjustments or send alerts, and this process does not require any user.

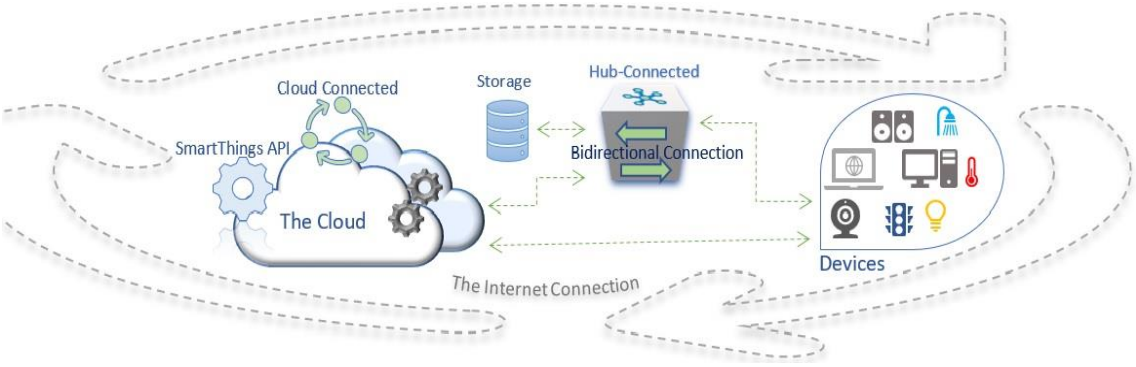


Figure 2.2.Hub hosted in the Cloud

Sensor data can give us much information about how different processes are performed. For example, the processes that take place within cities can be evaluated and controlled by these data. We focused on water consumption which is known as limited resources around the world, and smart sustainable cities with sensors can detect any problems with the water delivery process or consuming water. As Figure 2.3 shows, we can see the IoT process implementation of Sarpsborg municipality that they have applied the LoRaWAN sensors in the Microsoft Azure. The sensor network data can transfer to the Cloud using Azure IoT Hub, creating a connection between all devices and the Cloud.

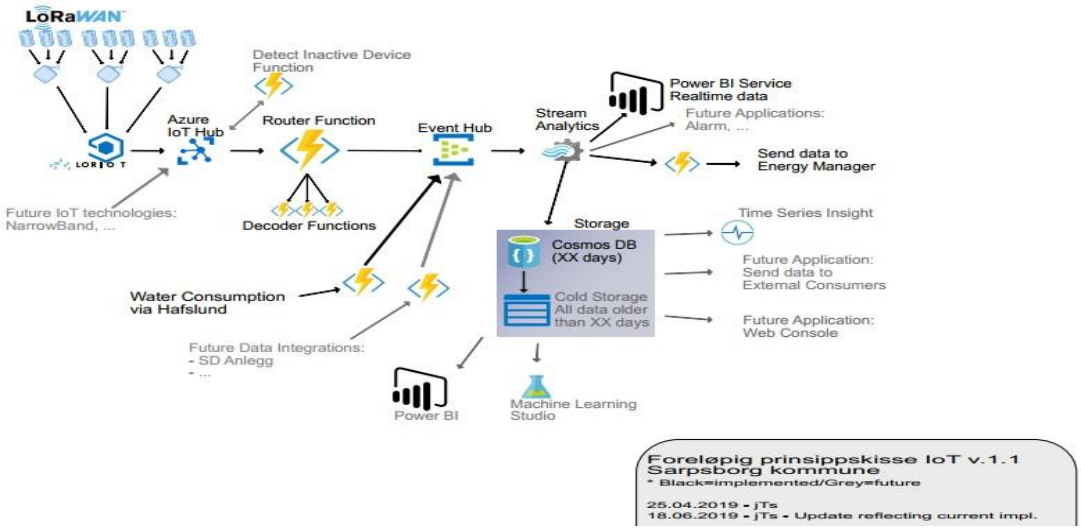


Figure 2.3. IoT process implementation of Sarpsborg municipality (Norway)

2.3 Microsoft Azure

Azure is one of the Cloud services that is created by Microsoft. Microsoft Azure is known as a storage service and Cloud computing platform with a lot of advantages. For example, Azure provides efficient storage that is safe, scalable, stable with a high level of availability [12]. Also, it helps users to make resources easily that are Cloud-based

like databases, virtual machines, and so forth. Users with various technology and tools in Azure [13] can create and deploy Cloud-based services and applications with different functions such as networking, computing, storage, and analytics.

2.4 What is Big Data?

These days governments, institutions, and companies integrate sensors using IT, and the use of this technology at a large scale generates huge amounts of data. With the help of IoT in different aspects of life, the use of sensors is expected to grow significantly. Although data collecting is doable through many IT technologies like IoT, handling various massive datasets is another issue. Therefore, identifying and classifying the obtained data is the most crucial step in managing, using, and visualizing data. At this point, we need to mention some characteristics of big data to apply the right tools and techniques to categorize and visualize the data.

2.5 Big Data Analytics (Characteristics and Techniques)

A general definition of big data is we collect vast volumes of data, access the inaccessible dataset, and developing technologies for collecting big data. Data with high velocity, volume, and variety are known as big data [14]. It is noteworthy that while some applications process data only at certain times of the day, others do data processing at all hours of the day. For example, real-time applications are part of this category [15]. These characteristics indicate our need to apply appropriate techniques for the process of classification and management of big data that can be achieved using algorithms, AI, and various software and hardware in the field of IoT.

2.6 Which Type of Data Exist

Here we talk about our dataset characteristics: our data is univariate time-series, low-dimensional dataset, and the data format is JavaScript Object Notation (JSON). Time-series data means the data has been recorded and sorted in the time, and data are affiliate to each other that these are two essential features of time-series data. If time sequences play a significant role in data or output results, these features should be considered in model construction. Because predictions are made based on models, and models build their forecast pattern based on observations recorded in past time-series. Univariate time-series data means the data is being recorded and observed at specified intervals. Finally, there is a single list of sequential data measurements that the time is an implicit variable in these types of datasets. The order is a vital feature for the events that depend on time because it affects data concepts, creates a proper model, and predicts accuracy [16]. Generally, when the number of features is smaller than the number of samples [17], we have a low-dimensional dataset. Dimension refers to the number of features (variables) of a dataset provided in the columns. For example, Device-ID, Measurement Time, Value, MeteringPointId, ... are some of the water consumption features in our study. Because our raw dataset contained only three main features, it was much smaller than

the number of samples. Therefore, our dataset for predicting the hourly water consumption was a low-dimensional dataset [18].

The Sarpsborg municipality dataset was time-series, and its type was JSON. Modern and new programming languages can generate, read, utilize, and analyze JSON data format. JSON was driven from JavaScript (Table 2.1). It is a standard and lightweight data-interchange format, a language-independent, serialized data transfer capability that is easy to understand for computers and humans. Due to these reasons, it has the main role in web services and web applications [19]. JSON includes excellent numbers of comfortable characteristics, and it is known as a perfect data exchange language. It is 1) a standard text-based with a language-independent feature that can be utilized by programmers in Java, JavaScript, Perl, Python, C family languages, and many others 2) easy production and analysis for machines 3) easy for writing and reading [20]. Therefore, along with web services growth, JSON's role is more critical, and it is more utilized.

```
public struct TimeSerie
 {      public string MeteringPointId;
        public string DeviceId;
        public string ProductName;
        public string SourceRegister;
        public double Value;
        public DateTime RegistrationTime;
        public DateTime MeasurementTime;
        public Quality Quality;
        public long Interval;
        public string SourceCustomerId;
        public string SourceVersion;
        public string ExportDataId;
        public string ExportDataType;    }
```

Table 2.1. Brief description of Sarpsborg Data

It should be mentioned that the Sarpsborg Municipality’s datasets are not yet considered big data due to the low diversity of features. But in the near future, due to the increase in the number of sensors in places of consumption, the number of data will also increase, and they can be considered as big data. On the other hand, in this study, we generated new features using feature engineering methods; in this way, we solved the issue of features’ low variety, and our dataset could be considered as big data.

2.7 Machine Learning

Machine learning as a subset of AI is widely used these days in various fields such as industry, health, environment, energy, and municipal utilities. Machine learning is quite well-known as an efficient technology in future prediction because of its ability to find data patterns from past data. Self-learned and automatic improvement through experience are two main remarkable features of machine learning: working with various types of data, applying different algorithms and statistical techniques, big data handling, data analysis, and future prediction. Figure 2.4 shows the machine learning lifecycle. First, we define the business problem and specify the prediction aim. Then we prepare the data collected and select the appropriate data in the analysis step for utilizing in machine learning. In the Model step, we try to build a model based on our target variable and select features that affect the target value and the prediction. In the next step of the machine learning process, this model makes a pattern from the sample or primitive data. When new data enters, the model trains the data to test the relation between new data and the primitive pattern for prediction.

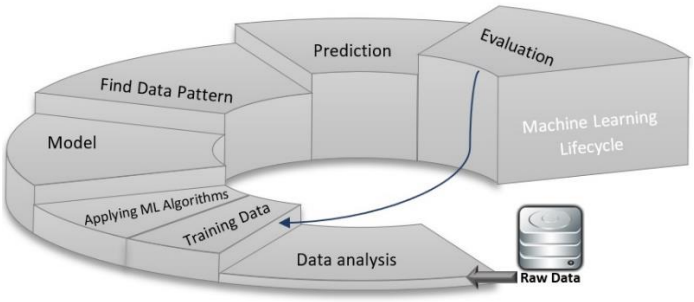


Figure 2.4.Machine Learning Lifecycle

2.7.1 Types of Machine Learning Algorithms

The first step in this section is about which type of machine learning tasks is suitable for our study for selecting the proper machine learning algorithms. Indeed, machine learning problems are divided into two tasks: supervised learning that works with the labeled data and unsupervised learning that works with unlabeled data (Figure 2.5). Labeled data means the input data or samples come with a label (tag) such as name, number, or type. Therefore, in this study, we describe the supervised learning task because we had labeled data. Supervised learning investigates the raw input data. When new data enters as an input, supervised learning algorithms try to produce the correct label for new data. Indeed, the supervised learning algorithms carry out this through the training data analysis and create a labeled output. This model predicts the future output based on available evidence. The evidence is available raw input or primitive sample of the labeled dataset that the predictive model has shaped based on them. Therefore, based on our dataset in this study, we had a supervised machine learning task that makes a predictive model.

Some combined regression models have been used in most of the research as a statistical tool because of their ability to forecast the target value with continuous values. The

regression model is based on supervised learning. There are different types of regression like Linear regression, Support Vector regression (SVR), Decision Tree regression, KNN regressor, AdaBoost regressor, Ridge regressor, and Random Forest regression.

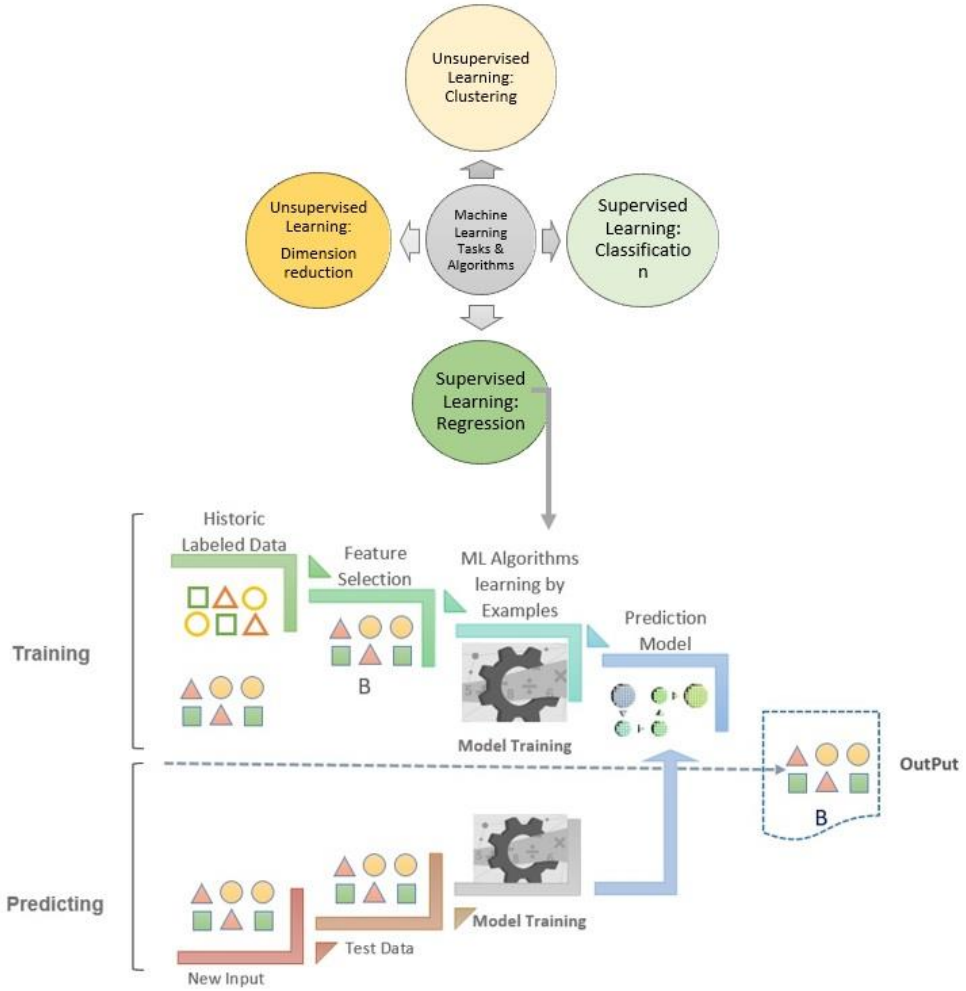


Figure 2.5. Machine Learning tasks and Supervised Learning process

2.7.2 Models and Algorithms

In this section, we describe some of the machine learning algorithms that we decided to apply for our study after first evaluating our dataset and investigating the result of different algorithms. It is mentioned that this is just a brief description of each algorithm because our study is not an Systematic Literature Review (SLR) on the functionality of each algorithm or its advantages or disadvantages. Therefore, we provide a short explanation based on their ability to give a generic perspective about what algorithms we used in this study based on our study approach or problem statement.

Support Vector Machine (SVM): One supervised machine learning algorithm is the SVM model that can analyze the data for both regression and classification. Although the SVM is a linear model, it can be used for both non-linear and linear models. This

analysis is done by SVM through a technique that is called Kernel technique. Kernel as a mathematical function is one of the SVM hyperparameters that try to find out the most optimal and efficient separating line or boundary by transforming the input dataset into two phases or dimensions [14]. When we can separate the input data into two sections, and they are separable, we utilize a hyperplane line to create two classes, as is shown in Figure 2.6.

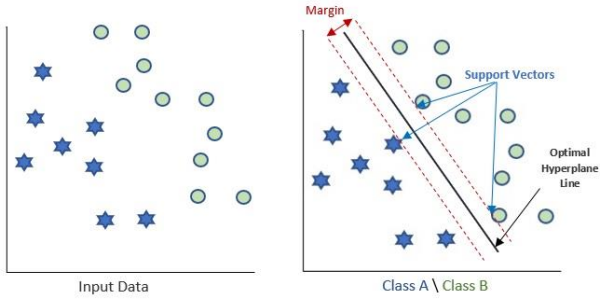


Figure 2.6.The input data is separated by hyperplane line

As is shown in the Figure 2.6, the solid black line is an optimal hyperplane line that the distance between two dotted black lines, and the optimal hyperplane line is called margin. The two dotted lines are two hyperplane lines that move between the nearest and optimal hyperplane lines. The closest data to the hyperplane lines are support vectors, and it can be claimed that often there is no data in the margin area when we use this method. But if the raw data or input data is not separable, the data is divided into two-dimension like illustrated in Figure 2.7.

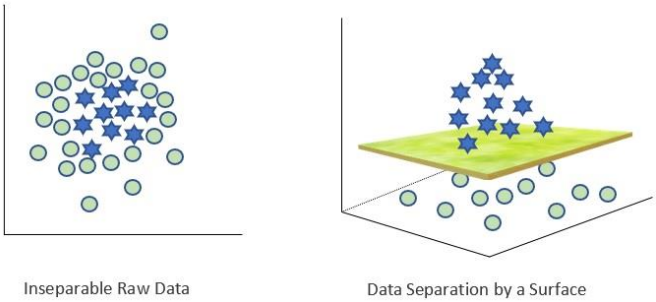


Figure 2.7.Data separation into two dimensions by a decision surface

The Kernel type can be Radial Basis Function (RBF: for non-linear problems), Polynomial Kernel Function, Linear Kernel, Sigmoid, Precomputed, Gaussian Radial Basis Function, and Gaussian Function. If we do not determine a specific type for the Kernel, the default type for the Kernel is considered RBF. The SVM parameters are the Kernel, degree, gamma, coef0, tol, C, epsilon, shrinking, cache_size, verbose, max_iter that can be modified or changed based on our dataset or model function [21].

Random Forest (RF): Another most popular machine learning model and algorithm is RF, a supervised learning and a tree-based algorithm (Figure 2.8). "Random" means this algorithm uses many different decision trees made randomly, and this huge number of trees creates a "Forest" of trees. One decision tree has a high level or amount of vari-

ance in the training set. At the same time, the RF uses several decision trees on one sample of the dataset, that the result of all the decision trees is the low level of variance. Indeed, the collection of confluences and the production of the decision trees in each sub-branch improve the algorithm performance. So, the result or output is gained based on the combination of multiple decision trees, not one decision tree.

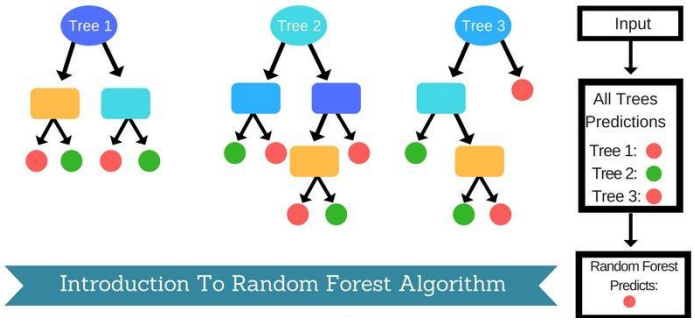


Figure 2.8.The functionality of Random Forest. The final result is the majority of voting (red ball) [22]

The method used by RF is the Bagging technique that includes Bootstrap and Aggregation phases (Figure 2.9). Each tree in the training phase is build based on learning from one sample of data points that are randomly selected. Bootstrap does resample through replacement which means every sample replaces with a random sample selected. Sometimes one sample can be repeated or used many times in the replacement process. RF in the regression model considers the mean of all the outputs as a final result or output that this process is called Aggregation (Figure 2.10). RF in the classification model produces the final output based on the majority vote.

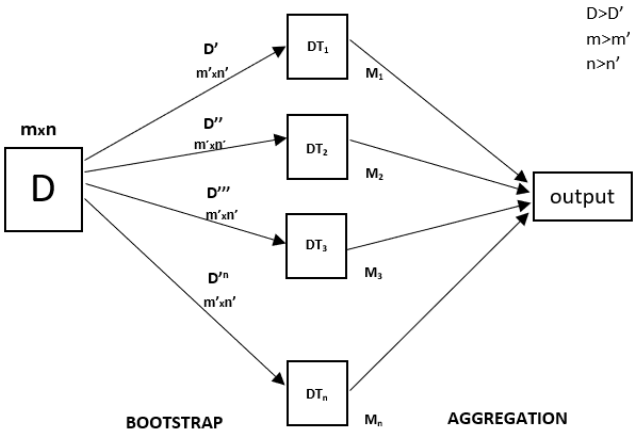


Figure 2.9. Bootstrap and Aggregation in the Random Forest [23]

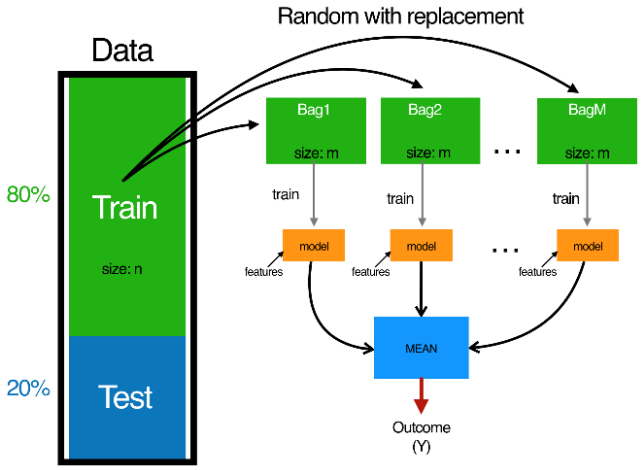


Figure 2.10.The process of Bagging regression model. The Mean means Aggregation [24]

The RF is a fast model in the data training phase because of its great number of decision trees, but it is known as a slow algorithm in prediction when the dataset is trained. Therefore, we should maybe choose other algorithms for run-time performance and real-time prediction. RF is a widely used model for most machine learning approaches. Some algorithms like the neural network algorithm can be better in some features such as better performance compared with the RF algorithm. But the neural network algorithm is time-consuming, while RF, with easy and quick development, is an efficient algorithm for various features like categorical, numerical, and binary, making it a flexible algorithm. Overall, RF is a fast, simple, robust, and diverse algorithm with easy and quick development that we can apply for both regression and classification tasks [25].

