A MINI-PROJECT REPORT

ON

“Air Quality Prediction and Mitigation web app using IOT and ML”

BY

Soham Athavale (A605)
Sarang Gajare (A625)
Mihir Kate (A647)
Armaan Moledina (A663)

Under the guidance of

Dr. Shikha Gupta

MANJARA CHARITABLE TRUST
RAJIV GANDHI INSTITUTE OF TECHNOLOGY
Juhu-Versova Link Road Versova, Andheri(W), Mumbai-53

Department of Computer Engineering

University of Mumbai
April 2024

MANJARA CHARITABLE TRUST

RAJIV GANDHI INSTITUTE OF TECHNOLOGY

Juhu-Versova Link Road Versova, Andheri(W), Mumbai-53

C E R T I F I C A T E

Department of Computer Engineering

This is to certify that

1. **Soham Athavale (A605)**
2. **Sarang Gajare (A625)**
3. **Mihir Kate (A647)**
4. **Armaan Moledina (A663)**

Have satisfactorily completed this project entitled

"Air Quality Prediction and Mitigation web app using IOT and ML"

Towards the partial fulfilment of the

THIRD YEAR BACHELOR OF ENGINEERING
IN
(COMPUTER ENGINEERING)

as laid by University of Mumbai.

Guide	**H.O.D.**
Dr. Shikha Gupta	**Prof. Sunil P. Khachane**

Principal

Dr. Sanjay Bokade

Project Report Approval for T. E.

This project report entitled "**Air Quality Prediction and Mitigation web app using IOT and ML***"* by ***Soham Athavale (A605), Sarang Gajare (A625), Mihir Kate (A647), Armaan Moledina (A663)*** is approved for Partial fulfilment degree *of* ***Computer Engineering***.

Examiners:

1_____

2._____

Date: 25/4/2024
Place: Rajiv Gandhi Institute of Technology

Declaration

We wish to state that the work embodied in this project titled **"Air Quality Prediction and Mitigation web app using IOT and ML"** forms our own contribution to the work carried out under the guidance of "**Dr. Shikha Gupta**" at the Rajiv Gandhi Institute of Technology.

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources that have thus not been properly cited or from whom proper permission has not been taken when needed.

Soham Athavale (A605) _____

Sarang Gajare (A625) _____

Mihir Kate (A647) _____

Armaan Moledina (A663) _____

Abstract

Ensuring the availability of clean and pure air has become a critical determinant in safeguarding the health and well-being of communities, as well as preserving the delicate balance of the ecosystem. The significance of maintaining pristine air quality extends beyond mere environmental concerns, encompassing multifaceted aspects of human health and ecological sustainability. However, the growing menace of air pollution has emerged as a formidable challenge, significantly impacting urban areas, particularly in rapidly developing nations such as India.

In response to the escalating concerns surrounding air pollution, the adoption of continuous monitoring and forecasting mechanisms has become indispensable for effectively managing and preserving ambient air quality. In this pursuit, the integration of machine learning-based prediction technologies has showcased its transformative potential, proving to be an invaluable asset in comprehensively examining and predicting the Air Quality Index (AQI) with a high degree of accuracy and efficiency.

The Air Quality Index (AQI) serves as a pivotal environmental metric, serving to holistically evaluate ambient air quality by integrating key pollutant parameters such as Sulphur dioxide (SO2), nitrogen dioxide (NO2), respirable suspended particulate matter (RSPM), suspended particulate matter (SPM), as well as PM2.5 and PM10 particulate matter. This comprehensive assessment not only aids in conducting precise and reliable public health assessments but also forms the basis for devising and implementing effective environmental management strategies and regulations.

The deployment of machine learning algorithms in predicting the AQI offers a dynamic approach to discern intricate patterns within air quality data, facilitating a comprehensive understanding of the complex dynamics of air pollution. This, in turn, empowers stakeholders and policymakers to formulate targeted and impactful interventions to mitigate the detrimental effects of air pollution effectively. With the overarching objective of developing a robust and accurate machine learning model, this project aspires to contribute to the creation of a sustainable and healthier living environment, ensuring the well-being and prosperity of present and future generations.

Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

CHAPTER 1

Introduction

1.1 Introduction Description

Air pollution is a pervasive environmental concern that results from the release of various pollutants into the Earth's atmosphere. It is a complex issue with far-reaching implications for both human health and the environment. The rapid rise of industries and urban centres has led to a surge in emissions, encompassing various pollutants such as particulate matter and greenhouse gases. Increased vehicular traffic and congestion in urban areas exacerbate the problem, with idling vehicles emitting harmful gases like carbon monoxide and nitrogen oxides.

A variety of pollution types, such as water pollution, air pollution, and soil pollution, are prevalent in the environment. Among them, air pollution takes precedence and requires prompt intervention because it directly affects human health through the inhalation of atmospheric oxygen.

Ill effects: Air pollution has dire consequences for human health, leading to a range of diseases and health conditions. Prolonged exposure to polluted air has been linked to respiratory disorders such as asthma, chronic obstructive pulmonary disease (COPD), and bronchitis, often aggravating these conditions and reducing overall lung function. Cardiovascular diseases, including heart attacks and strokes, are aggravated by the presence of air pollutants. Moreover, air pollution is a known carcinogen, increasing the risk of lung cancer. Vulnerable populations such as children, the elderly, and individuals with pre-existing health conditions are particularly susceptible to these health hazards. Furthermore, air pollution contributes to developmental issues in children, cognitive decline in the elderly, and adverse pregnancy outcomes. The repercussions reach beyond the individual level, affecting entire communities and placing a burden on healthcare systems. Addressing air pollution through effective measures, including accurate prediction using machine learning models, is imperative to mitigate the public health crisis associated with this environmental hazard.

Pollutants: Pollutants can be classified into two types: primary pollutants and secondary pollutants. Primary pollutants are:

Carbon dioxide (CO2): Carbon dioxide plays a significant role in air pollution by contributing to global warming. It is also known as also known as a greenhouse gas. It is released into the atmosphere by human respiration as well and due to the combustion of fossil fuels.

Sulphur oxide (SO): Sulphur dioxide (SO2) is produced when coal and petroleum are burned. It is emitted by various industries. When it reacts with nitrogen dioxide (NO2), it forms sulphuric acid (H2SO4) which is a major contributor to acid rain and is a leading cause of air pollution.

Nitrogen oxide (NO): Nitrogen dioxide (NO2), is the most common nitrogen oxide. It can result from thunderstorms and temperature fluctuations.

Carbon monoxide (CO): Carbon monoxide is generated from the combustion of coal and wood and is also emitted by vehicles. This colourless, odourless, and toxic gas contributes to smog formation, which makes it a significant primary pollutant in air pollution.

Chlorofluorocarbons (CFCs): Chlorofluorocarbons are released by air conditioners and refrigerators which react with other gases and harm the ozone layer. This depletion allows harmful ultraviolet rays to reach the Earth's surface thus posing risks to human health.

Particulate matter: Particulate matter originates from dust storms, forest fires, volcanoes, and similar sources in the form of solid or liquid particles, making a significant contribution to air pollution.

Secondary Pollutants are:

1. Ground Level Ozone: This troublesome substance materializes just above the Earth's surface as a result of a chemical dance between hydrocarbons and nitrogen oxide when sunlight makes an appearance.
2. Acid Rain: Another environmental concern emerges when sulfur dioxide engages in a chemical tango with nitrogen dioxide, oxygen, and atmospheric water. This interaction leads to the creation of acid rain, which can descend upon the ground in either dry or wet forms.

One crucial distinction sets Primary Pollutants apart from Secondary Pollutants: Primary Pollutants are those directly emitted into the atmosphere from their sources, while Secondary Pollutants come into being through reactions with primary pollutants or other atmospheric components.

The complex landscape of air pollution involves various culprits, and among them, PM 2.5 takes center stage. This fine particulate matter is recognized as a major contributor to air pollution. Researchers have harnessed the power of logistic regression and autoregression techniques to ascertain PM 2.5 levels. Notably, some scholars have shifted their focus from daily to hourly data prediction, adopting diverse algorithms to refine their predictions and leaving behind the daily forecasting of pollutant levels.

Air Quality Index: The Air Quality Index (AQI) is a standardized measurement which is used to assess the quality of air in a specific area, for public health and environmental purposes. It provides a numerical value that represents the overall level of air pollution and helps the public understand the potential health risks that are associated with the air they breathe. The AQI is calculated based on the concentrations of various air pollutants, including particulate matter (PM2.5 and PM10), ground-level ozone (O3), sulfur dioxide (SO2), nitrogen dioxide (NO2), and carbon monoxide (CO).

