Machine Learning based Prediction System for Detecting Air Pollution

Vineeta[1]
Department of Computer Science Engineering
AMC Engineering College
Bengaluru, India

Ajit Bhat[2]
Department of Computer Science Engineering
AMC Engineering College
Bengaluru, India

Asha S Manek[3]
Department of Computer Science Engineering
AMC Engineering College
Bengaluru, India

Pranay Mishra[4]
Sr. Consultant
Hitachi Consulting
Bengaluru, India

***Abstract*— In today's world, one of the most common problems we are facing worldwide is Air Pollution. Air of most cities is polluted these days and newer pollutants have also been added in the air rendering it more poisonous. Both human and natural activities can produce Air Pollution. Pollutants like Sulphur Oxides, Carbon Dioxide (CO2), Nitrogen Oxides, Carbon Monoxide (CO), Chlorofluorocarbon (CFC), Lead, Mercury etc. are being added in the air due to human activities. In this paper, data has been collected from the two sources in the Bengaluru region: government website and static sensors built using Arduino. The level of CO is measured using three machine algorithms namely Random Forest Regression (RFR), Decision Tree Regression (DTR) and Linear Regression (LR). The results show that RFR gives least error of the three and hence more accuracy. Command line interface has also been created to see the CO level prediction.**

Keywords—* *Air Pollution, Machine Learning, Carbon Monoxide, Random Forest Regression, Decision Tree Regression, Linear Regression

I. INTRODUCTION

Air quality in most of the cities in India has drastically decreased in recent years. Other than the common pollutant like Carbon Dioxide (CO_2), many newer pollutants like Nitrogen Dioxide(NO_2), Sulphur Dioxide(SO_2), Carbon Monoxide(CO) also has been added into the atmosphere. Most of the pollutants have harmful effects on our health. But CO is more hazardous. It is called as "Silent Killer" because it takes life quietly and quickly. It directly enters into blood and replaces the oxygen molecules thus depriving the brain and heart of necessary oxygen to function. If it is present in the air, it quickly goes into the blood causing symptoms like headache, flu, nausea, dizziness, confusion etc. As the level increases, person may get vomiting, unconsciousness and if exposure is too long it may result in damage of brain or death.

There are many sources of producing CO like incomplete burning of charcoal, coal or wood, burning of solid, liquid and gaseous fuels, running cars etc.

Carbon Monoxide also affects the ability of atmosphere to cleanse itself and responsible for smog and lower atmospheric ozone. Almost all the big cities in India are polluted by carbon monoxide (CO) which extends to as far as 10 kilometers from the surface of the earth in the troposphere. Pollution from vehicles is a major contributor to high carbon monoxide levels. However, researchers believe winds carry the CO produced by biomass burning in Africa and Southeast Asian countries to the Indian subcontinent, thus adding to the already high levels of the gas in the country's atmosphere.

As carbon monoxide due to vehicular pollution and other sources is released, it travels upwards. In monsoon as there is a lot of wind, CO particles reach heights of 10 kilometers as quickly as two hours. That is why in monsoon CO values are the highest in upper layers of the troposphere. In winter, it is the opposite as there is not much wind and the CO particles are closer to the Earth's surface

Concentration of CO can be measured in Parts Per Million(PPM). For example, 100 PPM of CO means that for every 9999,900 molecules of air, there are 100 molecules of CO. Another way of measuring CO concentration is Time Weighted Average (TWA). It measures individual's exposure to CO over time.

The U.S. Standards for CO are as follows: For 1 hour exposure-Maximum of 35 PPM of CO and for 8-hour exposure - Maximum of 9 ppm of CO.

Using pollution masks when air pollution is more than the specified rate can reduce premature deaths linked to air pollution. There are many consumer applications and services that broadcast what the air pollution is and has been. But there are very few that predict what air pollution will be in the future. Like weather conditions, individuals should care more about air pollution in the future, so that they can make decisions about their day/week.

In this paper we developed a system which trains a Machine Learning model using pollution data gathered from government sites and static sensors. The learnt model is used to estimate the air pollution for any day/time in Bengaluru city. It exposes the possibility of developing a Predictive Model for predicting the CO levels.

