Pollution Detection and Prediction System

Project Report

Submitted in partial fulfillment of the requirements for the

Degree of Bachelors in Engineering

by

Rohit Martires (Reg no. 1613032)

Swizel Monteiro (Reg no. 1613033)

Vranda Asolkar (Reg no. 1613001)

Vaylon Fernandes (Reg no. 1613017)

Under Guidance of

Dr. Shreyas Simu (Internal Guide)

Dr. Varsha Turkar (Internal Guide)

Mr. Vassant Salgaonkar (External Guide)

12/21/19

ACKNOLEDGEMENT

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend our sincere thanks to all of them. We are highly indebted to Our Director, Rev. Fr. Kinley D'Cruz for providing us the necessary infrastructure to carry out our project work with ease, Dr. Neena Panandikar, for giving us the opportunity to work on this project. We would like to express our gratitude to Dr. Varsha Turkar, Head of Department of Electronics And Telecommunication Engineering and Dr.Shreyas Simu project guides for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project, we would like to thank Mr. Vassant Salgaonkar of Enviroguru for his expert domain Knowledge, kind co-operation, encouragement and for giving us such attention and time. Our thanks and appreciations also goes to all the faculty members of Don Bosco College of Engineering for their expert advice and critical guidance during various phases of our project. We also are in great depth. to the ETC Lab Assistants, Technical staff and Librarian for assisting us in bring together the project and providing resources whenever required by us for our project work. Lastly, we place on records, a huge amount of thanks to Almighty, our family and friends for their blessings, inspiration and constant support and those who have helped us directly and indirectly for the accomplishment of the project.

Contents

List of Figures

Abstract

To Design and build a gadget/series of gadgets that are durable in highly polluted climates and resistant to a specified amount of temperature and pressure, for the purpose of pollution level monitoring, data collection using IOT, and to study and analyze said data using statistical models and machine learning algorithms (implemented to accommodate for change and scalability) to predict future results with outputs in the form of graphs, charts, tables and images.

1 INTRODUCTION

With the rapid growth of the economy, industrial activities are increasing more frequently, leading to a faster rate of pollution. This rate is only increasing and if not kept in check can cause harmful effects to mankind and other living organisms. These effects are amplified if no regulation is kept in effect. Environmental pollution is one of the most serious problems facing humanity and other life forms on our planet today, industrial pollution contributing a major share in it. Industrial pollution is generally referred to as the undesirable outcome when factories or other industrial plants emit harmful by-products and waste into the environment such as emissions to air or water bodies. The six major types of pollutants are carbon monoxide, hydrocarbons, nitrogen oxides, particulates, Sulphur dioxide and photochemical oxidants.

1.1 Project scope

The Paris agreement's central aim is to strengthen the global response to the threat of climate change by keeping a global temperature rise this century well below 2 degrees Celsius above pre-industrial levels. [5] Long term exposure to polluted air and water cause's chronic health problems making the issue of industrial pollution into a severe one. It also lowers the air quality in surrounding areas which causes many respiratory disorders affecting both lungs and heart. Not just humans, but the marine life is greatly deteriorating and affected with the extent of increasing industrial pollution. However, with effective measures, the ill effect of industrial pollution could be reduced significantly. The prevention and control of industrial pollution are highly encouraged by the government worldwide. Simple things like purchasing energy-efficient equipment and products made from recycled materials for the organization. Having industrial pollution control policies in place and guiding strictly upon them.

1.2 Motivation

The Draft Environment Laws (Amendment) Bill, 2015 was published by the Ministry of Environment, Forest and Climate Change (MoEFCC) on October 7, 2015. The objectives of the Draft Bill are to provide for "effective deterrent penal provisions" and to introduce "the concept of monetary penalty for violations and contraventions".There are no effective strict rules for pollution monitoring and control in industries yet. But the government of India is making industrial pollution control and monitoring laws more strict.

1.3 Objectives of the project

The main objective of the project is to monitor the pollutants emitted from an industry/factory and predict the future dispersion of these pollutants also to then output these results in the form of reports for industries/factories. To accomplish this, the main objective can be broken down into 3 sub-objectives that are

- To create a series of gadgets which can measure the emission parameters and the meteorological parameters and also that can withstand the environmental conditions present near the stack/chimney and transmit this information to a server that is Internet of things.
- To develop a module that uses machine learning models with data acquired from the cloud server to predict future emission parameter values.
- To create a module that simulates the movement of fluid particles (pollutant) in the air using air dispersion models with meteorological data.

2 LITERATURE SURVEY

In [7] the authors proposed a diagnostic model for calculating concentration distribution of a passive scalar in a built-up area Their model basically requires measurements of the wind velocity and direction at a certain reference height above the obstacles. Their model is effectively able to predict 3-d concentration distributions and is also able to identify concentration accumulation at specific and precise points. The proposed model succeeds in predicting concentration distribution both quantitatively and qualitatively. The model can also be further used to study many air pollution phenomena.

