Journal Homepage: -www.journalijar.com

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH (IJAR)

Article DOI:10.21474/IJAR01/17711
DOI URL: http://dx.doi.org/10.21474/IJAR01/17711

RESEARCH ARTICLE

ENVIRONMENTAL POLLUTION ANALYSIS AND PREDICTION OF INFLUENTIAL FACTORS: A DATA-DRIVEN INVESTIGATION

Md Redoan Hosen, Md Borhan Hosen, Afzalur Rahaman, Yasin Arafat and Abhijit Pathak
Department of Computer Science and Engineering, BGC Trust University Bangladesh, Chandanaish - 4381, Chattogram, Bangladesh.

Manuscript Info

Manuscript History
Received: 15 August 2023
Final Accepted: 18 September 2023
Published: October 2023

Key words:-
Big Data, Linear Regression, Hive, Map-Reduce, Machine Learning, Environment Pollution, Hadoop

Abstract

Environmental pollution is a pressing concern affecting humanity and other Earth species today. The escalation of air pollution due to heavy traffic density and industrial effluents in urban areas poses a significant threat to human life and the environment. The Air Quality Index (AQI) is employed as a crucial metric to gauge the extent of air pollution in a city. Apart from industrial and traffic emissions, meteorological factors, including wind speed, wind direction, temperature, humidity, and total precipitation, play a pivotal role in shaping a city's pollution levels. Mitigating the harm caused by air pollution is contingent on the ability to predict the factors influencing air pollutant levels. This predictive capability enables issuing advanced warnings to citizens or implementing precautionary measures for their protection. This paper focuses on forecasting temperature, humidity, wind speed, wind direction, and total precipitation in the town of Basel, Switzerland, on specific future dates, utilizing historical data maintained by local authorities. The methodology leverages Hive tools and MapReduce frameworks and employs machine learning techniques to predict future values of desired parameters. Various regression techniques, including Lasso, Linear Regression, Ridge Regression, and Polynomial Regression, were evaluated for their performance in predicting air quality factors. Polynomial Regression emerged as the preferred choice due to its superior performance. To enhance user accessibility, a user-friendly graphical interface (GUI) has been developed to input data and visualize predicted results. This study presents a comprehensive approach to address the critical issue of air pollution by providing accurate predictions of meteorological factors, thus enabling proactive measures to safeguard public health and the environment.

Introduction:-

The global rise in ground-level air pollutant concentrations is closely linked to economic development and societal progress. The proliferation of industries within countries has exacerbated air quality (AQ) conditions, necessitating the widespread deployment of AQ monitors to monitor and assess pollution levels. These monitors collect data from various devices and locations, rendering the analysis and prediction of future AQ conditions a formidable and time-sensitive task. Local environmental and health agencies now face the critical challenge of calculating and evaluating

Corresponding Author:- Abhijit Pathak
Address:- Assistant Professor,Department of Computer Science and Engineering, BGC Trust University Bangladesh.

air pollutant concentrations [1][2]. To tackle this challenge effectively, there is a pressing need for a technique capable of rapidly accessing and manipulating vast datasets. While several data mining techniques have been employed in the past, such as sampling, cluster analysis (CA), and Principal Component Analysis (PCA), each has its limitations when dealing with large databases:

1. PCA, which aids in visualizing data patterns and identifying correlations, is crucial for discerning potential emission sources.
2. Cluster analysis identifies grouping patterns among pollution control stations and presents results through Dendrograms [2].
3. Sampling, although used for data reduction, comes with various drawbacks, including the risk of information loss and variations in sample quality [3].
4. Feed-Forward Artificial Neural Networks (ANNs), widely used for predicting air pollutant concentrations, often suffer from issues like slow convergence rates and local minima [1].

In the era of big data, a popular term used to describe complex and extensive datasets, traditional data processing techniques fall short [4]. The generation of enormous volumes of data across all fields has led to the rapid adoption of big data analytics. This encompasses data analysis, capture, creation, search, sharing, storage, transfer, visualization, querying, and information privacy management [5]. Scientists, businesses, and governments alike encounter significant challenges when dealing with large datasets. As of 2018, a staggering 10 exabytes (10×10^18) of data were generated daily. Coping with the storage of such colossal datasets has been facilitated by the doubling of data storage capacity roughly every forty months since the 1980s. Traditional relational database management systems (RDBMS) and other conventional techniques struggle to handle big data [5]. Consequently, the evolution of MapReduce, a technique that harnesses massive parallelism across thousands of servers, has emerged as a crucial tool for managing and processing big data [6]. Map-Reduce is a relatively recent but widely embraced computing paradigm for analyzing extensive datasets. It is gaining momentum across various industries and academia, especially in domains requiring substantial data analysis. Its popularity arises from its effectiveness and user-friendly nature [3]. In this paper, we explore the simplicity and ubiquity of MapReduce as a solution to address the challenges posed by big data in the context of air quality monitoring.

Background Study:-

The study presented in the paper uses ML techniques to predict air quality in 23 Indian cities. The authors preprocessed the air pollution data by reducing the noise present in the data and solved the data imbalance problem with a resampling technique. They also selected key features through correlation analysis. The study employed five machine learning models to predict air quality: Gaussian Naive Bayes, Support Vector Machine, Random Forest, Decision Tree, and XGBoost. The performances of these models were evaluated and compared through established performance parameters. The authors show that the XGBoost model performed best among other models and achieved the highest linearity between predicted and actual data. The exploratory data analysis in the study aimed to find various hidden patterns present in the dataset. The researchers analyzed the statuses and trends of air pollutants over the past six years (2015-2020), explored the distribution of pollutants in the air along with the top-six polluted cities with their average AQI values, and estimated the top four pollutants which are directly involved in increasing the AQI values. This study can be used as a reference for researchers who are interested in using machine learning techniques to predict environmental pollution and identify influential factors [8].

This study proposes a combination of traditional statistical models and ML algorithms to predict PM 2.5 concentrations in Dhaka, Bangladesh accurately. The authors highlight the limitations of conventional statistical models when dealing with a large number of predictors, particularly when these predictors are dependent due to their increasing complexity. On the other hand, machine learning algorithms, such as Deep Neural Networks, Support Vector Regression, and Random Forest techniques, are emerging globally and are widely used to capture the complex relationships between parameters and demonstrate satisfactory results in estimating PM 2.5 levels. The study also emphasizes the importance of investigating the complex, nonlinear relationship between large datasets of environmental, meteorological, and air pollution factors. The authors suggest that machine learning can provide helpful information that government officials and policymakers can use to issue early alerts about air pollution incidents and protect the public from health risks. Overall, this study provides practical insights into the use of data-driven approaches to analyze and predict environmental pollution and its influential factors. It can serve as a helpful literature review for researchers interested in exploring similar topics [9].