The rest of the paper is organized as: Section II describes the related work, Section III provides the steps in our model and its implementation, Section IV describes the result from the model, Section V shows the comparison with other papers and Section VI concludes the paper.

II. RELATED WORK

D.J. Briggs et al. [1] have developed a methodology that maps traffic related air pollution within GIS environment. They have considered NO2 in Huddersfield, Prague and Amsterdam. Theirs results show good predictions of pollution levels.

V. Singh et al. [2] have proposed a system that estimates and interpolates daily ozone concentrations. This approach is based on a technique called cokriging.

In [3], M Mead et al. have deployed the sensor nodes in static network in the Cambridge (UK) area and mobile network. They have provided the results for quantification of personal exposure.

P. Dutta et al. [4] have proposed system where individual can measure their personal exposure using participatory sensors then groups to summarize their members' exposure.

K Hu et al. [5] have collected urban air pollution data with high spatial density by using many software applications and hardware devices. They have devised a web based tool and mobile app for the estimation and visualization of air pollution. Their system shows accurate exposure than the current systems.

V. Sivaram et al. [6] developed a project that uses many mobile sensors attached to vehicle to measure the air pollution concentration. The collected data is uploaded to user's mobile. Afterwards pollution maps are created that show the exposure history and accordingly the route can be planned to reduce the future exposure.

K B Shaban et al. [7] developed a system that uses motes equipped with gaseous and meteorological sensors. These communicates to an intelligent sensing platform that comprises of various modules. Mainly four modules have been used for receiving data, preprocessing and converting the data into meaningful information, predicting the pollutants and presenting the information through short message services, web portal and mobile app. They used three ML algorithm namely Support Vector Machine, MSP Model trees and Artificial Neural Networks.

D Hasenfratz et al. [8] have collected the measurements for more than a year through mobile sensor nodes. These nodes were installed on top of public transport vehicles in Zurich (Switzerland). From this obtained data, they developed regression models that create pollution maps with high resolution of 100m

Ke Hu et al. [11] have introduced a machine learning model that takes fixed station data and mobile sensor data and then estimate the air pollution for any hour on any given day in Sydney city. They have used seven regression models and ten-fold cross validation.

Arnab Kumar Saha et al. [12] have used have used cloud based Air Pollution Monitoring Raspberry Pi controlled System. They measured Air Quality Index based on five criteria pollutants, such as particulate matter, ground level ozone, Sulphur Dioxide, Carbon Monoxide and Nitrogen Dioxide using Gas Detection Sensor or MQ135 Air Quality.

Kavitha B C et al. [13] have deployed various IoT sensors on the industrial floor to collect the data and implemented a pollution monitoring system

A Orun et al. [14] have used artificial intelligence technique such as Bayesian Networks to establish relation between traffic and traffic related air pollutants.

Nitin Sadashiv Desai and John Sahaya Rani Alex [15] have measured CO and CO2 level in the air with GPS by using pollution detection sensor and uploaded into Azure Cloud Services.

E Suganya and S Vijayashaarathi [16] have proposed a system that uses Mobile Ad Hoc Network routing algorithm and monitors the travelling vehicles by using number of sensors. The collected data is stored in cloud network to access the information about levels of pollution.

III. SYSTEM DESIGN AND IMPLEMENTATION

The system follows a typical methodology as shown below:

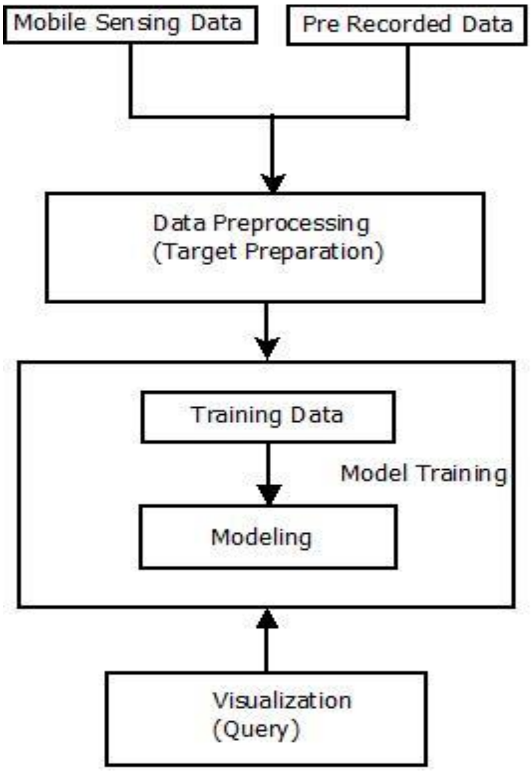


Fig. 1. System Flow Diagram

A. Data Collection

Data is collected from two sources:

- Static Sensor Circuit built using Arduino Uno, MQ7 Sensor, RTC Module and SD Card module.
- Open Source Government Website (www.data.gov.in) [9]

B. Sampling Data

This part is comprised of two separate steps:

(i) Data Preprocessing: This step involves cleaning raw data by removing null values, removing unwanted values, converting one feature into another, extracting other features from one feature etc.

(ii) Target Preparation: This step involves converting data to a form that is easy to handle for the Machine Learning models. This is achieved by performing one of the many methods of target preparation such as Standardization (converts data to have mean 0 and same standard deviation), Min-Max Normalization (converts data so they occur between any fixed interval) etc.

C. *Training Data*

This paper uses three Machine Learning models to predict air pollution rates namely LR, DTR and RFR.

D. *Estimation*

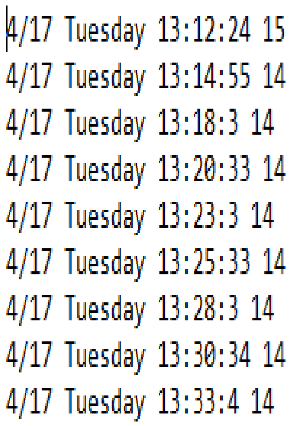
k-fold cross validation is a procedure to estimate the parameters (parameter tuning). In this, the original data sample is randomly partitioned into k equal sized groups. Out of the k groups, a first group is reserved as the validation data for testing the model and the remaining k-1 groups are used as training data sample. Each sample data is used in hold out set one time and then repeated k-1 times (the folds) to train the model. Single estimation result can be produced by averaging the k results from the folds. The benefit of this method over repeated random sub sampling is that all data samples are used for both validation and training, and each data sample is used precisely once for validation. In this system, k=10 i.e. 10-fold cross-validation is used.

E. *User Visualization*

For the user to interact with the program, a simple command line interface has been designed where the user types the day (a number ranging from 1 to 7, 1 being Monday and 7 being Sunday) and hour of the day (ranging from 9 to 21). After typing the details, the user hits the Enter key and the CO prediction is displayed on the screen.