The authors of [2] proposed a detailed approach to model dispersion which widely aims at combining the advantages of puff models and particle models. Their resulting model type is called Puff-Particle Model (PPM). In this PPM, they collectively simulated few hundred puffs in three dimensional space, in comparison to many thousand particles usually required in pure particle models. The overall PPM concept is quite simple : while puff growth is described by the concept of relative dispersion which accounts for eddies smaller than the puff, causing effect of meandering ie the variation between the trajectories of different puffs due to larger eddies those larger than the actual puff size and it is simulated by introducing puff-centre trajectories derived from particle trajectories from a particle model.

In [10] the authors proposed that the Lagrangian Monte Carlo particle dispersion models works very effectively for the atmospheric dispersion of effluents. They also mentioned that in order to incorporate the effect of vertical wind shear, the modified dispersion coefficient should be used with the existing Gaussian plume model. As an alternative, they came up with a much reliable 3D numerical model which may be used at a slightly greater computational cost. This numerical model can also be used in a complex topography region with hills and mountains, where mostly the conventional Gaussian models seem to be less effective and mainly not suitable.

The authors of [8] presented a model of NO emissions for a power plant boiler. They have modelled it from the extended Zeldovich mechanism and it is observed that it requires only a few physical parameters obtained from experiments. They used a set of fresh new test data to compare the simulated values with real measurements. It was observed that model performed well with real plant input variables. Their proposed model can also be used in other applications such as for optimising boiler operation and combustion control system design.

In [4] the authors proposed a model which provided an efficient polynomial network solution to the problem of tedious on-line monitoring of NOx emission from industrial boilers. They considered the effect of six variables and studied them using 3D CFD simulation model and used by polynomial networks for prediction of NOx and O2 in the exhaust flue. The prediction of NOx and O2 are both essential for efficient operation and functioning of the boiler while maintaining the NOx pollutant within a tolerable limit. The proposed soft sensor has a simple modular structure for low cost implementation which is greatly an advantage. The soft sensor can also be integrated with the boiler control system for optimization of boilers operation increasing the effectiveness.

In [15] the authors presented a model, to study on the factors to affect the PM-10 pollution and they developed a PM-10 prediction model using MLP neural network model. They used neural model especially because it has an advantage that there doesn't exist a need to analyse the input data before the data are used, like that in regression model. Also, to improve and optimize the performance of the proposed model , it required to shorten the learning period from year to quarter month and to learn and predict PM-10 with multiple networks according to the PM-10 levels.

The authors of [12] proposed a good solution to the complexity of air pollution. It was observed that use of a large number of sensors ensures monitoring accuracy, greatly reduces monitoring cost and makes monitoring data in monitoring area more perfect and systematic. They also noticed that addition of more meteorological factors would highly improve the prediction performance. They used past 5 years data for training the model, showing that large sample data can increase performance and train the model well. The overall experiment used IOT and neural network for air pollution monitoring and forecasting.

The authors of [13] proposed an Air quality monitoring system with and an early warning system, including an assessment module and a forecasting module. The model was able to successfully identify the major pollutants in two cities. The experimental results showed that the proposed model had the best accuracy and stability compared to GRNN, ENN, MCSDE-ENN and MCSDE-EEMD-ENN.

In [6] the authors compared research work on air quality evaluation based on big data analytics, machine learning models and techniques and also highlighted some future resource issues, needs and challenges. They stated that the accuracy of the air quality evaluation and assessment is affected by device faults, battery issues and sensor network. They mentioned due to this issue there is a strong need for research in data quality modelling and automatic real-time validation. They also mentioned that air in a city might be considered as a multi-level air system. The need for research and development of real-time air quality monitoring and evaluation and analysis on multiple levels was also highlighted. They said that smart cities in the future must support real time air quality monitoring, evaluation and prediction. This gives rise to the need to develop integrated and dynamic air quality model using hybrid machine learning models.

The authors of [1] proposed a IOT based air quality monitoring and prediction system. In this model sensor data of a humidity and temperature sensor (DHT11) and a gas sensor (MQ135) was stored on the cloud using a web-enabled microcontroller (ESP8266). This data was then processed (converted into CSV) and used to train the machine learning model and forecast the pollution rate. The machine learning algorithm used is called Long Short Term Memory (LSTM) which is a modification Recurrent Neural networks (RNN).

In [16] the authors proposed refined models for the prediction of hourly concentration of air pollution using meteorological data of previous days by formulating the prediction over a 24 hour period as a multi-task learning (MTL) problem. Their results showed that the proposed light formulation achieved the best result compared to Baseline, and heavy formulation.