XGBoost: When we want to talk about performance and speed for supervised learning tasks, XGBoost (Extreme Gradient Boosting) is another efficient algorithm that is a tree-based algorithm. XGBoost can be used for classification and regression tasks in machine learning challenges when we have a structured dataset with small or medium size. For example, the countless of decision trees causes to overfitting issue and model complexity. XGBoost algorithm can eliminate these problems through Ridge regression and Lasso regression [26]. This algorithm is capable of managing missing values by understanding the missing values' trend. This trend is gained through automatic "learning" from the best missing values in the "training" phase of the XGBoost algorithm. Using the automatic learning ability of XGBoost can also help to fix the problem of raw data sparse.

Furthermore, the XGBoost structure includes a Cross-Validation (CV) function that this ability means we do not need to import the CV function from Scikit-Learn library [27]. XGBoost algorithm follows the ensemble learning [28] method (Figure 2.11) to predict the distance between the predicted values and the actual values. In contrast with machine learning method that uses one hypothesis based on each data training phase (base learners or individual models), the ensemble learning method uses several learners

and make a combination of hypothesis to create a sample for more precise prediction or predictive model. The ensemble models include several base learners in which both training and testing phases are performed. In fact, because the base learners work based on a random guess, the XGBoost algorithms extract the poor performance of base learners from a combination of prediction of ensemble learners to gain excellent and precise final prediction.

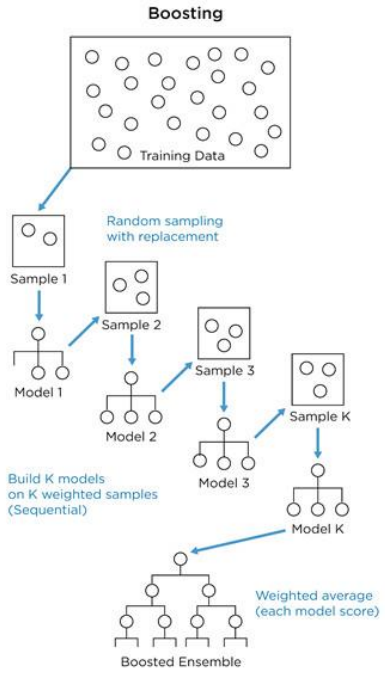


Figure 2.11.An ensemble learning method example [29]

AdaBoost Regressor: An ensemble method and Boosting are two essential features of the AdaBoost algorithm. This algorithm uses the ensemble method to grow trees in regular series in the training phase and tries to improve the weak classifications by using the Boosting feature (Figure 2.12). It does this by Boosting the combination of previous weak classifications and trying to set a new strong combination of previous weak classifications into the new classification to alleviate the problems of the previous poor classification in the new sample. Decision trees that grow using the Boosting method and form new classifications are called "stump". In this case, each tree is trained so that it pays particular attention only to the weaknesses and challenges of its previous tree. This model works based on this hypothesis that making a new model from the previous weak models can create a new powerful model that ensemble learning produces sequentially. In the regression problems, the AdaBoost algorithm computes and applies the Mean of these models made by Boosting and ensemble method [30].

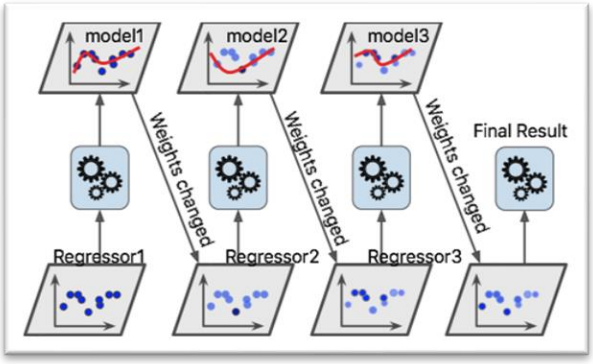


Figure 2.12.The new strong models are made by Boosting and ensemble learning techniques and the average of these models is used for regression as a final result or output [31]

Indeed, the output or final predictor is a combination of all several predictors, including their knowledge about the previous models or predictors. With this approach, each new model is more efficient than the previous model.

Ridge Regressor: The regularized shape of linear regression is Ridge regression, one of the supervised learning algorithms. Ridge is a model tuning algorithm that can analyze every dataset that has a multicollinearity problem. When there is a high correlation between some input variables with other variables in the regression model, the dataset has the multicollinearity problem. This algorithm uses the L2 penalty technique (adding a squared magnitude of the coefficient to the loss function) to shrink some parameters like coefficient for those input variables that do not influence the model prediction [25], [32]. By limiting the size of all coefficients, the L2 penalty method tries to make these ineffective parameters smaller and makes them zero or omitted. Also, it decreases the complexity of the model because of coefficient shrinkage. So, the Ridge algorithm with the L2 penalty method can prevent the multicollinearity problem [17], [33]. This method is useful for feature selection when we have a great number of features in the input dataset because it declines or removes ineffective features.

K- Nearest-Neighbors Regressor (KNN): One of the non-parametric algorithms initiated by Fix et al., 1951 [34] and then developed by Cover et al., 1967 [35] is the KNN algorithm (Figure 2.13) which is used for both classification and regression problems. Based on the performance of this algorithm, every data point gets a value or a weight. When a new data point is entered, the algorithm tries to find out how similar the new data point is to the training dataset points and assign a new value to this new input based on this similarity [36]. The KNN calculates the distance between the data points in the training set and the new input data point that is a new input or observation. This algorithm is sensitive to the scale of the dataset because it works based on distance. Therefore, before using this algorithm, we should consider the scale of our dataset. Because on a larger scale, it calculates the higher distances leading to the poor result. In this al-

gorithm, the K is an integer value and parameter that points to the number of all nearest neighbors in the most of voting process steps.

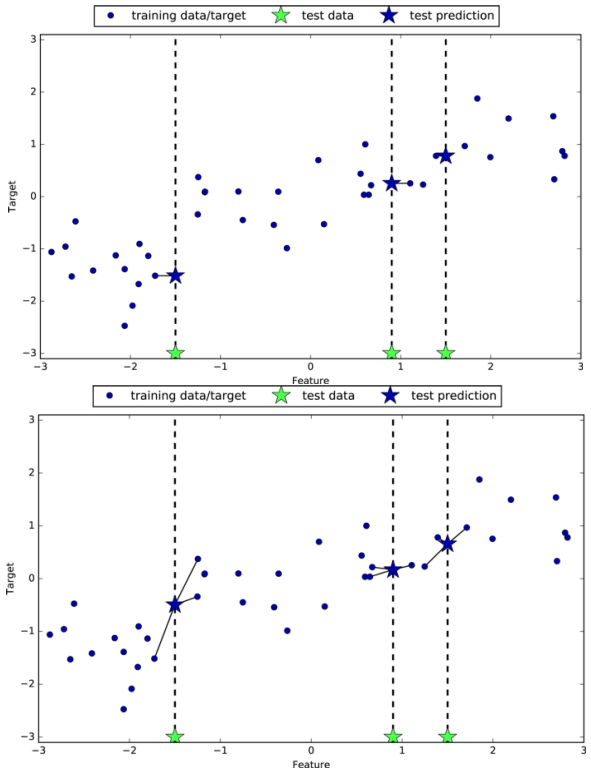


Figure 2.13. KNN regression plot by using n_neighbors = 1 / using more closest neighbor and prediction by computing the Mean of the relevant neighbors [37]

The important thing in the KNN for classification is that it calculates the Mode of nearest K neighbors, while in the regression, it computes the Mean of nearest K neighbors. The KNN can store all the training samples and forecast numerical target values based on distance functions. In fact, in both regression and classification models, KNN works based on the distance functions. The simple functionality of the KNN for regression is to compute the mean of the numerical target values of the KNN. As mentioned above, this algorithm stores all training instances in memory because it does not have any special training phase. This can be a great advantage for this algorithm that can make predictions without using the training phase. But the problem arises when this algorithm is computationally costly if the data is too large. Because this requires a lot of memory space and time to store all the training samples, it is also called a lazy algorithm due to not having a particular training phase and storing all the training samples. The lack of a special training phase and a non-parametric algorithm makes the KNN an efficient algorithm for non-linear datasets [38].

Long Short-Term Memory cells (LSTM): LSTM has been introduced to improve Recurrent Neural Networks (RNN). Therefore, first, we introduce the concept behind RNNs. RNNs are other types of Artificial Neural Network (ANN) in which the neurons

have connections to subsequent steps. Figure 2.14 demonstrates a simple RNN layer architecture. Like other ANNs, RNNs can have many hidden layers, or connections can have complex behaviors.

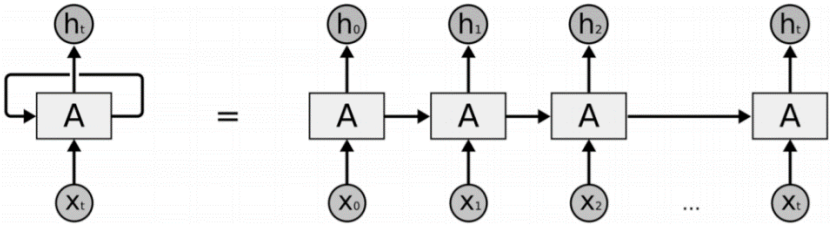


Figure 2.14.The RNN architecture. The figure shows an RNN layer (left) and its unfolded schema (right) [39]

In a standard ANN, the data goes through the input, hidden, and output layers, respectively. While in RNN, the hidden layer receives information from both the current time step input layer and the prior time step hidden layer. In this way, the RNN can keep the past or historical information [36]. Recurrent networks are widely used in sequential data like time-series problems because this kind of network can consider the non-linearity of sequences, preserve the previous state, and remember past events by connecting past and current neurons. This characteristic makes the RNN models very appropriate for time-series prediction problems [40].

By training RNNs using backpropagation, through time, the vanishing and exploding gradient problems will happen. The exploding case occurs when the gradient factor increases exponentially, making the model unstable because of a large change in the weights. On the other hand, the vanishing case is when the component decreased enormously. The weight coefficients become very small, near-zero in this condition, and the model does not learn anything during the training. For tackling these problems and improving the RNNs, some solutions were introduced; among them, LSTM was a successful approach [41]. The structure of LSTM is depicted in Figure 2.15, and the abbreviations are described in Table 2.2.

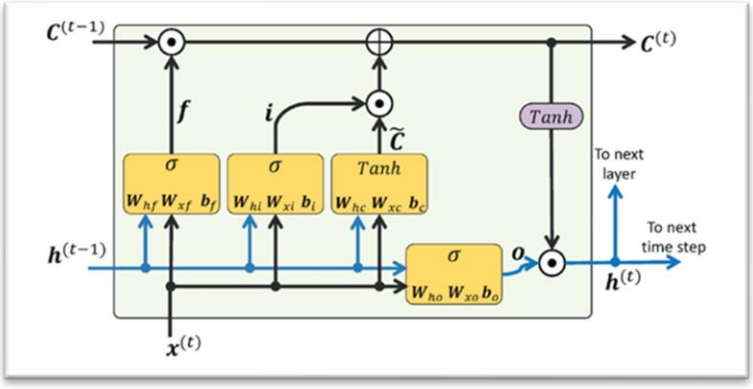


Figure 2.15. The LSTM structure [36]

Table 2.2. The abbreviations' description of LSTM	
$C^{(t-1)}$: the cell state from the previous time step (t-1) $C^{(t)}$: the cell state from the current time step (t) $x^{(t)}$: input data at current time step (t) $h^{(t-1)}$: hidden units' activation at previous time step (t-1)	$\oplus$: element-wise addition $\odot$: element-wise multiplication σ: sigmoid function Tanh: hyperbolic tangent function W: weight matrix b: bias vectors

The LSTM is composed of three computation units called gates:

- The input gate (i) is responsible for allowing the signal to update the "cell state" or not.
- The forget gate (f) makes the cell keep its past state or ignore it.
- The output gate (o) permits the cell state to influence other nodes in the layer or prevent that [42].

2.7.3 Regression Evaluation Metrics

There is the fact that the ability of machine learning models in future prediction should be evaluated by some statistical metrics or measurements. We can use various metrics in the regression models to estimate prediction accuracy. In our regression models, we used some metrics like:

Mean Absolute Error (MAE): measures the errors (differences) between predicted variables and the target, then calculates the absolute value of the average of the total errors of the predicted set [43]. Our study first estimates the MAE for each Device-ID based on time duration changing and choosing minimum MAE. After collecting all minimum MAE of all Device-IDs, we compare them and choose the least minimum MAE to find the best time duration for prediction. The lower the MAE value and the closer to zero, means that our model works better. Also, it is mentioned that we applied another type of MAE called RMAE (Root Mean Absolute Error), which is the value of the root of MAE.

Mean Squared Error (MSE): Another popular regression metric that we use in our machine learning models is MSE that calculates the sum of square differences (error) between predicted values and target variables [44]. In summary, the purpose of training the machine learning model is to reduce the amount of loss function to gain a prediction that is precisely equal to the actual value.

Root Mean Square Error (RMSE): Using RMSE (the root of MSE) helps find and handle the larger errors. Indeed, RMSE indicates how much our regression line is fit with the data points. The lower values of this measurement indicate better fit and higher accuracy for our predictive model [45].

Correlation Coefficient: This indicates how much variables relate to each other or how they relate. This value is a statistical measure and is always between (-1, +1). If the

linear correlation is very weak or the variables do not correlate, the Correlation Coefficient becomes (0). When the variables often move in the same direction, the Correlation Coefficient is (+1) because there is a perfect relationship and positive connection between variables, while (-1) shows the variables have a strong negative correlation or negative relationship [46]. This metric describes the dependency between our variables that prove how much of a change in one variable causes a change in another variable.

Variance score: There are three kinds of variance: residual, regression, and total variance. We utilize regression variance to investigate the degree of difference between actual data and our model. The goal of using this metric is to find the value error or difference of actual value from the mean of predicted data points through using the regression line rather than the mean to make the prediction. The Best possible value or score for variance is 1.0 and more than 60%. Lower values are worse and show that the data collected should be investigated or collected again. Perhaps some extra factors should be removed from the predictive model [47].

R-Squared (R^2 or coefficient of determination): R^2 calculates the proportion of variance to a dependent variable that is defined by variables in the regression model or independent variables. R^2 for the multiple regression represents how much the data points are close to the regression line. This statistical measurement describes how the variance of one variable can explain the variance of another variable. The R^2 value is the target variable variation value in the supervised learning that the linear model defines. This value is between 0 and 100%. The zero value means the model does not explain any variability of the target data. The 100% value shows that the model explains all variability of target value around its mean [48].

2.7.4 Hyperparameters Optimize Machine Learning Models

Hyperparameters are anything that is set before the training of the machine learning method begins. They are different from inner parameters. For example, in a neural network model, the weights are not hyperparameters because they are set and updated in the training process. The batch size or optimizer functions are hyperparameters since they are placed before training begins and do not change during the model training phase. Since they control the training algorithm behaviors directly, they are crucial in machine learning studies. Also, they have a fundamental impact on the model performance [49], [50]. Some simple machine learning models do not require any hyperparameters. While in some other algorithms, there are many hyperparameters, some may be dependent on the other ones. The execution time of model training and testing may depend on its hyperparameters configuration [51].

Hyperparameter Tuning (HPT): In machine learning, the process of finding hyperparameter values that have the highest performance concerning the execution time is called hyperparameter tuning (HPT) or optimization. This process is done before the training phase begins. There are a wide variety of hyperparameter iterations and combination options. In this regard, the HPT may be an exhaustive and time-consuming task [49]. Two main HPT methods exist: manual and automatic. Manual search performance

depends on the professional knowledge and experience of performers and should be done by expert users. This method cannot be applied when encountering high dimensional data or algorithms with many hyperparameters, and it is not reproducible easily. Automatic search methods are good choices to overcome these drawbacks. Among automatic search methods, Grid Search is a popular method. It is an exhaustive search and trains the machine learning algorithm with every possible value set of defined hyperparameters and provides the best combination with the best performance by evaluating the performance of models according to the predefined metric [50].

2.7.5 ReactJS

ReactJS is an open-source and frontend JavaScript library that is utilized to create a user interface. ReactJS is efficient and worthwhile due to its benefits and attributes. Some of its useful attributes are being declarative, fast, simple, flexible, scalable, building a web application, ability to communicate with old web servers like NGINX or Apache, ability to communicate with the backend like Rails, PHP, and letting you create a reusable and complex user interface from small parts of code (components) [52]. These remarkable traits lead every data scientist researcher to apply this frontend library to visualize JSON's data.

2.8 Research Questions and Methodology

In this study, the research section includes two parts. The first one comprises our research approach and methodology about the study's machine learning part. The second one is a brief literature review addressing the studies using ReactJs for JSON data visualization. In the first section, we use the new research methodology that is a combination of two methodologies, as we explain in the following.

2.8.1 First Section: Machine Learning Research Approaches

To achieve our goal of predicting the amount of water consumption, we shaped our research by investigating many studies about energy consumption in both IoT technology and machine learning techniques. Finally, we decided not to talk about both technologies because this study is not just the Systematic Literature Review. It is unnecessary to focus on all techniques to deal with this issue. So, we continued our study toward concentrating on the machine learning models and algorithms.

2.8.1.1 Research Question (RQs)

Research Question 1 What are the characteristics of the dataset used in the energy and water consumption studies?

Research Question 2 Which types of machine learning algorithms or models are efficient for analyzing water datasets and predicting water consumption?

Research Question 3 What are the other possible methods used in addition to Artificial Intelligence (AI) algorithms in energy and water studies?

Research Question 4 Which variables are influencing water consumption?

Research Question 5 What are the evaluation metrics for measuring the performance of models in water consumption studies?

2.8.1.2 Scholarly Sources and Search Strategy

2.8.1.2.1 Data Resources

In this section, the academic resources as are mentioned below were the basis of our research.

- ScienceDirect
- ACM Digital Library
- Springer Link
- IEEE Xplore Digital Library
- Hindawi
- Journal of Algorithms & Computational Technology

2.8.1.2.2 Search Term

After rounds of initial searches with various combinations of search terms, finally, we formulated the following search term, which was an efficient term for searching:

("SENSOR" AND ("BIG DATA" OR "MACHINE LEARNING") AND ("CONSUMPTION" AND "ENERGY" AND ("WATER" OR "ELECTRICITY")) AND ("CITY" OR "MUNICIPALITY"))

2.8.1.2.3 Search Process

Our search process is a combination of two techniques and includes four phases that results from the phases are described in Search Execution:

Phase 1. First, we reviewed the abstract, introduction, and summary of the related articles to our study. Then, we separated those papers that were more relevant to the subject matter studied. We finally transferred them into a reference manager known as Zotero.

Phase 2. Then for scrutiny review, we scanned all the resources obtained from the first phase accessible to explore the studies' details further. Due to a more precise investigation in this step, we reviewed a few resources related to our field of research.

Phase 3. Then we reviewed the remaining resources from previous phases based on the Systematic Literature Review (SLR) methodology to review and to perform our results. The results of this phase have been categorized in a data extraction form generated in an Excel file for streamlined accessibility.

Phase 4. In the final evaluation, we finalized our review by the combination of two techniques. To get closer to the studies that were precisely relevant to our research topic, we utilized the results of the SLR for doing Snowballing. As a result, we achieved exactly related studies in this area by searching for a few references from the previous phase.

2.8.1.3 Criteria as a Selection Tool

This step presents our criteria for selecting and choosing resources and categorising them into two sections: Inclusion and Exclusion Criteria (Table 2.3).

✓ Inclusion Criteria	⊗ Exclusion Criteria
✓ The studies which investigated the big data management obtained from sensor networks ✓ The papers which referred to at least one machine learning algorithms ✓ Studies related to sustainable smart cities ✓ Focus on energy consumption, especially water consumption	⊗ The studies before 2009 ⊗ Papers in a language other than English ⊗ Thesis, reports, books ⊗ The studies that are not relevant to our research like investigation security and Sensors' function ⊗ The studies that are not defined as reliable (such as web pages) ⊗ The inaccessible studies

Table 2.3. Inclusion and Exclusion criteria for Machine Learning studies

2.8.1.4 Research Methodology (SLR + Snowballing)

2.8.1.4.1 Search Execution

As Figure 2.16 shows, all the results have been achieved through the combined methods include the SLR and Snowballing technique on academic resources that we describe in the continuation of this section.

The important and time-consuming part of search execution was the 2nd step results (Full-text Scanning for Literature Review) that include choosing one technology between machine learning and IoT technologies. After investigating some IoT scientific papers, we decided to focus on machine learning methods in the SLR technique. Therefore, the number of results decreased because of removing IoT studies. The Snowballing technique helped us utilize the references of the most relevant studies to find other related studies in this area based on our research criteria. After further review by Snowballing method ability, 15 related sources were added to our resources to get closer to the subject under study (Table 2.4). The results of these 27 scientific papers focusing on the hourly water consumption prediction are fully described in chapter 3.