1.2 Organization of report

- **Ch.1 Introduction:** This chapter serves as an initial overview, providing context and background information about the importance of air quality prediction and the relevance of using machine learning techniques in this domain. It outlines the research problem, objectives, scope, and significance of the study. The introduction also includes a brief explanation of the structure of the report and an outline of the subsequent chapters.
- **Ch.2 Literature Review:** The literature review chapter critically analyzes existing research and studies related to air quality prediction and machine learning applications. It reviews relevant academic papers, articles, and other sources to provide a comprehensive understanding of the current state of research in the field. It discusses various methodologies, models, and approaches used in previous studies, highlighting their strengths, weaknesses, and gaps in knowledge.
- **Ch.3 Proposed System:** The proposed system chapter details the methodology and approach used for air quality prediction using machine learning. It outlines the specific algorithms, data sources, and features used in the predictive model. This chapter provides a step-by-step explanation of the data preprocessing, feature engineering, model training, and validation processes. It also describes the technical architecture and framework employed in developing the predictive system.
- **Ch.4 Results & Discussion:** The results and discussion chapter presents the findings and outcomes of the air quality prediction model. It includes the graphs, results and accuracy results of the machine learning model. This chapter interprets the results in the context of the research objectives and compares them with existing literature and benchmarks. It also discusses the implications of the findings, limitations of the study, and potential areas for further research or improvement in air quality prediction using machine learning techniques.

CHAPTER 2

Literature Review

2.1 Survey existing system

Kennedy Okokpujie in their research used naive forecast approach, Linear Regression and Gradient Boosting Algorithm to predict air quality [1]. Miss Ruchita Nehete in their research used Logistic Regression, Random Forest and Decision Tree algorithms to predict air quality index and concluded that classifier models perform better for air quality prediction [2]. Mrs.
A. Gnana Soundari in their research used linear regression and gradient boosting algorithm and concluded that gradient boosting algorithm works best for air quality prediction [3]. D. Iskandaryan in their research studied various other models used in other research papers such as neural network, regression ensemble and other models to predict and forecast air quality and concluded that PM 2.5 was the most important element affecting air quality [4]. T. Madan in their research compared various algorithms used by researchers to predict air quality such as Neural Network, Recurrent Neural Network, Logistic regression, Autoregression, Artificial Neural Network, Deep Belief Network, Random Forest, Multilayer Perceptron Regressor, Hybrid tree and light gradient boosting model, Extra trees, Extreme Learning Machine, XG Boost, Multilayer Perceptron, Gated Recurrent Neural Network and concluded that neural network and boosting models come out to be superior than other algorithms [5]. G. Kalaivani in their research compared various algorithms used by researchers to predict air quality such as Naïve Bayes, K-means Clustering, K Nearest Neighbor, State Vector Machine, Random Forest, Artificial neural network and other regularization and optimization techniques. It was concluded that Random Forest was the best technique, performing well for pollution prediction for data sets of varying size and location and having different characteristics. Its processing time was found much lower than the gradient boosting and multilayer perceptron algorithms. Its error rate was found to be the lowest among all the other studied algorithms. The processing time of Decision Trees was found to be the lowest, its error rate remained higher than most techniques [6]. R. Murugan in their research used Multi-Layer Perceptron and Random Forest algorithms for classification and concluded that Random Forest has higher accuracy for air quality prediction [7]. Yongliang Feng in their research used a deep learning model to predict air quality by LSTM algorithm [8]. Heydari in their research used algorithms such as LSTM, Elman Neural Network, Multi Verse Optimization Algorithm, Particle Swarm Optimization Algorithm and concluded that MI-LSTM-MVO when used in combination in deep learning give the best results for air quality prediction [9]. Gokulan Ravindiran in their research used LightGBM, Random Forest, CatBoost, Adaptive boosting (AdaBoost) regressor and Extreme gradient boosting (XGBoost) algorithms and concluded that the Random Forest and CatBoost algorithms outperformed other machine learning models like LightGBM, Adaboost, and XGboost in predicting AQI accurately [10].

2.2 Limitation existing system or Research gap

The following research gaps were found in the papers surveyed in general:

- Not taking latest data in the datasets
- Not taking data from live sensors but relying on past data
- Improper/no preprocessing of data
- Not taking data from various cities to identify patterns better

2.3 Problem Statement and Objectives

The problem statement of this project is to mitigate and reduce air pollution by employing machine learning algorithms and testing and training a ML model and then based on the accuracy of the algorithms, deploying the best algorithm to predict future air quality.

2.3.1 Objectives

- Mitigation Strategies: Develop mitigation strategies, such as traffic management recommendations and emission reduction plans, based on predictive models.
- Machine Learning Models: Implement machine learning models for air pollution prediction. Explore various ML algorithms, such as regression, decision trees, and neural networks, to identify the most accurate model.
- Evaluation and Testing: Evaluate the accuracy and reliability of the prediction models using historical data and real-world testing. Fine-tune the models to improve prediction performance.
- Data Collection and Sensors: Select and deploy appropriate sensors for measuring key air quality parameters, including PM2.5, PM10, NO2, CO, O3, and more. Establish a network of IoT devices to collect real-time data from various locations within the target area.

2.4 Scope

The scope of the project for Air Quality Pollution utilizing machine learning and IoT encompasses the environmental domain, with a focus on developing efficient and accurate models for air quality prediction and monitoring.

CHAPTER 3

Proposed System

3.1 Analysis/Framework/Algorithm

Data Collection:

• Historical air quality data is gathered from various monitoring stations.

• Geographical data, including monitoring station locations and land use information, is incorporated.

Data Preprocessing:

• Missing data and outliers are handled.

• Data cleansing and transformation processes are conducted. Data from various sources are merged and synchronized.

Feature Selection:

Relevant features that may affect air quality, such as meteorological variables, geographical factors, and historical pollution levels, are identified.

Model Building:

- Machine learning models, including Logistic regression, Decision Tree Classifier,KNN are used to predict air pollution levels.
- Historical data is used for training and testing the models.
- Real-time Data Integration:
- A pipeline will be established to collect and integrate real-time data from monitoring stations. The model is continuously updated with new data to maintain accurate predictions.
- Prediction and Visualization:
- Air pollution predictions are generated using the trained model.
- Model Evaluation:
- Model performance is assessed using appropriate evaluation metrics, such as RMSE and R-squared.
- Different models are compared to identify the most accurate one.
- Decision Tree Classifier was found to be the most effective in predicting the air pollution levels.

3.2 Details of hardware and software

3.2.1 Hardware requirements

Data Servers and Storage:

- High-capacity servers are essential for storing historical air quality data, meteorological data, and other relevant information.
- Distributed storage systems may be used to handle large datasets efficiently.

IOT Equipment

- MQ135 Gas Sensor
- ESP 8266

3.2.2 Software requirements

Data Processing:. Custom scripts or software for handling missing data and outliers.

Machine Learning and Predictive Modelling:

- Machine Learning Frameworks: Libraries like numpy, pandas and scikit-learn for developing and training predictive models.
- Statistical Software: Tools like R for statistical analysis and modeling.
- Custom Algorithms: KNN,Decision Tree Classifier and Logistic Regresssion for air pollution prediction.
- Software for automating the retraining of predictive models with new data.