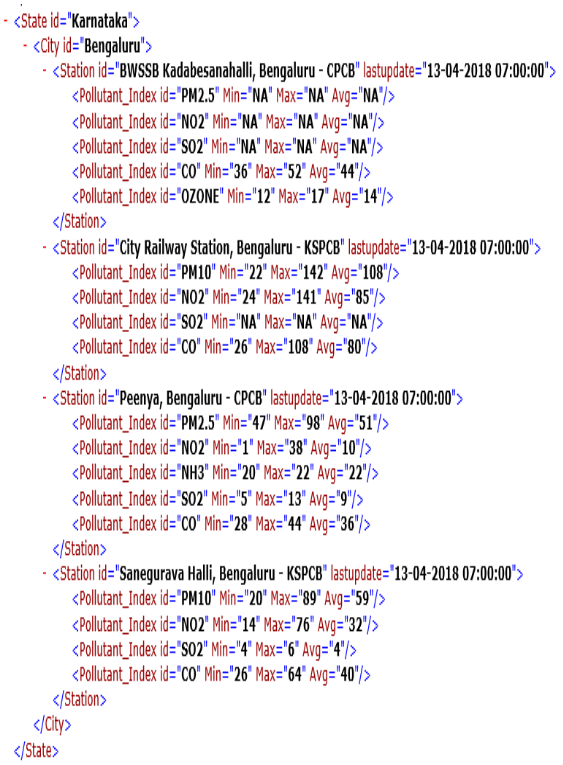
IV. RESULTS AND DISCUSSION

The raw data obtained from both sensor and government websites are pre-processed into structured dataset. The below figures show the raw sample data obtained from sensors and government websites.



```
4/17 Tuesday 13:12:24 15
4/17 Tuesday 13:14:55 14
4/17 Tuesday 13:18:3 14
4/17 Tuesday 13:20:33 14
4/17 Tuesday 13:23:3 14
4/17 Tuesday 13:25:33 14
4/17 Tuesday 13:28:3 14
4/17 Tuesday 13:30:34 14
4/17 Tuesday 13:33:4 14
```

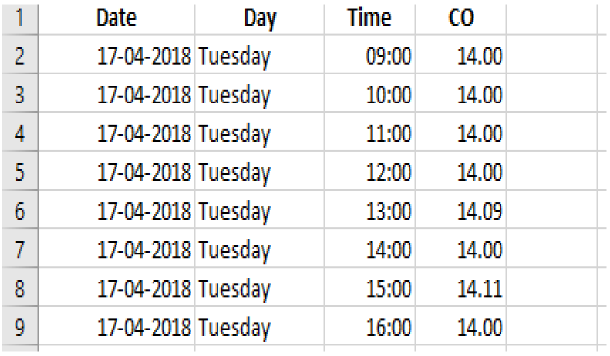
Fig. 2. Raw data obtained from sensor



```
- <State id="Karnataka">
  - <City id="Bengaluru">
    - <Station id="BWSSB Kadabesanahalli, Bengaluru - CPCB" lastupdate="13-04-2018 07:00:00">
        <Pollutant_Index id="PM2.5" Min="NA" Max="NA" Avg="NA"/>
        <Pollutant_Index id="NO2" Min="NA" Max="NA" Avg="NA"/>
        <Pollutant_Index id="SO2" Min="NA" Max="NA" Avg="NA"/>
        <Pollutant_Index id="CO" Min="36" Max="52" Avg="44"/>
        <Pollutant_Index id="OZONE" Min="12" Max="17" Avg="14"/>
      </Station>
    - <Station id="City Railway Station, Bengaluru - KSPCB" lastupdate="13-04-2018 07:00:00">
        <Pollutant_Index id="PM10" Min="22" Max="142" Avg="108"/>
        <Pollutant_Index id="NO2" Min="24" Max="141" Avg="85"/>
        <Pollutant_Index id="SO2" Min="NA" Max="NA" Avg="NA"/>
        <Pollutant_Index id="CO" Min="26" Max="108" Avg="80"/>
      </Station>
    - <Station id="Peenya, Bengaluru - CPCB" lastupdate="13-04-2018 07:00:00">
        <Pollutant_Index id="PM2.5" Min="47" Max="98" Avg="51"/>
        <Pollutant_Index id="NO2" Min="1" Max="38" Avg="10"/>
        <Pollutant_Index id="NH3" Min="20" Max="22" Avg="22"/>
        <Pollutant_Index id="SO2" Min="5" Max="13" Avg="9"/>
        <Pollutant_Index id="CO" Min="28" Max="44" Avg="36"/>
      </Station>
    - <Station id="Sanegurava Halli, Bengaluru - KSPCB" lastupdate="13-04-2018 07:00:00">
        <Pollutant_Index id="PM10" Min="20" Max="89" Avg="59"/>
        <Pollutant_Index id="NO2" Min="14" Max="76" Avg="32"/>
        <Pollutant_Index id="SO2" Min="4" Max="6" Avg="4"/>
        <Pollutant_Index id="CO" Min="26" Max="64" Avg="40"/>
      </Station>
    </City>
  </State>
```

