Real-Time NO2 monitoring alert using XGBoost

1st Aarushi Vyass
Amity School of Engineering and Technology
Amity University Mumbai
Mumbai, India
aarushvyas23@gmail.com

2nd Tomoyuki Koyama
Graduate School of Computer Science
Tokyo University of Technology
Tokyo, Japan
g21210247f@edu.teu.ac.jp

3rd Vishnu Vinodkumar
Amity School of Engineering and Technology
Amity University Mumbai
Mumbai, India
vishnuz1611@gmail.com

4th Takayuki Kushida
School of Computer Science
Tokyo University of TechnologyTokyo, Japan
kushida@acm.org

***Abstract*— Air pollution has become a major health hazard in developing countries. IoT enables to monitor air pollution and analyze measured values such as temperature. This study focuses on monitoring the real-time NO2 quality pattern in a given region. Using the Random Forest algorithm with XGBoost, not only a range of how air quality varies across regions can be determined but also enables a feature of alerting the user to wear a mask of a standard pertaining to the concentration in the air. This study shows a promising outcome for effective NO2 level monitoring and prediction for a smart city application.**

Keywords— IoT, Time Series prediction, NO2 sensing, Cloud Computing

I. INTRODUCTION

Air pollution is a major problem in several developing countries. India is the world's third-largest superpower due to reasons such as traffic congestion and biomass burning [1]. Exposure to air pollution over a long period of time is a threat to the human body [2]. Air pollution causes human health damage. Harmful substances such as NO_2 are inhaled from nasal passages over the respiratory tract [3]. When a human inhales the substances, the human cells are stimulated and act as a part of multicellular immune responses. As a result, the substances cause human diseases.

Nitrogen oxides (NOx) are one of the substances that cause health hazards [4]. The book describes the primary source of NOx as follows: “The main source of nitrogen oxides in cities is the combustion of fuels by motor vehicles and stationary sources such as industrial facilities.”. In fact, there are ten times more cars today than there were 50 years ago [5].

Motor vehicles support people to move anywhere and industrial facilities provide consumers with products. Therefore, it is necessary to balance the convenience of life with air pollution. Air pollution monitoring system protects people's health. The measured air quality value is stored on the system. When the value has exceeded the criteria or threshold, the system sends people alerts by email or short messages. Developing cities in Africa don’t have established air quality monitoring networks. The prospects of building the networks in the cities are uncertain [6]. IoT device is utilized for air pollution monitoring with wireless sensor network. It measures air pollution level in a specific region by sensors and transfers the air pollution level to centralized server over the internet. The centralized server predicts the next level of air pollution level by machine learning algorithms.

II. RELATED WORK

Pollutant gas level is measured in the form of PPM and is transmitted through GPRS and the location is transmitted using a global positioning system [7]. Saini et al. describe a systematic review of the Indoor Air Quality (IAQ) monitoring systems employing IoT devices [8]. The review is limited to the period contained between 2015 and 2020, while the main research questions have focused on 1) sensors used for IAQ monitoring, 2) supervised parameters, 3) microcontrollers used to articulate the sensors in the data gathering strategy, 4) preferred interfaces for air quality sensing, 5) communication technologies, 6) power requirements, and 7) details about the incorporated functionality. As a difference, our proposal is oriented to PM monitoring and the impact on people’s health (indoor and outdoor) specifically.

There are methods performing vaticination on PM2.5 and PM10 parameters and parameters available in the selected dataset and further trained on four cited algorithms to prognosticate PM2.5 and PM10 values for near-term days, calculating RMSE error rate between available dataset factual values and prognosticate values [9].

The approach uses public and private web services as well as a list of public websites to provide real-time meteorological, weather forecasts and air quality data for their forecasting [10]. Further enhancement of the fixed sensors, public transportation infrastructure such as buses has been used to collect air quality data [11].

In the field of air quality prediction, regression models are often employed for prediction. Another study proposes a multivariate linear regression model for forecasting PM2.5 over short time periods, which incorporates additional gaseous pollutants such as SO_2, NO_2, CO, and O_3 [12].