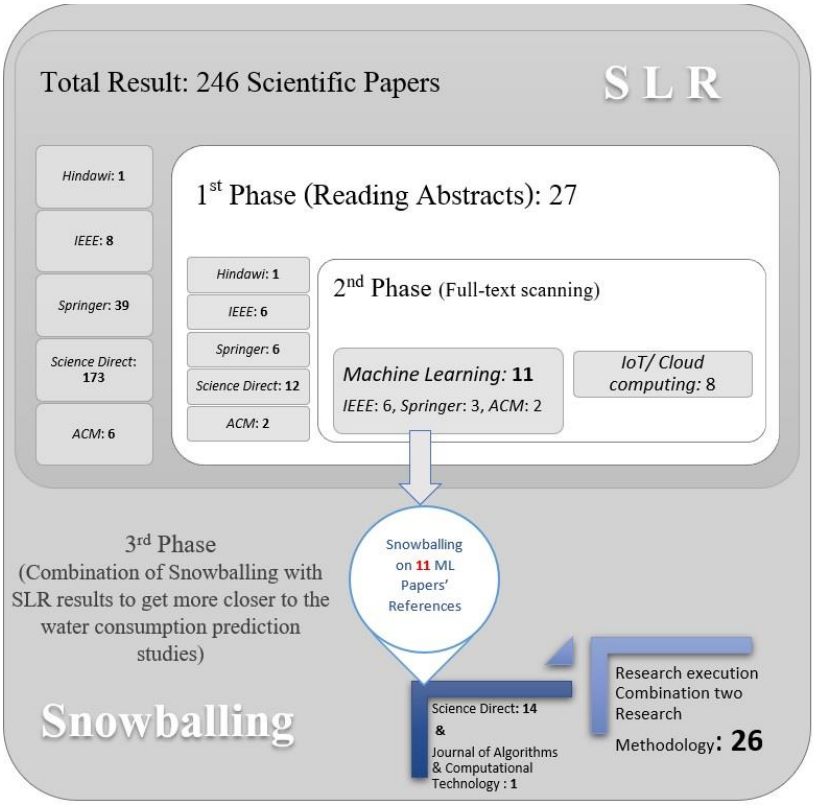


Figure 2.16. Our research methodology is a combination of SLR and Snowballing

Library	Full-Search Result	Abstract and Title Scanning	Full-text Scanning		Result of Snowballing on Machine Learning Papers' References
			Machine Learning Literature Results	IoT/ Cloud Computing Literature Results	
Hindawi	1	1	0	1	0
IEEE	8	6	6	0	0
Springer	39	6	3	3	0
Science Direct	173	12	0	4	14
ACM	6	2	2	0	0
Journal of Algorithms & Computational Technology	0	0	0	0	1
246		27	**11**	8	**15**
Research Execution			**26**		

Table 2.4. The result of Research Execution

2.8.2 Second Section: The ReactJs Research Approaches

2.8.2.1 Data Flow Display by the ReactJs

Facebook has developed ReactJs in JavaScript. That is a frontend web application and JavaScript library used as a graphical interface to display data. It has reusable components, which mean it can accept different arbitrary inputs and then show a React component as an output on the screen. Scalable framework, reusable UI components, stable code with regular updates are just some of the functional characteristics of ReactJs that make it an efficient interactive web app for users.

2.8.2.2 Scholarly Sources and Search Strategy

2.8.2.2.1 Data Resources

The results were collected from well-known academic research sources such as:

- ScienceDirect
- ACM Digital Library
- Springer Link
- IEEE Xplore Digital Library

2.8.2.2.2 Search Term

After trying different search terms, we reached desired results by this search terms about using the ReactJs functionality in data visualization.

("ReactJs "AND" DATA VISUALIZATION" AND "SENSOR" AND "MACHINE LEARNING" AND ("TIME-SERIES DATA" OR "JSON") AND "ENERGY" AND ("WATER" OR "ELECTRICITY") AND" CONSUMPTION")

2.8.2.2.3 Search Process

Phase 1. Among 41 studies found, we tried to select the papers relevant to our study's aims with a brief overview. Then we transferred articles with the relevant topics, abstracts, or introduction to our research to the Zotero.
Phase 2. We scanned all the relevant studies from the previous phase that we accessed to examine the obtained resources. Therefore, we reviewed a few numbers of studies to get closer to useful information and data.
Phase 3. The extracted data from relevant studies were transferred to the Excel sheets for quick access.

2.8.2.2.4 Criteria as a Selection Tool

Table 2.5 outlines our criteria for selecting and choosing resources related to the ReactJs studies.

✓ Inclusion Criteria	⊗ Exclusion Criteria
✓ The studies which investigated the ReactJs functionality ✓ The papers which referred to the sensor networks in smart cities ✓ Focus on JSON data visualization	⊗ The studies before 2009 ⊗ Discard papers in a language other than English ⊗ Thesis, reports, books ⊗ The studies that are not defined as reliable (such as web pages) ⊗ The inaccessible studies

Table 2.5. Inclusion and Exclusion criteria for ReactJs studies

2.8.2.3 Research Methodology (SLR)

2.8.2.3.1 Search Execution

To prove the ReactJS capabilities, we applied the SLR methodology to find papers with a similar context to our studies. Therefore, we achieved several efficient and persuasive studies to use ReactJS to visualize the JSON data we provide in section 3.2.

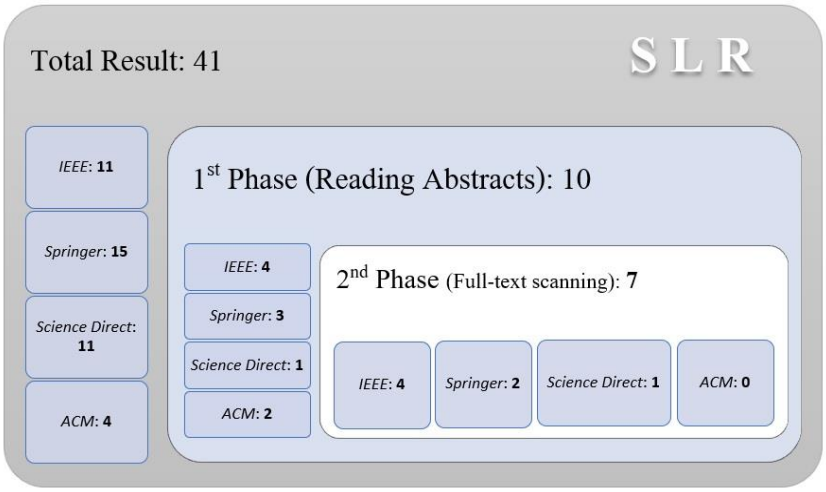


Figure 2.17. SLR methodology for ReactJs studies

Chapter 3 Related Work

3.1 Overview

Our research is based on energy consumption management with a focus on water consumption management. It should be mentioned that first, we considered some studies about power or other types of energy consumption management with machine learning approaches as the general scope of this study. Then we conducted further research to focus only on machine learning capabilities in predicting the amount of hourly water consumption.

HydroSense study by Froehlich et al., 2009 [53] provided a simple and low-cost solution for estimating a home's water consumption using a non-intrusive sensor in every valve. They continuously analyzed water pressure when every valve is closed or opened (especially in the kitchen sink, toilet, and shower). They measured the pressure of water waves in valves emitted to sensors, and they also tried to calculate the amount of water used in a piece of water infrastructure based on how large the pressure drop was. This study implemented a HydroSense sensor at any outlet or usual water spigot. It used machine learning approaches to evaluate the labeled data collected by sensors related to 10 houses in 4 cities with different plumbing systems, ages, and styles. They applied linear regression to analyze the stream of residual trials in the test set and the cross-validation technique and achieved 97.9% accuracy.

Somontina et al., 2018 [54] investigated a non-intrusive and single-point method to monitor the monthly water consumption of a house. The purpose of their study was summarized in 3 sections: measuring home water consumption in real-time with a non-intrusive sensor, identifying fixture and faucet, and calculating the cost and volume of household water consumed in one month. RF as a machine learning algorithm was applied in this study to measure the amount of water consumed, that the accuracy presented by this algorithm was 92.9%.

One review based on using the state-of-the-art application of machine learning methods was conducted by Seyedzadeh et al., 2018 [55] to predict buildings energy consumption. They hypothesized that increasing the energy efficiency of new buildings could reduce the level of global warming risk. They investigated machine learning models such as ANN, SVM, Gaussian-based regressions, and clustering for finding the most suitable model to improve the energy performance. They found out that ANN has been vastly utilized in energy prediction, and it is efficient for data relates to temperature and humidity prediction. Although ANN is an efficient tool for energy modelling with reliable forecasting in buildings, the structured energy modelling by ANN cannot support a local smallest problem because it requires the precise choice of sampling from samples, the precise choice from network structure, and accurate setting of parameters. In contrast, SVM is powerful to create a model with a few samples and parameters. Gaussian process (GP) and SVM, through few parameters, can present acceptable performance, and clustering as an unsupervised learning method can classify the buildings based on their different features instead of solely utilizing their type or structure.

Another study by Sornam et al., 2018 [56] presented information about the importance of data mining approaches to create greener and smarter structures in smart cities. This research reviewed different data mining algorithms to predict energy consumption to have a greener environment and smarter buildings. For data analysis, they followed seven steps that one of them was data mining to discover an efficient pattern. They stated that 1) SVM can be efficient for classification of non-linear and linear data, 2) Decision Tree is a robust algorithm for rules' extraction from data collected by sensors, 3) Neural Network is used for commentary improvement of the trained network, 4) meta-algorithms like ensemble method through the ability of the combination of some machine learning algorithms just in one model can decline variance and enhance the level of accuracy in predictions, 5) Sliding window can be a useful method for analyzing the flow of data collected by sensors, and as a result, the data mining with the utilization of machine learning techniques can tackle huge dataset.

Fernández et al., 2016 [57] studied the role of big data on the management of energy efficiency in smart homes. They stated that according to a 2011 European Commission statement, economic planning could reduce the amount of energy consumed in buildings by customers by up to 40 percent of total energy consumption. In this study, the role of different machine learning algorithms was investigated to find the most suitable model for managing big data which predicted users' weekly energy consumption. Their purpose was to examine various machine learning techniques on raw data produced by the smart home to collect useful information for energy efficiency enhancement. Their study's structure was shaped on four modules; one of these modules was machine learning which included three sections 1) applying a supervised classifier, clustering techniques, and some weighted algorithms with 74 % accuracy for recognition data used by each device 2) ability to investigate and process the recorded data about the users' energy consumption helps to specify consumption patterns to give some suggestions to other users who are acting like this to modify energy consumption habits, and 3) prediction of this energy consumption pattern with 90% accuracy by using machine learning techniques. The experiment conducted in this study explained how the big data process could manage the various huge volume of datasets by machine learning support to categorize, store, and analysis the information based on needs. Furthermore, the techniques and methods examined for using and evaluating data generated in smart homes are also generalizable to other smart environments similar to the smart homes in the project.

The study by Vafeiadis et al., 2017 [58] applied machine learning approaches for occupancy recognition about the data collected by smart meters such as water or power consumption sensors in an internal environment. Their goal was based on an experiment on the water and power sensors dataset to determine occupation status by expressing two states, such as presence or absence. In this experiment, the amount of water or electricity used by residents is considered as a measure of occupation because the consumption showed residents are in the building or not. Some machine learning algorithms which were used in this study are Decision Tree (with AdaBoost) with the best accuracy around 80.94%, SVM-POLY (the Polynomial) with 79.83% accuracy, SVM-RBF with 80.06% accuracy, Random Forest with 80.23% accuracy, and ANNs (the backpropagation algorithm) with 80.21% accuracy. As a result, they stated that using machine learning abilities and techniques can have satisfactory results for dealing with the occupancy recognition challenge.

Li et al., 2009 [59] investigated the energy consumption of 59 households in China. Their goal was based on using machine learning algorithms to predict energy that was consumed by residents annually. They considered 50 houses as sample sets and the rest of the nine houses as a testing sample for applying three neural networks models such as GRNN (General Regression Neural Network), BPNN (Backpropagation Neural Network), RBFNN (Radial Basis Function Neural Network), and SVM model. The experiment conducted in these 59 houses showed that both machine learning models were efficient for predicting electricity consumption. The prediction accuracy by GRNN was better than other Neural Networks models, and SVM had the best with an error of less than 10%. As a result, they stated that the "*structure risk minimization (SRM) principle, which is the most outstanding feature of SVM, is implemented to minimize the upper bound of the generalization error rather than the training error, which is applied in neural networks*" [59].

Zhao et al., 2010 [60] utilized Energy Plus software to model energy consumption for several buildings. They implemented the machine learning techniques like the SVM algorithm and the Gaussian RBF Kernel to create a prediction model. Their target was to save energy by predicting energy consumption based on complicated parameters in buildings. Some of these important involvement parameters of each house included Cooling type, Air infiltration, Thermal Zones, Structure, Duration, some facilities like light or water heater, Heating type, People, Fenestration surface, Building Shape, and Location. In fact, in this research, they examined several SVMs in a simultaneous execution, the RBF for the training phase and SVR. They applied the Pisvm tool to increase the speed of the model training phase on a huge dataset. They conducted three research that first tested the ability of SVMs techniques to predict and analyze the energy consumption by every building. Then they investigated several buildings considering their structural details, and finally, in the third step, parallel SVMs were examined on a huge dataset of several different buildings. They stated that although this assessment based on simulation building with the historical dataset is not enough and investigation with real-time energy data in different kinds of houses could be better to have more accurate and better results close to the actual situation; this experiment proved that SVMs and SVR had the best results for this study. Indeed, in machine learning techniques for improving accuracy, the number of collected data should be increased. Following that, the time for analyzing this huge dataset will increase, and the training phase takes a long period for exploring all the buildings' datasets. To enhancement this challenge, they chose to decrease the learning time by using parallel SVMs execution. The SVM was the best approach to analyze a huge energy dataset by reducing the average training sets, and SVR had a good result for evaluating the amount of energy consumption by selecting the most influential parameters in their study.

One Review of sustainable and renewable energy was conducted by Zhao et al., 2012 [61] that investigated the effect of some factors on energy consumption prediction in buildings. Their goal was based on this issue that some factors such as Heating, ventilation, and air conditioning system change their behavioral function, occupancy, and lighting create complicated conditions for the high level of accuracy in the energy consumption prediction. Therefore, their reasearch investigated previous studies that provided some AI models (ANN and SVM algorithms), simple or complex engineering models, and statistical methods. After evaluating the studies about energy consumption

prediction, they presented the results of their study in one table, which included some features such as the complexity of a model, the simplicity of applying, the speed of performance, required input parameter, and the level of accuracy.

Techniques	Results
Simplified Engineering	This technique includes a high level of accuracy and complexity. Moreover, it is easy for applying and needs simple parameters as the input.
Elaborate or complex Engineering	However, the level of accuracy is remarkably high. The level of complexity is also high with low running speed, and it is not easy for applying and needs input with detail.
SVM	Although this technique is not easy for applying with a low speed of running, with a high level of complexity, and utilizes historical data, the level of accuracy is remarkably high.
ANN	ANN is not easy for applying with a high level of complexity and utilizes historical data parameters as the input but has high running speed with a high level of accuracy.
Statistical	This technique is easy for applying with fair accuracy, fair complexity and uses historical data parameters as the input. Moreover, it includes high running speed.

Another research in 2009 has investigated the effect of using SVM to improve energy efficiency by predicting the energy consumed through a cooling load in buildings. Hou et al. [62] examined the ability and possibility of SVM and neural network algorithms on a real HVAC (Heating, ventilation, and air conditioning) system in Nan Zhou (China) for predicting the cooling load. The challenging part of HVAC parameters assessment is that the parameters are different in type and time. HVAC is a non-linear system in which all these items should be considered for the cooling load assessment in every house. Based on this goal, they assessed the SVM capability by considering two important factors, C and ε, conducted by a step-by-step search method based on the RBF Kernel. Implementing the SVM showed that δ2 and C have the main role in the output of this algorithm, so they should be chosen with the ideal value because the SVM is affected by the value of both. Their study had an excellent result solely with 4% for the mean value of the error.

Ahmad et al., 2014 [63] conducted a review about applying the SVM and ANN algorithms to predict electricity energy consumption in buildings. They stated that these two methods had remarkable results in the energy consumption prediction in previous research. The combination of these methods can be another approach to increase the level of prediction accuracy. Therefore, their goal was to compare the result of this combination which includes GLSSVM (Group Least Square SVM), with some single algorithms like ANN, GMDH (Group Method of Data Handling), and LSSVM (Least Square SVM). Their research showed that based on GMDH and LSSVM's ability to analyse and predict with the most accuracy, the combination of these could manage every nonlinear issue and increase the level of time-series prediction accuracy in energy consumption for buildings.

Benedetti et al., 2016 [64] examined an innovative methodology about using adaptive algorithms and ANN in energy consumption. One methodology for several goals such as controlling energy consumption by automation system activation, maintaining the model's accuracy implemented over time, providing two ways to re-enable automatic recovery, and careful evaluation of retraining ways over time, that all the methods are based on ANN algorithms applying. Continuous analysis and updating of data are very important items in managing energy consumption, which requires workforce, effort, and time. Although to deal with this issue, machine learning techniques can be an efficient approach to resolve the complicated problem, the reliability and accuracy of the machine learning model face a downtrend over time. Therefore, they examined an innovative approach to improve this defect to utilize one automation system for controlling and estimating energy consumption exactly. In three parts, first, they investigated and trained three various structures of ANN to choose the best structure for creating an efficient tool for controlling energy consumption. And finally, they provided a method to estimate how long it was possible to maintain the model's reliability and accuracy with a minimum of data collected because the massive dataset is not always accessible. They stated some reasons for decreasing the accuracy in models for energy consumption were based on an unexpected alteration of the external situation, behavior changes in the system’s structure that happened by itself and taking a long duration after training. Their results proved that their methodology included the advantages that the ANN structure made was the most efficient structure for their purpose. With the ability to automate the operation, this methodology could investigate the defects and identify the model for detecting a sudden problem.

Another ANN model applied by Wong et al., 2010 [65] investigated for energy consumption in office buildings. They examined this machine learning algorithm to evaluate the amount of energy consumed by devices that were worked with electrical energy every day. In this study, Energy Plus was applied to simulate energy consumption and create the database structure. Nine input variables were considered for use in the ANN model divided into three categories: one parameter for specifying the day of the week, four-parameter for outdoor climate situations, and four-parameter assigned to the office building envelope. Four nodes were considered at the external layer of the ANN model for evaluating the amount of energy consumed by chilling devices, heating system, lighting, and every device that were worked with the electrical energy. To assess the hydrological model's ability to predict energy consumption, they investigated the results of the Nash–Sutcliffe model efficiency coefficient that were 0.994, 0.940, 0.993, and 0.996. Since the best returns occur when the NSE is equal to one of the nearest digits to one, they obtained the best result in this study that showed the perfect fit of the model with the data collected.

Brentan et al., 2017 [66] proposed a hybrid model for online urban water demand forecast in the short term. This model consisted of two components: an offline model using SVR as a prediction base and an online Fourier series process for adjusting prediction deviation. Using an SVR model could describe the general behavior of the daily demand, but not the pattern at the peaks (max or min values). Besides, by launching the Adaptive Fourier series (AFS) on the SVR model, which updates the forecast result in about real-time, they could improve offline model predicted values. In this research, a Grid Search Method (GSM) was developed to tuning the hyperparameters (C and ϵ).

Then the deviation between predicted values and observations was computed. Since human behavior is different on various days of the week, water consumption is different on weekdays compared to weekends. Also, the holidays have different patterns. This was taken into account in this paper by including the calendar information.

A sample of 570 days of data was used for the training phase, and then the data of the following 400 days were taken for prediction. Afterwards, a new input of 140-day samples of water demand data was used for the model validation process. Finally, for testing the model performance, fresh 30-day data was used. Furthermore, the cross-correlation between weather variables (temperature, air humidity, wind velocity, and rain) and water demand time-series was investigated, which showed that the correlations between these variables exist. The MAE% (Mean Absolute Percentage Error), the RMSE, and the R^2 were used to assess the final output of the proposed model (SVR prediction and the AFS). The largest deviations were observed at the maximum and minimum points. In the end, the comparison between the result of the suggested model with the real water demand showed that the AFS was successful in forecasting the deviations, and it improved the results of the offline model. Here, constant CPU time for the AFS model (16.5 seconds) and the algorithm agility allowed near real-time prediction. The authors claimed that this hybrid model would be a major tool for water utilities because the online feature supports the performance of water distribution systems (WDSs), leading the operators to execute efficiently and save water.

Another research on forecasting hourly urban water demand was done by Herrera et al., 2010 [67]. They used a series of predictive models for forecasting water demand. The data they used was non-linear time-series data. Therefore, they have applied several machine learning algorithms suitable for non-linear predictions such as ANN, PPR (Projection Pursuit Regression), MARS (Multivariate Adaptive Regression Splines), SVR (with RBF Kernel) and Random Forest. The ANN model consisted of one hidden layer in a feed-forward neural network and a back-propagation process. Furthermore, they have tried to launch a simple heuristic model as a baseline for comparing other more complex algorithms based on the weighted demand profile arising from the exploratory data analysis. This model consisted of two parts: the first part reflected the regular behavior while the second part adjusted this early prediction. In this work, besides water consumption information, the values of weather variables such as temperature, wind velocity, millimetres of rain, atmospheric pressure were considered. To predict future demand using past data, they have proposed a procedure by designing a Monte Carlo simulation to evaluate the performance of predictive models. For evaluating models' performance, RMSE and MAE metrics were used as well as two non-dimensional metrics being more responsive to systematic errors, named the Nash–Sutcliffe efficiency and a modified version of Nash–Sutcliffe. Moreover, they exploited the advantage of using Monte Carlo simulation to ensure that estimations of metrics are unbiased and various model construction strategies were considered to carry out the structure of data shifts. In this research, the outcomes showed that accurate results were achieved by the SVR model, followed by MARS and PPR, and Random Forest, while the performance of neural networks was disappointing.