In [3] the authors conducted a comparative study of machine learning methods including NARX, ANN and SVR which been employed for air pollution prediction and the NARX was selected as the optimum one. The effective parameters for air pollution prediction have been determined in this research. This research used daily data of the pollutants. They also mentioned that the quality of the proposed model can be significantly improved if hourly data is implemented .Considering the importance of air pollution problem, they recommended that the number of air pollution measuring stations increases so as to allow for a better fit on the air pollution prediction.

The authors of [14] proposed a DNN-based approach to predict air quality. For this purpose they adopted a novel distributed fusion architecture to fuse heterogeneous urban data, which can simultaneously capture the individual and holistic effects from all influential factors affecting air quality. This approach achieves a higher accuracy in both general cases and sudden changes.

The authors of [9] developed a dynamic model to estimate source emissions and predict contaminant concentration in closed spaces. This was realised by using Extended Kalman Filter(EKF) algorithm and least square method based on an established variable –structural contaminant model. Their model could realize to track and real-time predict contaminant concentration, and identify source emission rate accurately and efficiently.

3 GAP ANALYSIS

The previous work in this field included setting up monitoring stations to measure the amount of pollutants in the atmosphere .This was done using network of sensors topologically arranged as a grid.[12] These sensors are usually placed in a large area for e.g. around a city or a large industry. The placement of sensors around a large area can reduce the accuracy of the model. This happens because of external factors like weather, noise, etc. affect the accuracy of the sensors to a large extent. Also, these models can't accurately reflect the extent of pollution a particular industry is causing due to the large area. Some previous models also predicted pollutant concentration for the future based on machine learning algorithms. Also there were some models that predicted dispersion of pollutants from a point source i.e. the area to which the pollutants will spread to.[11] No previous models have linked the emission rate of the stack and the dispersion from the source(stack) together i.e. they have not been predicting the emission rate and calculating the spread in the same model.

The proposed model will however predict emission rate at the stack and use it as a source to calculate dispersion using air dispersion models. This will be done by placing sensors at the source (stack). Placing the sensors at the source will help the sensors take more accurate readings as the sensors will not be affected by external factors to a large extent as compared to the previous models. This will allow us to take more accurate readings and hence make more accurate predictions of future emission rates.

4 METHODOLOGY

In figure 1: methodology flow diagram, the overall setup consists of network of sensors that will be mounted in a specific industry, the data collected from these sensors will be stored on the server. These sensors measure the air parameters in terms of ambient air as well as stack emission. On this data, we apply various machine learning algorithms for prediction of emission rate. The air dispersion models are then applied on the predicted emission rate to calculate the dispersion of pollutants from the source that is at the stack level. The entire system is basically divided into two broad categories:

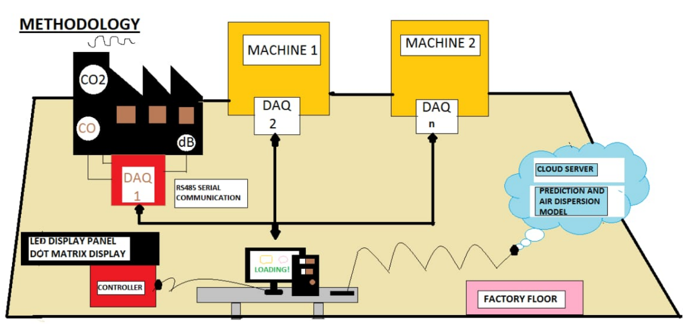

Figure 1: Methodology flow diagram

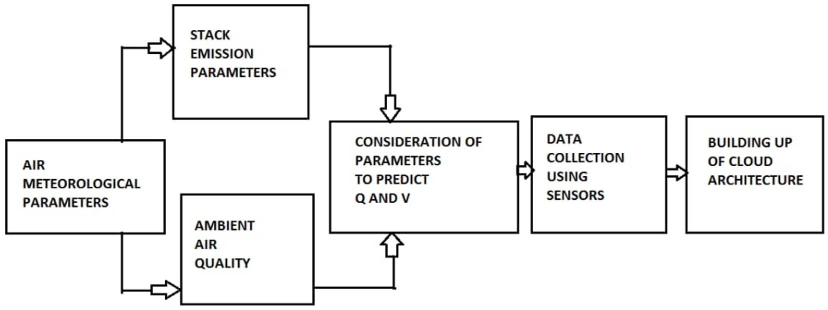

Figure 2: IOT process flow

- Internet of things: the iot process flow begins with o Air meteorological parameters, which are to be measured using various sensors. On the basis of guidelines laid by central pollution control board (cpcb) ,it is divided mainly into ambient air parameters and stack emission parameters.
  - Ambient air parameters, refers to concentrations of pollutants in the air, typically outdoor air. The criteria for this is based upon protection of human health, crops, ecosystems as well as for planning and other useful purposes.
  - Stack emission parameters, parameters required to be monitored in the stack emissions using continuous emission monitoring systems, are industry specific.
  - Consideration of parameters to predict q and v, we considered all the air meteorological parameters inorder to predict the value of v(velocity of wind) and q(emission rate).
  - Data collection using sensors, with the help of various sensors employed in a specific industry, huge amount of data was collected.
  - Building up of cloud architecture, the data collected from the sensors is uploaded and stored on the cloud setup. The sensor network is connected to the cloud server