The author outlines the importance of air quality monitoring and preservation in industrial and urban areas and the adverse effects of air pollution caused by transportation, electricity, fuel use, etc. The authors mention several parameters; one of the parameters used is PM10 concentration, which is measured using a DustTrak meter and is used to calibrate the algorithm for air quality evaluation. The authors also mention atmospheric reflectance and sun radiation as parameters used in the newly developed algorithm for air quality evaluation. In addition to these parameters, other air pollutants are also measured and used in air quality evaluation. Several air quality monitoring stations provide readings for NO2, SO2, CO, and O3. Big data analytics and machine learning models use these parameters to conduct air quality evaluation and prediction. The authors also discuss the challenges and future research needs in air quality evaluation. One of the challenges is the issue of data quality and validation, which affects the accuracy of air quality evaluation and assessments due to device faults, battery issues, and sensor network communication problems. Another challenge is the nature of the dynamic wind flow, both single-input time series and multiple-input time series, and dynamic quality impacts on different atmospheric levels. Future research needs include big data quality assurance research in data quality modeling, automatic real-time validation methods, and tools to increase the accuracy of air quality evaluation [10]. The study focuses on three main air pollutants: O3, PM2.5, and SO2. These variables contain air temperature, relative humidity, wind speed and direction, wind gust, precipitation collection, visibility, dew point, wind cardinal movement, pressure, and weather conditions. The authors used machine learning algorithms to predict air pollutant concentrations, including random forest, gradient boosting, and deep learning models. They found that the deep learning model outperformed the other models regarding accuracy and efficiency. The authors also used feature importance analysis to identify the most influential factors affecting air pollutant concentrations. They found that temperature, wind speed, and relative humidity were the most important factors for predicting O3 concentrations, while wind speed, temperature, and pressure were the most important factors for predicting PM2.5 concentrations. For SO2 concentrations, wind speed, temperature, and wind direction were the most important factors. Overall, the study demonstrates the potential of machine learning algorithms for predicting air pollutant concentrations. The authors suggest that their approach could be used to develop real-time air quality forecasting systems that could help mitigate air pollution's adverse health effects [11]. Based on meteorological data, the authors use machine learning algorithms to predict the levels of various air pollutants, including CO, O3, NO2, SO2, PM 2.5, and PM 10. They highlight the importance of continuous monitoring of air quality due to the hazardous effects of air pollution on human health and the ecological balance. The authors identify gaps in the literature, noting that previous studies have only implemented the prediction of PM2.5. In contrast, the authors propose a technique to predict all the pollutants mentioned above with the help of meteorological data for better prediction. The paper presents a flow chart for the proposed approach, which involves collecting data from pollution monitoring stations using machine learning algorithms to predict future pollutants based on past data. The paper discusses the use of different machine learning algorithms, including linear regression, decision tree, and random forest, to predict air quality index and the levels of various air pollutants. The results depict that the random forest algorithm gives a better prediction of the air quality index compared to linear regression and decision tree algorithms. For the SO2 prediction, the prediction probability of linear regression is 0.125, the decision tree is 0.8060, and the random forest regression is 0.856 [12].

Big Data Analysis:-

The exponential growth in data accumulation has presented a significant challenge: how to effectively analyze vast datasets. The field of big data analytics is a dynamic one, marked by constant evolution and a strong interest in emerging technologies like MapReduce, Hadoop, Hive, and extensions of MapReduce for existing relational database management systems.

Initially, big data was characterized by three primary attributes: volume, variety, and velocity. However, over time, it has expanded beyond these initial three Vs and is now recognized by ten different characteristics. These characteristics serve as a framework for understanding both the challenges and benefits of big data initiatives. They are presented in Table 1 as follows:

Table 1:- Challenges and advantages of big data initiatives.

Sl. No.	**Characteristic**	**Features**	**Challenges and Skill Responses**
i.	Volume	Tracks data generated over recent years	The internet has led to a tremendous increase in global data production, resulting in large volumes of data. The NoSQL approach presents a significant challenge.
ii.	Velocity	High-speed data production	The rapid growth of IoT contributes to increased data

		growth	production velocity.
iii.	Variety	Data ranges from unstructured to structured formats with redundancies	Flexible data models have evolved due to changes in data collection methods, posing stark differences from traditional models and language query processing.
iv.	Variability	Consists of data from multiple disparate sources and types	Data is collected and loaded into databases at inconsistent speeds by Big Data technologies.
v.	Veracity	Focuses on data source reliability	The increment in volume, velocity, variety, and variability has led to a decline in veracity (confidence or trust in the data).
vi.	Value	The analysis increases data value and aids in consumer behavior analysis	Challenges arise when transforming raw data into valuable information for decision-making and business needs.
vii.	Validity	Refers to data accuracy and correctness for its intended use	Ensures the reliability, consistency, and quality of the data in use.
viii.	Vulnerability	Raises concerns about Big Data security	Any data breach can have serious consequences.
ix.	Volatility	Essential to consider data unpredictability	Rules are needed to ensure data availability and quick retrieval. Storing and retrieving big data become costlier and more complex.
x.	Visualization	Implements dynamic visual changes	Existing data visualization tools face restrictions due to in-memory technology, limited scalability, functionality, and extended response times.

This table provides an overview of the characteristics of big data, the features associated with each, and the challenges faced, along with suggested skill responses for managing these challenges [13].

In Hadoop, the MapReduce programming model is used to process and analyze data. The primary components of the Map-Reduce framework are the Mapper and Reducer classes, and these classes are used to define the logic for processing data.

An overview of the Map-Reduce framework's operation is provided below:

1. Mapper Class (‘org.apache.Hadoop.mapreduce.Mapper’):

- The **‘Mapper’** class is responsible for the map phase of the Map-Reduce job.
- It takes a set of input key-value pairs and processes them to generate a set of intermediate key-value pairs.
- The **‘Mapper’** class defines the ‘**map’** method, which you need to implement. This method is called once for each input key-value pair, and you can write custom code to process the data.
- The output from the ‘**map’** method is a set of intermediate key-value pairs.