Another approach has applied the Extreme Gradient Boosting (XGBoost) algorithm to predict hourly PM2.5 concentrations in China and compared it with the results from the random forest, support vector machine, linear regression and decision tree regression, and demonstrated the best performance of the XGBoost algorithm in air quality forecasting [13].

III. SOLUTION APPROACH

The proposed approach can be divided into four main phases as described in this section. Figure.1 represents how the plan of action, data management, visualization and implementation are carried out in the afore mentioned four phases:

1. IoT Device Architecture
2. Real-time monitoring using Cloud Computing

3. Pre-Processing of Acquired Data
4. Prediction Algorithms

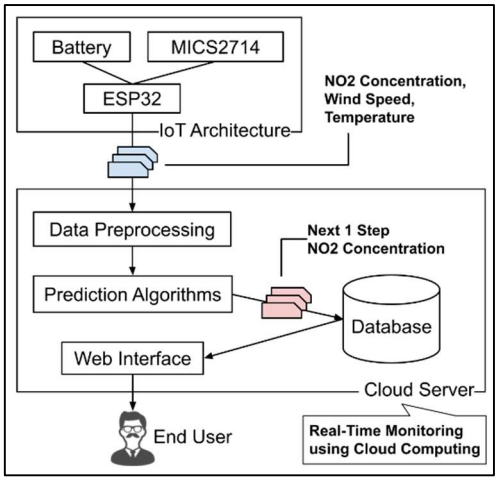

Figure 1. Proposed solution approach

A. IoT Device Architecture

Functions of the sensors are described as following:

- Nitrogen Dioxide Sensor: Nitrogen dioxide sensor (MICS 2714) reacts to the presence of NO_2 and measure levels within the range 50 ppb to 10 ppm. Impedance changes as a result of a catalytic reaction, allowing it to be used in a voltage divider configuration. It is a robust sensor, which can be used in harsh environment. Detection of the pollution gas is achieved by measuring the sensing resistance of the sensor. The sensor resistance increases in the presence of NO_2.
- ESP-32 NodeMC: Its main function is to process the sensor data and send it over the internet to the cloud server. ESP32 is to be programmed using Arduino IDE.
- Battery: We use a power bank with a capacity of 7,000 mAh. The overall power consumption of the setup is close to 1A.

B. Real-Time Monitoring using cloud computing

The proposed system runs on top of the IoT sensor and transmits collected data back to a curated system:

- Database: We design the database to store the collected real time sensor values from fixed IoT sensor. The data-fields are: 1) time 2) NO_2 reading and 3) Air quality.
- Cloud Server using ThingSpeak and Data Mapping
- User-Interface

The system sends data to the server at regular intervals over the internet. Similar to connecting any device to an accessible Wi-Fi network, the ESP32 connects to an available Wi-Fi field, which in turn connects to the internet. The data is saved in the server database and is shown in real time as a graph. This strategy allows for monitoring and informing the user of any issues or errors. Using this strategy, the IoT architecture may submit data to the server, allowing the user to monitor many systems in different places with no effort. ThingSpeak[1] tracks the levels. ThingSpeak is a cloud-based IoT analytics platform solution that allows you to gather, view, and analyze live data streams. ThingSpeak delivers real-time visualizations of data sent to it by data collecting devices. With the ability to execute code in ThingSpeak, real-time data analysis and processing can be done.