3.3 System Flow

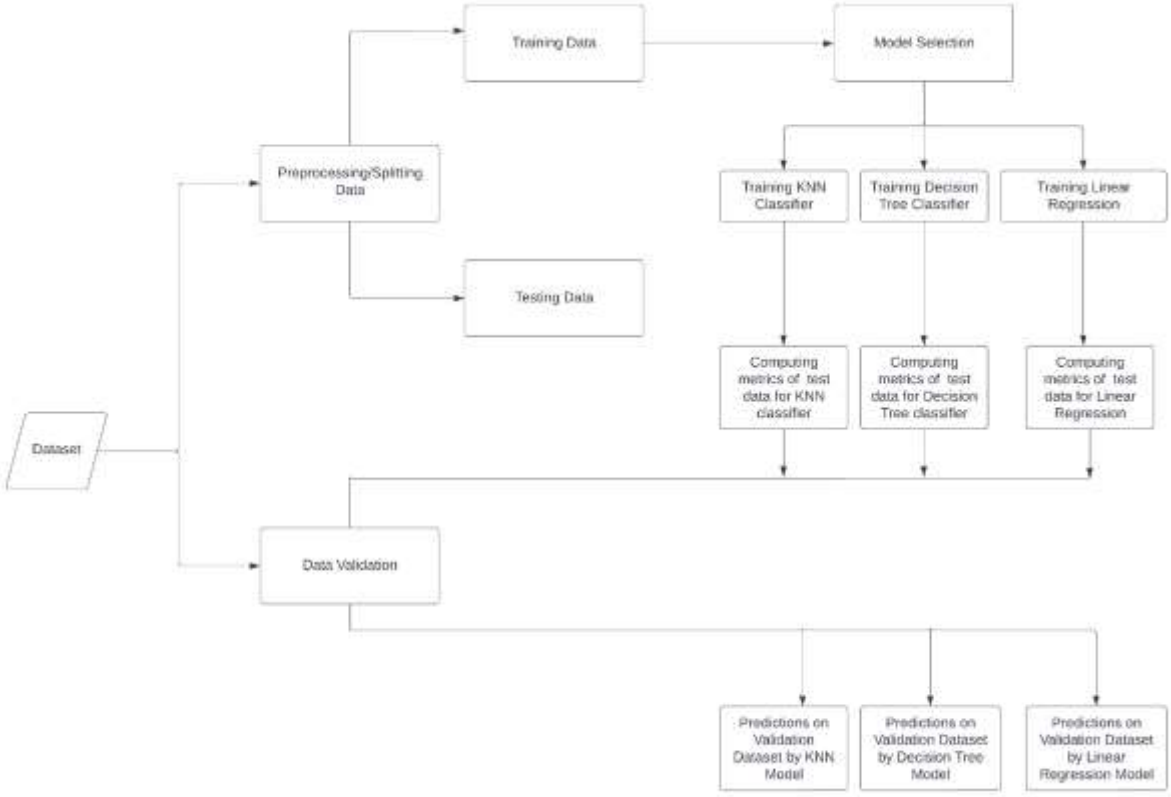

Fig 3.1. System Flow of Machine Learning Model for AQI Prediction

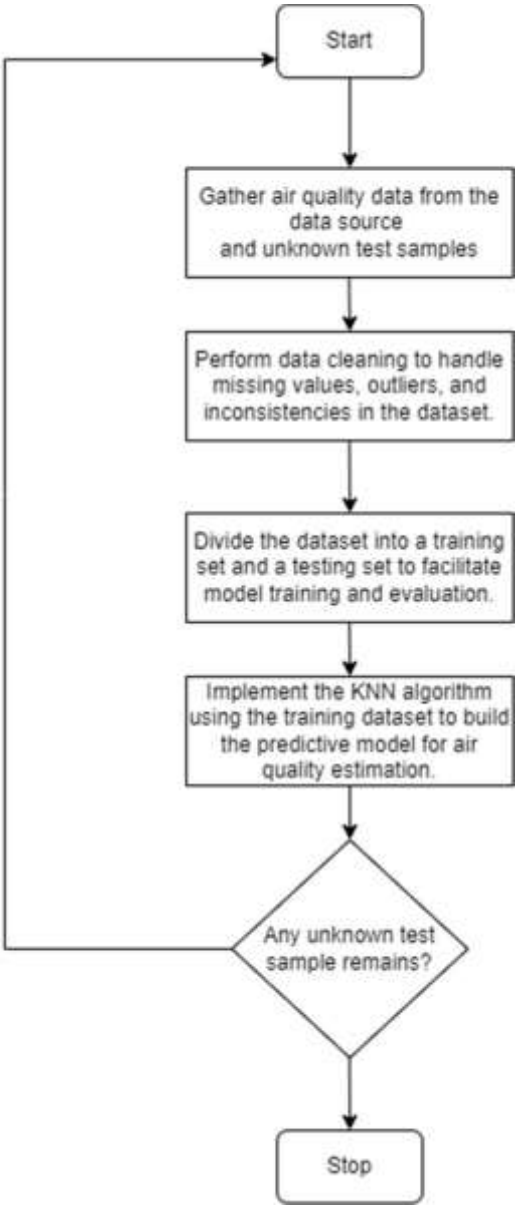

Fig 3.2. System Flow for KNN Algorithm

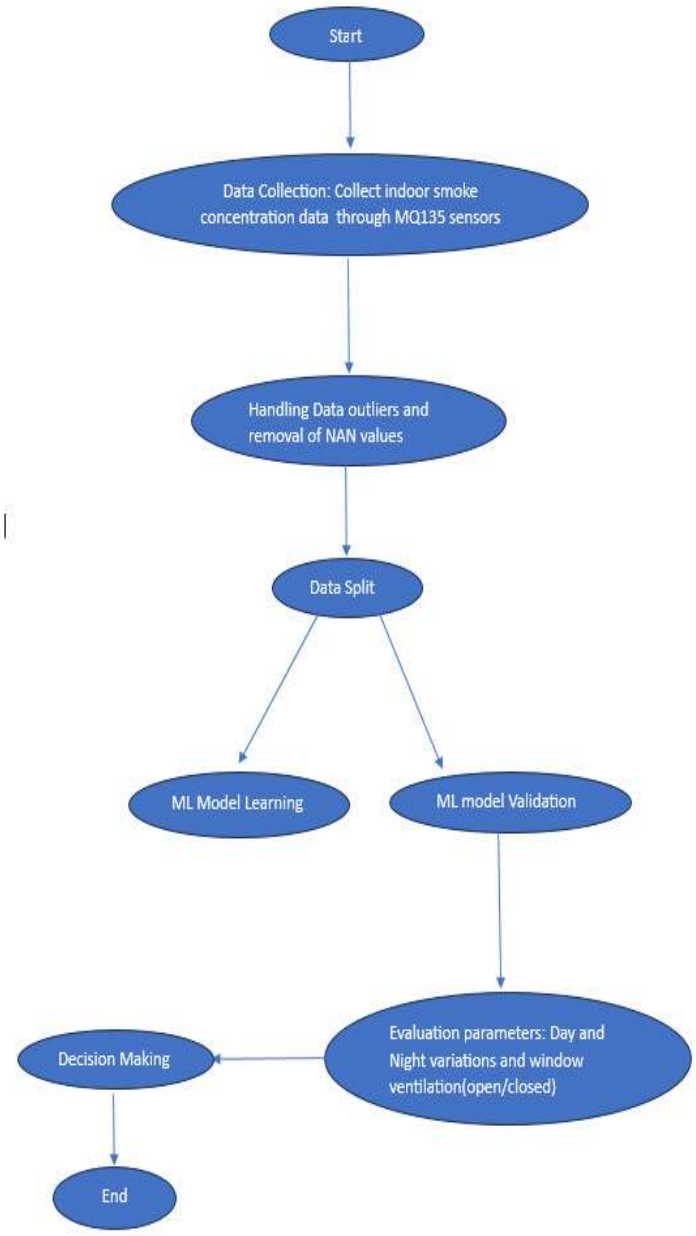

Fig 3.3. System Flow for IOT Data Processing and ML Model evaluation

3.4 Methodology/Procedure

- **Data Collection:**

  Data Collection Parameters:
- Sensor Data
- Day
- Time
- Window_Ventilation

Data Preprocessing:
Handle missing data using interpolation methods or fill forward/backward. Outlier detection and removal using statistical methods (e.g., Z-score). Synchronize data from different sources into a unified time series format. Aggregate data over a time interval (e.g., hourly or daily) to create time-series datasets.

Feature Selection and Engineering: Identify relevant features such as meteorological variables ({MD_t}) and historical pollution levels ({AQD_t-1, AQD_t-2, ...}) as predictors. Create additional features, e.g., day of the week, time of day, and station proximity to traffic or industrial areas.

Model Selection: Choose a machine learning model, such as a regression model (e.g., Linear Regression) or a time series model (e.g., ARIMA or LSTM), to predict air quality.
Define the model as follows: $Y_t = f(X_t, \theta)$, where Y_t is the predicted air quality at time t, X_t is the feature vector at time t, and θ represents the model parameters.

Model Training: Split the dataset into training and validation sets. The training set is used to train the model, while the validation set is used for model evaluation. Train the model using the training data. For a linear regression model, the formula is: $Y_t = \beta0 + \beta1X1_t + \beta2X2_t + ... + \varepsilon_t$, where $\beta0$, $\beta1$, $\beta2$, ... are coefficients and ε_t is the error term.

Real-time Data Integration: Develop a data pipeline to collect real-time data from monitoring stations and update the dataset. Perform real-time predictions using the updated model.

Prediction and Visualization: Use the trained model to predict air pollution for the next time step (e.g., the next hour). Visualize predictions and historical data on maps and time series graphs for public access.

Model Evaluation: Assess the model's performance using evaluation metrics such as Root Mean Squared Error (RMSE) or Mean Absolute Error (MAE).

Compare different models to determine the best-performing one.

Algorithms: Three algorithms have been implemented in this project: K Nearest Neighbor, Logistic Regression and Decision Tree.

Each algorithm is discussed below:

KNN(K Nearest Neighbour): KNN(K Nearest Neighbour) is an algorithm in machine learning used for classification. It is a supervised machine-learning algorithm and can also be used for regression. For detecting air pollution we can use this algorithm. It works in the following way:

1. Data Collection: For performing the detection of air pollution we first need a dataset. This dataset can include things like SPM, RSPM, SO2, NO2, CO, PM2.5. One more thing is that a label-high/low/medium for each data point would be needed.

2. Data Preprocessing: It means to clean the data and to preprocess it. An example would be to handle missing values, feature engineering or selection could be performed to increase model performance.

Choosing K: This is a crucial part of performing KNN, it determines how many k neighbours would be there when making predictions. It is an important step because it could impact our model's performance and may require hyperparameter tuning.

3. Splitting Data: There would be a need to split the dataset, into a training dataset and a testing dataset. The training dataset is used to train our model and the testing dataset is to test our model's performance.

4. Training the KNN Model: KNN, unlike other machine learning algorithms, does not involve traditional testing instead it uses memory to store the dataset and while predicting calculates distances in the dataset.

5. Predictions: Suppose in a new dataset the live air pollution data calculates the distance between that point and the K nearest neighbour based on Euclidian distance, it then assigns a class label or predicts a value based on the majority or average class value of the K nearest neighbour.

6. Evaluation: To evaluate our model we have used R squared. On both training and testing data. The accuracy score, and kappa score has been used to evaluate our model.

7. Optimization: A need may arise to fine-tune the model by adjusting the choice of distance metric or other parameters to improve the model's performance.

8. Deployment: Once all the above-mentioned processes have been completed KNN model can be integrated into the air pollution monitoring system.