Chen et al., 2016 [68] also studied urban water consumption prediction by implementing a conjunction model (named W-RFR) of multiple Random Forest algorithms composed of Random Forest regression (RFR) and wavelet transform. The First Raw

data were divided into low and high-frequency components with DWT (Discrete Wavelet Transformation). Then the RFR was applied using each sub-series. Finally, the summation of all predicted time-series was considered as the final output. MAPE (Mean Absolute Percentage Error), R (Correlation Coefficient), TS (Threshold Static), and NRMSE (Normalized RMSE) were used for evaluating the performance of the model. It is concluded that the W-RFR model could forecast the daily urban water consumption more accurately than the single model and capture the basic dynamics of water consumption.

Zhang et al., 2018 [69] proposed an effective method in forecasting water table depth to help make management decisions. In this research, 14 years of time-series data were used. Then a Lasso regression was applied to the data to select important variables. In this way, only five variables were selected from the original data. The proposed model was employed to forecast water table depth from the input variables: monthly water diversion, evaporation, precipitation, temperature, and time. In this study, the authors developed a two-layer LSTM-based model that contained an LSTM layer with another fully connected layer on top of the LSTM layer. Also, a dropout method was applied in the LSTM layer. After that, the results from this proposed model were compared to the traditional FFNN (feed-forward neural network) model and a Double-LSTM model. The 14 years of data were divided into two sets: 12 years as the training set and the rest as the validation set. In this study, RMSE and R2 scores were used for measuring performance. By comparing the results gained from the proposed model with the results obtained by launching the FFNN model, this study showed that the proposed model could learn past information well and achieve acceptable scores. Moreover, a comparison of scores gained by the Double-LSTM model and the proposed model proved that the proposed model had more robust results on time-series and the dropout method successfully prevented overfitting. Therefore, the authors concluded that their model could play an essential role in studying water table depth prediction, specifically when obtaining the data is difficult.

Nasser et al., 2020 [70] presented a system for both data acquisition and prediction of short and long-term water consumption. In this context, the designed system consisted of two main parts. In the first part, and for data acquisition, a smart water meter was used to send data to the Cloud, and a solution for real-time data gathering was presented. The aggregated datasets were analyzed based on machine learning techniques to forecast water demand in the second part. They modelled three different LSTM architectures with one, two, and three recent time steps to predict the next time step. The results showed that the LSTM architecture with three inputs performed better than other architectures. Moreover, SVR (with a Gaussian RBF Kernel) and RF models were launched on similar datasets. The authors stated that the reason to select SVR and RF is that SVR is a popular method in water demand prediction problems; at the same time, RF is a successful algorithm in time-series prediction problems. The MAE, RMSE, and MAAPE (Mean Arctangent Absolute Percentage Error) accuracy metrics were used as evaluation criteria. According to the results, the LSTM with three inputs outperformed the other LSTM architectures as well as the SVR and the RF models.

Dufour et al., 2016 [71] studied heating and hot water consumption predictions in two levels: anticipation level and reactive level. The first one proposed a forecast for every hour in one day, while the latter provided the forecast for the next hour. The supervised

learning methods were utilized, composed of tree ensemble predictors. The model consisted of 30 decision trees, each trained on a different randomly selected subseries. Therefore, there existed an ensemble model of different decision tree models as output models. The majority voting was used for the final prediction. In this paper, the authors have focused on predicting hot water consumption by the heating and hot water consumption data. According to the results, the proposed model performed well on this dataset. For the heating prediction model, several variables were included in the prediction of the heating, such as the hour, the heating consumption the previous hour, the maximum heating consumption the previous hour and the solar radiation predicted. For forecasting the hot water consumption, the determinant variables were the hour, the hot water consumption the previous hour. The prediction results showed a correct estimation of about 91% for the given data, which showed an acceptable prediction in this context.

Another approach to hourly water consumption prediction was made by Candelieri et al., 2015 [72]. They followed the prediction of hourly periodicity water demand in the short term by a data-driven, self-learning approach. This approach was composed of two sequential phases. In the first phase, the time-series clustering was done to define a limited set of general patterns. The clusters were identified according to Calinski-Harabatz and Silhouette measures. After that, in the second phase, the SVR models for each cluster was launched to gain one prediction model. Thus, several SVR models were performed for prediction. Each cluster was assumed as a separate dataset in this stage, and all the SVR algorithms had the same input data. This approach was proposed at both aggregated and individual levels. In this study, the MAPE was used to measure the performance of the model. In this work, the size of the dataset was limited and was a barrier in identifying patterns and seasonality of data. The authors concluded that although more data is needed, this approach is reliable.

Ju et al., 2014 [73] used SVR models to forecast total water requirement. Two solutions were proposed forecasting by SVR and ARMA model and the other forecasting by only time-series analysis (ARMA). In this paper, the authors presented four different models via SVR. Then, they measured the performance of models by the MSE criteria. Correlation analysis was used to remove some determining variables with less influence on the total water requirement to use fewer determining variables. In the first SVR model (with 29 determining variables), the total water requirement was defined by the determining variables of the same year. The second model (with 29 determining variables) was the total water requirement defined by the determining variables of next year. The third model (with eight determining variables) was that the total water requirement was defined by fewer determining variables of the same year and using correlation analysis. The fourth model (8 determining variables) was also a model of the total water requirement defined by fewer determining variables of last year and using correlation analysis. Comparing the results of these four models showed that the first model had the best results. So that, they selected the first model as their predictive model, and the predicted values were compared to the results from the ARMA model directly. They concluded that both models could be a reasonable basis for predicting total water requirements on the given dataset.

Another study about making a new system to predict hourly water consumption was done in 2019. The system designed by Bejarano et al. [74] took historical water consumption data as input and delivered future water consumption forecasts as output. The

system consisted of two parts: a data pre-processing part and a prediction part. In the prediction part, two models were launched: GCRFs (sparse Gaussian Conditional Random Fields) and LSTM. Both algorithms used similar datasets as train and test sets and predicted water consumption in 12 hours (in the future) from the data of the past 24 hours. The authors claimed that an attractive characteristic of the designed system was that it only needed data of the past 24 hours, which made the proposed system computationally efficient during test time. Moreover, in this study, the performance of these two algorithms was compared with two baseline models: ARIMA (Auto-Regressive Integrated Moving Average) and linear regression. According to the results, both algorithms could learn and capture non-linear dependencies, making the prediction more robust. Also, the performance evaluations showed that the designed system outperformed the baseline model's performance (ARIMA and linear regression models).

Walker et al., 2015 [75] presented an ANN-based model to predict the hourly water consumption of households. In this paper, the authors followed the highest possible accuracy with fewer possible inputs. Thus, a mix of actual and statistical values of domestic water consumption was used as inputs. In this context, for predicting water usage at time t, an initial configuration consisting of three inputs were considered: the water consumption at the previous timestep ($t-1$), the average consumption during the last seven days, and the current hour of the day. The predictive model consisted of a one hidden layer ANN algorithm, and the number of its neurons were defined by experiment to minimize this number. Since the study's goal is to predict the water usage at the next time step, the output layer contained one output neuron with a sigmoid activation function. Besides, an algorithm known as EA (evolutionary algorithm) was used for optimizing network weights. Also, a leave-one-out cross-validation with eight folds was employed. Although the model could follow the data trend, it failed to match precisely at the peaks. They tried to improve their model performance by adding standard deviation as new input, and in another try, they used only real historical values. The results were similar in all cases, and the model could not precisely predict the peak consumption. The authors claimed that this inaccuracy was because of noise in the dataset, and the results were affected by such a noise. Based on our perception of this study, a lack of accuracy happened in timesteps when the water consumption was considerably high. This study shows that applying efficient methods of removing noise is essential in water consumption studies because the water consumption data is naturally noisy.

Romano and Kapelan, 2014 [76] presented a methodology for adaptive WDF (Water Demand Forecasting) based on the water demand analysis. They designed a data-driven and self-learning DFS (Demand Forecasting System) by employing EANNs (Evolutionary ANN). In this study, the two WDF methods and four scenarios were launched to assess the DFS self-learning ability. The DFS was composed of four main modules: the data pre-processing module, the ANN optimization module, the ANN building module, and the WDF module (EANN). The first module provided the raw data, and then the ANN optimization module automatically chose the optimal ANN input configuration and ANN parameters. Finally, the ANN building and WDF modules were applied to launch the EANN model and make predictions. In this research, two different approaches were considered: the ensemble EANN (eEANN) and the recursive EANN (rEANN). In the first one, multiple models were launched parallelly to predict demands for different hours of the day. In contrast, one model with a fixed horizon (for example, one hour)

was used recursively, in the latter. Moreover, four scenarios were examined; In scenario 1, the ANN optimization module was used with launching the DFS updating weekly. In scenario 2, the ANN optimization module was applied without performing the DFS updating weekly. In scenario 3, the model performed with launching the DFS updating weekly but without the ANN optimization module. Lastly, none of the ANN optimization modules and the weekly DFS updating module was employed in the fourth scenario. The results showed that the proposed framework had good-quality predictions with less possible human involvement. It was also observed that the ensemble EANN (eEANN) performed slightly better than the recursive one (rEANN). However, it should be noticed that rEANN still had relatively good results, and its implementation required little effort. Thus, it could be considered a useful model. The authors concluded that they proposed a generic model that could be used at different horizons (short or long-term) and with varying periodicities and the possibility of adding more variables. They also stated that this methodology could be applied in other water demand forecasting studies due to its generic characteristic.

An ANN-based model to forecast the residential water end-use demand was developed by Bennett et al., 2013 [77]. To achieve this, first, they tried to define the main influencing factors on residential water demand. Then, an ANN model was employed, and the results were evaluated by some criteria such as RMSE, coefficient of determination (R^2 score), the Absolute Relative Error (ARE), Average Absolute Error (AAE), Mann–Whitney Wilcoxon (MW) P-value. In summary, the proposed model was utilized to forecast consumption in these categories: toilet demand, clothes washer demand; shower demand; dishwasher demand; tap demand; and total internal demand. In the proposed methodology, three ANNs were employed: one radial basis function network and two feed-forward networks with a backpropagation approach, the activation function of a hidden layer was Sigmoid activation, and in the output layer, a linear activation was used. According to the results, all the models except the bath demand were producing a moderately accurate prediction. It is observed that these categories were responsible for more than half of the observed variance: the dishwasher demand, the clothes washer demand, and total internal demand. This study demonstrated that using ANN-based methodology was suitable for producing residential water demand end-use prediction models. Also, it was shown that the proposed model could be used in water demand reduction retrofit programs.

Tamang and Shukla, 2019 [78] explored the water consumption for dairy plants to optimize the water demand forecasts. The water usage from several units was used as the input data for the SVR model with RBF Kernel. The SVR algorithm was used because both the classification and regression models can be launched. In this study, 85% of the samples were used in the training step and 25% in the testing step. They evaluated the result of their model by using three metrics as R^2 score, MSE, and RMSE. The predicted values were compared to the real consumption values, and it was observed that the predictions by the SVR model were very close to the actual values. SVR performed well on small train and test samples and outperformed other statistical algorithms requiring more datasets. Also, it worked effectively compared to a neural network.

All in All, by the literature review, we perceive that various algorithms were exploited to predict water consumption/demand. The study of water consumption prediction is

a branch of time-series forecast modelling. The regression methods have been used to predict time-series as the basic models for a long time. However, by the evolution of machine learning algorithms, the comparisons between the performance of statistical-based models (like regression methods) and the performance attained using the machine learning algorithms were done, which proved that typically the latter had better results. The time-series data is non-linear and complex, and machine learning methods are capable of modelling non-linearity. In this context, it should be noticed that although SVR algorithms and Tree-based algorithms were very successful in this field, the simple ANN models have not obtained superior results compared to other machine learning algorithms and even the traditional statistical methods. As we examine in the related studies, despite successful modelling by machine learning algorithms, there is still a gap, which is associated with the fact that the models mentioned above cannot capture the sequential dependency between samples; therefore, we can see that the recent studies utilized the RNN models like LSTM which showed the best achievements. Moreover, it is discovered that using hybrid models consisting of a combination of different types of algorithms as well as ensemble models was a useful approach to gain accurate predictions. It should be noted that according to our investigations about water consumption management, we did not observe such research in Norway to have been done on examining hourly water consumption prediction by machine learning and deep learning algorithms.

3.2 The ReactJs Related Work

Table 3.1 briefly describes what others have done about applying ReactJs to visualize the JSON dataset in their studies.

Study Title	Authors	Journal	Application
IoT Personal Air Quality Monitor	Sean Mc Grath et al. 2020	IEEE	In this study, the structure was based on a LoRa network, and Google Cloud stores the time-series data that are JSON. They used the ReactJs ability to display the Heatmap component. The air quality data from the Cloud storage was accepted by the refillable user interface's ReactJs component as an arbitrary input. And the output was the appropriate scaled Heatmap point displayed by the ReactJs [79].

Study Title	Authors	Journal	Application
Selena: a Serverless Energy Management System	Florian Huber et.al	IEEE	They tried to record and display energy usage from various energy resource data to find out the Co2 emission level. The ReactJs as the main JavaScript library was chosen to visualize existing resources. For example, users can select the desired location like a specific climatic zone or resource then visualize the data flow about humidity, temperature, and so forth [80].
A LoRa Mesh Network Asset Tracking Prototype	Emil Andersen et al. 2020	IEEE	Due to the useful feature of the LoRa in sending data over long distances, they made a tracking application. Therefore, they applied the ReactJs with Hook-based architecture to visualize the data related to the location and tracking, including geographic visualization of nodes' position [81]. In this study, coordinate data was in JSON format.
Low-Cost Smart House Implementation with Sensory Information Analysis and Face Recognition	Serik Zhilibayev, et.al	IEEE	They used the ReactJs as a web interface for JSON dataset visualization to display the data flow from sensors located in every room about the level of gas, light, temperature, and humidity [82].
A framework for using calibrated campus-wide building energy models for continuous planning and greenhouse gas emissions reduction tracking	Shreshth Nagpal et al. 2019	Science Direct	In this study, the web application structured by ReactJs receives and displays the JSON data. Indeed, when the user selects a specific building, the data present the annual energy usage level in every building. Changing the color gradient shows the type of building and the amount of energy [83].
Occurrences Management in a Smart-City Context	Mário Ferreira et al. 2019	Springer	This study used recorded urban events data like water, garbage, energy, roads, traffic, environment, forest, and so forth to anticipate the urban events in the near future by using AI algorithms and machine learning abilities. They created a web application to warn people about likely events in the city. The ReactJs as a graphical visualization tool was selected considering the reusability of some code to depict an overview of the urban events [84].

Study Title	Authors	Journal	Application
IoT-Based Air-Pollution Hazard Maps Systems for Ho Chi Minh City	Phuc-Anh Nguyen, et.al. 2019	Springer	This study included gateways that have communication with sensor nodes network, and after collecting data by the sensors, gateways sent the data to the Cloud using the internet connection. The ReactJS displayed the data collected about CO, temperature, dust, CO2 concentration to visualize the amount of these suspended particles in the air for users. Therefore, the users can select a specific gateway to access information about the level of air pollution in different points of the city on the map [85].

Table 3.1.The summary of the ReactJS studies

It should be mentioned that, although we raised the issue of the use of ReactJs, we did not apply it in our study because the machine learning part was very time-consuming, and our focus was on this issue. Another reason for not using ReactJs was that the suggestion about the ReactJs in this study was just a remarkable road map for other research that would work on developing an efficient tool for visualizing the energy data like water consumption data in the JSON data type.

Chapter 4 Methodology

4.1 Strategy & Analytic Approach

It is important to specify the goals of the project and the project's context. To achieve the goals, we should use an efficient strategy. This strategy helps us identify problems well, leading us towards appropriate answers for the issues expressed in the study. Therefore, we utilized the Foundational methodology structure [86], [87], the most common model in the data science area. The data science methods are a set of phases that can be repeated to achieve the best result. An analytic approach is chosen based on the type of questions. These steps work based on an ideal analytical approach to discovering the most accurate results and answers. Figure 4-1 shows our study road map based on this methodology.

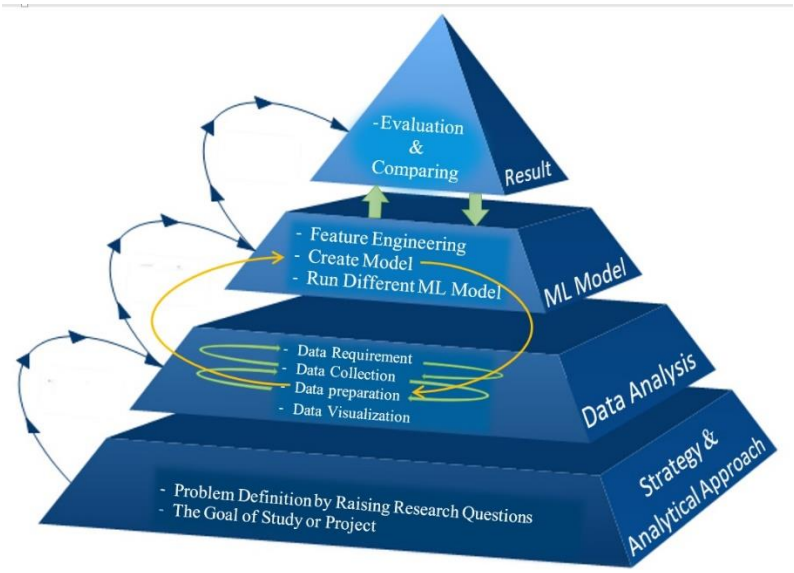


Figure 4.1.Our methodology structure based on the Foundational methodology structure

In this section, we express the problems by raising Research Questions in the context of machine learning techniques and statistics. Based on the research done in previous studies mentioned in the literature review section and answering the Research Questions, it was decided to use machine learning methods for data management and future prediction in the first phase of our study. This study was supposed to predict the future consumption of Sarpsborg city by hourly investigating the water consumption rate that, for example, we can apply a regression algorithm. Our objective was to examine the effect of some features based on hourly water consumption that we were producing by feature engineering in the data preparation section. Our approach can be used to compare some seasons like summer and winter, weekends, working days, and the national holidays as some features, along with the investigation of their effect on water consumption. We followed

the steps below to determine and predict the average hourly water consumption and achieve this goal by machine learning models.

4.2 Data Analysis

The first step is to consider the data structure as the main foundation of every machine learning process that can keep us on the right track to create the most efficient prediction model. In the previous phase, the chosen analytic approach determines the Data Requirement, Data Collection, and Data Presentation (Visualization) that we explain all these three items in Figure 4.2.

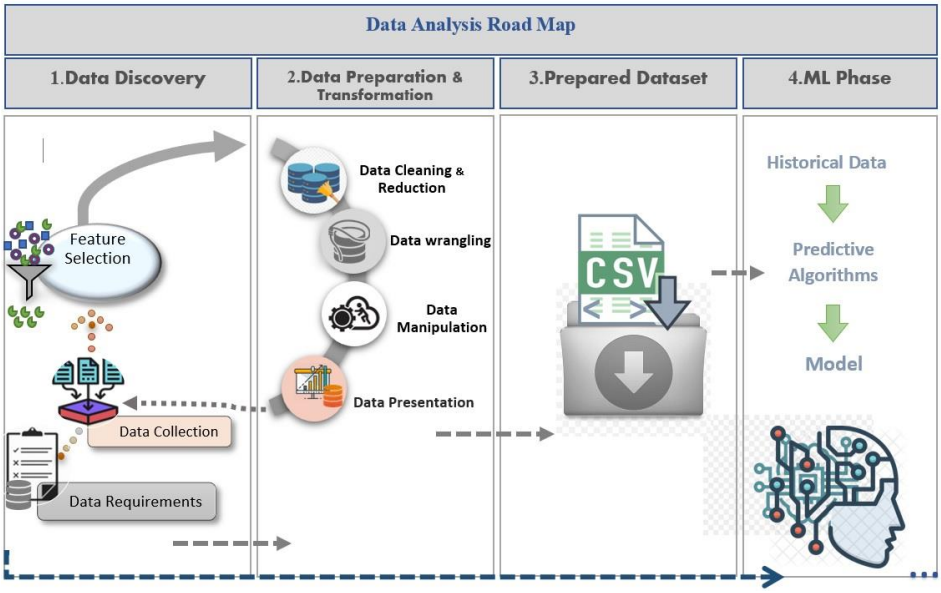


Figure 4.2.Data Analysis Road Map

4.2.1 Data Discovery

4.2.1.1 Data Requirement

We investigated what information we need, how we can access this information or data, what type of data we have access to, why we need this dataset, and who has access or responsibility to the recorded data. The answers to these questions based on the aim of our study can be expressed in the following. We paid attention to information and data format content based on our research and the analytic approach. In this project, we required information about the rate of hourly water consumption by every house or industrials' location based on the date and time. Although the Sarpsborg municipality had main responsibility and access to the recorded water dataset and they gave confidential private access to us for using available data on the Azure Cloud, almost always, we received the recorded data in the CSV format; therefore, we did not need to refer to the Azura regularly. As

the type of independent variable was categorical, we required this hourly categorical dataset to investigate the flow of consumption and predict the consumption of water at a specific time. Therefore, the required data are the same as the properties of the recorded water data, such as the volume or amount of consumption that is considered a feature. Then we gathered water data that includes some water consumption features such as measurement time, water consumption value, and the Device-IDs.