[1] https://thingspeak.com/

C. Pre-Processing of Acquired Data

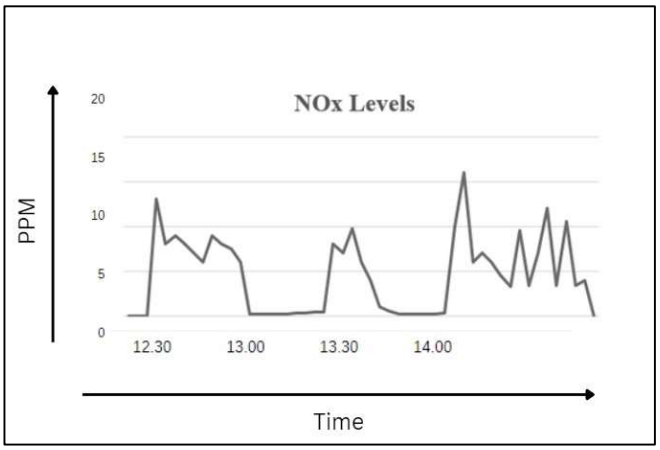

Figure 2. NO_2 data values prior to pre-processing

Pre-processing helps removing any noise, missing values, from the acquired data to develop a robust prediction model. As shown in figure 2. the data reading recorded by the sensor for NOx in Mumbai from 10th November to 12th November 2022, is then retrieved from the ThingSpeak sever, includes hourly information on NO_2 level conditions. Prior to use in the prediction algorithm, the dataset is converted to a time series dataset to solve a supervised learning problem.

Outlier detection: The process involves identifying data-points that fall outside an expected distribution pattern. Measured samples with a discrete difference beyond the interval [-0.5, 2] are removed.

Interpolation: We choose Gaussian Process Regression (GPR) as our interpolation method as it helps in reaching thebest prediction accuracy in our experiments.

Data Normalization: Since data is measured at different scales, we need to normalize the sensor measurements between 0 and 1.

Evaluation Metric: Root Mean Square Error (RMSE) calculates the square root of the mean for the square of the differences between the predicted and actual values. It is calculated as:

$$RMSE = \sqrt{\sum_{i=1}^{n} (P_i - A_i)^2 / n}$$

where n is the number of samples, P_i and A_i are the predictedand actual values, respectively.

Coefficient of determination (R2): This parameter evaluates the association between actual and predicted values. It is determined as:

$$R^2 = 1 - \frac{\sum_{i-1}^{n} (A_i - P_i)^2}{\sum_{i=1}^{n} (A_i - \bar{A})^2}$$

where n is records count, A is the mean measured value of the pollutant. R^2 is a descriptive statistical index. It is either 1, 0 or negative if prediction value is worse than the baseline model.

D. Prediction Algorithms

While working on an aspect of it, the statement was confronted with the problem of choosing between a `

Random Forest and a XG Boost. Random Forests in XGBoost (XGBRF) Gradient-boosted decision trees and other gradient-boosted models will have been trained with either XGBoost or Random Forests. This training is possible because they have the same model representation and inference, but their training algorithms are different. XGBoost can use Random Forests as a base model for gradient boosting or can be used to train standalone Random Forests. XGBRF training focuses on the standalone random forest. This algorithm is a scikit-learn wrapper included in the opensource library of XGBoost. NARX (Non-linear AutoRegression with eXogenous Input) Model is mostly used to model time series. It is a nonlinear version of the autoregressive model incorporating extrinsic (outside) input, determines the output in a linear relationship with its previous values. omparing the present value of a time series to prior values of the same series, as well as current and previous values of the driving series. A function exists that translates input values to output values. The model works by inserting input features from successive time steps and grouping past time steps in parallel in the exogenous input sequence. Each of these features can be delayed by time steps. Such a model can be stated algebraically as:

$$y_t = F(y_{t-1}, y_{t-2}, y_{t-3}, \dots, u_t, u_{t-1}, u_{t-2}, u_{t-3}, \dots) + \varepsilon_t$$

y : various of interests
u : eternally declined variable
ϵ : error term
F : non-liner function

IV. IMPLEMENTATION DETAILS

Our deployment and collection of data is based in Mumbai, India. We then explain how the data is being processed using the device and how we store and transmit this information. Further, we explore the user interface, that will display the collected information to the user.