Decision Tree Classifier: A decision tree classifier represents a machine learning technique employed for the identification of air pollution, particularly for the purpose of categorizing air quality using diverse features and parameters. Decision trees are renowned for their ease of comprehension, interpretability, and visualization, rendering them a suitable option for this objective. The subsequent actions to be executed encompass the following:

1. Data Collection: A dataset containing historical air quality data has been gathered, including features such as:

i. Sampling date
ii. State
iii. Location
iv. Agency
v. Type (Residential or Industrial)
vi. Nitrogen dioxide (NO2) levels
vii. Sulphur dioxide (SO2) levels
viii. Respirable Suspended Particulate Matter (RSPM) levels
ix. Suspended Particulate Matter (SPM) levels
x. Location monitoring station

2. Data Preprocessing: In the model, the missing values problem has been addressed, categorical variables have been encoded, and the dataset has been divided into training and testing subsets.

3. Decision Tree Model: A decision tree classifier has been built using a machine learning library - scikit-learn in Python.

4. Model Evaluation: To evaluate the model R squared has been used on both training and testing data. Accuracy score, and kappa score has been used to evaluate the model.

5. Tuning and Pruning: Decision trees can be prone to overfitting, so tuning hyperparameters like the maximum depth of the tree and minimum samples per leaf, has been considered to optimize the model's performance.

6. Deployment: Once all the above-mentioned processes have been completed Decision Tree model can be integrated into the air pollution monitoring system.

7. Monitoring and Maintenance: Regularly update and retrain the model with new data to ensure it remains accurate over time, as air quality conditions may change.

Linear Regression:

Linear Regression is a useful approach for addressing air pollution detection and classification objectives, especially when the goal is to forecast the probability of particular air quality conditions or pollution levels using a range of environmental variables. Here's how Linear Regression finds application in the sphere of air pollution monitoring:

1. Data Collection: The same dataset is used as mentioned in Decision Tree Classifier.

2. Data Preprocessing: The data is subject to cleaning and preprocessing as part of the research

process. This phase involves handling missing data, normalizing or scaling features, encoding categorical variables, such as meteorological conditions, and partitioning the dataset into training and testing subsets.

2. Binary Classification: A binary classification problem is defined, wherein the target variable assumes binary values, typically denoting "good" or "poor" air quality. Historical air quality indices or predefined expert thresholds are utilized to label the dataset accordingly.

3. Model Training: Linear Regression is employed for training a classification model. The primary objective is to predict the probability that a given combination of environmental features corresponds to either "good" or "poor" air quality. Linear Regression models this probability using a logistic (sigmoid) function.

4. Feature Selection: The research entails the identification of environmental features with the greatest relevance for air quality prediction. Linear Regression provides coefficients for each feature, signifying their impact on the classification decision. Features with higher absolute coefficients exert a more substantial influence.

5. Model Evaluation: The performance of the Linear Regression model is assessed using pertinent evaluation metrics, including accuracy, precision, recall, F1-score, and ROC-AUC. These metrics serve as criteria for gauging the model's effectiveness in classifying air quality conditions.

6. Threshold Selection: An appropriate probability threshold is chosen to classify air quality conditions effectively. For instance, air quality may be categorized as "good" when the predicted probability surpasses a specified threshold and as "poor" otherwise.

7. Real-time Prediction: Following model development and evaluation, the trained Linear Regression model is deployed in a real-time air quality monitoring system. Continuous input of fresh environmental data into the model yields real-time predictions regarding air quality conditions.

Further, two more regressor algorithms were used after collecting data from the IOT device:

Decision Tree Regression

Decision Tree Regression offers a robust solution for modeling and predicting continuous variables, such as predicting house prices based on various house attributes. Here's how Decision Tree Regression is applied in the context of predictive modeling for continuous outcomes

1. **Data Collection**: Utilizes the same dataset as mentioned in previous models, tailored for regression rather than classification. The dataset could include features like square footage, number of bedrooms, location, and age of the house.

2. Data Preprocessing: Involves cleaning and preparing the dataset for the regression model. This step includes handling missing values, possibly transforming skewed data, and splitting the dataset into training and testing subsets for model validation.
3. Continuous Variable Prediction: The problem is framed as predicting a continuous value (e.g., house price) rather than classifying into categories. The Decision Tree Regression model predicts the value based on the input features by learning simple decision rules inferred from the data features.
4. Model Training: A Decision Tree Regressor is trained on the dataset. It constructs a tree-like

model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. It splits the dataset into subsets based on feature values, aiming to reduce variance or bias in predictions.

5. Feature Importance: One advantage of Decision Tree Regression is its ability to automatically select the most informative features for making predictions. The importance of each feature is calculated based on how effectively it reduces the variance of the predictions.
6. Model Evaluation: The performance of the Decision Tree Regression model is evaluated using metrics suitable for regression tasks, such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared (R^2). These metrics help in understanding the accuracy and predictive power of the model.
7. Pruning: To prevent overfitting, where the model learns the training data too well and performs poorly on unseen data, pruning techniques are applied. Pruning reduces the size of the decision tree by removing parts of the tree that do not provide additional power in predicting the target variable.
8. Real-time Prediction: Once trained and evaluated, the Decision Tree Regression model can be deployed to make predictions in real-time. For instance, it can provide instant estimates of house prices given current market conditions and house attributes, aiding both sellers and buyers in making informed decisions.
9. Visualization: Decision Tree models have the added benefit of being easily visualized and understood, even by those with no background in data science. This transparency allows stakeholders to see exactly how predictions are made, increasing trust in the model's outputs.

Gradient Boost Regressor

Gradient Boost Regressor is a powerful and flexible machine learning technique for regression tasks, leveraging the principle of boosting weak learners, typically decision trees, into a strong predictive model. Here's an overview of its application:

1. Data Collection: The dataset required for Gradient Boost Regressor should ideally encompass a diverse range of input features relevant to the target variable. For instance, in energy consumption forecasting, features might include weather conditions, time of day, historical usage patterns, and demographic data.
2. Data Preprocessing: Critical steps such as handling missing values, encoding categorical variables, feature scaling, and dividing the dataset into training and testing sets are undertaken to prepare the data for the model.
3. Continuous Variable Prediction: Gradient Boost Regressor is employed to predict a continuous outcome, such as the energy consumption of a building. It incrementally builds an ensemble of weak decision tree models in a stage-wise fashion to minimize a loss function.
4. Model Training: Training involves sequentially adding weak learners, each correcting its predecessor, to improve the model's accuracy. The gradient descent algorithm is used to minimize the prediction error in each step.
5. Feature Importance: This model inherently performs feature selection by allocating more weight to the most predictive features. The importance of each feature is derived from how much it contributes to reducing the overall prediction error.
6. Model Evaluation: Evaluation metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared (R^2) assess the accuracy and generalizability of the Gradient Boost Regressor. Cross-validation techniques are often employed to ensure the model's robustness.
7. Parameter Tuning: The performance of Gradient Boost Regressor can significantly depend on the settings of its parameters, such as the number of trees, learning rate, and depth of trees. Hyperparameter tuning, possibly through grid search or random search, is crucial to optimize the model.
8. Real-time Prediction: Once optimized and validated, the model is ready for deployment in real-world applications, providing real-time predictions with high accuracy and efficiency.

Random Forest Regressor

Random Forest Regressor is an ensemble learning method for regression that operates by constructing a multitude of decision trees at training time and outputting the mean prediction of the individual trees. Here's a breakdown of its use in predictive modeling:

1. Data Collection: Begins with gathering a comprehensive dataset that includes a wide range of variables potentially influencing the target variable, such as predicting the sale price of vehicles based on their features (age, mileage, brand, etc.).
2. Data Preprocessing: Essential preprocessing steps include cleaning the data, dealing with missing values, encoding categorical variables, normalizing or standardizing features, and splitting the data into training and testing subsets.
3. Continuous Variable Prediction: Random Forest Regressor is utilized to predict continuous outcomes by constructing multiple decision trees during training and outputting the average prediction of the individual trees for a more accurate and stable prediction.
4. Model Training: It involves creating a large number of decision trees that are trained on random subsets of features and data points. The randomness helps in making the model more robust against overfitting.
5. Feature Importance: An advantage of using Random Forest is its capability to evaluate the importance of each feature in making accurate predictions. This is achieved by looking at how much the tree nodes, which use that feature, reduce impurity across all trees in the forest.
6. Model Evaluation: The model's performance is evaluated using regression metrics such as MAE, MSE, and R^2. Utilizing out-of-bag (OOB) error estimates can also provide an internal evaluation mechanism without needing a separate test set.
7. Handling Overfitting: Random Forest has built-in mechanisms to handle overfitting through its ensemble approach. However, parameters like the number of trees, max depth, and max features need careful tuning to balance bias-variance tradeoff.
8. Real-time Prediction: After training and fine-tuning, the Random Forest Regressor can be deployed to predict outcomes based on new data in real-time, benefiting various applications from stock price forecasting to real estate valuation.