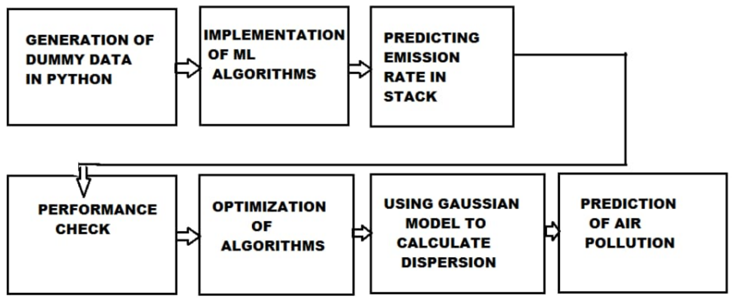

Figure 3: Machine Leaning process flow

- Machine learning: the machine learning process flow begins with
  - Generation of dummy data in python, huge amount of data, in bulk is generated using python. The dummy data was not truly random, it was correlated with various meteorological air parameters so that the machine could be trained well. It was observed that by adding more meteorological factors, the prediction performance is greatly improved and large sample data can improve performance and train the model well.
  - Implementation of ml algorithms, machine learning algorithms were then implemented on the created dummy data to predict the value of q-emission rate and v-velocity of wind. For this purpose, the data was divided into training set (80%) and test set (20%). If the error reduces for training as well as test data, the process is continued. Else if the error reduces only for training data, but increases for test data then the process is terminated. This could greatly increase chances of generalizability of the algorithm.
  - Performance check, was then conducted on the predicted emission rate. The mean square error was measured in each case to check for accuracy.
  - Optimization of algorithms, various algorithms were optimized in such a way so as to reduce the error as minimum as possible thereby increasing the accuracy of prediction. On this basis, the best algorithm was selected.
  - Using Gaussian model to calculate dispersion, the extent of pollution spread .we chose Gaussian dispersion model since it was the most optimal model in terms of computing power.
  - Prediction of air pollution, in this manner the entire process of prediction of pollution and calculation of its spread is done.

4.1 Machine Learning Algorithms

Based on previous work the various machine learning algorithms were compared based on the pollutants considered, accuracy, simplicity, robustness, flexibility, sensors used, and other factors that affect the choice of models. The following proved to have best results hence these were used:-

1. Kth Nearest neighbors (KNN)-In this machine learning model similar things are considered to be in close proximity to each other. The value of K indicates the nearest neighbors that are taken for consideration. This similarity criterion can be based on distance or other similar factors. It uses a database that is separated into several classes to predict the classification of a new sample point. It is extended as a regression technique wherein the output of the point is the average or median value of its nearest neighbors. In KNN the training data required is very less, which means there is a lack of generalization, from (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/).

2. Support Vector Regression (SVR)- SVR is a supervised machine learning algorithm that is being used for both classification and regression problems. In this algorithm, a data point is plotted in n-dimensional space after which it is classified by finding the plane that will differentiate the classes very well. The hyperplane which is considered is a linear separator of any dimension (line, plane, hyperplane). The training points are used in decision function and are called support vectors, from (https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/).

3. Random Forest- The decision tree is a decision support tool. It uses a graph in the form of a tree to show possible consequences. Random forest operates by constructing multiple decision trees during the training phase. The decision of the majority of the trees is chosen by the random forest as the final decision. When a training data set is considered with targets and features, some set of rules are formulated, these rules are used to perform predictions. It identifies the most important features out of all the available features in the training dataset. The larger the no. of trees the more accurate the results will be. Random forest classifier handles missing values and overfitting problem doesn't exist, from (https://medium.com/@Synced/how-random-forest-algorithm-works-in-machine-learning-3c0fe15b6674).

4. Multilinear regression (MLR)- This technique uses several explanatory variables to predict the outcome of the response variable. A linear relationship is modeled between these independent and response variable (dependent). Prediction about one variable is done based on the information about the other variable. In this case, the independent variables should not be too highly correlated with each other, (https://www.investopedia.com/terms/m/mlr.asp).