2. Reducer Class (‘org.apache.Hadoop.mapreduce.Reducer’):

- The **‘Reducer’** class handles the reduce phase of the Map-Reduce job.
- It takes the intermediate key-value pairs generated by the ‘**Mapper’** class and processes them to produce the final output key-value pairs.
- The **‘Reducer’** class defines the ‘**reduce**’ method, which you need to implement. This method is called once for each unique intermediate key, and you can write custom logic to aggregate and process the values associated with that key.
- The output from the ‘**reduce**’ method is the final set of key-value pairs, which is typically the result of the Map-Reduce job [14].

Here's a simplified example of a Mapper and a Reducer class in Java for a word count Map-Reduce job:

Mapper Class:

```
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;
public class WordCountMapper extends Mapper<LongWritable, Text, Text, LongWritable> {
public void map(LongWritable key, Text value, Context context) throws IOException, InterruptedException {
// Tokenize the input text and emit key-value pairs for each word
```

```
        String[] words = value.toString().split(" ");
        for (String word: words) {
context.write(new Text(word), new LongWritable(1));
        }
    }
}
```

Reducer Class:

```
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

public class WordCountReducer extends Reducer<Text, LongWritable, Text, LongWritable> {
    public void reduce(Text key, Iterable<LongWritable> values, Context context) throws IOException,
InterruptedException {
        // Sum the values for each word to get the word count
        long sum = 0;
        for (LongWritable value: values) {
            sum += value.get();
        }
context.write(key, new LongWritable(sum));
    }
}
```

Data processing in Hadoop is based on key-value pairs, where a key function is an identifier for its corresponding value. The Map-Reduce API offers robust capabilities for bulk processing, high availability, and parallel processing of massive datasets. The Map-Reduce workflow is a multi-step procedure, and the final output is typically stored in the Hadoop Distributed File System (HDFS) with data replication for fault tolerance. The coordination and management of Map-Reduce jobs across the Hadoop cluster are overseen by components like the Job Tracker and Task Tracker. Here is a concise description of these elements [16]:

Job Tracker: This component is fundamental to the Hadoop ecosystem. These are its principal responsibilities:

1. Scheduling and managing Map-Reduce jobs across various nodes in the Hadoop cluster.
2. Tracking the progress of each Map and Reduce task.
3. Ensuring fault tolerance by rescheduling tasks in case of failures.
4. Managing the allocation of resources to different tasks.
5. Maintaining an overview of the entire job's progress.

Task Tracker: Task Trackers are worker nodes in the Hadoop cluster. They are responsible for executing the actual Map and Reduce tasks as directed by the Job Tracker. Key functions of Task Trackers include [17]:

1. Receiving and executing tasks from the Job Tracker.
2. Reporting the status and progress of tasks back to the Job Tracker.
3. Handling data locality, meaning they try to process data that is already present on the same node to reduce data transfer over the network.
4. Managing task execution and resource allocation.
5. The Map-Reduce workflow in Hadoop involves multiple phases, including data input, mapping, shuffling, reducing, and final output. During these phases, data is processed in a distributed and parallel manner, making Hadoop an effective tool for managing large-scale data processing projects.

The final results of a Map-Reduce job are typically stored in the Hadoop Distributed File System (HDFS), a distributed and fault-tolerant file storage system designed to handle the massive data generated and processed within the Hadoop cluster. Data replication within HDFS ensures data durability and availability, even in the face of node failures.

Data in the Map-Reduce paradigm is organized as key-value pairs, creating a logical association between two data elements: the key and its corresponding value. The key (denoted as 'k') serves as an identifier for the associated value. The Map-Reduce API offers robust capabilities for tasks such as batch processing, ensuring high availability, and facilitating parallel processing of substantial data volumes. The Map-Reduce workflow proceeds through a series of distinct phases, which are integral to the data processing operation. The final results are typically stored in the Hadoop Distributed File System (HDFS), a storage system known for its data replication to ensure fault tolerance. The orchestration of Map-Reduce jobs is managed by the Job Tracker, a pivotal component within a Hadoop cluster. This key role includes the scheduling and monitoring of all Map-Reduce tasks running on various nodes across the cluster. The Job Tracker maintains oversight of the entire Map-Reduce job lifecycle. Execution of the actual map and reduced tasks is carried out by the Task Tracker. This component performs the computational work, processing the data according to the Map-Reduce paradigm. Task Trackers are distributed across the Hadoop cluster to achieve parallel processing and efficient data analysis. The Hadoop ecosystem, characterized by these elements, offers a scalable and reliable framework for distributed data processing and analysis, making it an invaluable tool for handling substantial datasets and demanding data processing tasks [5].

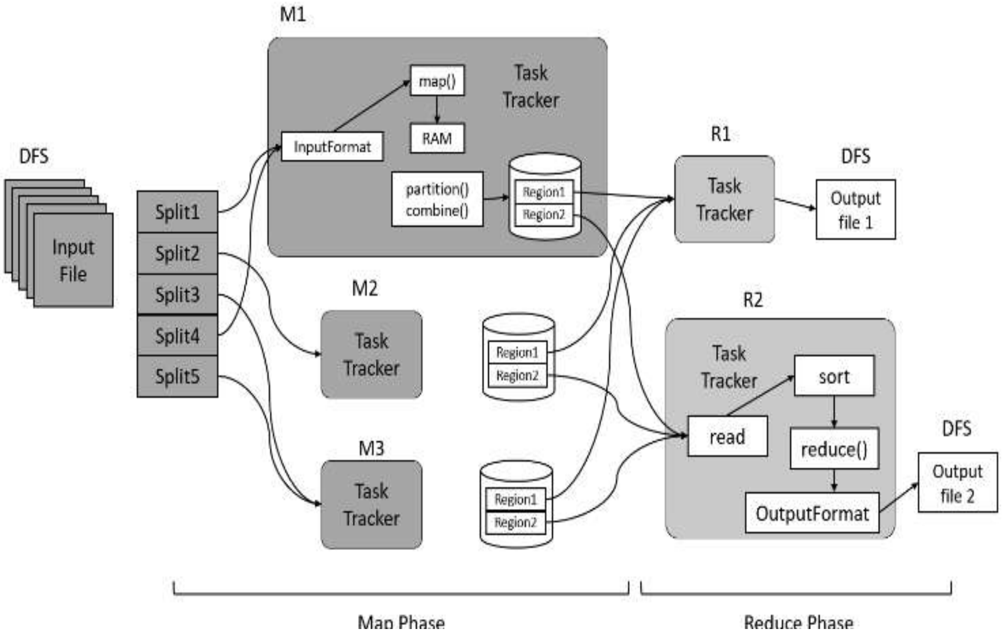


Fig.1:- HadoopMap-ReduceArchitecture.