Internet of Things equipment used in this model:

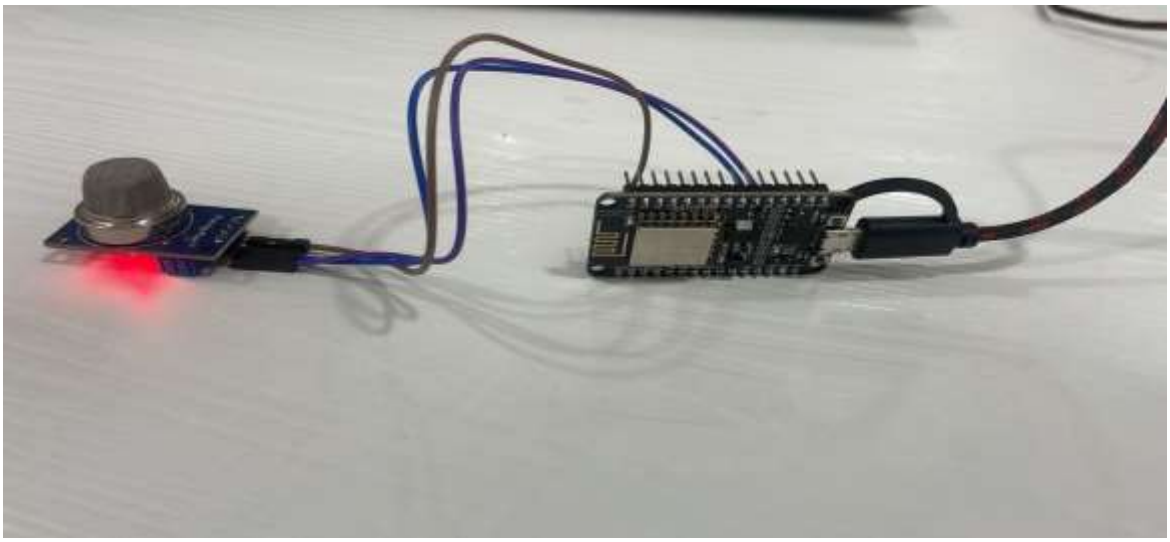

Fig 3.4. Assembled IOT Equipment

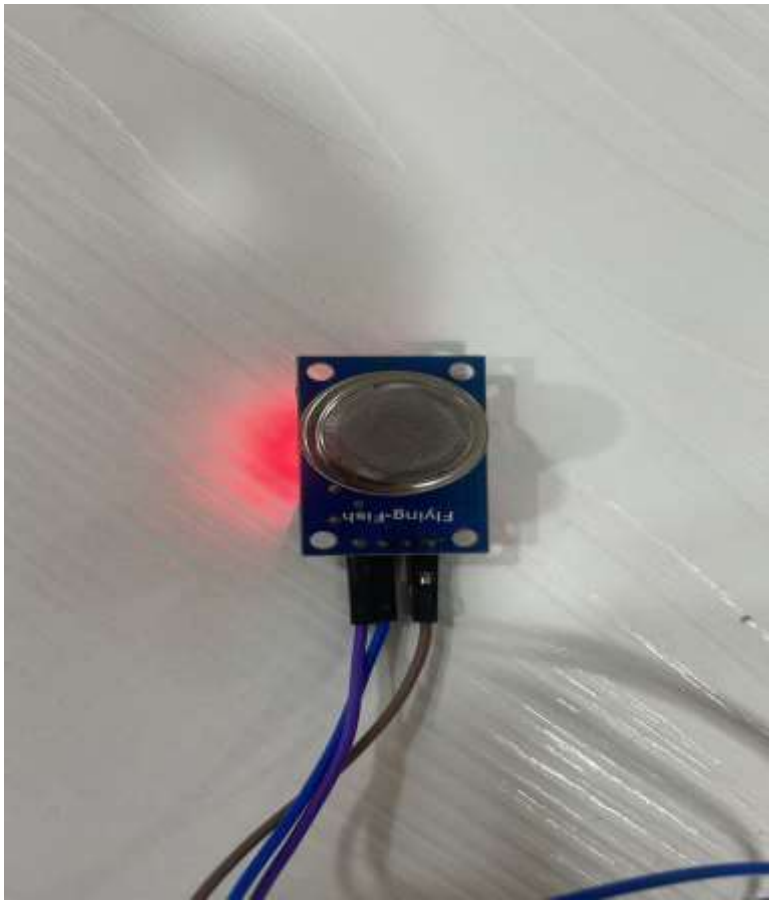

Fig 3.5. MQ135 Gas Sensor

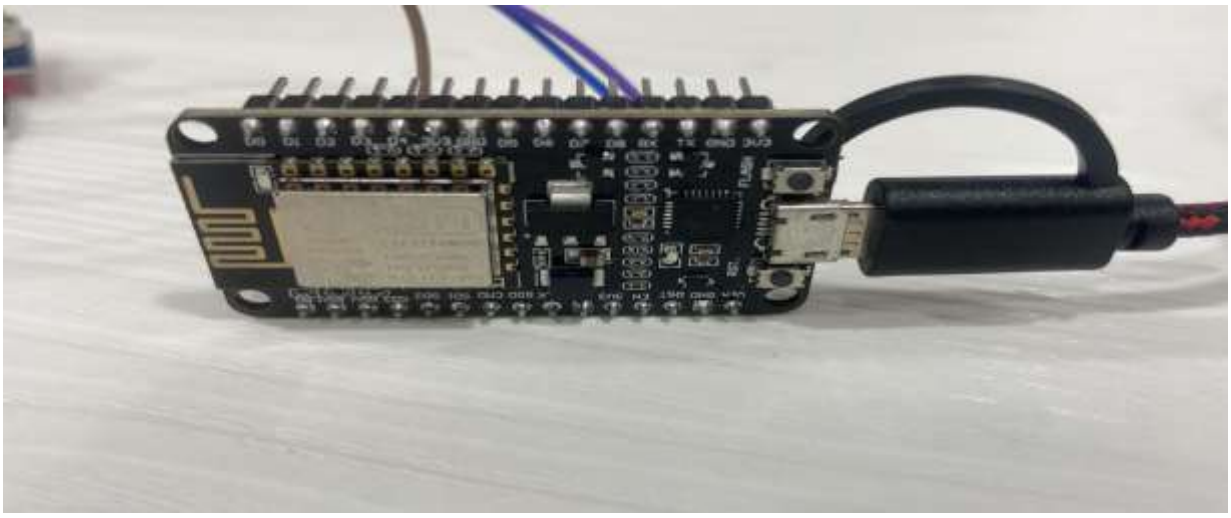

Fig 3.6. ESP 8266

ESP 8266

- ESP 8266: ESP 8266 is a Wi-Fi microchip which has a built in TCP/IP software.
- It enables microcontrollers to connect to the internet
- It has been connected to the computer using a USB Data Transfer Cable

MQ135

- MQ135: MQ135 is a gas sensor
- It is used to detect various harmful gases such as ammonia, benzene and CO2 and for air quality monitoring.
- It has been connected to ESP 8266 using connecting wires

Live data collection involves continuously gathering real-time information from sensors, such as the MQ135, and transmitting it to the ESP8266 microcontroller for processing and analysis. In this setup, the ESP8266 serves as the central hub for data aggregation and management. The MQ135 sensor is deployed in indoor locations of interest it could be our kitchens, master bedroom, bathrooms wherever we wish to continuously monitor gas concentrations, including smoke particles. As the sensor detects changes in gas levels, it generates analog signals proportional to the detected concentrations. These analog transmitted to the ESP8266 microcontroller via the A0 pin connection.

Upon receiving the analog data from the MQ135 sensor, the ESP8266 microcontroller processes the information using programmed algorithms. This may involve converting analog signals to digital format, performing data filtering or calibration, and extracting relevant parameters related to smoke detection.

Once processed, the data is ready for transmission over Wi-Fi networks. The ESP8266 microcontroller utilizes its built-in Wi-Fi capabilities to establish a connection to the internet and transmit the processed data to remote servers, cloud platforms, or other IoT devices.

The live data collection process enables real-time monitoring and analysis of environmental conditions, including the presence of smoke particles. The continuous nature of live data collection ensures that any changes or trends in gas concentrations are promptly detected and addressed.

The Blynk mobile application serves as the interface for monitoring and controlling the data collection process remotely. Through the Blynk app, users could access real-time sensor readings, view historical data trends, and configure parameters such as data sampling frequency.

Data Parameters: The collected data encompassed various parameters essential for smoke detection analysis, including:

- Smoke density measurements from the sensors.
- Timestamps indicating the time of data acquisition.
- Status indicators for window openings, distinguishing between open and closed states.

CHAPTER 4

Results and Discussion

4.1 Results

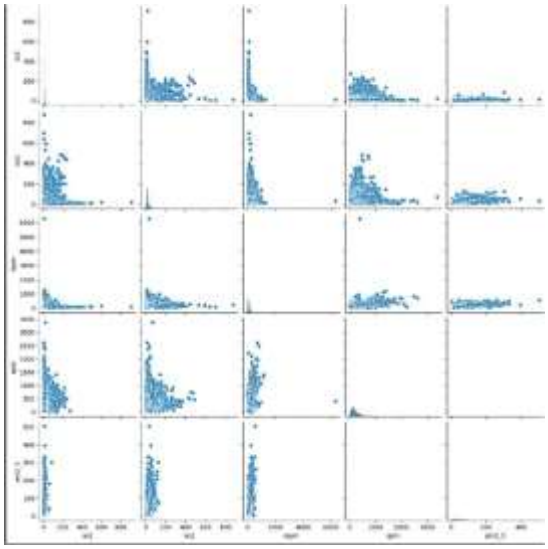

Fig 4.1. Scatter Plot of Variables

Fig 4.1 shows a scatter plot.which shows the relationship between all variables (no2, so2, spm, rspm) with each other.