5. Neural Network- it is a model that mimics the way the human brain operates. A neuron in a neural network is a mathematical function that collects and classifies information according

to some specific architecture. A neural network contains layers of interconnected nodes. Each connection is associated with a weight that is multiplied with the input value. Each neuron has an activation function that defines the output of the neuron which is used to introduce non-linearity in the network model, from (https://www.investopedia.com/terms/n/neuralnetwork.asp).

These Machine learning algorithms were implemented using python and the mean square error of each of these was measured to check for accuracy.

4.2 Air Dispersion Model

4.2.1 Gaussian Dispersion Model

Gaussian dispersion model [11] is a steady-state system, the concentration of pollution downwind from a source is treated as spreading outward from the stack. A point source was considered somewhere in the air where a pollutant is released at a constant rate Q (kg/s). The wind is blowing continuously in a direction x (measured in meters from the source) with a speed U (m/s). The plume spreads as it moves in the x-direction such that the local concentrations C(x,y,z) (kg/m3) at any point in space form distributions which have shapes that are Gaussian or normal in planes normal to the x-direction i.e. the concentration of the pollution will be maximum at the source and will gradually disperse where this dispersion follows a gaussian curve. The maximum concentration is in the direction of the wind and there will be lateral dispersion on the y-z plane. The more the dispersion area, the more diluted will be the pollutant.

4.2.2 Lagrangian Model

The Lagrangian model determines the trajectory of air pollutants. The pollutants are tracked as air parcels which move along trajectories determined by wind field, the buoyancy and turbulence effects. The trajectory calculations are based on ordinary differential equations(ODE) instead of partial differential equations which are used in the gaussian dispersion model. The estimation of the concentration field is given by the final distribution of a large number of particles. The particles can be tracked from the source area to the area of reception. The computational cost of these models is independent of the output grid resolution and hence they are very efficient for short-range modeling. For long-range simulations, however, the computational cost increases as there is a need to calculate a large number of single trajectories. Also, a large amount of computational power is required to handle a large number of puffs.

4.2.3 Puff-Plume Model

The puff-plume model maintains the advantages of the puff models and also of the plume models. The pollutant particles are grouped in clusters. These clusters are treated as Gaussian puffs which are dispersed using the concept of relative diffusion. The center of mass of each puff follows a stochastic trajectory. The particle trajectories of the Lagrangian stochastic dispersion model gives the trajectory of the puffs.

Why Gaussian model?

The Gaussian dispersion model has been selected as it has relatively less computational time as compared to the Lagrangian model in which the computational time increases significantly with an increase in distance.

4.3 Dataset and study area

The objective of the prediction module is to predict the emission rate at the stack/chimney. To build and test the prediction model, 1000 dummy data points were generated using Gaussian distribution to randomly generate values for the feature set with guidance from an industry expert, this data was later split into training and testing that is 80% and 20% respectively. The feature set is chosen on the basis of what factors will correlate to the emission rate and the type of pollutants emitted from the stack. The feature set consists of independent variables: day, month, type of industry, size of the industry, and output efficiency of industry and dependent variable: emission rate.

- Type of industry: what the industry/factory produces which will correlate to what gases are emitted out of the stack/chimney, this will be in the form of labeled classes.
- Size of the industry: how big is the industry/factory which will correlate how much the maximum is outputted, this will be represented in the form of a scale from 1 to 10.
- Output efficiency: the amount of output it produces each day divided by the total amount of output it can ideally produce.
- Emission rate: this is defined as the amount of pollutants released from the stack per unit time.

The prediction module is also used on the collected meteorological data to predict the air velocity and direction, this is then applied to calculate the dispersion of pollutants in the air.

The feature set of this prediction model consists of independent variables: day, month, ambient temperature, ambient pressure, moisture content and dependent variables: the air velocity and air direction.

Day and month are used to timestamp data points also temperature, pressure and moisture content of the ambient air is correlated to the velocity and direction of air.

5 RESULTS

The prediction module and air dispersion module was implemented in python programing language using Spyder Integrated development environment.

The prediction module was tested on the dummy data set generated then the precision was measured by finding the deviation of the predicted value from the actual value using mean square error. In an attempt to optimize the models the parameters of the models were tuned and the response (mean squared error) was graphed out.

The air dispersion module was implemented to calculate the concentration from a stationary steady state point source i.e. stack and the results were mapped onto a grayscale bitmap image where the pixel color represents the concentration.