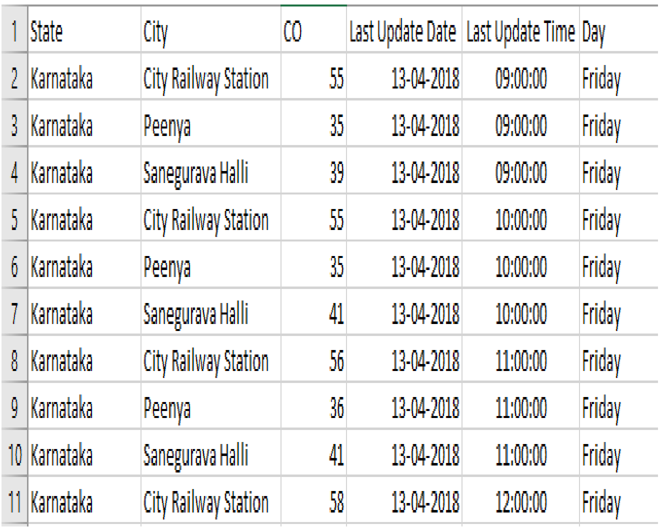
Fig. 3. Raw sample data obtained from government websites

The below figures show the pre-processed dataset obtained from both sensor and government websites.



1	Date	Day	Time	CO
2	17-04-2018	Tuesday	09:00	14.00
3	17-04-2018	Tuesday	10:00	14.00
4	17-04-2018	Tuesday	11:00	14.00
5	17-04-2018	Tuesday	12:00	14.00
6	17-04-2018	Tuesday	13:00	14.09
7	17-04-2018	Tuesday	14:00	14.00
8	17-04-2018	Tuesday	15:00	14.11
9	17-04-2018	Tuesday	16:00	14.00

Fig. 4. Pre-processed dataset sample obtained from sensor



1	State	City	CO	Last Update Date	Last Update Time	Day
2	Karnataka	City Railway Station	55	13-04-2018	09:00:00	Friday
3	Karnataka	Peenya	35	13-04-2018	09:00:00	Friday
4	Karnataka	Sanegurava Halli	39	13-04-2018	09:00:00	Friday
5	Karnataka	City Railway Station	55	13-04-2018	10:00:00	Friday
6	Karnataka	Peenya	35	13-04-2018	10:00:00	Friday
7	Karnataka	Sanegurava Halli	41	13-04-2018	10:00:00	Friday
8	Karnataka	City Railway Station	56	13-04-2018	11:00:00	Friday
9	Karnataka	Peenya	36	13-04-2018	11:00:00	Friday
10	Karnataka	Sanegurava Halli	41	13-04-2018	11:00:00	Friday
11	Karnataka	City Railway Station	58	13-04-2018	12:00:00	Friday

Fig. 5. Pre-processed dataset sample obtained from government websites

The below figures show comparison between CO and independent variables like City, Time, Day.

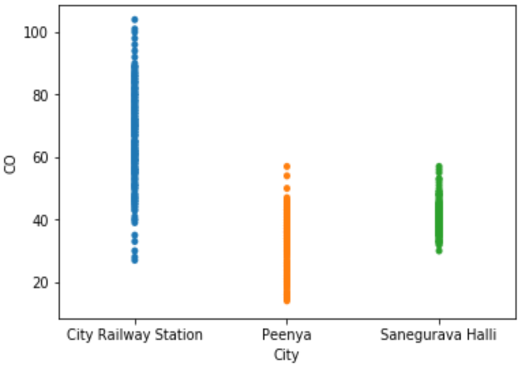


Fig. 6. CO levels vs City for data obtained from Government website.