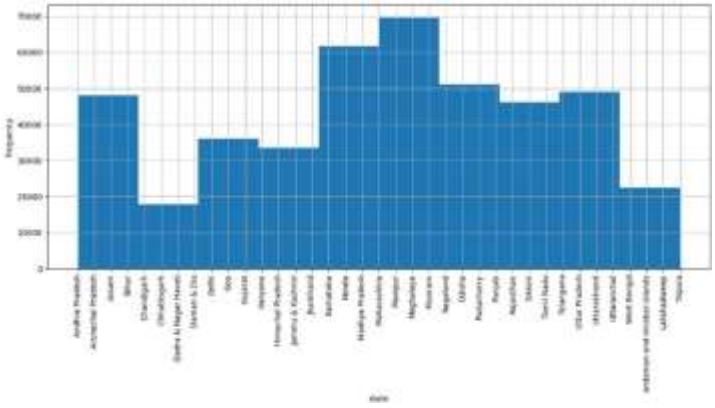

Fig 4.2. Frequency of data from each state

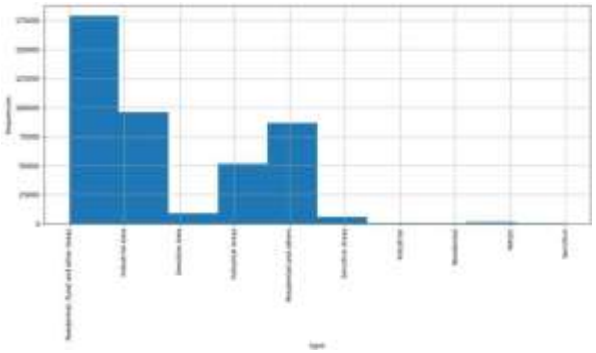

Fig 4.3. Frequency of data based on type of area

Fig 4.3 shows the frequency of air pollution data with respect to type of area (residential areas, rural areas, industrial areas and so on).

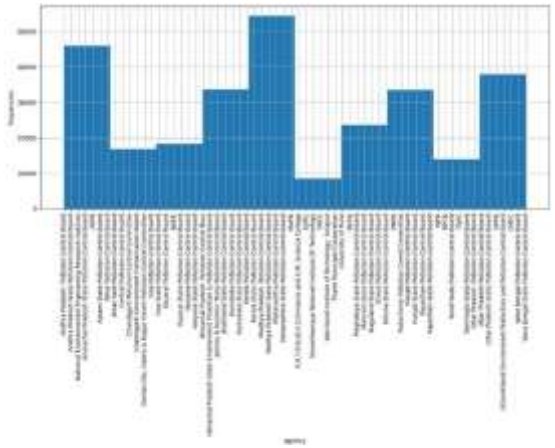

Fig 4.4. Frequency of Data from each State's Pollution Control Board

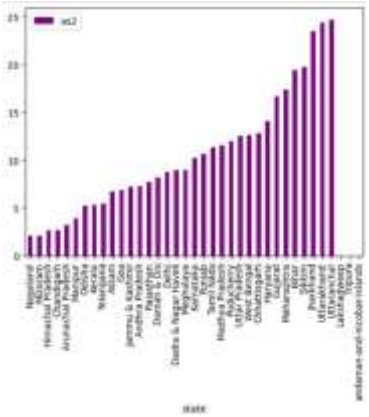

Fig 4.5. SO2 levels of each state

Fig 4.5 shows the levels of SO2 in ppb (parts per billion).

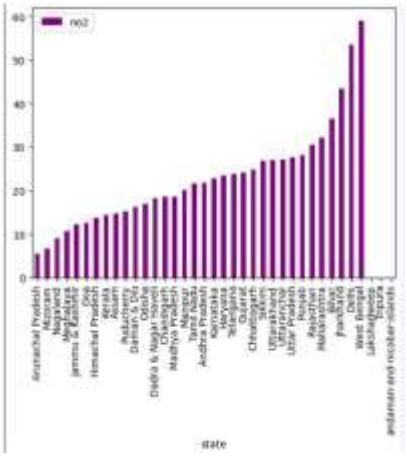

Fig 4.6. NO2 levels of each state

Fig 4.6 shows the levels of NO2 in ppb (parts per billion).

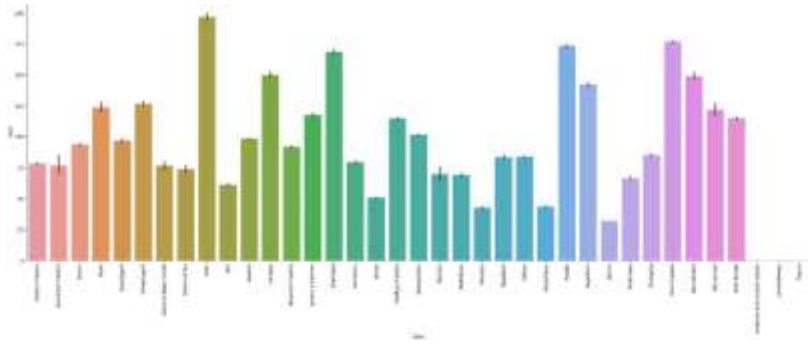

Fig 4.7. RSPM levels of each state

Fig 4.7 shows the levels of RSPM in μg/m^3

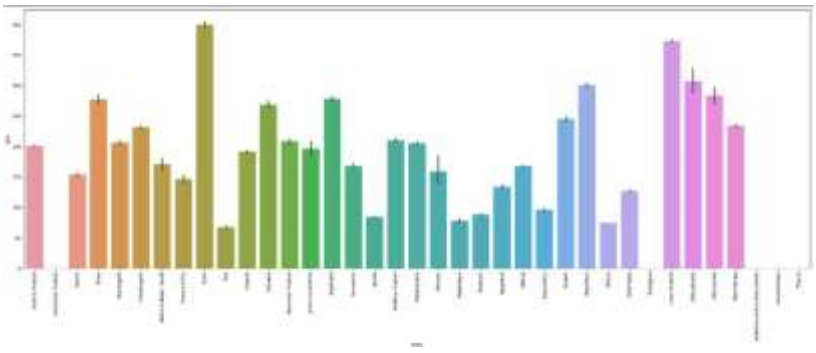

Fig 4.8. SPM levels of each state

Fig 4.8 shows the levels of SPM in μg/m^3

Fig 4.9 Air pollution detection

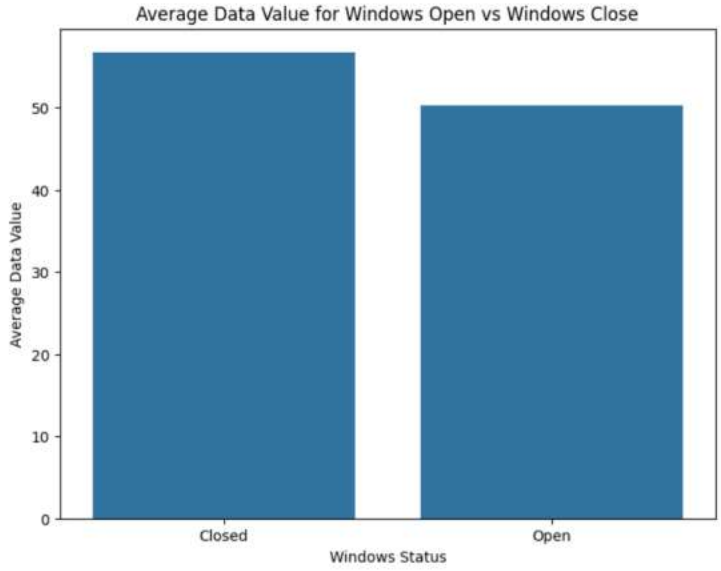

Fig 4.10 Sensor Data Value for Windows Open vs Windows Close

```
#avg predicted values for weekday

# Filter the data for each day of the week
for day in range(1, 6):
    day_data = data[data['Day'] == day]
    X_day = day_data[['Time', 'Day', 'Windows_open']]  # Include necessary features
    day_pollution_prediction = tree_model.predict(X_day)
    print(f"Predicted pollution level for Day {day}:", day_pollution_prediction[0])
```

```
Predicted pollution level for Day 1: 54.0
Predicted pollution level for Day 2: 53.0
Predicted pollution level for Day 3: 54.0
Predicted pollution level for Day 4: 54.0
Predicted pollution level for Day 5: 54.0
```

Fig 4.11 Predicted values for weekdays

```
[ ] #avg predicted values for weekend

    # Filter the data for each day of the week
    for day in range(6, 8):
        day_data = data[data['Day'] == day]
        X_day = day_data[['Time', 'Day', 'Windows_open']]  # Include necessary features
        day_pollution_prediction = tree_model.predict(X_day)
        print(f"Predicted pollution level for Day {day}:", day_pollution_prediction[0])
```

```
Predicted pollution level for Day 6: 54.0
Predicted pollution level for Day 7: 51.0
```

Fig 4.12 Predicted values for weekends

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeRegressor
from sklearn.metrics import mean_squared_error