Machine Learning:-

Machine Learning plays a pivotal role in recent advancements. It diverges from traditional programming by utilizing learning algorithms to extract insights from data and acquire the capability to perform new tasks. These machine learning algorithms are typically categorized into three distinct groups: supervised, unsupervised, and semi-supervised. We can illustrate these categories in the following way [18]:

Supervised Learning

Provided:

Input: (x1, x2, ...)

Output: (y1, y2, ...)

The goal is to discover a function that accurately models the relationship between the input and output in a way that allows

for generalization. Depending on the nature of the problem, the output can take one of two forms:

i) It can represent a class label (in cases involving classification).

ii) It can represent a real number (in cases involving regression).

Unsupervised Learning

Provided:

Input: (x1, x2, ...)

Output: However, no target output is provided.

In situations where target output labels are absent, various methods can be employed, including:

i) Density estimation, which involves estimating an underlying Probability Density Function to make predictions.

ii) K-means clustering, used for classifying unlabeled data with real numerical values.

iii) K-mode clustering, suitable for classifying unlabeled data with categorical values and categories.
These methods are applied to uncover patterns or structures within the data when explicit output targets are not given [29].

Semi-supervised Learning
It involves the estimation of a function using both labeled and unlabeled data. As a result, this approach is often costlier to acquire labeled data, whereas obtaining unlabeled data is a more cost-effective process.

In practice, supervised algorithms are predominantly employed. Researchers have proposed various machine learning algorithms, yet no single algorithm can universally address all problems. A suitable way to describe learning algorithms is as follows: they involve creating and refining a target function (f) that can link input values (X) to an output variable (Y), represented as Y = f(X) [5]. In essence, learning refers to predicting Y values based on given X values. Consequently, the most commonly utilized model of machine learning focuses on learning to map or predict Y values based on a limited set of X examples. This approach is aptly referred to as predictive modeling or predictive analysis, with the primary goal of making highly accurate predictions about future values [15].

Factors Affecting the Environment and Pollution:-
Environmental pollution, categorized into various forms, represents one of the gravest challenges facing humanity and the planet's ecosystems today. This predicament arises from the excessive exploitation of natural resources, outstripping nature's ability to replenish itself, and consequently leading to pollution of the air, land, sound, and water.

In urban areas, particularly, the confluence of heavy traffic volumes and industrial emissions has escalated air pollution to a critical hazard for human well-being. Air quality in cities is typically gauged using an "Air Quality Index" (AQI), a metric that emphasizes the health impacts experienced within a short time frame after inhaling contaminated air. The Environmental Protection Agency (EPA) regulates AQI based on five major air pollutants, which are mandated by the Clean Air Act. These pollutants include:
(a) Carbon monoxide
(b) Sulfur dioxide and nitrogen dioxide
(c) Ground-level ozone
(d) Particle pollution (also known as particulate matter) [18].

Particulate matter, categorized as PM10 (particulate matter with a diameter of 10 micrometers or less) and PM2.5 (particulate matter with a diameter of 2.5 micrometers or less, often referred to as fine particles), are key components of air pollution. To illustrate, 40 fine particles could span the width of a human hair [19]. Air quality is not solely influenced by industrial activities, traffic emissions, and construction projects; meteorological factors such as temperature, humidity, wind speed, wind direction, and total precipitation also exert a significant influence. Observations indicate that when temperatures are low, humidity is high, and wind speed is sluggish, AQI tends to rise. This is because air pollutants become trapped and struggle to disperse under such conditions. Conversely, during periods of high temperatures,low humidity, and brisk winds, pollutants disperse swiftly, leading to a marked decrease in AQI [28].

The adverse impacts of air pollution can be mitigated substantially by predicting the likely levels of air pollutants. Such predictions could enable warnings to be issued to residents or the implementation of precautionary measures, thereby reducing the detrimental effects of air pollution on urban populations.

Project Description
In light of the information outlined earlier, this project aims to harness the capabilities of big data and machine learning methodologies to forecast the meteorological conditions of a specific city using historical data for that location. The project unfolds through the following steps:

Accumulation of Meteorological Data: Gather historical meteorological data for a town in Switzerland, specifically Besil, covering crucial parameters such as temperature, humidity, wind speed, wind direction, and total precipitation over the past five years.

Data Preprocessing:
Prepare and clean the acquired data to ensure its accuracy and consistency.

Data Processing Using Hadoop and Hive:
Utilize Hadoop and Hive, powerful data processing tools, to efficiently process and summarize the input data.

Machine Learning Approach - Polynomial Regression:
Employ a machine learning technique known as Polynomial Regression to model and predict future values of meteorological parameters for the city, encompassing temperature, humidity, wind speed, wind direction, and total precipitation.

Early Warning System:
With the assistance of predicted meteorological parameters, government agencies and relevant authorities can proactively offer solutions and precautionary measures to mitigate the impact of adverse weather conditions on the well-being of the city's residents. This project leverages the potential of big data and machine learning to enhance meteorological predictions and, in turn, facilitate timely responses and measures to safeguard the populace from the effects of changing weather conditions [27].

Existing System
In the year 2000, Seisint Inc. developed a distributed file-sharing framework based on C++. This framework was designed for the storage and querying of data. It could store and distribute all three categories of data, including structured, semi-structured, and unstructured data, across multiple servers. Querying of data was accomplished using ECL, a modified version of C++, which required applying a schema during the reading process to structure the stored data when conducting queries. In 2004, LexisNexis acquired Seisint Inc., and in 2008, it acquired ChoicePoint, Inc., along with its high-speed parallel processing platform [21].

The high-speed parallel processing platform from ChoicePoint and the ECL-based system were merged into High-Performance Systems in 2011, and the resulting platform was open-sourced under the Apache v2.0 License. In 2004, Google introduced the MapReduce concept.

In 2017, R. A. Ramadan provided an overview of big data tools available in the market. Additionally, various data storage and management tools and data analytics platforms were discussed. Storage and management tools included Cloudera, MongoDB, Apache HBase, Hypertable, Hive, and others. Data analytics platforms encompassed Hadoop, MapR, and IBM data analytics. However, the most popular technology used for data analytics across diverse application domains remains Hadoop MapReduce [20].

Kaur-Jindal's survey in 2017 explored prominent machine learning algorithms, including Support Vector Machines (SVM), Artificial Neural Networks (ANN), Decision Trees, Naïve Bayes, regression, and K-means. The paper also highlighted suitable application areas for these techniques. A brief description of the key algorithms is provided below:

Support Vector Machine (SVM):
A widely popular technique for machine learning, primarily employed for classification. It operates on the principle of maximizing the margin between classes, effectively reducing classification errors.