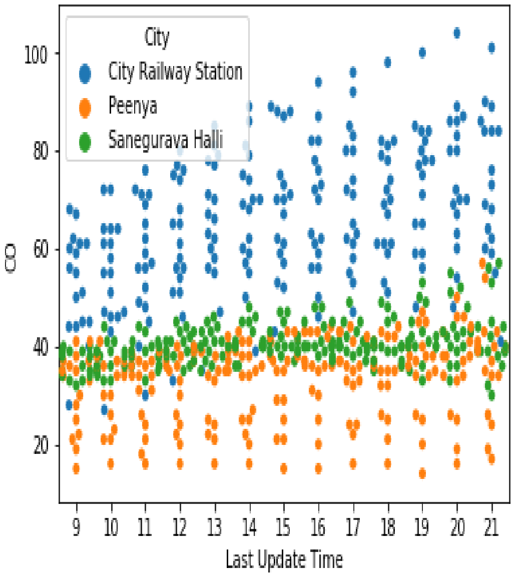


Fig. 7. CO levels vs Time for data obtained from Government website

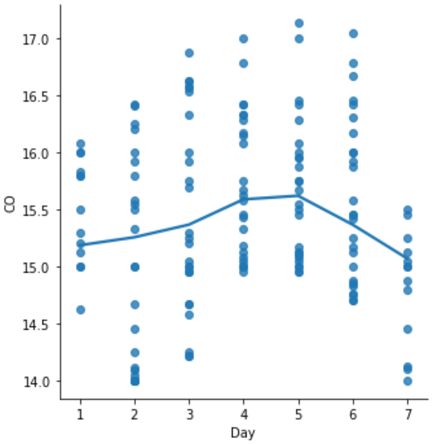


Fig. 8. CO levels vs Day for data obtained from static sensor circuit

TABLE I. ROOT MEAN SQUARE ERROR (RMSE) OBTAINED IN EACH MODEL

Machine Learning Model	***Error***	***Accuracy***
Linear Regression (LR)	13.8275	86.1725
Decision Tree Regression (DTR)	11.6576	88.3424
Random Forest Regression (RFR)	9.7527	90.2473
Linear Regression (LR)	13.8275	86.1725

V. PERFORMANCE ANALYSIS

Arnab Kumar Saha et al. [12] have used have used cloud based Air Pollution Monitoring Raspberry Pi controlled System. They measured Air Quality Index based on five criteria pollutants, such as particulate matter, ground level ozone, Sulphur Dioxide, Carbon Monoxide and Nitrogen Dioxide using Gas Detection Sensor or MQ135 Air Quality.

But their results don’t show Carbon Monoxide level and also accuracy has not been computed. In our work we used MQ7 sensor and government site for measuring the CO level and computed the accuracy using three machine learning algorithms giving the maximum accuracy as 90.2%.

E Suganya and S Vijayashaarathi [16] have proposed a system that monitors the level of NO2, Humidity, Temperature, CO by using NO2 sensor, Humidity Sensor, Temperature Sensor and CO Sensor respectively but in the results CO level and accuracy is not shown. In our results we have shown both CO level and accuracy.

Nitin Sadashiv Desai and John Sahaya Rani Alex [15] have not mentioned any algorithm on which the pollution metrics were predicted. Only Azure Machine Learning Service is used. Accuracy component is missing. In our system we have used different Machine Learning algorithms to compute the accuracy.

Kavitha B C et al. [13] have used the sensors to sense the level of various gases in the industrial floor. They just monitored the amount of pollution but accuracy is missing. Our system shows both accuracy and CO level.

A Orun et al. [14] have established relation between traffic and traffic related air pollutants namely, SO_2, NO_2 and CO. They have developed a Bayesian predictive model using Bayesian classifier with classification accuracies 85%, 78% and 81% respectively for the above-mentioned pollutants. Our work considers only one main pollutant namely CO and assesses the CO level using three Machine Learning Algorithms and it gives the accuracy as 90.2%.

TABLE II. COMPARISON OF PERFORMANCE OF PROPOSED APPROACH WITH VARIOUS METHODS

Authors	**Pollutants**	**Sensor /Dataset Used**	**Algorithm/ Technology Used**	**Accuracy**
Arnab Kumar Saha et al.	Ground Level Ozone, Particulate Matter, CO, SO_2 and NO_2	LM393, MQ135, DHT11	Raspberry – pi/Calculates Air Quality Index	-
Suganya et al.	NO2, Humidity, Temperature, CO	Humidity Sensor, NO2 Sensor, Temperature Sensor, CO Sensor	MANET	-
Nitin Sadashiv Desai et al.	CO and CO_2	MQ7, MQ11	Beagle Bone Black	-
Kavitha B C et al.	CO, LPG, Methane, Butane	MQ135/6/7, DHT11	Raspberry-pi/IoT Shield	-
A Orun et al.	SO_2, NO_2 and CO, Temp, Air Pressure etc.	Dataset	Bayesian Network	SO_2- 85% NO_2-78% CO-81%
Our System	CO	MQ7 and dataset	Three ML Algortihms- LR, DTR and RFR	CO-90.2%

VI. CONCLUSION

In this paper, we have proposed a novel machine learning based system to estimate dense air pollution using historical data both from wireless sensor and government monitoring sites. We chose three regression models (Linear Regression, Random Forest Regression and Decision Tree Regression) and compared the estimation performances. We selected Random Forest Regression (RFR) as the machine learning algorithm because of its low error index. We applied RFR algorithm in our system for its optimal air pollution levels estimation performance.