# Load the dataset from the CSV file into a DataFrame 'df'
df = pd.read_csv("/content/Uno.csv")

# Convert the 'Day' column to a datetime type
df['Day'] = pd.to_datetime(df['Day'])

# Convert datetime values to numerical representations (number of days since the minimum date)
df['Numeric_Day'] = (df['Day'] - df['Day'].min()).dt.days

# Filter the dataset to include only rows where 'Windows Open' is 0
data_zero_windows_open = df[df['Windows_open'] == 0]

# Assuming 'Sensor_Data' is the target variable to be predicted
X = data_zero_windows_open[['Numeric_Day']]  # Features
y = data_zero_windows_open['Data']  # Target variable (Data)

# Create and train a Decision Tree Regressor model
model = DecisionTreeRegressor()
model.fit(X, y)

# Predict the 'Data' when 'Windows Open' is 0 for all days
predicted_data = model.predict(X)

print("Predicted 'Data' when 'Windows Open' is 0 for all days:", predicted_data.mean())
```

```
Predicted 'Data' when 'Windows Open' is 0 for all days: 56.75409836065572
```

Fig 4.13 Predicted values for when Windows are Open

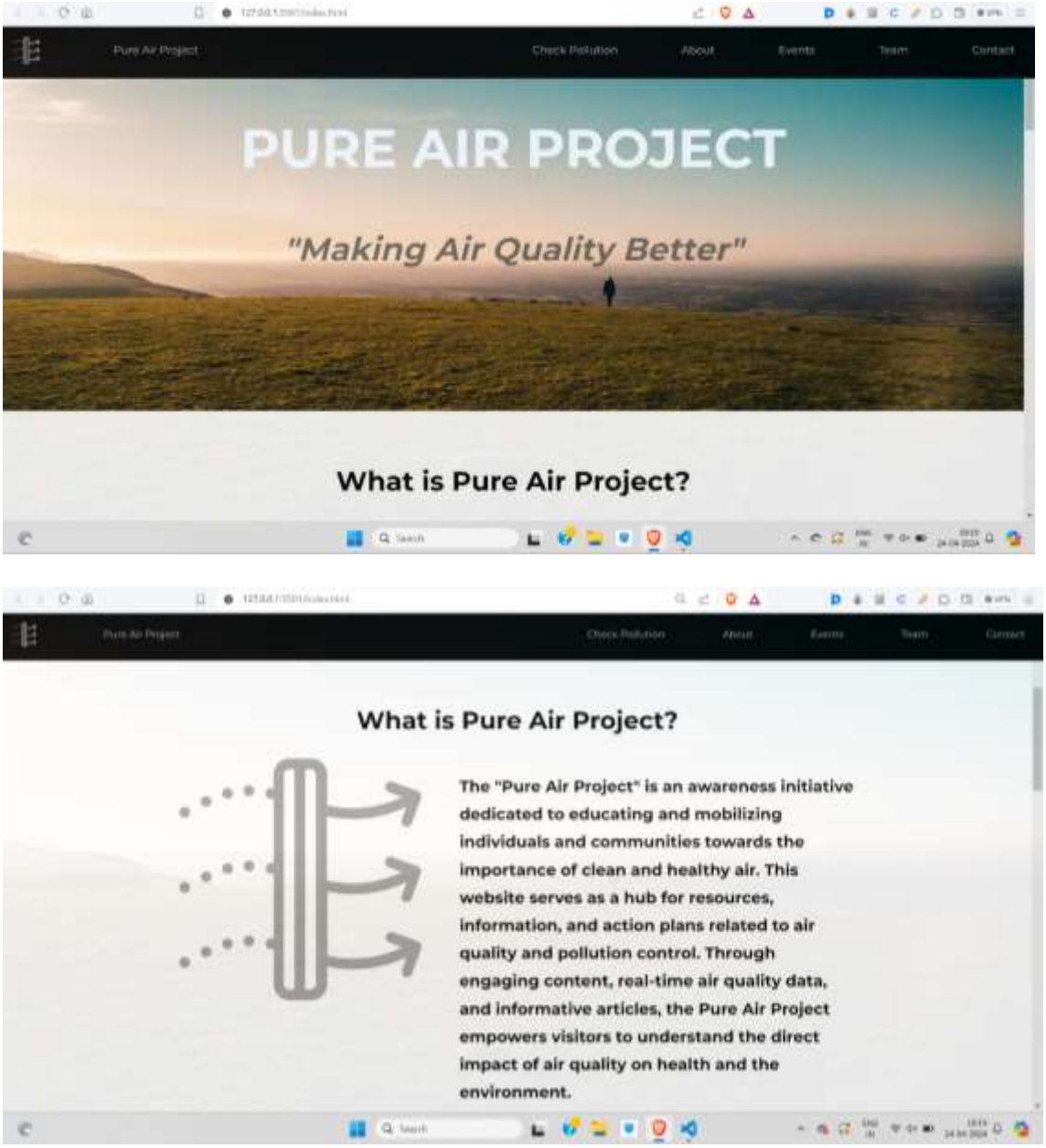

Fig 4.14 Home page of the Website

The Home Page of the website conveys the aim and purpose of the project

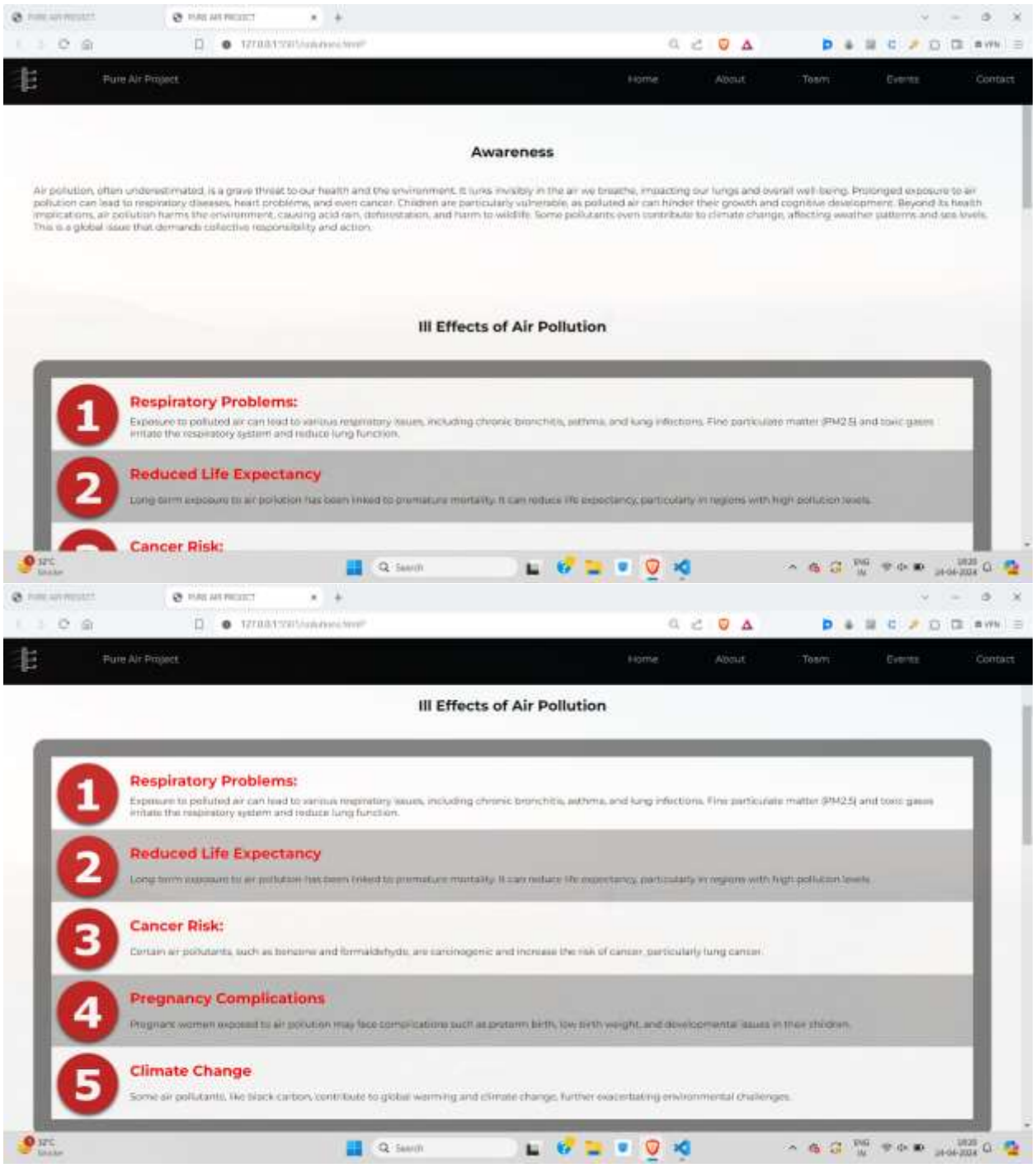

Fig 4.15 Webpage about Ill Effects of Air pollution

This Page of the website tells about the various ill effects of air pollution such as Respiratory problems, reduced life expectancy, climate change, etc.

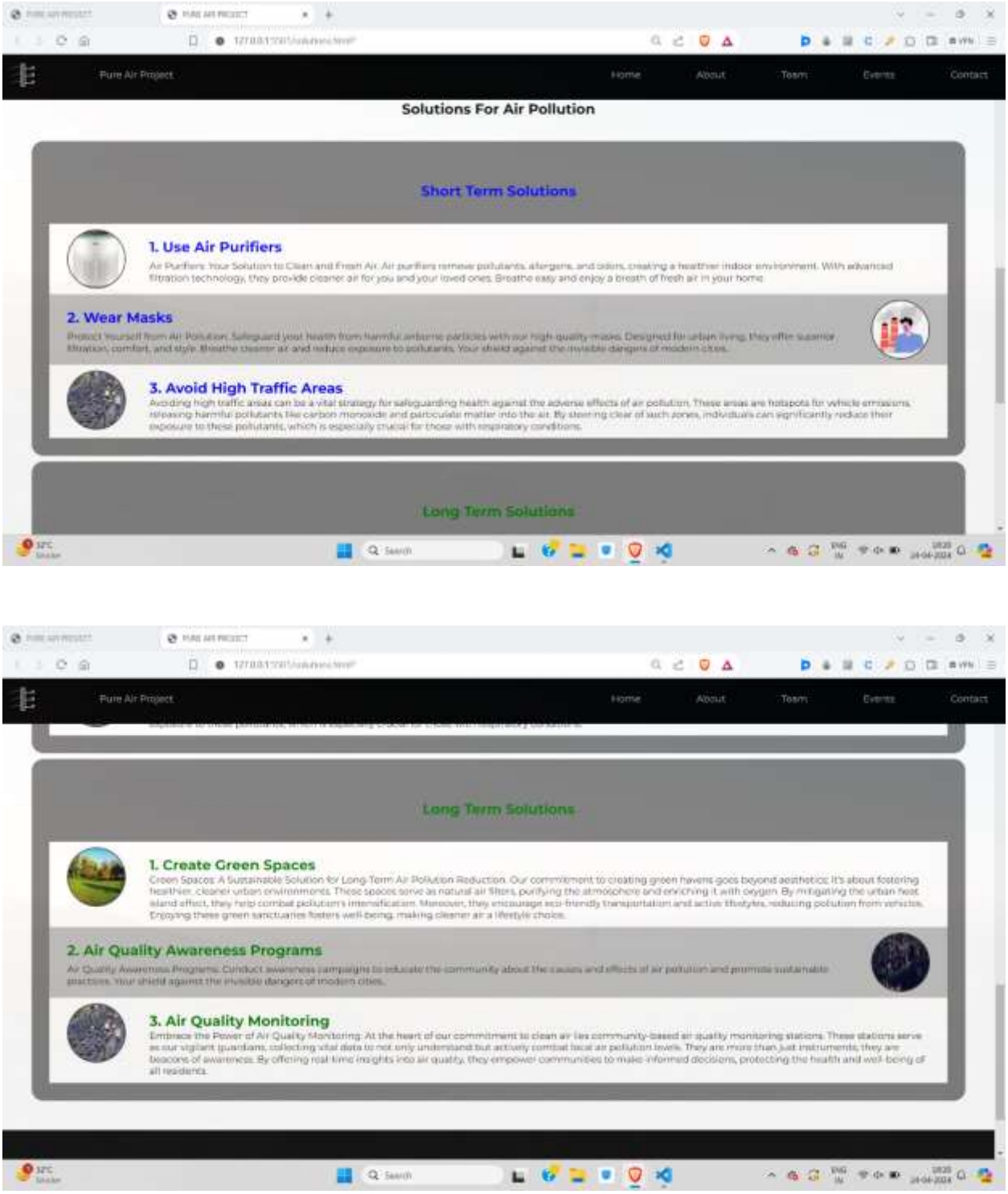

Fig 4.16 Webpage for Solutions of Air Pollution

This webpage tells about the various short term and long term solutions of air pollution

4.2 Discussions

K Nearest Neighbor, Linear Regression and Decision Tree Classifier algorithms were trained and tested in the model. After training and testing, the following results were obtained:

Table 4.1 Comparative Study of all algorithms

Algorithm used:	**Model accuracy on train**	**Model accuracy on test**	**Kappa score**
Linear Regression	98.8539%	98.8587%	97.9453%
Decision Tree Classifier	99.99%	99.9867%	99.9762%
KNN	99.9626%	99.9367%	99.8863%

In the three algorithms used on the dataset obtained by the IOT equipment readings, random forest regressor performed the best.

Table 4.2 Comparative Study of Regressor Algorithms

Algorithms used	**Mean Squared Error**
Decision Tree Regressor	0.39
Gradient Boost Regressor	0.91
Random Forrest Regressor	0.3725

From the predicted data, it was also found that there is higher indoor pollution when windows are closed.

CHAPTER 5

CONCLUSION AND FUTURE WORK

In the future, collaboration with governments can be done. By harnessing the power of data analytics, sensor technology, and policy advocacy, we can develop robust predictive models that enable more effective air quality management. Governments bring regulatory authority, access to critical data, and the ability to enact and enforce air quality standards. With predicted air quality data represented in an area wise graphical format, governments can plan policies better and in advance. For instance, better traffic management can be done by governments, diverting traffic from more polluted areas to relatively lesser polluted areas if they have access to predicted air quality data for the upcoming days.

Implementing an area-wise Air Quality Index (AQI) system, supported by comprehensive data, brings about tremendous convenience and fosters a proactive approach to safeguarding public health and the environment. By making AQI information easily accessible for specific locations, individuals and authorities can take necessary measures with precision and timeliness. This not only streamlines response efforts during periods of poor air quality, but also encourages preventative actions to improve local air conditions.