Artificial Neural Network (ANN):
Inspired by the structure of the human nervous system, ANN attempts to replicate the functioning of neurons. Understanding ANN requires knowledge of the neural cell's operation.

Decision Tree:
A data structure that organizes attributes by sorting and using their values for classification. It consists of nodes representing attribute groups and branches indicating accepted node values.

Naïve Bayes:
Developed primarily for text classification, it is also applied in clustering tasks. It generates probabilistic models in the form of Bayesian networks based on the probability of occurrences [22].

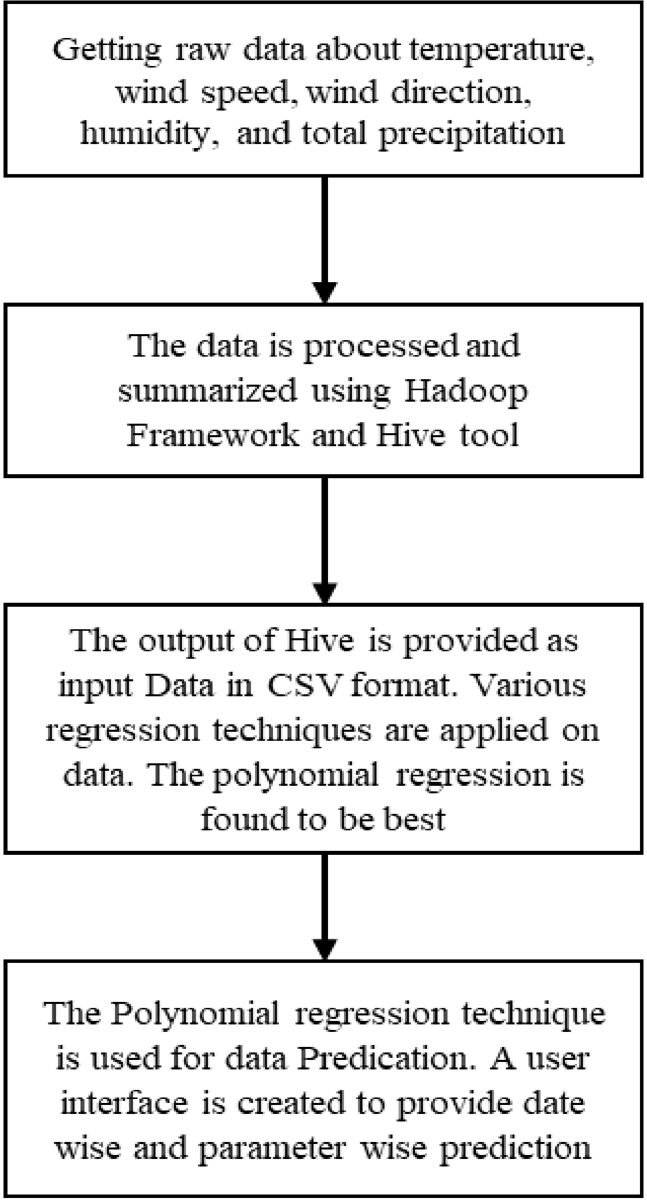
Regression:
This model utilizes existing data to create a customized model for predicting output values and features based on input values. However, it necessitates training data for model development.

K-Means Clustering:
An unsupervised learning technique that automatically forms groups of similar data points. Named "k-Means" because it creates k distinct clusters, with the center of each cluster being the mean value of the data points within that cluster [23].

Methodology:-

Workflow of Proposed System

This project proceeded through several key steps, as depicted in Figure 2:



Getting raw data about temperature, wind speed, wind direction, humidity, and total precipitation

The data is processed and summarized using Hadoop Framework and Hive tool

The output of Hive is provided as input Data in CSV format. Various regression techniques are applied on data. The polynomial regression is found to be best

The Polynomial regression technique is used for data Predication. A user interface is created to provide date wise and parameter wise prediction

Fig. 2:-Workflow of the process.

a) Data Retrieval: We collected five years' worth of data (2013-2019) encompassing variables such as wind speed and direction, temperature, humidity, and total precipitation for Besil, a town in Switzerland [28]. The data was freely available for download from a website [24].

b) Data Processing with Hive: The acquired data was then input into the Hive tool, using the Map-Reduce technique. Hive further processed and condensed the data into a more usable format.
c) Regression Techniques: To determine the most suitable machine learning technique, we explored four regression methods, including linear regression, polynomial regression, ridge regression, and Lasso.
d) Optimal Technique Selection: After conducting tests, it became evident that the polynomial regression technique produced the most promising results. Consequently, we opted for polynomial regression for data prediction.
e) Applying Polynomial Regression: We applied the polynomial regression technique to the output data generated by Hive, which was saved in a .CSV file. This file served as the input data for the machine learning algorithm.
f) User Input and Prediction: To facilitate predictions, users are prompted to provide a specific date and specify the meteorological parameter they seek to predict. The system then generates the predicted value for the requested meteorological parameter as the output [25].

Selection of Appropriate Algorithm
The training set and the testing set are two separate groups from which we divided the data. On the input data, we applied the following regression techniques to determine the best strategy for making future predictions:
a) Linear Regression
b) Lasso Regression
c) Ridge Regression
d) Polynomial Regression

Both the training and testing phases were executed utilizing all four of these techniques. Notably, it was observed that Polynomial Regression exhibited the lowest error in comparison to the other three techniques. This suggests that Polynomial Regression is the most promising method for making accurate future predictions based on the given data [26].

Prediction Methodology:-

Our approach for making predictions using our dataset involves the following steps:

a. **Data Import:** We began by importing our dataset file and stored it within a variable named "dataset."
b. **Feature Selection:** We chose a specific field or feature from the dataset for model training. This selection could pertain to parameters such as temperature or total precipitation.
c. **Application of Prediction Algorithm:** We applied a prediction algorithm to our selected feature. In this case, we employed Polynomial Regression.
d. **Model Training:** To enable our model to make accurate predictions, we trained it using the training data and the chosen algorithm. This training process is executed using the "fit" keyword.
e. **Prediction:** With a trained model in place, we utilized the "predict" keyword to predict the desired output when presented with test data.
f. **Error Margin Evaluation:** We assessed the accuracy of our predictions by computing the error margin, often using metrics like the mean squared error. This evaluation is facilitated through the "sklearn.metrics" library.
g. **Iterative Refinement:** We iteratively refined the process, cycling through steps 3 to 5 multiple times until we identified the algorithm that consistently produced the lowest error margin.

