Journal of Soft Computing Paradigm (ISSN: 2582-2640)
www.irojournals.com/jscp/

Smart Environment: AI-Driven Predictions and Forecasting of Air Quality

S R Mugunthan

Associate Professor, Department of Computer Science and Engineering, Sri Indu College of Engineering and Technology, Hyderabad, India

E-mail: srmugunth@gmail.com

Abstract

Addressing the critical issue of air quality in the Coimbatore region, this study introduces a novel approach for continuous monitoring and forecasting of air pollution. By utilizing the Internet of Things (IoT) technology integrated with Artificial Intelligence (AI) methods, this research focuses on monitoring and forecasting three major pollutants such as Ozone (O3), Ammonia (NH3), and Carbon Monoxide (CO). The proposed IoT-based sensor nodes collect the real-time data and give the resultant data as an input to the Naive Bayes (NB) for classification and Auto-Regression Integrating Moving Average (ARIMA) for optimization. The optimized model parameters are obtained and then validated by using performance metrics like Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). Deploying a machine learning algorithm on a Raspberry Pi-3, the proposed system ensures efficient monitoring and forecasting of air pollutants 24/7 through an online open-source dashboard.

Keywords: Internet of Things (IoT), Artificial Intelligence (AI), Naive Bayes (NB), Auto-Regression Integrating Moving Average (ARIMA), Raspberry Pi-3, Sensor Nodes, Air Pollutants.

1. Introduction

Air pollution, a ubiquitous environmental challenge, poses significant threats to human health, emphasizing the critical need for air quality monitoring and assessment. The adverse effects of air pollution on human health are diverse, ranging from respiratory issues and

DOI: https://doi.org/10.36548/jscp.2023.4.005
Received: 06.12.2023, received in revised form: 29.12.2023, accepted: 14.01.2024, published: 22.01.2024

cardiovascular diseases to more severe conditions like lung cancer. In today's society, where urbanization and industrialization are on the rise, understanding and addressing air quality concerns have become vital [1]. The Air Quality Index (AQI) serves as a potential evaluation metric in this regard, providing a quantitative measure to communicate the quality of air and associated health risks to the public. AQI value plays a crucial role in raising awareness, guiding regulatory measures, and empowering individuals to make informed decisions about their activities and well-being in the face of increasing air pollution challenges [2]. As societies strive for sustainable development, the emphasis on air quality monitoring and the AQI becomes increasingly crucial for safeguarding public health and fostering a healthier and more robust global community.

The integration of various state-of-the-art technologies and Machine Learning (ML) models has revolutionized the process of air quality monitoring, particularly in terms of the analysis of particulate matter such as PM2.5 and PM10 [3]. The Unmanned Aerial Vehicles (UAVs) equipped with advanced sensors have recently emerged as a potential solution for collecting real-time, spatially distributed air quality data by gaining a comprehensive understanding of various pollution dynamics [4]. On the other hand, the machine learning models, including time series techniques like Naive Bayes (NB), Random Forest (RF) and Auto regressor, have been deployed for forecasting pollutant concentrations and assessing temporal patterns. These models utilize historical data to predict future air quality, aiding in proactive decision-making and pollution control measures. Additionally, the calculation of the Air Quality Index (AQI), a crucial metric for quantifying overall air quality, integrates diverse pollutant data, providing a comprehensive overview of the potential health risks to the public. This multidimensional approach, integrating UAV technology, machine learning models, and time series analysis represents an effective framework for advancing air quality monitoring and facilitating more informed environmental management strategies [5].

In the field of air quality monitoring, the synergy of Internet of Things (IoT), Wireless Sensor Networks (WSN), NodeMCU, Thingspeak, and Google Colab has enabled a paradigm shift in the process of data collection, analysis, and distribution. IoT technologies, enabled by devices like NodeMCU, establish a seamless connection of air quality sensors, allowing for the real-time transmission of data to centralized platforms. WSN further enhances the efficiency by providing a network of interconnected sensors, ensuring comprehensive spatial coverage.

Platforms such as Thingspeak serve as robust data repositories, aggregating information from various sensor nodes. The integration of Google Colab, a cloud-based collaboration tool, empowers to analyze and visualize extensive datasets, offering valuable insights into air quality trends and patterns. These technologies not only facilitate prompt detection of pollutants but also enhances the accessibility and usability of air quality data, fostering a more informed and responsive approach to environmental monitoring and management.

The main focus of this study lies in the development and implementation of an Internet-of-Things (IoT) framework for Air Pollution Monitoring and Forecasting with minimal manual intervention. This research study outlines the process of data collection, sensor data validation, pre-processing, and the implementation of a machine learning model with subsequent validation using performance metrics. The optimized model is then deployed in an Edge device, specifically a Raspberry Pi 3, and a User Interface (UI) is designed using Firebase on the Google Cloud Server. The result is a remote dashboard that facilitates real-time monitoring of both live and forecasted air pollutants, enhancing the overall efficiency of the air quality monitoring system while minimizing the environmental impact and operational costs.