4.2.1.2 Data Collection (DEFA Structure)

After the data requirement phase, when we collected all data based on the study's requirements about water consumption prediction mentioned in the previous section, we had a clear perspective of what things we need to create a model based on the goal of our study. Based on what we want to predict and how we want to do that, we considered how much the data is available and how we could gain the data required. We spent time discovering the data and data resources to achieve the best result for the expressed problem in the study. Therefore, we gathered the data from different resources to classify and organize according to the recorded date and date of data collection. Once we collected and organized the data recorded, we searched related data to enter the analysis phase. This step helped us create one record from the past data, and by using data analysis, we could find iterative patterns in the past events that lead us towards future predictions. Indeed, we created our predictive model based on these data patterns to predict the future of water consumption in Sarpsborg city. It can be claimed that the efficient predictive model is built based on the quality of data collected and the data prepared. Figure 4.3 shows the DEFA process for data gathering, which is used in this study.

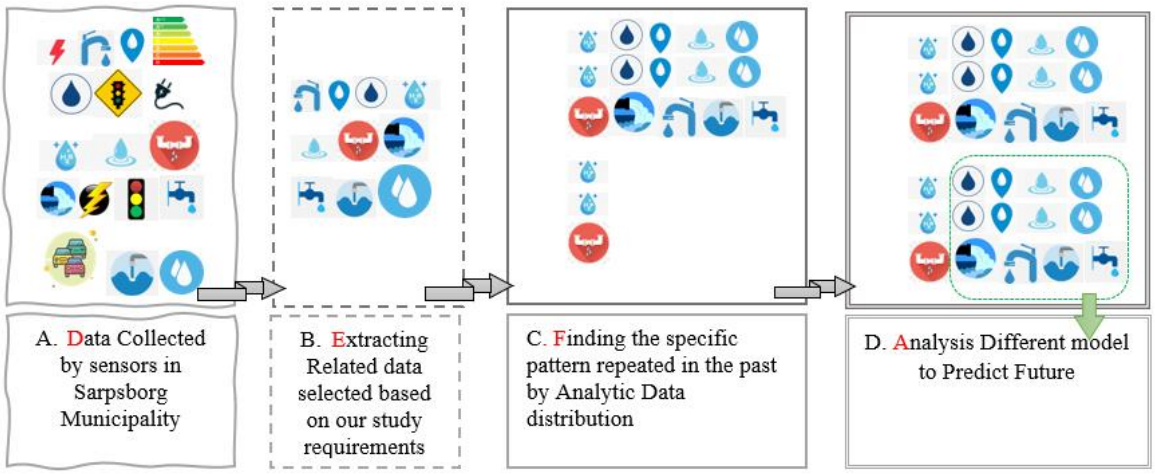


Figure 4.3.DEFA structure, the data collection process. (modified from [88])

If necessary, we revised the available data and tried to gain more data. For example, the data collected by the Sarpsborg municipality was the JSON data type collected by the LoraWan network sensor in 247 locations like houses and some industrial places. This dataset was recorded on the Azure Cloud computing service, and Data collecting has started from the sixth month of 2020 so far that the data was recorded hourly. In this

study, we wanted to investigate the effect of some seasons on water consumption. So, we needed more data about water consumption in the first and second half of the year to assess the impact of cold and hot weather on water consumption and collected more data from resources. In this situation, we went back to the data source, LoraWan Sensors, filled gaps, and collected the required data to improve our dataset to get accurate answers to the study questions. In the following steps with data analysis, integrating, reducing, and formatting, we could prepare the data to enter the next phase, the modelling step.

4.2.1.3 Feature (Variable) Selection

There are many input variables that we should choose the most relevant to our target variable. Extraneous input variables are time-consuming in the computational process, mislead algorithm processing flow, and distract its trend. Therefore, by reducing redundant, unnecessary, and irrelevant input variables, we can improve the performance and accuracy of the machine learning model. Also, we can decrease the computational process time. The focus should be on the attributes related to the expressed problem solution in the study. The more appropriate data with the model leads us towards higher accuracy in the prediction. Thus, we used the feature selection technique to prevent losing time and reduce the quality of model performance.

There are two types of variables known as target variables (supervised) and indirect variables (unsupervised). Target variables have a leading role in prediction, while indirect variables do not have any influential output variables in prediction. In our study, we considered and chose the date and time (Measurement Time), the amount of water consumption (Value), and the identification number of each sensor (Device-ID) as significant features or variables.

4.2.2 Data Preparation and Transformation

After data collection and data assessment, it is time to transform the raw data into data that algorithms can use in the machine learning model. This step is a wondering trip for understanding the context of data until future predictions. So, we should find the right data. The most important, complicated and time-consuming task in machine learning projects is to consider the data structure because collected data includes an unexpected range of values, missing values, incorrect combination of data, and so forth. We require data preparation to transform the raw data into accurate and acceptable data for machine learning models and future predictions. Therefore, data pre-processing is necessary for every machine learning project with data cleaning, data reduction, data editing, and data wrangling steps.

4.2.2.1 Data Cleaning and Reduction

Many factors cause data to include incorrect values. A critical step is detecting, removing defective data, and trying to fill the gaps. It includes removing one row or column of data or replacing new values. Filling in missing values by using a default value, detecting, and removing duplicate data records, covering private and sensitive information or data, and matching the data based on the requirements mentioned in the study are some types of data cleaning. Data Reduction is about removing missing values or null that means we

face empty values, question marks in cells, or empty cells. Also, we should identify anomalies that are unexpected values or containing errors and remove them. In our study, we removed some duplicate rows of data. However, we did not remove zero values because it could show the amount of water consumption that might be zero in that specific hour in a house or industrial place.

4.2.2.2 Data Wrangling

Data wrangling (Data Munging) is the process of manipulating data (Figure 4.4). It is the change in the nature of the data like mapping data, changing the data distribution, and changing the format of raw data to another form (for example, Categorical to Binary in our study by One-Hot encoding) that is more useful and worthwhile to be utilized in the analysis phase. Indeed, Data Munging is an operation of data normalization, data aggregation, format updating, and data visualization. There are various types of data, like Categorical or Continuous Variables: in our study, all independent variables such as Sensor ID, weekdays, weekends, … were categorical and required the one-hot encoding, while the amount of water consumption (the dependent variable) did not require the one-hot encoding.

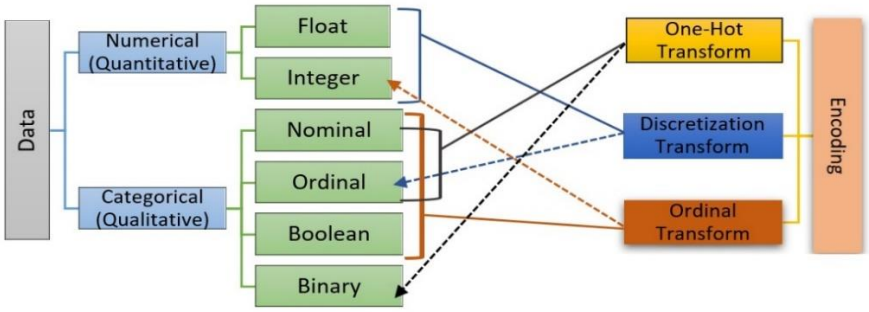


Figure 4.4.Data Categories and Transforms by Encoding

4.2.2.3 Data Manipulation (Feature Engineering)

Data manipulation or feature engineering can include dividing one column into several columns, removing some columns, data aggregating, adding new columns as new features, and so forth. Data enrichment is another type of data formatting that includes joining data, connecting data, and adding data to limit data with basic. With this method, we have a rich and valuable resource of a new dataset to improve the quality of the decision process in the machine learning predictive model. Based on the goal of our study, sometimes it is necessary to create new features. First, we had four original features (Device-ID, the Value, and the Measurement Time) in this study. We applied feature engineering by converting one of our features (the measurement time) into four features (year, month, day, and hour) as a new data frame. Also, we defined two new features that are weekends and weekdays. These were used in the training phase for considering the correct days in the water consumption prediction. For example, if the selected date for prediction was the weekend, only the weekends were chosen for the training phase, not the weekdays, and

vice versa. Finally, the final prepared dataset details to enter the training phase were based on Table 4.1.

An Overview of Final Prepared Dataset			
Samples	Features before Data Manipulation	Features after Feature Engineering	Length of Time-series
1.300.000 Rows	3 main Features:	8 Features:	2019-2020
	Device-ID, Value, and Measurement Time	Device-ID, Value, year, month, day, hour, weekends, and weekdays	

Table 4.1. The final dataset used in the Training and Testing phases

4.2.2.4 Data Normalization

For data quality improvement, normalization is an efficient technique. When there is a considerable difference between the values of features, the feature with a larger value intrinsically affects the prediction result. So, we apply normalization as one of the data preparation steps in machine learning to put all variables on the same scale. This scale adjustment is made without losing data or distorting the amplitude of each value. Also, this technique can be helpful for some algorithms like ANN and KNN that do not pay any attention to the data distribution. Thus, we used the normalization technique to put all our variables in the same range, optimize data integrity, and decline data redundancy.

4.2.2.5 Data Presentation (Visualization)

Data visualization or presentation is a simple display of trends and data patterns in charts, graphs, or tables. A good presentation of data leads us to correct interpretation of the relationship between data. This allows us to have a correct analysis of the data to predict the future. So, this is an important step in any project because the better the visualization of the data, the more data we can interpret and assess. Instead of looking through many rows of data, we can look at the summary of data in a chart or a graph. Visualization helps us understand the trend of data and transfer it simply and more understandable to others by a simple and clear picture of what is happening in a huge dataset process. There are different types of data visualization based on the data type. We should consider the effective factors on the data visualization, for example, the time factor, because time has a vital role in the trend of the dataset process, and the passage of time affects the dataset pattern. By considering the effect of the time factor, we determined how much water was used at defined time intervals. Also, there are various types of charts for data presentation or visualization like line charts, histograms, scatter plots, pie charts, bar charts, and so forth. We chose the related chart with the goal of our study. We exploited Heatmap to show the minimum MAE and R^2 score result using plot_confusion_matrix from Scikit-Learn library [27], and it helped us choose the best time point in both time intervals (Appendix A). To illustrate the water consumption prediction procedure based on the machine learning techniques, we used a cross-functional flowchart to describe different stages of how

they relate to each other. Figure 4.7 and Table 5.2 to Table 5.5 provide the results of our applied algorithms based on using hyperparameters. As a result, it sent us important messages about the data, which showed us the importance of this step.

4.3 Machine Learning Models

In this step, the actual machine learning modelling was done. We exploited different regression algorithms to find the most efficient outcome and express how the variables relate to each other and how some variables affect others. Furthermore, regression is a useful technique when we aim to predict a future value based on a new collection of predictors.

4.3.1 Predictive Modelling

For future prediction, we need a statistical model that predicts future behavior based on a dataset that enters as a new input. This statistical model is built by a mathematical approach which is called a predictive model.

4.3.1.1 Algorithms Selection

We used supervised machine learning models for our structured dataset to make a prediction model based on the labeled target variable. But before choosing proper algorithms, we compared some algorithms to choose the best ones. So, we applied different regression algorithms on our dataset to estimate and compare them based on the result of 4 metrics in our machine learning methodology (as we introduced them in the Background section), such as variance score, MAE, R^2, and RMSE.

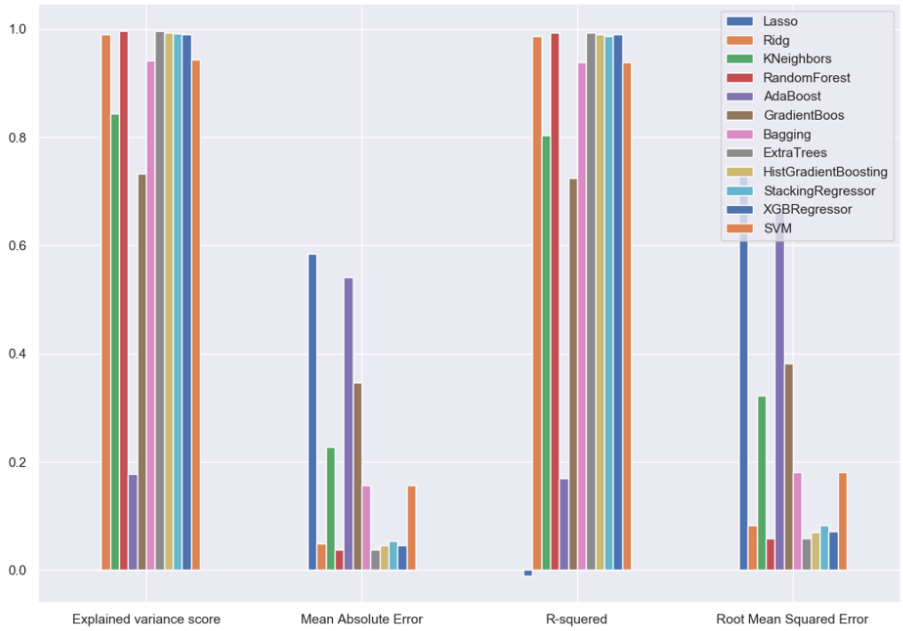


Figure 4.5.Initial Machine Learning models' evaluation on our dataset

As is shown by Figure 4.5, the best models' results belong to SVR, RF, XGBoost, KNN regressor, AdaBoost regressor, Ridge regressor with the highest value in variance score and R^2 that their values are around 1.0. Also, these selected algorithms have the best score in MAE and RMSE, that their result is near zero. Therefore, we applied these six algorithms and neural networks (LSTM) for water consumption prediction.

The most accurate and efficient model includes a proper training dataset and testing dataset. The process that we feed the dataset into the machine learning algorithms to train the model is called “Training the dataset”. “Testing the dataset” is a process for accuracy validation of the machine learning models. An unseen dataset is used as the test dataset, which is not used in the model training phase.

4.3.1.2 Train Models

After choosing proper algorithms and preparing the data, the model should be trained by our desired algorithms through historical input data. This is done to find the relationship between the prediction target and independent variables. Our methodology for splitting the dataset into train and test was based on hourly water consumption. We divided our dataset to 80% for the training set and 20% for the testing set. It means we choose one specific time for water on a specified day and date to predict water consumption value. To describe which part of the past data we train and test, we will explain in detail in the next section.

4.3.1.3 Model Prediction and Deployment

The model prediction is a process of giving input test data to our trained model. We use the regression metrics (defined in the Background section) to evaluate the accuracy of the model output. Our model was trained and tested based on the structure depicted in Figure 4.6, showing the process used for splitting the dataset into train and test datasets in this study.

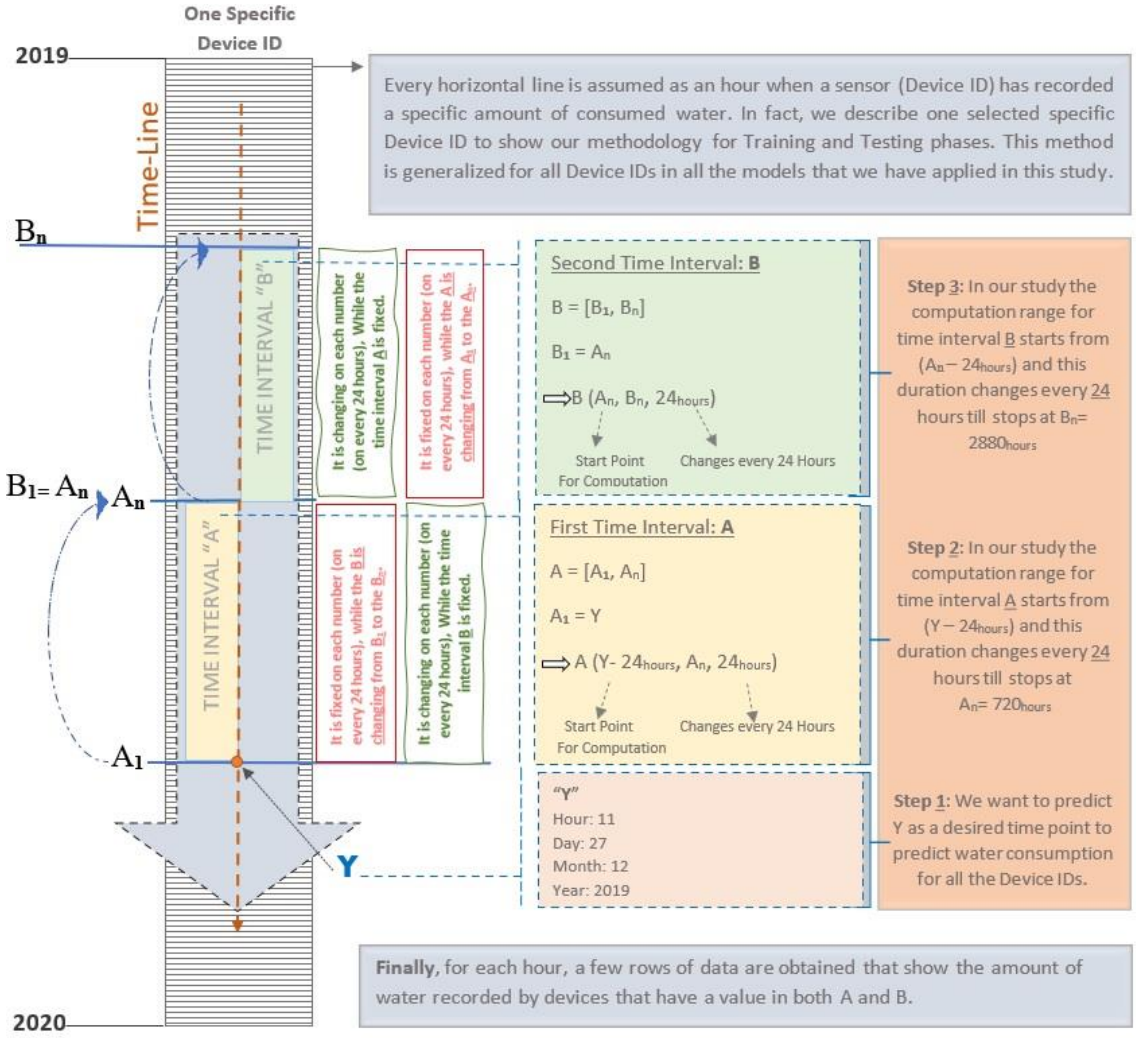


Figure 4.6.Our Method to split the dataset for Training and Testing phases.
The hourly water consumption is shown on the time axis.

First, we describe our idea based on the measurement of one sensor (one specific Device-ID) to show how to split the dataset for training and testing data; then, this method has been generalized to all the Device-IDs and the main dataset. This model is implemented to predict the hourly water consumption on the time axis. This means that we select a time point Y from the data we prepare to predict, which includes: To reach this point and predict the amount of water consumption at this hour, we must train and test on past data. At this stage, the time axis is divided into two time intervals. One is called A and the other B. Time interval A is the number of hours we move back and forth from point Y on the time axis. The important point is that the time interval A is a variable interval that starts from A_1 = 24, exactly one day before the point Y, and goes back to the equal point A_n = 720 hours, equivalent to one month, and the recorded data for water consumption. Every 24 hours at a time point equal to Y. The second time interval, which we call B, starts from time point B_1 = An and can continue until B_n (in our study, B_n equals 2880 hours). Time interval B is for learning from previous examples. In this

course, data is trained at intervals (every 24 hours) to see at what intervals each model or algorithm gives us a better forecast. This method of training and testing data helps us to understand which periods each algorithm makes more accurate predictions of our historical data, which algorithms for short time intervals, and which algorithms are appropriate and efficient for long intervals. The function of the model for training and testing is as follows:

The model first keeps A time intervals fixed every 24 hours, and for each point in time interval A, B time intervals change every 24 hours and trains the data. In the second step, interval B is kept constant by the model, and for each point in B time intervals, interval A is changed every 24 hours, and the data is trained. The important point is that there is only one row of data at each time point for each hour, which shows the amount of water recorded by Device-IDs, which at one time had values in both A and B. This means that the number of rows of data being trained will be less than the total amount of data prepared because for each time point in interval A, there must be a recorded amount of water consumption in interval B (and conversely) so that we can have a row of data. Therefore, we must choose a comfortable date (time point) for prediction and select the appropriate time duration based on the hour (for time intervals A, B) that gives us enough dataset volume for the training and testing phase. Because if we do not pay attention to these two issues, the model tells us, “You should choose a larger dataset” as an error about the selection of uncomfortable time prediction or duration, not about the structure of the algorithm or model. Indeed, in the result of each algorithm, interval "A" will tell us how many hours is good. We go backwards to find the best period in the past time for selecting a proper time duration ("B") for the training phase.