Following this comprehensive prediction exercise, we concluded that Polynomial Regression was the most effective method, delivering the least error margin. Therefore, we have chosen Polynomial Regression as the model for predicting future data.

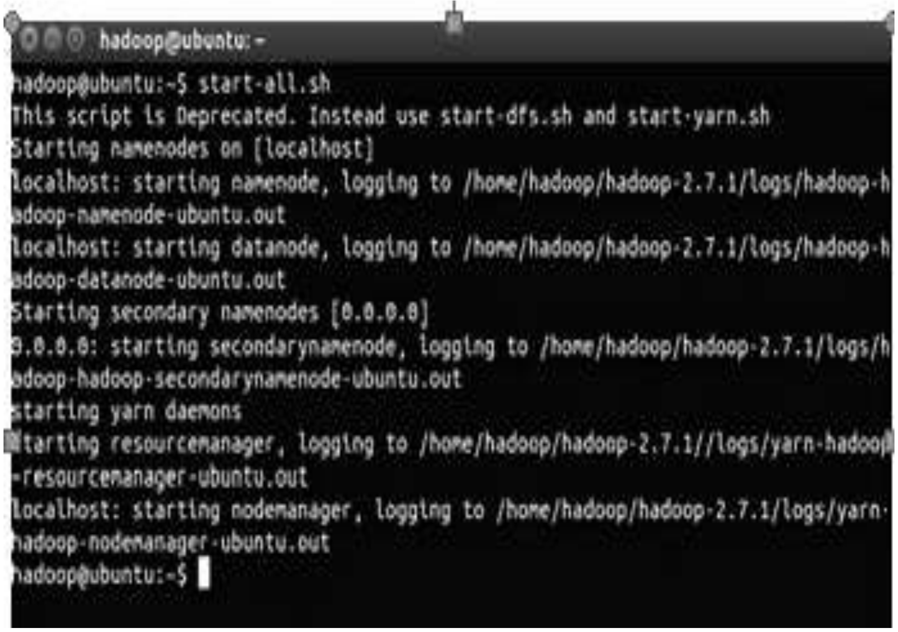
Results and Discussions:-

Input Data

We have selected the Swiss town of Besil as our study location to forecast meteorological variables, such as wind speed and direction, temperature, humidity, and total precipitation, about air pollution and the Air Quality Index (AQI). Our dataset includes the most recent five years of data, from 2013 to 2019, and it includes important factors like wind direction and speed, temperature, humidity, and total precipitation. This data was obtained from the website referenced as [28].

Data Processing

To facilitate the data processing, it's essential to activate the Hadoop framework. The precise queries for initiating the Hadoop framework are presented in the following Figure 3:

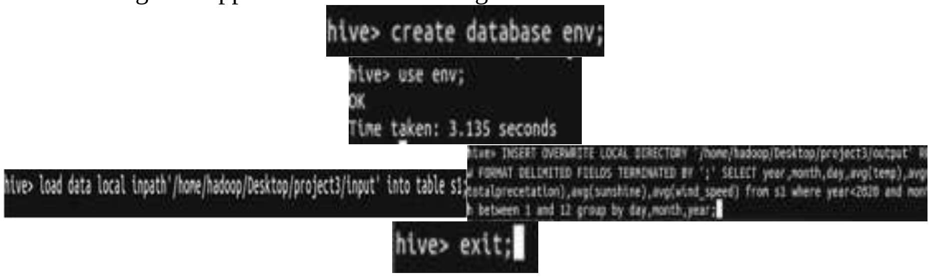


```
hadoop@ubuntu: ~
hadoop@ubuntu:~$ start-all.sh
This script is Deprecated. Instead use start-dfs.sh and start-yarn.sh
Starting namenodes on [localhost]
localhost: starting namenode, logging to /home/hadoop/hadoop-2.7.1/logs/hadoop-h
adoop-namenode-ubuntu.out
localhost: starting datanode, logging to /home/hadoop/hadoop-2.7.1/logs/hadoop-h
adoop-datanode-ubuntu.out
Starting secondary namenodes [0.0.0.0]
0.0.0.0: starting secondarynamenode, logging to /home/hadoop/hadoop-2.7.1/logs/h
adoop-hadoop-secondarynamenode-ubuntu.out
starting yarn daemons
starting resourcemanager, logging to /home/hadoop/hadoop-2.7.1//logs/yarn-hadoop
-resourcemanager-ubuntu.out
localhost: starting nodemanager, logging to /home/hadoop/hadoop-2.7.1/logs/yarn-
hadoop-nodemanager-ubuntu.out
hadoop@ubuntu:~$ 
```

Fig. 3:- Commands to start the Hadoop Framework.

The real data processing must now be done with the Hive tool. The commands for the following actions are shown in the figure 4 below:

a. Making a database; b. Using that database; c. Making a table in the needed format. d. Reading input data from a file into a table e. Processing the supplied data and f. Exiting Hive



```
hive> create database env;
```

```
hive> use env;
OK
Time taken: 3.135 seconds
```

```
hive> load data local inpath'/home/hadoop/Desktop/project3/input' into table s1;
```

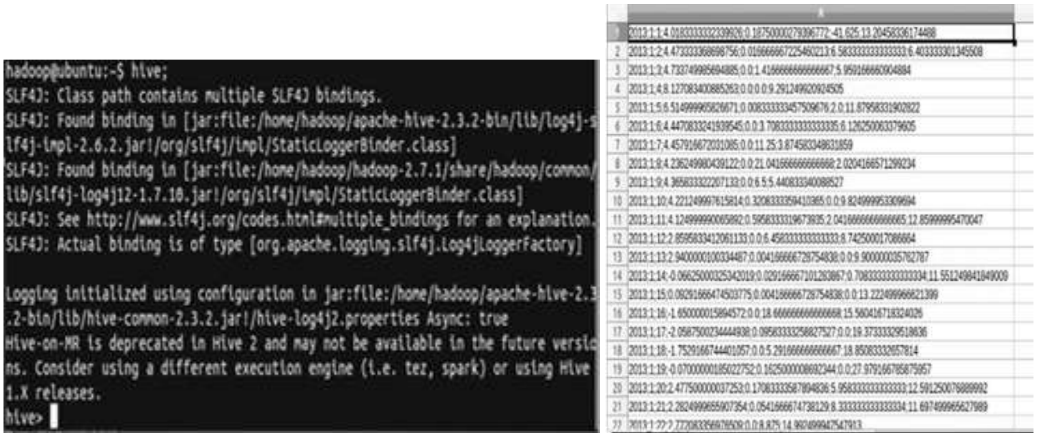
```
hive> INSERT OVERWRITE LOCAL DIRECTORY '/home/hadoop/Desktop/project3/output' RO
W FORMAT DELIMITED FIELDS TERMINATED BY ';' SELECT year,month,day,avg(temp),avg(
totalprecetation),avg(sunshine),avg(wind_speed) from s1 where year<2020 and mont
h between 1 and 12 group by day,month,year;
```

```
hive> exit;
```

Fig. 4:- Commands to start Hive.