The information obtained through the sensor is transmitted to the network by using sensors and a suitable hardware device. IoT is low power wide area networking Technology that is developed to enable efficient communication by providing much wider coverage with long battery life and lower cost. Other environmental threats similar to the targeted NO_2 like PM10, PM2.5, temperature, humidity, and several other chemicals have also proven to be easily detected by the IoT node. Devices and sensors are the components of the connectivity layer. These smart sensors continuously collect data from the environment and transmit it to the next layer. IoT devices and low-power wide-area networks make it a lot easier to get and disseminate ultra- local data. The proposed system consists of main components as shown in figure 1. IoT hardware connected to nodes for detecting various NO_2 metrics, consisting of sensors coupled to a Wi-Fi enabled microcontroller unit ESP32 (NodeMCU).
MicroPython is an open-source Python programming language that runs on small embedded development boards, via Arduino development platform. It enables to program of the device with clean and precise code. Further, a central cloud for monitoring the condition of various parameters is set for values supplied to a trained model. The proposed system is further connected to a web application, the frontend, using React and JavaScript library for building user interfaces where the website sends prompts to the connected device.

V. RESULT

The project proposed hybrid NARX architecture that incorporates a machine learning algorithm that predicts atmospheric NO_2 data. The use of edge devices to predict NO_2 levels in air is critical because they allow for faster response in the event of air pollution incidents or in the event of insufficient internet connectivity or in a remote location.

The performance of the machine learning algorithm used in this work was investigated by applying them to the same dataset. Fast prediction algorithms are preferable to work efficiently on edge devices. The results would show that XGB related methods are fast and the best method for both efficiency and accuracy is NARX/XGB.

As for future scope, we intend on full scape deployment by making use of MICS 2714 sensor capturing NO2 levels, NARX architecture with a deployed deep learning system containing XGBoost algorithm ready for pre-processing. Further a ThingSpeak hybrid cloud architecture connected as backend not only to a website but also a mobile application notifying users for mask safety.

Project URL: https://youtu.be/oqzjpLL5fnQ

REFERENCES

[1] Khot, R., & Chitre, V. (2017, March). Survey on air pollution monitoring systems. In 2017 international conference on innovations in information, embedded and communication systems (ICIIECS) (pp.1-4). IEEE.

[2] Mannucci, P. M., & Franchini, M. (2017). Health effects of ambient air pollution in developing countries. International journal of environmental research and public health, 14(9), 1048.

[3] Glencross, D. A., Ho, T. R., Camina, N., Hawrylowicz, C. M., & Pfeffer, P. E. (2020). Air pollution and its effects on the immune system.Free Radical Biology and Medicine, 151, 56-68.

[4] McGranahan, G., & Murray, F. (Eds.). (2012). Air pollution and health in rapidly developing countries. Earthscan.

[5] Fenger, J. (2009). Air pollution in the last 50 years–From local to global. Atmospheric environment, 43(1), 13-22.

[6] Khot, R., & Chitre, V. (2017, March). Survey on air pollution monitoring systems. In 2017 international conference on innovations in information, embedded and communication systems (ICIIECS) (pp.1-4). IEEE.

[7] Moses, L. (2020, November). IoT enabled Environmental Air Pollution Monitoring and Rerouting system using Machine learning algorithms. In *IOP Conference Series: Materials Science and Engineering* (Vol. 955, No. 1, p. 012005). IOP Publishing.

[8] Saini, J., Dutta, M., & Marques, G. (2020). Indoor air quality monitoring systems based on internet of things: A systematic review. International journal of environmental research and public health, 17(14), 4942.

[9] Zhang, D., & Woo, S. S. (2020). Real time localized air quality monitoring and prediction through mobile and fixed IoT sensing network. IEEE Access, 8, 89584-89594.

[10] Zheng, Y., Yi, X., Li, M., Li, R., Shan, Z., Chang, E., & Li, T. (2015, August). Forecasting fine-grained air quality based on big data. In Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining (pp. 2267-2276).

[11] Hsu, Y. C., Dille, P., Cross, J., Dias, B., Sargent, R., & Nourbakhsh, I. (2017, May). Community-empowered air quality monitoring system. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (pp. 1607-1619).

[12] Shaban, K. B., Kadri, A., & Rezk, E. (2016). Urban air pollution monitoring system with forecasting models. IEEE Sensors Journal, 16(8), 2598-2606

[13] Application of the XGBoost Machine Learning Method in PM2.5 Prediction: A Case Study of Shanghai Jinghui Ma1,2,3, Zhongqi Yu2,3*, Yuanhao Qu2,3, JianmingXu2,3,4, Yu Cao2,3