5.1 Machine Learning module

Different machine learning models were studied and implemented on the same dataset and the results were compared. The following table shows the range of error in the respective models while the parameters were tuned

Sr no.	Name of ML model	Parameters tuned	min error $\frac{m^3}{hr}$	max error $\frac{m^3}{hr}$	avg. error $\frac{m^3}{hr}$
1	Random Forest	• maximum depth • maximum features • minimum sample split • minimum sample leaf • n estimators	5	24	14.5
2	Multi-linear perceptron regression	• activation function • number of neurons • solvers	2.2	2.7	2.45
3	Support vector regression	• kernel	10	10.8	10.4
4	K-nearest neighbor	• weight • algorithm	3	14	8.5
5	Multi-linear Regression	none	10	10	10

The following graphs shows the mean square error while the parameters are being tuned for the respective models

- Random forest The random forest model was tuned on its parameters that are n_estimators, min_sample_leaf, max_depth, min_sample_split, max_features

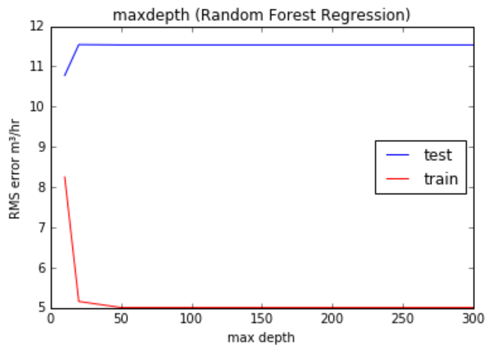

Figure 4: parameter tuned max depth

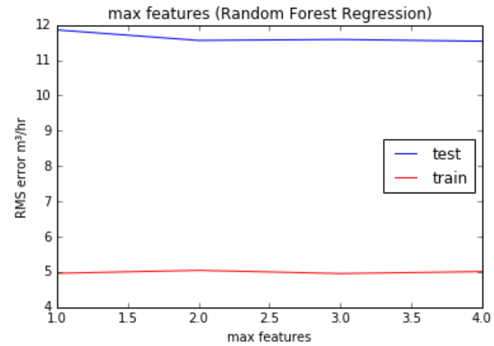

Figure 5: parameter tuned max features

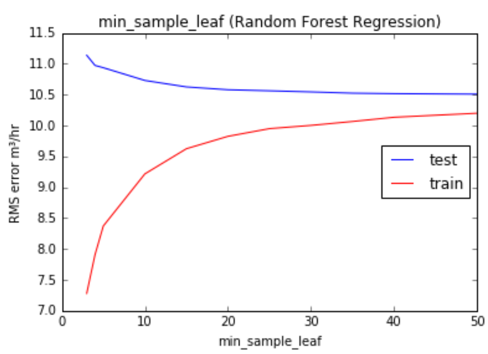

Figure 6: parameter tuned min sample leaf

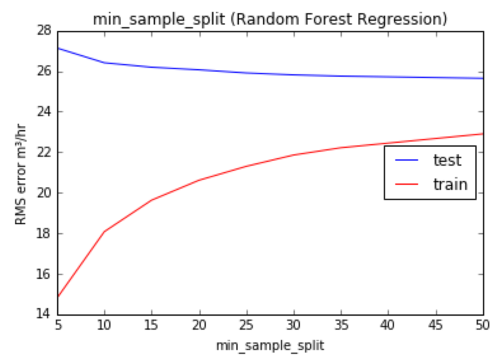

Figure 7: parameter tuned min sample split

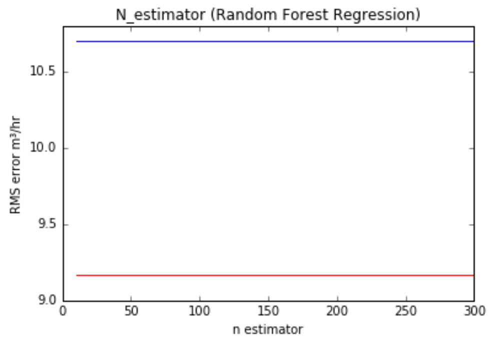

Figure 8: parameter tuned n estimators

- In random forest we can see that tuning the n estimators, max features and max depth doesn't vary the error by any significant value but tuning min sample leaf and min sample split does impact the error.

- Multi-linear perceptron regression

Here the model is tuned on activation function, number of neurons and type of solvers