Output Data

Upon executing these commands, the meteorological data for Besil Town, which was obtained from meteolube.com, has been processed to meet the required format. The data has been averaged daily for all the specified parameters within the designated time range. The resulting data is available in .CSV format, making it suitable for input into a machine learning algorithm. A sample of this output data is illustrated in Figure 5.



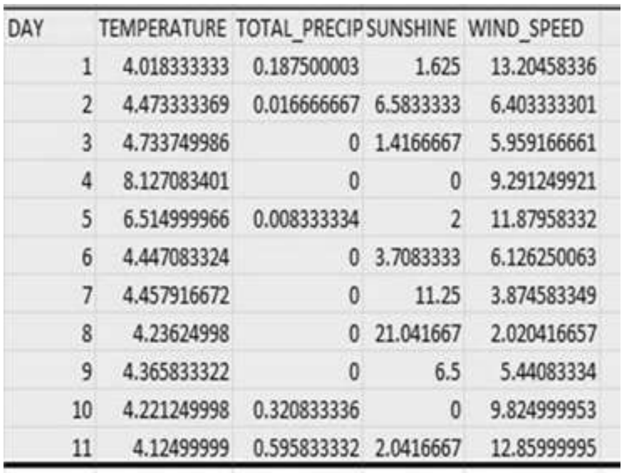
```
hadoop@ubuntu:~$ hive;
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/hadoop/apache-hive-2.3.2-bin/lib/log4j-s
lf4j-impl-2.6.2.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/hadoop/hadoop-2.7.1/share/hadoop/common/
lib/slf4j-log4j12-1.7.10.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]

Logging initialized using configuration in jar:file:/home/hadoop/apache-hive-2.3
.2-bin/lib/hive-common-2.3.2.jar!/hive-log4j2.properties Async: true
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versio
ns. Consider using a different execution engine (i.e. tez, spark) or using Hive
1.X releases.
hive>
```

```
1  2013:1:1:4.0183333322239926:0.18750000279396772:1.625:13.20458336174488
2  2013:1:2:4.473333368698756:0.016666667225460213:6.583333333333333:6.403333301345508
3  2013:1:3:4.733749985694885:0.0:1.4166666666666667:5.959166660904884
4  2013:1:4:8.127083400885263:0.0:0.0:9.291249920924505
5  2013:1:5:6.514999965826671:0.008333333457509676:2.0:11.879583319028822
6  2013:1:6:4.447083324193954 5:0.0:3.7083333333333335:6.126250063379605
7  2013:1:7:4.457916672031085:0.0:11.25:3.874583348631859
8  2013:1:8:4.236249980439122:0.0:21.041666666666668:2.020416657129234
9  2013:1:9:4.365833322207133:0.0:6.5:5.440833340088527
10 2013:1:10:4.221249997615814:0.3208333359410365:0.0:9.824999953309694
11 2013:1:11:4.124999990065892:0.5958333319673935:2.0416666666666665:12.859999954700047
12 2013:1:12:2.8595833412061133:0.0:6.458333333333333:8.742500017086664
13 2013:1:13:2.9400000100334487:0.004166666728754838:0.0:9.900000035762787
14 2013:1:14:-0.06625000325342019:0.029166667101283867:0.7083333333333334:11.551249841849009
15 2013:1:15:0.08291666474503775:0.004166666728754838:0.0:13.222499966621399
16 2013:1:16:-1.650000015894572:0.0:18.666666666666668:15.560416718324026
17 2013:1:17:-2.0587500234444938:0.09583333258827527:0.0:19.373333295186636
18 2013:1:18:-1.7529166744401057:0.0:5.291666666666667:18.850833326057814
19 2013:1:19:-0.07000000185022752:0.1625000008692344:0.0:27.979166785875957
20 2013:1:20:2.477500000037253:0.17083333587894836:5.958333333333333:12.591250076889992
21 2013:1:21:2.2824999665907354:0.0541666674738129:8.333333333333334:11.697499965627989
22 2013:1:22:2.772083356878509:0.0:8.875:14.992499947547913
```

Fig 5:- Processed Output Data.

The initial data obtained from meteoblue.com was in hourly intervals, spanning the last five years. We established a Hive database, imported this data into a table, and subsequently conducted data processing. This processing entailed computing daily averages for all parameters, including temperature, wind speed, wind direction, humidity, and total precipitation. The resultant processed data serves as the input for our machine learning algorithm, which is tasked with forecasting future values of these parameters.

For prediction purposes, the input data originally received from meteoblue.com, provided on an hourly basis, has been transformed and summarized through the Hive tool to a daily frequency. A sample of this input data is illustrated in Figure 6.



DAY	TEMPERATURE	TOTAL_PRECIP	SUNSHINE	WIND_SPEED
1	4.018333333	0.187500003	1.625	13.20458336
2	4.473333369	0.016666667	6.5833333	6.403333301
3	4.733749986	0	1.4166667	5.959166661
4	8.127083401	0	0	9.291249921
5	6.514999966	0.008333334	2	11.87958332
6	4.447083324	0	3.7083333	6.126250063
7	4.457916672	0	11.25	3.874583349
8	4.23624998	0	21.041667	2.020416657
9	4.365833322	0	6.5	5.44083334
10	4.221249998	0.320833336	0	9.824999953
11	4.12499999	0.595833332	2.0416667	12.85999995

Fig. 6:- Input Data for Machine Learning and Prediction.

Conclusion:-

The project at hand is a comprehensive endeavor that involves the analysis, processing, and summarization of historical meteorological data, encompassing parameters such as wind speed and direction, temperature, humidity, and total precipitation for a specific town. The execution of these tasks is made possible through the application of the MapReduce framework and Hive, ensuring efficient data management and processing. The processed data is then harnessed as input for a machine learning algorithm, with a specific focus on employing polynomial regression to predict the values of the aforementioned meteorological parameters for a user-defined date. This selection of polynomial regression as the predictive model follows a rigorous evaluation process that considers alternative techniques, including Lasso, Linear Regression, and Ridge Regression. The availability of a user-friendly interface, which ensures accessibility and simplicity of use for a varied range of users, is a key aspect that improves the project's utility. There are two promising directions for future developments of the concept. First, the project can benefit from the incorporation of a machine-learning technique selection menu, enabling users to choose from a range of available machine-learning methods, aligning with their specific needs or preferences. This enhancement enhances adaptability and customization. Secondly, as part of an expanded scope, the project can delve into the prediction of the Air Quality Index (AQI) for a city on a designated date. This would necessitate the analysis and forecasting of both historical meteorological data and AQI data. Predicting AQI is of paramount importance in addressing air pollution concerns and improving environmental quality in urban areas. By considering these future enhancements, the project aims to become more versatile and applicable to a wider array of users and scenarios, thereby increasing its overall value and impact.