4.3.2 Prediction Model

Here, we demonstrate the procedure of the water consumption prediction model by machine learning techniques. The first three stages of this process are fully described in the previous sections. There are two important points about this model. The first point is about the second stage that, if the model finds the selected date for prediction is a weekend, the model only chooses the weekends for use in the training phase when moving backwards on the timeline. That means the model ignores all the weekdays for the training phase. Also, this strategy works in the same way for weekdays by ignoring the weekend when the dataset is used in the training and testing phases. It causes our prediction to be more accurate because, for weekdays, it only considers the weekdays and vice-versa.

Another important point is about the fourth stage to achieve the most accurate and efficient model; we use hyperparameters in the evaluation step for each algorithm. Indeed, we applied hyperparameters to improve our models when the model evaluation results were not desirable. Figure 4.7 describes all the stages in detail.

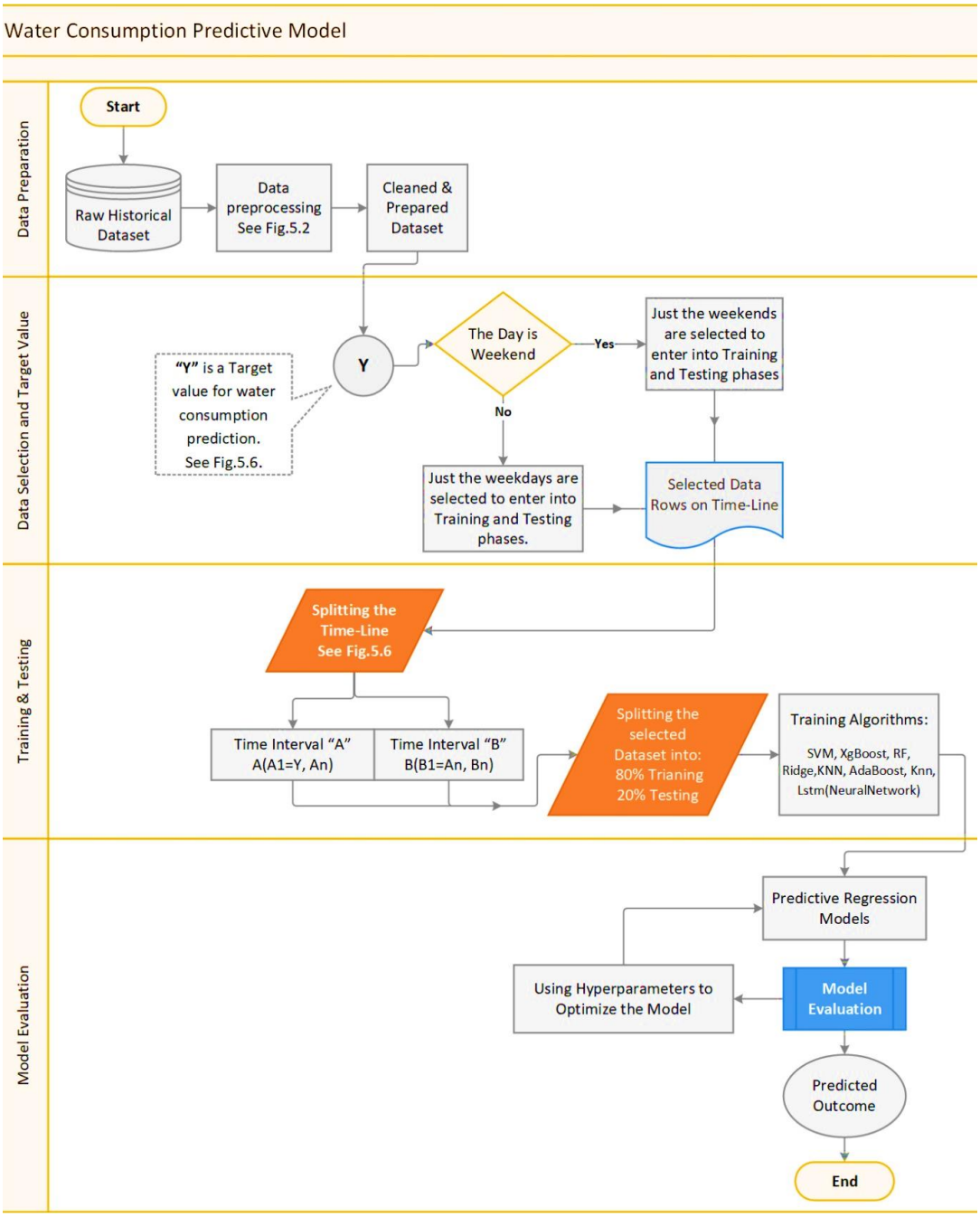


Figure 4.7.Water Consumption Prediction procedure

Chapter 5 Results and Evaluation

We describe our result in detail after applying our different algorithms and models to the water consumption dataset (Table 5.2 to Table 5.5). Figure 5.1 specifies which types of algorithms are efficient for what period. Although the result of some algorithms in this study needed to be optimized by hyperparameter tuning, some of our models had instinct good results based on their original structures and without using the hyperparameter tuning. So, we did not use the HPT for all the models. Furthermore, one of the assumptions to do regression is that the dataset is normalized. In this study, the result of each model has achieved through normalized data was done using the data preprocessing step. Another important issue about our models' evaluation is considering the R^2 score results as the main result for each algorithm more than the minimum MAE results. Our decision about considering the result of the R^2 score for evaluation is because all MAE results are very small and between 0 and 1 even after applying the hyperparameter tuning. Although a very small MAE value may be a sign of overfitting, this is not necessarily the case. It can depend on whether we have normalized and brought our numbers between 0 and 1. So usually, MAE also comes between 0 and 1 and can be a very small number. But when the goal is to compare models with different algorithms, there is no problem because that number does not represent the real error and the error in the scale is normalized.

Consequently, we consider the proper hours in the timeline based on the optimal R^2 score results that are more trustable based on the reasons we mentioned above then check these numbers on the MAE results' plots to present which types of regression algorithms in the hourly water consumption prediction are suitable for what time period. Indeed, the minimum MAE in our study has been achieved in two steps. It means that in the first step, the model calculates the MAE for all the possible match points of A and B for each Device-ID (sensor). The MAE results of each device are stored in an Excel sheet. Then the model chooses the most minimum MAE of each Device-IDs and brings all the most minimum MAE of all Device-IDs on one plot to show the best minimum MAE for time intervals A and B. The goal of this study is not only to calculate the MAE for each Device-ID but also to find the best time intervals for using the most capable algorithm in that specific time interval to predict the water consumption in the most accurate possible way.

5.1 SVM Result

The results of using the SVM regressor (Table 5.1) showed that the best choice of past data is when we go back a little and are closer to the time of consumption forecast. Nevertheless, using and investigating the past data based on both hour points (1560 and 2496) that are bigger than half of the whole examined time (2880 hours), we will gain a better result if we select a long period of past data.

Method 1	SVM	HPT	Result
Without Applying HPT	The time interval “A”: (24,720,24) The time interval “B”: (720,2880,24) The best points with the appropriate results belong to: The time interval “A”: 96 h The time interval “B”: (2064)h	NO	The results without HPT: “A”: 96hour “B”: (2064) hour R^2 score is 0.7 as an appropriate positive fit, and the minimum MAE is 0.01 that is near zero.

Table 5.1. The results of the applied SVM model with details

5.2 AdaBoost Regressor Result

Table 5.2 outlines the result of performing the AdaBoost regressor. The outcomes proved that for selecting past data, if we move back in the time as much as possible and move away from the time of consumption prediction, and then if we examine a lengthy period of time from past data, we get a desirable result.

Method 2	AdaBoost_Regressor	HPT	Result
1st step	The time interval “A”: (24,720,24) The time interval “B”: (720,2880,24) >> The result was not appropriate.	NO	not acceptable : means very good or being extremely out of range
2nd step	So, we chose the better duration for (HPT): “A”: (336, 552, 24) “B”: (1656, 2112, 24)	Prams: {'learning_rate': 1, 'loss': 'square', 'n_estimators': 50} From ['n_estimators': 50] to ['n_estimators': 100] the Results goes out of range, so we used the 50 value for the 'n_estimators'.	not acceptable
3rd step	After the first (HPT), To choose just the best point of times, not the duration of time: “A”: [432, 576, 552, 456] “B”: [1080, 1632, 1656, 1726]	Prams: {'learning_rate': 1, 'loss': 'square', 'n_estimators': 50}	The best results are: “A”: 552 hour “B”: 1726 hour R^2 score is -0.9 as a perfect negative fit, and the minimum MAE is 0.2 that is near zero.

Table 5.2.The results of the applied AdaBoost model with details

5.3 Ridge Regressor Result

The results related to performing the Ridge regressor algorithm (Table 5.3) showed that to select past data if we move back a little in time and get closer to the time of consumption forecast, but in the next step (which is to use and investigate the past data), if we select a long time duration from past data, we will have a better outcome.

Method 3	Ridge_Regressor	HPT	Result
1st step	The time interval "A": (24,720,24) The time interval "B": (720,2880,24) >> The result was not appropriate.	NO	not acceptable : means very good or being extremely out of range
2nd step	So, we chose the better duration for (HPT): "A": (216, 312, 24) "B": (1464, 2520, 24)	Prams: {'alpha': 0.3, 'fit_intercept': True, 'max_iter': 1000, 'normalize': True}	The best results are: "A": 264 hour "B": 2160 hour R^2 score is 0.67 as a perfect positive fit, and the minimum MAE is 0.1 that is near zero.

Table 5.3. The results of the applied Ridge Regressor model with details

5.4 RF Result

Table 5.4. shows the results of the RF algorithm, demonstrating that we should move back in time as much as possible and move away from the time of consumption forecast for selecting past data. And then, if in the same time period, we select a large range of past data, we will get better results.

Method 4	RF	HPT	Result
1st step	The time interval "A": (24,720,24) The time interval "B": (720,2880,24) >> The result was not appropriate.	NO	not acceptable
2nd step	So, we chose the better duration for (HPT): "A": (384, 696, 24) "B": (1896, 2760, 24)	Prams: {'bootstrap': False, 'criterion': 'mse', 'max_depth': None, 'max_features': 'auto', 'min_samples_leaf': 4, 'min_samples_split': 2, 'n_estimators': 3}	The results with HPT: "A": 456 hour "B": 2520 hour R^2 score is - 0.83 as an appropriate positive fit, and the Minimum MAE is zero.

Table 5.4. The results of the applied RF model with details

5.5 KNN Regressor Result

According to the KNN regressor results in Table 5.5, the best selection for the past data is moving back a little in time and being closer to the time of consumption prediction, but in the next step (which is to use and investigate the past data), if we select a long time duration from past data, we will achieve the most proper result.

Method 5	KNN	HPT	Result
1st step	The time interval "A": (24,720,24) The time interval "B": (720,2880,24) >> The result was not appropriate.	NO	not acceptable
2nd step	So, we chose the better duration for (HPT): "A": (648, 672,696) "B": (1824, 1872,2448)	Prams: {algorithm='brute', leaf_size=10, n_jobs=-1, n_neighbors=2}	The results with HPT: "A": 312hour "B": 1968 hour R^2 score is - 0.72 as an appropriate negative perfect fit, and the minimum MAE is 0.3 that is near zero.

Table 5.5. The results of the applied KNN model with details

5.6 XGBoost Result

By observing the results of the XGBoost algorithm in Table 5.6, we achieved the important point that moving back in time as much as possible and moving away from the time of consumption prediction can be a suitable choice to attain an acceptable outcome. And then, investigating a lengthy time period of past data in that same time period determines which time period this algorithm works better and efficiently.

Method 6	XGBoost	HPT	Result
Without Applying HPT	The time interval "A": (24,720,24) The time interval "B": (720,2880,24) The best points with the appropriate results belong to: The time interval "A": 576 The time interval "B": 2560	NO	The results without HPT: "A": 576 hour "B": 2568 hour R^2 score is 0.9 as an appropriate positive fit, and the Minimum MAE is zero.

Table 5.6. The results of the applied XGBoost model with details

5.7 LSTM Result

Based on the literature, some studies [69], [70], [74] applied the LSTM model for water consumption prediction and water depth forecasting and achieved good results. Therefore, we chose LSTM as a candidate model besides other algorithms used in the related studies. To decline the computation time, we launched our models on the university server with powerful hardware configurations. Computation intensity and time to train each of the algorithms varies. In our case study, we can rank the algorithms based on their training time, as follows: 1. Ridge Regressor, 2. SVR, 3. AdaBoost Regressor, 4. XGBoost, 5. KNN Regressor, 6. Random Forest, and 7. LSTM algorithm.

Within three months, we performed the LSTM model several times. It was running for 15 consecutive days the first time, and after 15 days, the system was restarted without any results. Then we started to change some parameters to reduce the calculation time and get the result for the LSTM algorithm. Therefore, we adopted the value of epochs, batch_size, and learning_rate. Still, each time it took 6 to 7 days. It was stopped because of insufficient memory or other problems such as network interruption on the server or sudden system reset due to long computational load. These are common problems and may usually occur when running heavy computational programming for a long duration. Thus, we could not achieve any final result for LSTM.

We can explain that this problem happened because of the long computation time of LSTM and our program. As stated in the literature [36], the LSTM is a combined model since it saves information to re-inject the past data into the network. This process happens several times regarding the number of hidden layer nodes. Therefore, the main drawback of LSTM is that it is not a computationally efficient algorithm. To clarify the second reason, we should point out that we used a combined procedure to decide how much past data we should include in our analysis. As the algorithm has to consider the two time periods in the past each time and enter a combination of common points in both time periods into the train and test stage, it requires repeated sampling from each time point, training, and testing that does not well fit with the LSTM algorithm structure. In the literature review, the studies used LSTM with a more straightforward procedure. For instance, Bejarano et al. [74] used the past 24 hours data as the input data to the LSTM. In contrast, we used “For Loops” to form the input data of our algorithms. Using “For Loops” was inevitable to reach our research objective of examining the models based on their ability to predict the near or distant data. Using “For Loops” made all of our algorithms to be time-consuming as we saw that Ridge and SVR (which are popular because of their simplicity) took, respectively, 30 hours and 50 hours to provide the results. As mentioned earlier, we first used the algorithms' primary forms without hyperparameter tuning to decrease the running time. If one algorithm was efficient in its basic form, we did not follow the hyperparameters tuning step. If not, we used only the best prediction points of that specific model for HPT instead of using the whole dataset. Overall, we can conclude that LSTM is not an efficient and suitable algorithm for our case study.

5.8 Comparison of Algorithms

Here, we present the different algorithms on the timeline based on the hour assessment to show which types of algorithms are appropriate for which time durations as an efficient algorithm for predicting the hourly water consumption (Figure 5.1). The results of the R^2 score and MAE metrics are measured based on one data in a special time point, not the whole dataset, because we want to find the best duration of time in both A and B to get results about this fact that which algorithms are efficient for which duration of time.

As the result of algorithms shows, to gain a good result by the SVM, we should choose a short period for going backwards (means time interval "A"). Furthermore, in selecting a time duration for dataset training, we should also select a short period to achieve a proper result. Although the results show the Ridge algorithm needs to choose a short period for time interval "A", it should choose a slightly longer time ("B") in the past to have a better result for predicting the hourly water consumption at the specified time. Regarding the result of both AdaBoost, XGBoost, and RF, this is obvious that they can achieve good prediction results when we select the data at farther and larger intervals. About the XGBoost and RF, as it progresses further and further into the distant past into past data, a better prediction result can be gained. The results of the KNN algorithm proved that for selecting past data if we move back a little in time and are closer to the time of consumption prediction, but in the next step (which is to use and investigate the past data), if we select a long-time duration from past data, we will get a better result.

To summarize, we had to choose every 24 hours calculation to find the match point of data in both A and B instead of using different random time calculations to gain more precise prediction and enough data rows for the training phase. Indeed, the model tries to find the data for both A and B at the same point in time, and because of this, the numbers of data rows decrease. So, if we choose data based on different random hours like 3 or 10, and so forth, we will not have enough rows to test our predictive model.

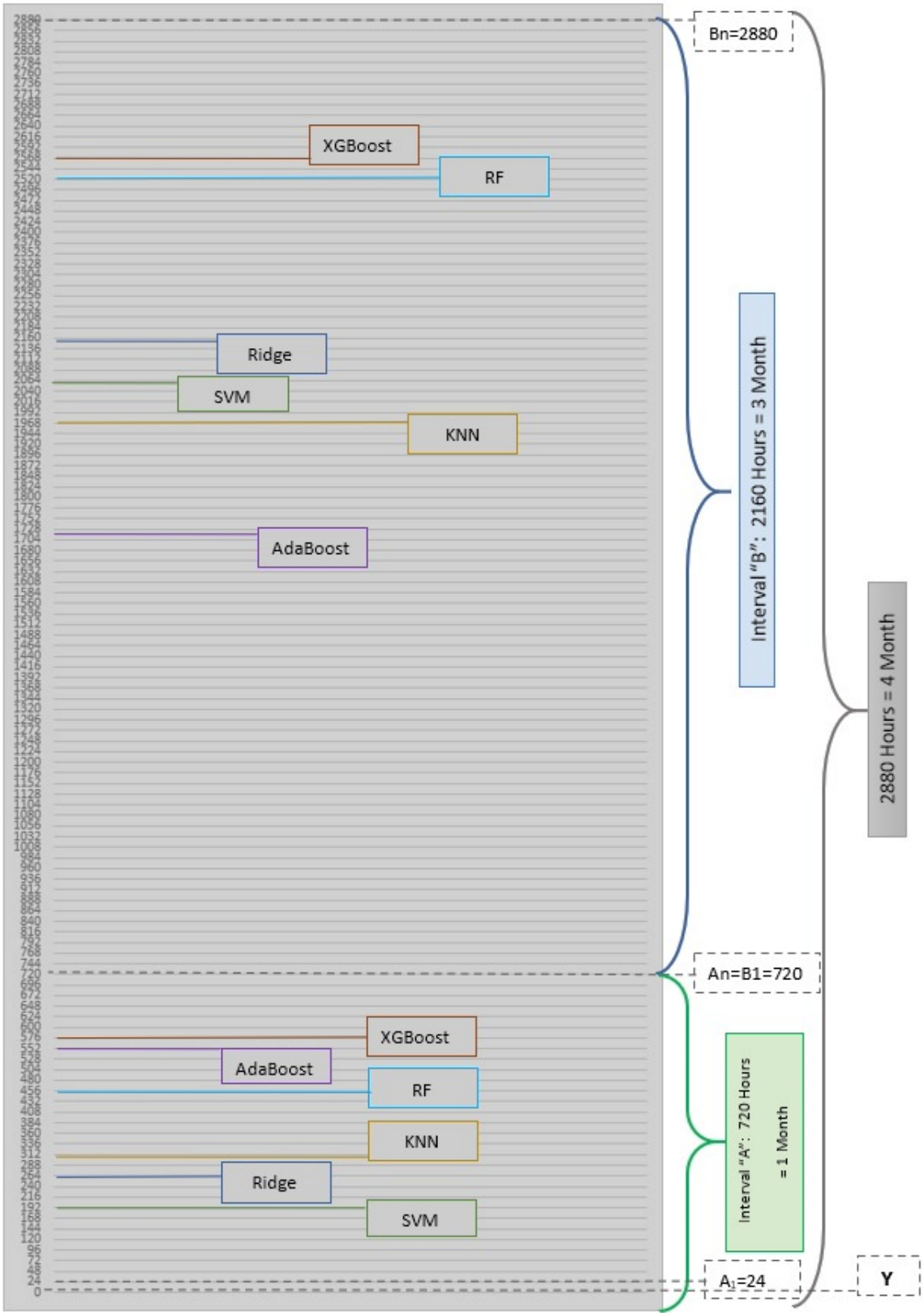


Figure 5.1. Comparison of Algorithms' results

Chapter 6 Discussion

Here we can point out some significant cases to summarize the weaknesses of the previous related studies and highlight our work's contribution in improving these weaknesses. In machine learning modelling, it is needed to pre-process the data before designing models for predictions. According to [36], data pre-processing can be done in several areas. Among them, dealing with missing data is of high importance because it can influence the accuracy of predictive models. As we can see in related studies [58], [71], [74], [76], [78] data pre-processing is included as an independent step. Nevertheless, in these studies, it is not mentioned clearly what they meant by the term “missing values” (null values or zero values). In the present research, like other studies, we considered data pre-processing as the first step in which we removed duplicate data. On the other hand, we kept zeros as valuable data, which showed us no water consumption at that specific time. In this way, we were able to improve the model accuracy.

Brentan et al. [66] applied an SVR method for forecasting the water demand, and they observed that this model could not capture the extreme points (peaks) of water consumption value. For fixing this problem, they added an adaptive Fourier time-series to the SVR model. Similarly, Walker et al. utilized an ANN model, and their model could not predict the peaks accurately. The authors concluded that these inaccurate results in peaks might be related to the noise in the input dataset. Whereas, in our research, we have not encountered such a result in either SVR or ANN models because we normalized our data before feeding it into the models.

Feature engineering is a way of making more robust features and increasing the accuracy of the prediction. In machine learning modelling [36], feature selection helps reduce the execution time and improve model accuracy. The new features creation approach is what we have done in this research to make meaningful features such as year, month, day, hour, weekends, weekdays and investigate their effect on the water consumption model performance.