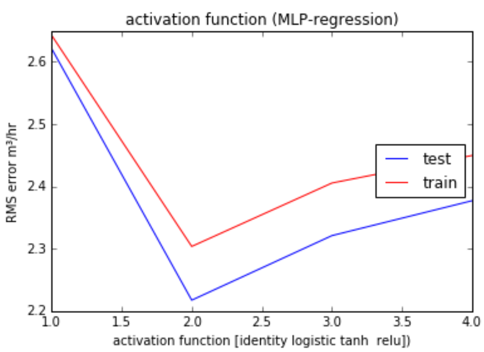

Figure 9: parameter tuned activation function

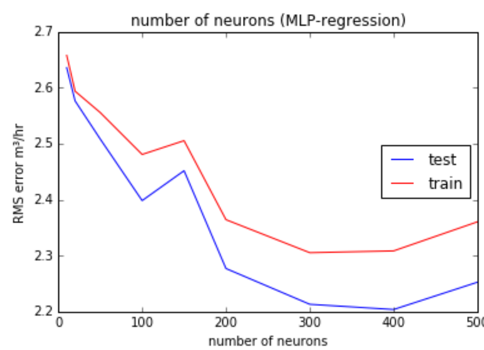

Figure 10: parameter tuned number of neurons

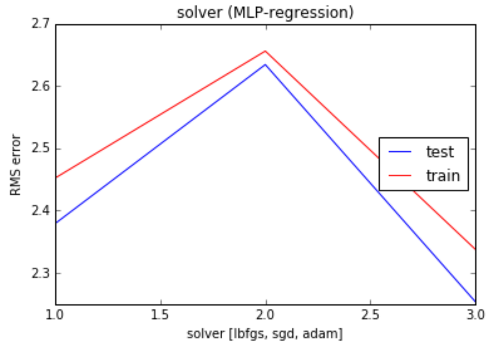

Figure 11: parameter tuned solver

In mlp the results shows that after tuning activation function, number of neurons and type of solvers the error change is lies between 2.2 and 2.6 which is not much significant

- Support vector regression

This model is only tuned on its kernel property

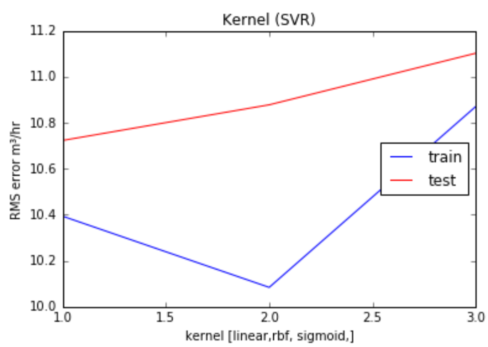

Figure 12: parameter tuned kernel

The SVR model was tuned on its kernel property and the following results show that the rbf kernel had the least error

- K-nearest neighbor

This was tuned on its weight and algorithm parameter, which are used for clustering

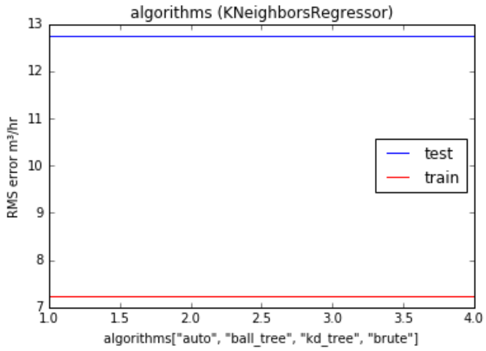

Figure 13: parameter tuned algorithm

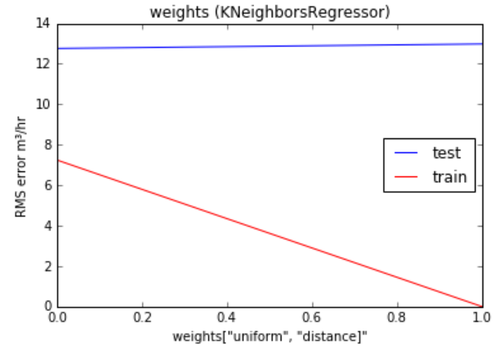

Figure 14: parameter tuned weights

This KNN model was tuned on its algorithm and weight parameter used to cluster data points the results shows that varying both these parameters does not change the error by any significant amount

- Multi-linear Regression

This algorithm is an extension of linear regression with added support of higher dimensions in feature space

Figure 15: predicted and actual data using samples form the test data

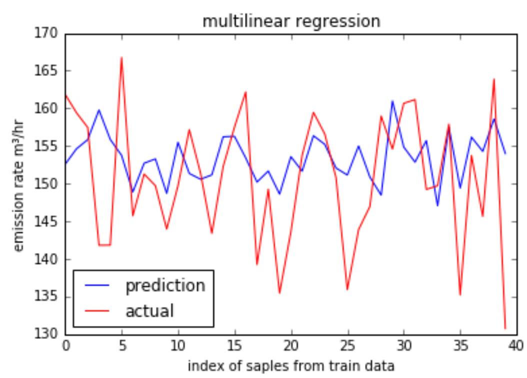

Figure 16: predicted and actual data using samples from the train data

The above results shows that the predicted values closely follows the actual values with an average error of about $10m^3/hr$

The results of the prediction module clearly shows that the multi-perceptron regression model has the least mean square error out of all the models also we can see that the error using test data loosely follows the error using training data i.e. the area between both plots over parameter tuning does not change much which is a good indication that overfitting is not occurring. Whereas multi-linear regression has the most error due to its simplistic design