References:-

1. J. Zhang and W. Ding, "Prediction of Air Pollutants Concentration Based on an Extreme Learning Machine" in Int J Environ Res Public Health. 2017 Feb; 14(2): 114.

2. David Núñez-Alonso, Luis Vicente PérezArribas, Sadia Manzoor, and Jorge O. Cáceres, "Statistical Tools for Air Pollution Assessment: Multivariate and Spatial Analysis Studies in the Madrid Region" in Journal of Analytical Methods in Chemistry Volume 2019, Article ID 9753927 https://doi.org/10.1155/2019/9753927.

3. An integrated approach for CURE clustering using map-reduce technique* S Maitrey, CK Jha - Proceedings of Elsevier. ISBN, 2013.

4. V.K. Jain, "Big Data and Hadoop", Khanna Book Publishing, 2017.

5. Sindhi K., Parmar D., Gandhi P. (2019) A Study on Benefits of Big Data for the Healthcare Sector of India. In: Mishra D., Yang XS., Unal A. (eds) Data Science and Big Data Analytics. Lecture Notes on Data Engineering and Communications Technologies, vol 16. Springer, Singapore, 02 August 2018.

6. https://intellipaat.com/tutorial/hadoop-tutorial/big-data overview/

7. Seema Maitrey and C.K. Jha, MapReduce: Simplified Data Analysis of Big Data / Procedia Computer Science 57 (2015) 563 – 571.

8. Kumar, K., &Pande, B. P. (2023). Air pollution prediction with machine learning: a case study of Indian cities. International Journal of Environmental Science and Technology, 20(5), 5333-5348.

9. Islam, A. R. M. T., Al Awadh, M., Mallick, J., Pal, S. C., Chakraborty, R., Fattah, M. A., ... &Elbeltagi, A. (2023). Estimating ground-level PM2. 5 using subset regression model and machine learning algorithms in Asian megacity, Dhaka, Bangladesh. Air Quality, Atmosphere & Health, 1-23.

10. Kang, G. K., Gao, J. Z., Chiao, S., Lu, S., &Xie, G. (2018). Air quality prediction: Big data and machine learning approaches. Int. J. Environ. Sci. Dev, 9(1), 8-16.

11. Zhu, D., Cai, C., Yang, T., & Zhou, X. (2018). A machine learning approach for air quality prediction: Model regularization and optimization. Big data and cognitive computing, 2(1), 5.

12.Pasupuleti, V. R., Kalyan, P., & Reddy, H. K. (2020, March). Air quality prediction of data log by machine learning. In 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS) (pp. 1395-1399). IEEE.

13.Thị, N.; Thanh, N.; Hung, B.Q.; Kế, L.C.; Hưng, L.V.; Hà, P.V.; Thành, Đ.N.; Bằng, P.H.; Chức, M.Đ.; Hà, L. Air Pollution Monitoring and Warning System. In Proceedings of the National Gis Conference, Ha Noi, Vietnam, 12 November 2014.

14.S. Ghemawat et al. The Google file system. ACM SIGOPS Operating Systems Review, 37(5):29–43, 2003.

15. https://stats.stackexchange.com/questions/517/unsupervised- supervised-and-semi- supervised-learning

16.https://towardsdatascience.com/a-tour-of-the-top-10- algorithms-for-machine-learning- newbies-dde4edffae11

17.https://www.sciencedirect.com/topics/earth-and- planetary-sciences/environmental-pollution

18.http://www.npi.gov.au/resource/particulate-matter-pm10- and-pm25

19.R. A Ramadan, "Big Data Tools an Overview", International Journal of Computer & Software Engineering, vol. 2, No.125, pp. 115, 2017.

20. S. Kaur, S. Jindal, "A survey on Machine Learning Algorithms", International Journal of Innovative Research in Advanced Engineering (IJIRAE), Vol. 3, No. 11, pp. 6-14, 2016

21. A. Dey, "Machine Learning Algorithms: A Review in International Journal of Computer Science and Information Technologies, Vol. 7 (3), 2016, 1174-1179.

22. Krzysztof Siwek, StanisławOsowski, "Data mining methods for prediction of air pollution", in July 2016. DOI: 10.1515/amcs-2016-0033.

23. Elangasinghe, M. A., Singhal, N., Dirks, K. N., Salmond, J. A., & Samarasinghe, S. (2014). Complex time series analysis of PM10 and PM2.5 for a coastal site using artificial neural network modeling and k-means clustering. Atmospheric Environment, 94(0), 106–116. https://doi.org/10.1016/j.atmosenv.2014.04.051.

24. Marko Debeljak, SašoDžeroski, “Decision Trees in Ecological Modelling”, in the book: Modelling Complex Ecological Dynamics, January 2011, DOI: 10.1007/978-3-642- 05029-9_14
25. SeunDeleawe, Jim Kusznir, Brian Lamb, and Diane J. Cook, “Predicting Air Quality in Smart Environments”, in J Ambient Intell Smart Environ. 2010; 2(2): 145–152, doi: 10.3233/AIS-2010-0061
26. Ruiyun Yu, Yu Yang, Leyou Yang, Guangjie Han and Oguti Ann Move, “RAQ–A Random Forest Approach for Predicting Air Quality in Urban Sensing Systems”, received: 30 September 2015; Accepted: 7 January 2016; Published: 11 January 2016.
27. Kingsy Grace, Manimegalai Rajkumar, M.S. Geetha Devasena, Baseria N. Raabiathul, “Air pollution analysis using enhanced K-Means clustering algorithm for real-time sensor data” in November 2016, DOI: 10.1109/TENCON.2016.7848362.
28. https://meteoblue.com.
29. Maitrey, S., Chandiramani, C., Gupta, R., &Somvanshi, D.. Analyzing and Predicting Factors Affecting Environmental Pollution. URL: https://www.scribd.com/document/463074143/Analyzing-and-Predicting-Factors-Effecting-Environmental-Pollution