In almost all studies reviewed in this field, the variable that specifies the type of day (whether being a holiday or not) has been considered [60], [65]–[67], [71], [72], [78]. Bejarano et al. [74] and Romano et al. [76] augmented the calendar information to their models to investigate the difference between the result of models by determining the holidays and regular weekdays. In our research, we used a new variable to show weekends and weekdays. Our model could distinguish between days as the weekends or weekdays and only select the specific days based on weekends or weekdays for the training phase. This approach increases the model precision in our study and decreases the execution time because it only calculates the special days based on the type of days.

Based on a comparison between previous studies' results, various deep learning methods, machine learning models, or statistical algorithms have been used individually or as a combined model (a hybrid model) of several algorithms. Differences in exploiting different machine learning methods and approaches show that a group of researchers gained comfortable results by applying one specific method in a study. In contrast, in another study, the same method did not yield acceptable results. For example, in studies [77] and [65], the use of the ANN method showed excellent results in predicting wa-

ter demand and energy consumption based on the observed data, while this method has not presented an acceptable output in the [75] and [59] studies. It can be because of different model hyperparameters used in various studies. Our research followed an optimizing approach to improve our model's result and decrease the HPT method's computation time. This method contained two phases. The first phase was about gaining the result of the model without changing the original structure of each algorithm. If the result of the model (R^2 score and MAE) was not acceptable, we selected the most optimal time points (time interval) that had the most optimal results in the first phase to apply the hyperparameter on these specific points in the second phase. Ignoring the weaker time points and using hyperparameter only on the optimal points can save computational time in the HPT process and increase model prediction accuracy.

In the water consumption prediction field, often, the dataset is a large time-series dataset. Even though the model execution time is a crucial factor when working on big data, none of the reviewed studies points out how they considered decreasing the computation time. Whilst we have taken into consideration to propose the ways of reducing model execution time. As described before, we designed the model that works separately for the different types of the day (weekends or weekdays). Also, to face the pure result of each model to understand how the structure of an original machine learning algorithm can influence the prediction results in this study, we did not use any HPT in the basic modelling. We applied the HPT in the second phase, only for the optimum acceptable data points of the first phase.

Some previous studies concluded that the inclusion of more past data decreased the prediction efficiency. In the study by Herrera et al. [67], two models were exploited. In one of these experiments, they only considered eight weeks of historical data, and in the other one, all the data was fed into the model as input. The Result showed that using more data made the performance worse, and too old data can be harmful to the forecasting models. Bejarano et al. [74] also examined the importance of the length of input data. They observed that the inclusion of 24 hours of past data was sufficient to capture the water consumption patterns while having more data as the input did not improve the model prediction ability. In this study, when we used the past dataset, we tried to use "For Loops" to find the time points that randomly have a recorded amount of water consumption in both periods to state the fact which types of algorithms work well and optimally to use which historical data (near prediction or distant) in the past. It is worthwhile to describe the design process that we followed in this study. First, we performed several regression algorithms in their basic format to see how they behave with data selected from different times and different sizes of the past. We could feed the data in different time periods into the models using "For Loops" and observe their results. In this context, if a model performance was acceptable, we used the basic format of that algorithm without any hyperparameter tuning. On the other hand, if a model did not perform well, we conducted a hyperparameter tuning process only on the time periods where the given model had satisfactory scores. We used the basic algorithms without any optimization because, in this way, they were able to show their real capability in forecasting this kind of data without any added correction factor or adjustment.

In the following, we answered Research Questions based on the literature review in detail.

RQ 1 What are the characteristics of the dataset used in the energy and water consumption studies? The dataset analyzed in water consumption studies is a time-series dataset that is an event observation in chronological order for a specific period. The temporal continuity of data recorded over a period in the past is a message from the past that gives us the behavioral pattern of an event to predict the future trend of the same event. The time-series dataset has some significant characteristics like the complexity of the relation between the past and present data. Researchers use different machine learning approaches to examine and understand the relationship between new recorded and past data that analyze the complexity of these relationships to build a model for predicting the future of events' trends. It should be mentioned that in some cases besides the time-series dataset, the researchers investigate the effect of other external variables on the prediction like weather conditions, temperature, or the population. In our study, we desired to examine the effect of population on the water consumption prediction, but Sarpsborg municipality could not give us this information because of confidentiality. So, we applied other features as mentioned in the previous sections for water consumption prediction. There is a fact about the quality of the dataset or variables how much the raw dataset is qualified to enter the machine learning process. Therefore, dataset preprocessing is a crucial step in every machine learning project like water consumption prediction studies to fix noisy data, outliers, missing values, and so forth. For domestic water consumption prediction, Walker et al. [75] considered statistical information derived from real meters beside the recorded data to empower the prediction model's generalisability to new input data despite the noise in data. In another study, Bejarano et al. [74] applied linear regression to dispel three percentage missing values in their dataset.

Moreover, in many cases, the input dataset is big data with many samples and variables. Although having large data helps make accurate predictions, it makes data visualization and model execution an intricate work that requires different approaches based on the dataset type. One example can be the study performed by Tamang and Shukla [78] to record the water consumption dataset by the smart meters. The dataset contained 90-day daily water consumption data collected from dairy plants considered big data because of the high volume of the dataset. They used the time-series clustering technique ability to distinguish similar recorded trends of consumption during weekends and weekdays. They applied the predictive algorithms on these two clusters on weekends and weekdays to decrease the time computation for the big dataset prediction model like what we have done in our predictive machine learning model. These factors are managed by machine learning abilities that we describe some of the machine learning functions in the second Research Question.

The researches have proved that the huge volume of data collected by sensors in smart homes can be stored, managed, backed up in a smart environment like the Cloud technology as remote storage. The Cloud storage is a fast implementation, elastic solution, cost-effective, resilient, scalable, and able to manage the stored data from outside the site that the admin has quick access to available data through special connection to the private network or the public internet similar to the Sarpsborg municipality that uses the Microsoft Azure Cloud. Continuous analysis, storing, updating, and using the data are very important factors in data management. The studies showed us that the most common characteristic of big data like high velocity, wide variety, high volume are just

some of the big data attributes that state our requirement to apply appropriate approaches like machine learning techniques for big data classification and management.

RQ 2 Which types of machine learning algorithms or models are efficient for analyzing water datasets and predicting water consumption? Indeed, the prediction ability of machine learning based on mathematical algorithms and models distinguishes it from other technologies and, needless to say, how much machine learning techniques have acted efficiently and powerful in presenting an accurate model for forecasting the future of everything in recent years. Therefore, we point out some of its capabilities associated with our research goals. The efficiency of the machine learning algorithms depends on the forecast horizon. The short-term prediction is helpful in management, while long-term prediction is a requirement for the design phase. This is the main issue that we have considered in our results investigation: which types of algorithms are more efficient and useful for short-term or long-term predictions. In addition, the algorithms can be viewed as being linear or non-linear models. Linear models are simple methods and can be understood easily, while non-linear models are complicated.

Based on the machine learning function, the larger the amount of data or information we give to the model, the prediction will be more accurate. But the important thing is that as the number of data increases, more time is spent analyzing, and in the training phase, the computation will be complicated. One solution for this issue, based on the study of Hai Xiang Zhao, Frédéric Magoulès in 2010 [60] on energy consumption prediction, uses parallel algorithm execution to decrease the average training sets. In our study, for reducing the time computation, we applied a technique to divide the days of the week into the workdays and weekends separately that the most accurate prediction is achieved using the least time computation in training phases. Water consumption time-series has non-linear patterns; therefore, the non-linear models are a better choice for having high accuracy in our predictions since they can present more generic models.

It can be claimed that a practical solution, according to most reviewed studies, is improving the process of some machine learning algorithms or combining them. Accordingly, most reviewed studies have applied non-linear algorithms individually or combined with other models as a hybrid system. For instance, Brentan et al. [66] proposed a hybrid model with a perfect fit for the collected data. The proposed model consisted up of an SVR algorithm combined with adaptive Fourier time-series for short-term forecasting. Moreover, in the study by Herrera et al. [67], they utilized various non-linear algorithms for a short-term forecast. The algorithms were ANN, PPR, MARS, SVR, and RF. The SVR model outperformed other types of algorithms, followed by MARS, PPR, and RF, respectively. Another study for predicting daily water consumption forecast was done by Chen et al. [68]. They exploited multiple RF models and attained more accurate results with this conjunction model than a single RF model. Furthermore, Dufour et al. [71] proposed a flexible and easily implemented solution utilizing multiple decision trees. The final output showed an acceptable performance of 91% correct prediction. Tamang et al. [78] explored the water demand forecasts optimization by applying an SVR model. The evaluation metrics showed that the SVR model predicted the water consumption well on the small dataset and outperformed the ANN model, however in our study, we interpret the SVM algorithm as an efficient algorithm for using the past data close to the forecast time, not just based on the value obtained in the future forecast model.

In addition to the mentioned algorithms covering the non-linearity aspect of time-series data, there are other advanced methods to capture the seasonality and periodic nature of water consumption historical data. These two characteristics of time-series datasets (seasonality and periodicity) show that the past time steps influence the present one. The Recurrent Neural Network methods such as LSTM can perform well on historical data in which the data points' values are related to each other. Given this, Zhang et al., 2018 [69] predicted the water table depth using a two-layer LSTM-based model. This model had very robust results on time-series. Another study in this context was performed by Nasser et al. [70]. They designed a system for both short and long-term water consumption prediction and used LSTM architecture. They compared the output of the proposed system with the results of launching SVR and Random Forest models on a similar dataset. According to the results, LSTM had higher evaluation scores than the other two traditional models. Likewise, Bejarano et al. [74] provided a system for predicting future water consumption using an LSTM architecture. The results showed that LSTM surpassed the other methods (the linear regression and ARIMA models). We applied LSTM in our study like other studies in this area for the water consumption prediction, but we did not gain any result, and this algorithm is not efficient for our study goal.

RQ 3 What are the other possible methods used in addition to Artificial Intelligence (AI) algorithms in energy and water studies? Besides AI methods (machine learning and deep learning algorithms), engineering methods and statistical methods are also used in some studies to predict energy and water consumption.

The engineering methods use the physical functions and thermal dynamics to calculate the energy consumption of a building. Many software tools have been developed by using the formulas of engineering methods. However, due to their complex formulas and many variables, engineering models require a large amount of input information, which is time-consuming and difficult to collect.

On the other hand, the statistical methods simply utilize regression to correlate energy consumption with the influential factors to apply these models to historical data. Still, lack of accuracy and no flexibility are two drawbacks of these statistical methods.

An alternative method to fill these gaps in engineering and statistical methods can be AI models that are powerful for making robust predictions. In addition, some of the studies used a combination of AI methods with statistical models or engineering models or both. There are reviewed studies in which a combination of engineering methods and AI methods are applied. For example, Zhao et al. [60] used Energy Plus software for raw data visualization and analysis; after that, they used SVM for making a prediction model. Likewise, Wong et al. [65] exploited EnergyPlus software for simulating the data and detecting the determinants. Some other studied used a combination of statistical models and AI models or compared the performance of these two methods. For example, Li et al. [59] investigated several statistical and AI methods for studying buildings' energy consumption.

Similarly, Herrera et al. [67] investigated hourly water demand models' performance by comparing several statistical and AI models such as PPR, MARS, ANN, SVR, and Random Forest. The result of their study demonstrated that the SVR outperformed other algorithms. Bejarano et al. [74] designed a system consisting of two models of GCRFs and LSTM. They compared these two models with ARIMA and linear regression mod-

els. They concluded that the designed system showed better performance scores than the other single methods.

Although this evidence could help us to improve our knowledge about the differences and abilities of each mentioned method above, in our study, we aim to compare the difference between the ability of each machine learning algorithm that the focus is on the accuracy of their prediction is based on the choice of an efficient period of the historical data relative to the time to be predicted, not the difference in the methods used. The experiments conducted on sensor data from energy resources led us to the right track to step on choosing efficient algorithms for our study.

The results of our study showed that the SVM's capability with its SRM feature in using parallel execution could decrease the learning time and is powerful in creating a model with a few samples and parameters. ANN is an effective approach for a minimum of data collected when we face the lack of available massive datasets, and it is used for commentary improvement of the trained network. Various structures of ANN apply to choose an efficient tool for controlling the process when the trend of the reliability and accuracy of the machine learning model decreases. The sudden occurrence of any change in the structure of the algorithm by itself or in the system's external environment during the operation of the model can be time-consuming after the training process. Still, it is manageable by alteration of the ANN structure. Also, RF and Decision tree algorithms with high accuracy can be another efficient algorithm for dealing with the occupancy recognition challenge or rules' extraction from data collected by sensors.

To summarize, the review that we conducted on previous research proved that it is a challenging choice to choose the best algorithms or machine learning model because every study was structured based on various conditions with a high diversity of parameters as the input. But the unique and special issue about all of them is the machine learning techniques that have had the most efficient performance for energy consumption management, like water consumption. Therefore, our proposed methodology shapes based on Random Forest, SVM, ANN algorithms (LSTM), XGBoost, KNN regressor, AdaBoost regressor, Ridge regressor to achieve acceptable results in this study.

RQ 4 Which variables are influencing water consumption? Almost all investigated studies about water consumption considered 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 [60], [65]–[67], [71], [72], [74], [76], [78]. In addition, they have 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, some of the researchers explored the role of external variables on water consumption. Brentan et al. [66] analyzed the relationship between rain, temperature, air humidity, and wind velocity to water consumption. They found out that there were meaningful correlations between these variables and water consumption. In another study on hourly water demand prediction by Herrera et al. [67], the information of daily weather variables was considered in addition to water consumption values. They studied the climate variables such as temperature, wind velocity, rain, and atmospheric pressure impact the water demand. The temperature was the most relevant external variable to the water demand behavior among all the mentioned factors. Zhang et al. [69] chose water diversion, precipitation, evaporation volume, and temperature to predict

water table depth. Ju et al., 2014 [73] investigated 29 important variables from three categories of climate, social and economic aspects, i.e., precipitation, annual mean temperate, annual frost-free period, evaporation, population, and so forth to assess influencing factors on water requirement.

In our study, because we had a short period of water consumption dataset (around one year), the prediction or investigation of the influence of the holidays or regular days was impossible since the machine learning techniques requires of long-time data history (many years) to train and test the dataset. But we used the difference of the day types for the training and testing phase to achieve the most accurate prediction for water consumption, as we described in the previous sections.

RQ 5 What are the evaluation metrics for measuring the performance of models in water consumption studies? Various evaluation criteria have been applied in the performance assessment of different models in energy and water consumption studies. In some researches in this field, the accuracy metric has been used to estimate the performance of the classification model. For example, Fernández et al. [57] in the energy efficiency study in smart homes, Vafeiadis et al. [58] in the occupancy recognition examination, and Zhao et al. [61] regarding the factors influencing the forecast of energy consumption in buildings, used accurate scores as a determining factor to evaluate the performance of their model. In addition, Ahmad et al. [63] for the model performance evaluation of predicting electricity energy consumption in buildings, Benedetti et al. [64] for energy consumption assessment, and Walker et al. [75], for hourly water consumption prediction of households utilized accuracy metric to evaluate the performance of their algorithms.

On the other hand, in the regression studies and specifically in the field of water consumption prediction, other metrics have also been used. For instance, Brentan et al. [66] used the MAE, R^2 score, and the RMSE to assess the result of their proposed model. Moreover, MAPE, R, TS, and NRMSE have been used to study urban water consumption prediction by Chen et al. [68]. Another example is Zhang et al. [69] that used RMSE and R^2 scores for computing the efficiency of their proposed method to prove the strong learning ability of their LSTM model structure. Bennett et al. [77] used several criteria such as RMSE, R^2, ARE, AAE, and MW P-value metrics for assessing their model prediction ability in forecasting the household water end-use consumption. Finally, Nasser et al. [70] used MAE, RMSE, and MAAPE to measure their model effectiveness about the ability of LSTM compared with SVM and RF in the water demand future forecast. We applied MAE and R^2 scores to evaluate the hourly water consumption prediction results based on our supervised machine learning models and time-series dataset.

Chapter 7 Conclusion and Future Work

In conclusion, our approach could cover the weaknesses of the previous studies and presented a new decision criterion for selecting the proper algorithm in the water consumption prediction studies. It must be stressed that the contribution of our work is that we introduced a valuable determinant for several popular algorithms instead of comparing various machine learning models' performance. We introduced a measure for selecting the appropriate model, for a given dataset, in terms of the time period and the size of the dataset. To give an example, we detected that the SVR method achieved admissible results on the dataset in the short distance of the prediction time. In contrast, the AdaBoost algorithm performed well on older data in the distant past. Therefore, based on the size and volume of your available water consumption time-series dataset, our study's result can help you choose the most efficient algorithms for the hourly water consumption prediction to get the most relevant and accurate results.

Suppose we have a huge volume of the dataset from water consumption. In that case, we will use the random calculation instead of every 24 hours calculation to find the match point of data in both A and B and have more rows to enter in the training dataset phase. Also, if we have more recorded data in many years, we can investigate the impact of special holidays in the summer or winter on water consumption future prediction. These two items were impossible because the Sarpsborg municipality has started this project newly. We face a lack of information and dataset for investigating these items on the water consumption future prediction. Therefore, there is the possibility to work on this issue, and this view can be achieved in the near future after some years, not right now.

Bibliography

[1] J. Wu, L. Ping, X. Ge, Y. Wang, and J. Fu, "Cloud Storage as the Infrastructure of Cloud Computing," in *2010 International Conference on Intelligent Computing and Cognitive Informatics*, Jun. 2010, pp. 380–383. doi: 10.1109/ICICCI.2010.119.

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

[3] TheWorldCounts, "Average Daily Water Usage." https://www.theworldcounts.com/stories/average-daily-water-usage (accessed Jun. 24, 2021).

[4] C. Pichery, "Sensitivity Analysis," in *Encyclopedia of Toxicology (Third Edition)*, P. Wexler, Ed. Oxford: Academic Press, 2014, pp. 236–237. doi: 10.1016/B978-0-12-386454-3.00431-0.

[5] H. Ahvenniemi, A. Huovila, I. Pinto-Seppä, and M. Airaksinen, "What are the differences between sustainable and smart cities?," *Cities*, vol. 60, pp. 234–245, Feb. 2017, doi: 10.1016/j.cities.2016.09.009.

[6] "Sarpsborg," *Wikipedia*. Jun. 20, 2021. Accessed: Jun. 24, 2021. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Sarpsborg&oldid=1029580120

[7] C. Höfer and G. Karagiannis, "Cloud computing services: Taxonomy and comparison," *Journal of Internet Services and Applications*, vol. 2, pp. 81–94, Jan. 2010, doi: 10.1007/s13174-011-0027-x.

[8] A. Lavric, "LoRa (Long-Range) High-Density Sensors for Internet of Things," *Journal of Sensors*, vol. 2019, p. e3502987, Feb. 2019, doi: 10.1155/2019/3502987.

[9] A. J. Wixted, P. Kinnaird, H. Larijani, A. Tait, A. Ahmadinia, and N. Strachan, "Evaluation of LoRa and LoRaWAN for wireless sensor networks," in *2016 IEEE SENSORS*, Oct. 2016, pp. 1–3. doi: 10.1109/ICSENS.2016.7808712.

[10] A. Whitmore, A. Agarwal, and L. Da Xu, "The Internet of Things—A survey of topics and trends," *Inf Syst Front*, vol. 17, no. 2, pp. 261–274, Apr. 2015, doi: 10.1007/s10796-014-9489-2.

[11] "Azure IoT Hub Documentation." https://docs.microsoft.com/en-us/azure/iot-hub/ (accessed Jul. 19, 2021).

[12] J. Barnes, *Microsoft Azure Essentials Azure Machine Learning*, 1st Edition. Microsoft Press Store, 2015. [Online]. Available: https://www.microsoftpressstore.com/store/microsoft-azure-essentials-azure-machine-learning-9780735698178

[13] "Azure documentation," Jun. 24, 2021. https://docs.microsoft.com/en-us/azure/ (accessed Jun. 24, 2021).

[14] 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.