However, there is still much to do. The proposed model performs better for large training set. But it takes more time to train model. Use of mobile sensors can help to build system to generate air pollution map. Estimation accuracy can be increased in future by introducing meteorological factors such as wind speed and weather in the system. We can also use other Machine Learning models like Artificial Neural Networks to assess the CO level. Future work can also contain predictions for other pollutants as well such as Nitrogen Dioxide, Sulphur Oxides, etc.

REFERENCES

[1] D. J. Briggs et al, *"Mapping urban air pollution using GIS: a regression-based approach,"* International Journal of Geographical Information Science, vol. 11, no. 7, pp. 699–718, 1997.

[2] V. Singh et al., "A cokriging based approach to reconstruct air pollution maps, processing measurement station concentrations and deterministic model simulations," Environmental Modelling and Software, vol. 26, no. 6, pp. 778 – 786, 2011.

[3] M. Mead et al., "The Use of Electrochemical Sensors for Monitoring Urban Air Quality in Low-Cost, High-Density Networks," Atmospheric Environment, vol. 70, pp. 186–203, May 2013.

[4] P. Dutta et al., "Common Sense: Participatory Urban Sensing Using a Network of Handheld Air Quality Monitors," in Proc. SenSys Demonstration, Berkeley, CA, USA, Nov 2009.

[5] K. Hu et al., *"Design and evaluation of a metropolitan air pollution sensing system,"* IEEE Sensors Journal, vol. 16, no. 5, pp. 1448–1459, March 2016.

[6] V. Sivaraman et al., *"Hazewatch: A participatory sensor system for monitoring air pollution in sydney,"* in Local Computer Networks Workshops (LCN Workshops), 2013 IEEE 38th Conference on, Oct 2013, pp. 56–64.

[7] K. B. Shaban et al., *"Urban air pollution monitoring system with forecasting models,"* IEEE Sensors Journal, vol. 16, no. 8, pp. 2598–2606, April 2016.

[8] D. Hasenfratz et al., *"Pushing the spatio-temporal resolution limit of urban air pollution maps,"* in Pervasive Computing and Communications (PerCom), 2014 IEEE International Conference on, March 2014, pp. 69–77.

[9] Open Source Government website: *https://data.gov.in/*

[10] CO information website: *https://www.detectcarbonmonoxide.com/co-health-risks/*

[11] Ke Hu et al., "*HazeEst: Machine Learning Based Metropolitan Air Pollution Estimation from Fixed and Mobile Sensors*" DOI 10.1109/JSEN.2017.2690975, IEEE Sensors Journal

[12] Arnab Kumar Saha et al., "A Raspberry Pi Controlled Cloud Based Air and Sound Pollution Monitoring System with Temperature and Humidity Sensing", 2018 IEEE

[13] Kavitha B C et al., "IOT Based Pollution Monitoring System Using Raspberry-Pi", 2018, Vol 118 International Journal of Pure and Applied Mathematics

[14] A. Orun et al., "Use of Bayesian Inference Method to Model Vehicular Air Poillution in Loacal Urban Areas", Transportation Research Pard D 63 (2018) , pp. 236-243, Elsevier

[15] Nitin Sadashiv Desai and John Sahaya Rani Alex, "IoT Based Air Pollution Monitoring and Predictor System on Beagle Bone Black", 2017 International Conference on Nextgen Electronic Technologies, IEEE

[16] E Suganya and S Vijayashaarathi et al., "Smart Vehicle Monitoring System for Air Pollution Detection using WSN", 2016 International Conference on Communication and Signal Processing, IEEE