The positive impact of this approach extends to the overall well-being of the population. It raises awareness about the immediate environment, prompting residents to make lifestyle adjustments, such as wearing masks during periods of high pollution or reducing outdoor activities. Moreover, it incentivizes local authorities to prioritize air quality management and enact policies for cleaner air. In the long term, this leads to improved health outcomes and a higher quality of life, as cleaner air translates to fewer respiratory illnesses, enhanced cardiovascular health, and a more sustainable environment.

REFERENCES

[1] Kennedy Okokpujie, Etinosa Noma-Osaghae, Odusami Modupe, Samuel John, Oluga Oluwatosin, “A Smart Air Pollution Monitoring System,” International Journal of Civil Engineering and Technology (IJCIET) Volume 9, Issue 9, pp. 799–809,2018.

[2] Miss Ruchita Nehete, Prof. D. D. Patil, ”Air Quality Prediction Using Machine Learning,” International Journal Of Creative Research Thoughts - IJCRT, Volume 9, Issue 6, pp. G365-368, 2019.

[3] Mrs. A. Gnana Soundari, Mrs. J. Gnana Jeslin M.E, Akshaya A.C, “Indian Air Quality Prediction And Analysis Using Machine Learning,” International Journal of Applied Engineering Research ISSN 0973-4562, Volume 14, Number 11, 2019, pp. 181-187,2019.

[4] D. Iskandaryan, F. Ramos, and S. Trilles, “Air Quality Prediction in Smart Cities Using Machine Learning Technologies based on Sensor Data: A Review,” Applied Sciences, vol. 10, no. 7, p. 2401, 2020.

[5] T. Madan, S. Sagar and D. Virmani, "Air Quality Prediction using Machine Learning Algorithms –A Review," 2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Greater Noida, India, pp. 140-145, 2020.

[6] G. Kalaivani and P. Mayilvahanan, "Air Quality Prediction and Monitoring using Machine Learning Algorithm based IoT sensor- A researcher's perspective," 2021 6th International Conference on Communication and Electronics Systems (ICCES), Coimbatre, India, pp. 1-9, 2021.

[7] R. Murugan and N. Palanichamy, "Smart City Air Quality Prediction using Machine Learning," 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, pp. 1048-1054, 2021.

[8] Yongliang Feng, "Air Quality Prediction Model Using Deep Learning in Internet of Things Environmental Monitoring System", Mobile Information Systems, vol. 2022, 2022.

[9] Heydari, A., Majidi Nezhad, M., Astiaso Garcia, D. et al. Air pollution forecasting application based on deep learning model and optimization algorithm. Clean Techn Environ Policy 24, 607–621, 2022.

[10] Gokulan Ravindiran,Gasim Hayder, Karthick Kanagarathinam, Avinash Alagumalai, Christian Sonne, “Air quality prediction by machine learning models: A predictive study on the indian coastal city of Visakhapatnam,” Chemosphere, Volume 338, pp. 1-10, 2023.

Acknowledgement

We wish to express our sincere gratitude to **Dr. Sanjay U. Bokade, Principal** and **Prof.S. P. Khachane , H.O.D.** of Department Computer Engineering of Rajiv Gandhi Institute of Technology for providing us an opportunity to do our project work on **" Air Quality Prediction and Mitigation web app using IOT and ML"**.

This project bears on imprint of many peoples. We sincerely thank our project guide **Dr. Shikha Gupta** for her guidance and encouragement in carrying out this synopsis work.

Finally, we would like to thank our colleagues and friends who helped us in completing project work successfully.

1. A605 Soham Athavale
2. A625 Sarang Gajare
3. A647 Mihir Kate
4. A663 Armaan Moledina