[15] M. Jaradat, M. Jarrah, A. Bousselham, Y. Jararweh, and M. Al-Ayyoub, "The Internet of Energy: Smart Sensor Networks and Big Data Management for Smart

Grid," *Procedia Computer Science*, vol. 56, pp. 592–597, Jan. 2015, doi: 10.1016/j.procs.2015.07.250.
[16] P. Moral and P. González, "Univariate Time Series Modelling," in *Computer-Aided Introduction to Econometrics*, J. R. Poo, Ed. Berlin, Heidelberg: Springer, 2003, pp. 163–224. doi: 10.1007/978-3-642-55686-9_4.
[17] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*, 2nd edition. New York, NY: Springer, 2016.
[18] J. Xia *et al.*, "LDSScanner: Exploratory Analysis of Low-Dimensional Structures in High-Dimensional Datasets," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 236–245, Jan. 2018, doi: 10.1109/TVCG.2017.2744098.
[19] F. Pezoa, J. L. Reutter, F. Suárez, M. Ugarte, and D. Vrgoc, "Foundations of JSON Schema," *WWW*, 2016, doi: 10.1145/2872427.2883029.
[20] "JSON." https://www.json.org/json-en.html (accessed Jun. 24, 2021).
[21] G. Loosli, S. Canu, and C. S. Ong, "Learning SVM in Kreĭn Spaces," *IEEE Trans Pattern Anal Mach Intell*, vol. 38, no. 6, pp. 1204–1216, Jun. 2016, doi: 10.1109/TPAMI.2015.2477830.
[22] S. Polamuri, "How the random forest algorithm works in machine learning," *Dataaspirant*, May 22, 2017. https://dataaspirant.com/random-forest-algorithm-machine-learing/ (accessed Jun. 29, 2021).
[23] V. Kumar, "Random forests and decision trees from scratch in python," *Medium*, Sep. 10, 2019. https://towardsdatascience.com/random-forests-and-decision-trees-from-scratch-in-python-3e4fa5ae4249 (accessed Jun. 24, 2021).
[24] "Bagging." https://www.ub.edu/cursosR/files/bagging.html (accessed Jun. 29, 2021).
[25] M. Kuhn and K. Johnson, *Applied Predictive Modeling*, 1st ed. 2013, Corr. 2nd printing 2018 edition. New York: Springer, 2013.
[26] 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.
[27] "scikit-learn: machine learning in Python — scikit-learn 0.24.2 documentation." https://scikit-learn.org/stable/ (accessed Jul. 16, 2021).
[28] Z.-H. Zhou, "Ensemble Learning," in *Encyclopedia of Biometrics*, S. Z. Li and A. Jain, Eds. Boston, MA: Springer US, 2009, pp. 270–273. doi: 10.1007/978-0-387-73003-5_293.
[29] "Many Heads Are Better Than One: The Case For Ensemble Learning," *KDnuggets*. https://www.kdnuggets.com/many-heads-are-better-than-one-the-case-for-ensemble-learning.html/ (accessed Jun. 29, 2021).
[30] Y. Freund and R. Schapire, "Experiments with a New Boosting Algorithm," in *International Conference on Machine Learning, Bari*, 1996, pp. 148–156. [Online]. Available: http://dl.acm.org/citation.cfm?id=3091696.3091715
[31] R. Madan, "Gradient boosting Vs AdaBoosting — Simplest explanation of boosting using Visuals and Python Code," *Analytics Vidhya*, Nov. 25, 2019. https://medium.com/analytics-vidhya/gradient-boosting-vs-adaboosting-simplest-

explanation-of-how-to-do-boosting-using-visuals-and-1e15f70c9ec (accessed Aug. 11, 2021).
[32] A. Nagpal, “L1 and L2 Regularization Methods,” *Medium*, Oct. 14, 2017. https://towardsdatascience.com/l1-and-l2-regularization-methods-ce25e7fc831c (accessed Jun. 24, 2021).
[33] J. Brownlee, “How to Develop Ridge Regression Models in Python,” *Machine Learning Mastery*, Oct. 08, 2020. https://machinelearningmastery.com/ridge-regression-with-python/ (accessed Jun. 24, 2021).
[34] 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.
[35] 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.
[36] 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.
[37] I. Muhajir, “K-Neighbors Regression Analysis in Python,” *Medium*, Feb. 08, 2020. https://medium.com/analytics-vidhya/k-neighbors-regression-analysis-in-python-61532d56d8e4 (accessed Jun. 24, 2021).
[38] A. C. Müller and S. Guido, *Introduction to Machine Learning with Python: A Guide for Data Scientists*, 1st edition. Sebastopol, CA: O’Reilly Media, 2016.
[39] “Understanding LSTM Networks -- colah’s blog.” http://colah.github.io/posts/2015-08-Understanding-LSTMs/ (accessed Jun. 24, 2021).
[40] Y. Li, Z. Zhu, D. Kong, H. Han, and Y. Zhao, “EA-LSTM: Evolutionary attention-based LSTM for time series prediction,” *Knowledge-Based Systems*, vol. 181, p. 104785, Oct. 2019, doi: 10.1016/j.knosys.2019.05.028.
[41] R. Pascanu, T. Mikolov, and Y. Bengio, “On the difficulty of training Recurrent Neural Networks,” *arXiv:1211.5063 [cs]*, Feb. 2013, Accessed: Jun. 24, 2021. [Online]. Available: http://arxiv.org/abs/1211.5063
[42] Z. Wu and S. King, “Investigating gated recurrent neural networks for speech synthesis,” *arXiv:1601.02539 [cs]*, Jan. 2016, Accessed: Jun. 24, 2021. [Online]. Available: http://arxiv.org/abs/1601.02539
[43] 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.
[44] R. Dua, M. S. Ghotra, and N. Pentreath, *Machine Learning with Spark - Second Edition*, 2nd Revised edition. Birmingham Mumbai: Packt Publishing, 2017.
[45] R. J. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*. Lexington, Ky: OTexts, 2013.
[46] A. K. Sharma, *Text Book of Correlations and Regression*. Discovery Publishing House, 2005.
[47] H. E. A. Tinsley and S. D. Brown, Eds., *Handbook of Applied Multivariate Statistics and Mathematical Modeling*, 1st edition. San Diego: Academic Press, 2000.

[48] S. Glantz and B. Slinker, *Primer of Applied Regression & Analysis of Variance*, 2nd edition. New York: McGraw-Hill Education / Medical, 2000.
[49] J. N. Basalyga, C. A. Barajas, M. K. Gobbert, and J. Wang, "Performance Benchmarking of Parallel Hyperparameter Tuning for Deep Learning Based Tornado Predictions," *Big Data Research*, vol. 25, p. 100212, Jul. 2021, doi: 10.1016/j.bdr.2021.100212.
[50] J. Wu, X.-Y. Chen, H. Zhang, L.-D. Xiong, H. Lei, and S.-H. Deng, "Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimizationb," *Journal of Electronic Science and Technology*, vol. 17, no. 1, Art. no. 1, Mar. 2019, doi: 10.11989/JEST.1674-862X.80904120.
[51] M. Claesen and B. De Moor, "Hyperparameter Search in Machine Learning," *arXiv:1502.02127 [cs, stat]*, Apr. 2015, Accessed: May 24, 2021. [Online]. Available: http://arxiv.org/abs/1502.02127
[52] "React – A JavaScript library for building user interfaces." https://reactjs.org/ (accessed Jun. 24, 2021).
[53] 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.
[54] J. A. B. Somontina, F. Carlo C. Garcia, and E. Q. B. Macabebe, "Water Consumption 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.
[55] S. Seyedzadeh, F. P. Rahimian, I. Glesk, and M. Roper, "Machine learning for estimation of building energy consumption and performance: a review," *Visualization in Engineering*, vol. 6, no. 1, p. 5, Oct. 2018, doi: 10.1186/s40327-018-0064-7.
[56] 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.
[57] 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.
[58] 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.
[59] 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.
[60] 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.

[61] 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.
[62] Z. Hou and Z. Lian, “An Application of Support Vector Machines in Cooling Load Prediction,” in *2009 International Workshop on Intelligent Systems and Applications*, May 2009, pp. 1–4. doi: 10.1109/IWISA.2009.5072707.
[63] A. S. Ahmad *et al.*, “A review on applications of ANN and SVM for building electrical energy consumption forecasting,” *Renewable and Sustainable Energy Reviews*, vol. 33, pp. 102–109, May 2014, doi: 10.1016/j.rser.2014.01.069.
[64] M. Benedetti, V. Cesarotti, V. Introna, and J. Serranti, “Energy consumption control automation using Artificial Neural Networks and adaptive algorithms: Proposal of a new methodology and case study,” *Applied Energy*, vol. 165, pp. 60–71, Mar. 2016, doi: 10.1016/j.apenergy.2015.12.066.
[65] S. L. Wong, K. K. W. Wan, and T. N. T. Lam, “Artificial neural networks for energy analysis of office buildings with daylighting,” *Applied Energy*, vol. 87, no. 2, pp. 551–557, Feb. 2010, doi: 10.1016/j.apenergy.2009.06.028.
[66] B. M. Brentan, E. Luvizotto Jr., M. Herrera, J. Izquierdo, and R. Pérez-García, “Hybrid regression model for near real-time urban water demand forecasting,” *Journal of Computational and Applied Mathematics*, vol. 309, pp. 532–541, Jan. 2017, doi: 10.1016/j.cam.2016.02.009.
[67] 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.
[68] G. Chen, T. Long, J. Xiong, and Y. Bai, “Multiple Random Forests Modelling for Urban Water Consumption Forecasting,” *Water Resources Management: An International Journal, Published for the European Water Resources Association (EWRA)*, vol. 31, no. 15, pp. 4715–4729, 2017.
[69] J. Zhang, Y. Zhu, X. Zhang, M. Ye, and J. Yang, “Developing a Long Short-Term Memory (LSTM) based model for predicting water table depth in agricultural areas,” *Journal of Hydrology*, vol. 561, pp. 918–929, Jun. 2018, doi: 10.1016/j.jhydrol.2018.04.065.
[70] A. A. Nasser, M. Z. Rashad, and S. E. Hussein, “A Two-Layer Water Demand Prediction System in Urban Areas Based on Micro-Services and LSTM Neural Networks,” *IEEE Access*, vol. 8, pp. 147647–147661, 2020, doi: 10.1109/ACCESS.2020.3015655.
[71] L. Dufour, D. Genoud, B. Ladevie, J.-J. Bezian, F. Cimmino, and S. Genoud, “Economic Interest of Heating and Hot Water Prediction System for Residential District,” in *2016 30th International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, Mar. 2016, pp. 827–832. doi: 10.1109/WAINA.2016.174.
[72] A. Candelieri, D. Soldi, and F. Archetti, “Short-term forecasting of hourly water consumption by using automatic metering readers data,” *Procedia Engineering*, vol. 119, pp. 844–853, Jan. 2015, doi: 10.1016/j.proeng.2015.08.948.
[73] X. Ju, M. Cheng, Y. Xia, F. Quo, and Y. Tian, “Support Vector Regression and Time Series Analysis for the Forecasting of Bayannur’s Total Water Requirement,”

Procedia Computer Science, vol. 31, pp. 523–531, Jan. 2014, doi: 10.1016/j.procs.2014.05.298.

[74] 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.

[75] D. Walker, E. Creaco, L. Vamvakeridou-Lyroudia, R. Farmani, Z. Kapelan, and D. Savić, "Forecasting Domestic Water Consumption from Smart Meter Readings Using Statistical Methods and Artificial Neural Networks," *Procedia Engineering*, vol. 119, pp. 1419–1428, Jan. 2015, doi: 10.1016/j.proeng.2015.08.1002.

[76] M. Romano and Z. Kapelan, "Adaptive water demand forecasting for near real-time management of smart water distribution systems," *Environmental Modelling & Software*, vol. 60, pp. 265–276, Oct. 2014, doi: 10.1016/j.envsoft.2014.06.016.

[77] C. Bennett, R. A. Stewart, and C. D. Beal, "ANN-based residential water end-use demand forecasting model," *Expert Systems with Applications*, vol. 40, no. 4, Art. no. 4, Mar. 2013, doi: 10.1016/j.eswa.2012.08.012.

[78] A. Tamang and S. Shukla, "Water Demand Prediction Using Support Vector Machine Regression," in *2019 International Conference on Data Science and Communication (IconDSC)*, Mar. 2019, pp. 1–5. doi: 10.1109/IconDSC.2019.8816969.

[79] S. McGrath, C. Flanagan, L. Zeng, and C. O'Leary, "IoT Personal Air Quality Monitor," in *2020 31st Irish Signals and Systems Conference (ISSC)*, Jun. 2020, pp. 1–4. doi: 10.1109/ISSC49989.2020.9180199.

[80] F. Huber, N. Körber, and M. Mock, "Selena: a Serverless Energy Management System," in *Proceedings of the 5th International Workshop on Serverless Computing*, New York, NY, USA, Dec. 2019, pp. 7–12. doi: 10.1145/3366623.3368134.

[81] E. Andersen, T. Blaalid, H. Engstad, S. Røkenes, and F. T. Johnsen, "A LoRa Mesh Network Asset Tracking Prototype," in *2020 15th Conference on Computer Science and Information Systems (FedCSIS)*, Sep. 2020, pp. 501–510. doi: 10.15439/2020F99.

[82] S. Zhilibayev, A. Ziyashev, A. Zhelambayeva, A. Yessembayev, A. Yazici, and E. Ever, "Low cost smart house implementation with sensory information analysis and face recognition," in *KST 2020 - 2020 12th International Conference on Knowledge and Smart Technology*, Jan. 2020, pp. 91–96. doi: 10.1109/KST48564.2020.9059401.

[83] S. Nagpal, J. Hanson, and C. Reinhart, "A framework for using calibrated campus-wide building energy models for continuous planning and greenhouse gas emissions reduction tracking," *Applied Energy*, vol. 241, pp. 82–97, May 2019, doi: 10.1016/j.apenergy.2019.03.010.

[84] M. Ferreira, J. Ramos, and P. Novais, "Occurrences Management in a Smart-City Context," in *Distributed Computing and Artificial Intelligence, Special Sessions, 15th International Conference*, Cham, 2019, pp. 113–120. doi: 10.1007/978-3-319-99608-0_13.

[85] P.-A. Nguyen, T.-R. Le, P.-L. Nguyen, and C. Pham-Quoc, "IoT-Based Air-Pollution Hazard Maps Systems for Ho Chi Minh City," presented at the Context-Aware Systems and Applications, and Nature of Computation and Communication.

8th EAI International Conference, ICCASA 2019, and 5th EAI International Conference, ICTCC 2019, My Tho City, Vietnam, November 28-29, 2019, Proceedings, Dec. 2019. Accessed: Jun. 24, 2021. [Online]. Available: https://eudl.eu/doi/10.1007/978-3-030-34365-1_6

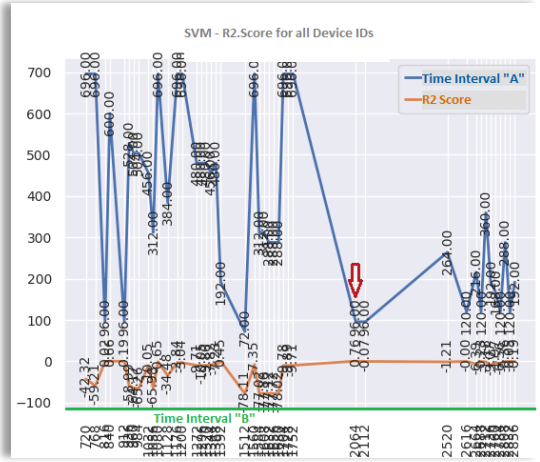
[86] B. Zohuri and F. Mossavar-Rahmani, *A Model to Forecast Future Paradigms: Volume 1: Introduction to Knowledge Is Power in Four Dimensions*. CRC Press, 2019.

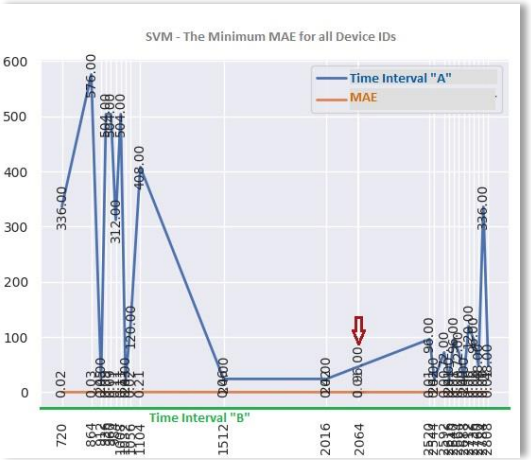
[87] J. B. Rollins, "Foundational methodology for data science." https://whitepapers.theregister.com/paper/view/4593/foundational-methodology-for-data-science (accessed Jul. 20, 2021).

[88] S. Brice, "Collecting Data for Deep Learning Development," *Analytics Vidhya*, Nov. 15, 2020. https://medium.com/analytics-vidhya/a1c43c7e8713 (accessed Aug. 14, 2021).

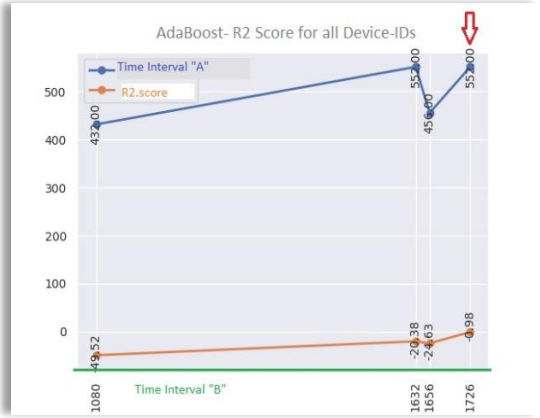
Appendix A The Results of all Runs

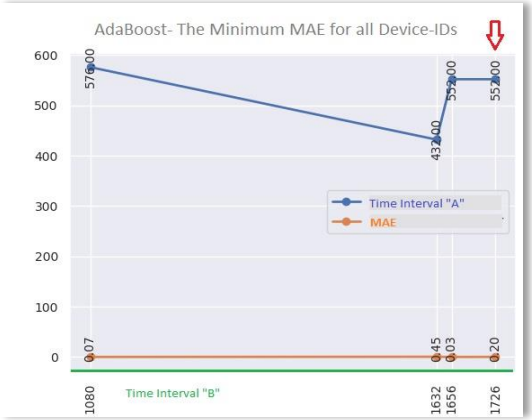
- **SVM Algorithm Results**



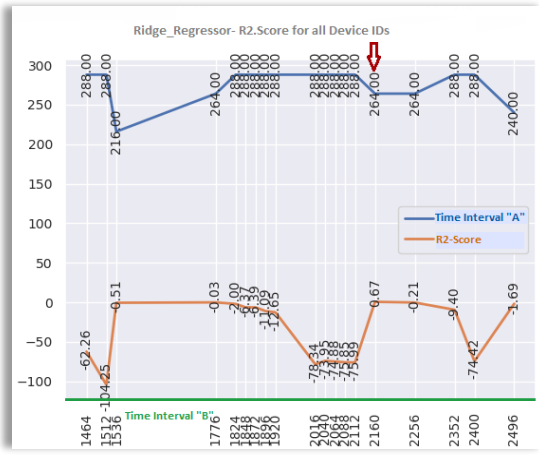


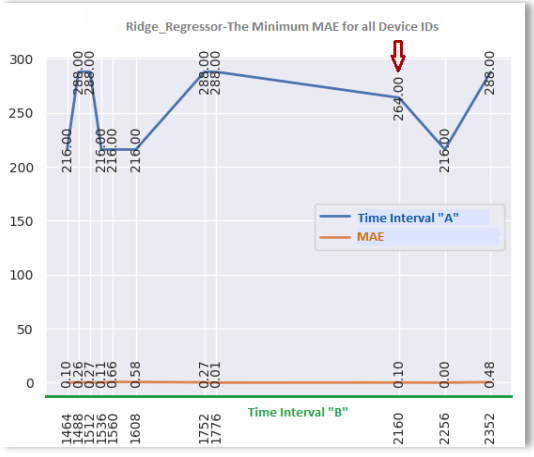
- **AdaBoost Algorithm Results**



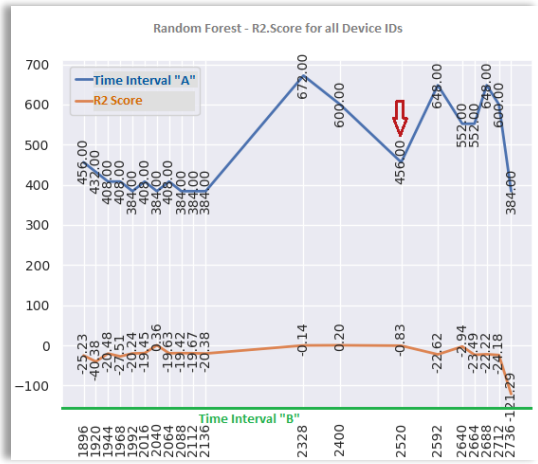


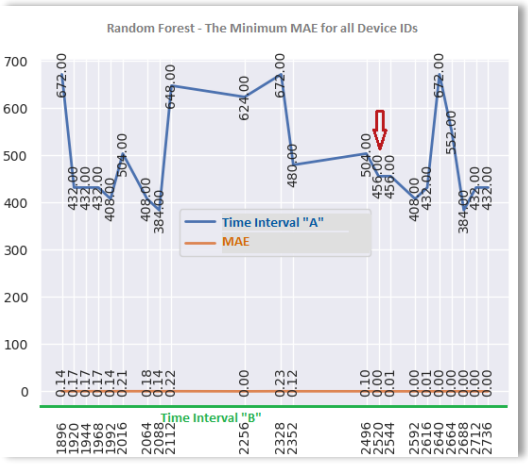
- **Ridge Regressor Algorithm Results**



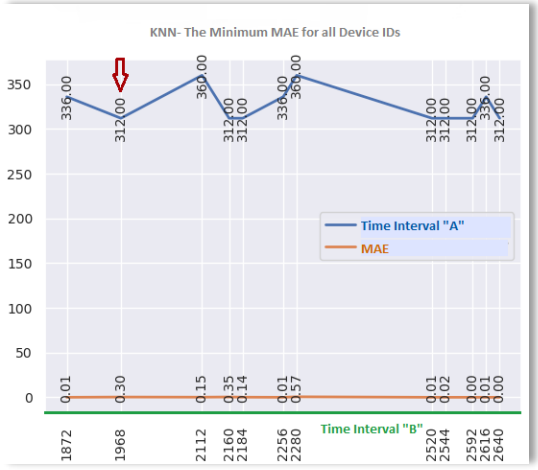


- **RF Algorithm Results**





- **KNN Algorithm Results**



- **XGBoost Algorithm Results**