5.2 Air Dispersion module

To calculate the concentration of the pollutant after emission from the stack, Gaussian air dispersion model was used for simulation, this was implemented using python programing language

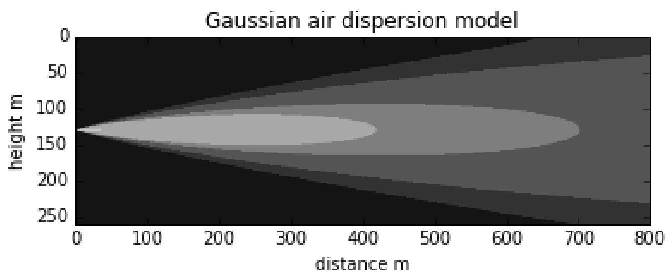

Figure 17: Gaussian air dispersion model

the white region represents regions with high concentration and black with low concentration

6 CONCLUSION

The pollution detection and prediction system which consists of 3 main modules of which 2 were successfully implemented that is

- Air dispersion module: in the module the Gaussian air dispersion model was implemented using python in spyder IDE.
- Prediction module: in this module 5 machine learning models were implemented and compared which include (random forest, multi-linear perceptron, knn, svr, multi-linear regression) from these the multi-linear perceptron model was seen to have the least error.

Therefore 2/3 of the system was designed and successfully built.

References

[1] Temesegan Walelign Ayele and Rutvik Mehta. Air pollution monitoring and prediction using iot. In *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, pages 1741–1745. IEEE, 2018.

[2] Peter De Haan and Mathias W Rotach. A novel approach to atmospheric dispersion modelling: The puff-particle model. *Quarterly Journal of the Royal Meteorological Society*, 124(552):2771–2792, 1998.

[3] Mahmoud Reza Delavar, Amin Gholami, Gholam Reza Shiran, Yousef Rashidi, Gholam Reza Nakhaeizadeh, Kurt Fedra, and Smaeil Hatefi Afshar. A novel method for improving air pollution prediction based on machine learning approaches: A case study applied to the capital city of tehran. *ISPRS International Journal of Geo-Information*, 8(2):99, 2019.

[4] Moustafa Elshafei, Mohamed A Habib, and Mansour Al-Dajani. Prediction of boilers emission using polynomial networks. In *2006 Canadian Conference on Electrical and Computer Engineering*, pages 823–827. IEEE, 2006.

[5] Robert Falkner. The paris agreement and the new logic of international climate politics. *International Affairs*, 92(5):1107–1125, 2016.

[6] Ganganjot Kaur Kang, Jerry Zeyu Gao, Sen Chiao, Shengqiang Lu, and Gang Xie. Air quality prediction: Big data and machine learning approaches. *International Journal of Environmental Science and Development*, 9(1):8–16, 2018.

[7] H Kaplan and N Dinar. A lagrangian dispersion model for calculating concentration distribution within a built-up domain. *Atmospheric Environment*, 30(24):4197–4207, 1996.

[8] N Li and S Thompson. A simplified non-linear model of nox emissions in a power station boiler. 1996.

[9] Hongquan Qu and Liping Pang. A dynamic method to estimate source emission rate and predict contaminant concentrations. In *2009 2nd International Congress on Image and Signal Processing*, pages 1–3. IEEE, 2009.

[10] S Raza, R Avila, and J Cervantes. A 3d lagrangian particle model for the atmospheric dispersion of toxic pollutants. *International journal of energy research*, 26(2):93–104, 2002.

[11] Kulshresth Singh. Air pollution modeling. *International Journal of Advance Research, Ideas and Innovations in Technology*, 4(3), 2018.

[12] Chen Xiaojun, Liu Xianpeng, and Xu Peng. Iot-based air pollution monitoring and forecasting system. In *2015 International Conference on Computer and Computational Sciences (ICCCS)*, pages 257–260. IEEE, 2015.

[13] Zhongshan Yang and Jian Wang. A new air quality monitoring and early warning system: Air quality assessment and air pollutant concentration prediction. *Environmental research*, 158:105–117, 2017.

[14] Xiuwen Yi, Junbo Zhang, Zhaoyuan Wang, Tianrui Li, and Yu Zheng. Deep distributed fusion network for air quality prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 965–973. ACM, 2018.

[15] SH Yu, YS Koo, EY Ha, and HY Kwon. Pm-10 forecasting using neural networks model. In *2008 International Conference on Computational Intelligence for Modelling Control & Automation*, pages 426–429. IEEE, 2008.

[16] Dixian Zhu, Changjie Cai, Tianbao Yang, and Xun Zhou. A machine learning approach for air quality prediction: Model regularization and optimization. *Big data and cognitive computing*, 2(1):5, 2018.