The research flow of the article is as follows: Section 2 presents a detailed literature review, section 3 includes the proposed block diagram and system specifications, section 4 presents the implementation of Naïve Bayes and ARIMA models for optimization, section 5 presents the simulated results and illustrates the real-time analysis and online dashboard and finally section 6 concludes the proposed research work.

2. Literature Review

In the existing research literature [6], various Artificial Intelligence (AI) and Machine Learning (ML) are proposed to predict the air quality. Recently, the time-series based computational models are used to predict the Air Quality Index (AQI) value. The recent research works on AQI based air monitoring includes, the implementation of mobile air monitoring system; neural network based AQI prediction and monitoring [7]; 360-degree long-range environment and AQI monitoring using Unmanned Aerial Vehicles (UAVs) [8]. Recently, the wireless sensor networks are used to enable the real-time monitoring of particulate matters like PM2.5 and PM10. In urban cities, these are easily monitored by using

UAVs. As a recent technological advancement, real-time air quality monitoring using Gaussian models has enabled a fine-tuned power efficient air quality detection method. This methodology remains as a major benefit to smart cities where both ground and aerial sensing data plays a major role. The utilization of Autoregressor, ARIMA, and neural network models plays a vital role in forecasting and predicting the future values [9].

Monitoring the air pollution by using Internet of Things (IoT), temperature, and gas detection sensors have increasingly automated the process of detecting harmful gases in the air. Recently, Raspberry Pi boards and IoT has automated the weather forecasting and monitoring process by reducing the power consumption and increasing the overall system efficiency [10]. In India, Ammonia (NH3), Ozone gas (O3), and Carbon Monoxide (CO) contributes a major part to the air pollution. Hence, there is an increasing need to monitor these gases by setting up a ground station [11]. On the other hand, these can also be monitored from a satellite station but the differential error occurs around 15-25%. However, the major challenge is that the data cannot be obtained in continuous manner and as a result the computed results will not be sufficient for monitoring the air quality. Moreover, setting up and maintaining a proper ground station is also a tedious task [12].

By considering all the challenges stated above, this study has moved forward to proposed a low-cost IoT and Machine Learning based Air Pollution Monitoring System. Here, the efficient remote monitoring option is enabled by utilizing the Raspberry Pi-3 board. The research gap is addressed by incorporating Machine Learning (ML) algorithm for optimization purpose.

3. Methodology

The proposed hardware model includes different components to collect different air pollutants such as ammonia gas, ozone gas and carbon monoxide by using sensors such as GG-NH3, MQ-131 and TGS5342 respectively.

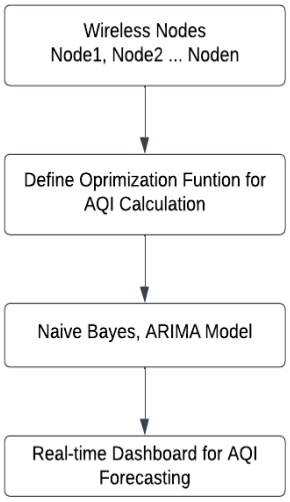

Figure 1. Proposed Flowchart

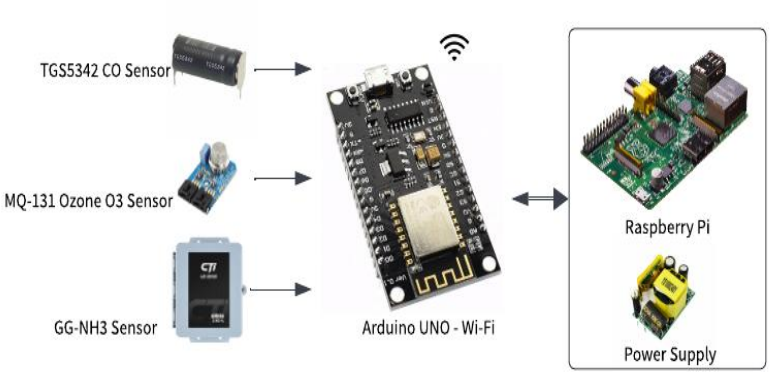

Figure 2. Proposed Hardware Prototype Model

Followed by the sensor calibration process, the sensor data are linked with Arduino UNO-Wi-Fi to transmit the data wirelessly to Raspberry Pi 3. Functioning as a local server and edge device, the Raspberry Pi 3 stores the transmitted data.

Here, the Raspberry Pi 3 embeds all data pre-processing and machine learning algorithm execution process using Python code. Subsequently, an online dashboard has been developed with the remote monitoring of both live and forecasted air pollutant data. The proposed model enhances the efficiency of data management and accessibility while utilizing the capabilities of Arduino, Raspberry Pi 3, and IoT infrastructure.

4. Machine Learning based Air Quality Forecasting

Here, the machine learning algorithms such as Naïve Bayes (NB) and Auto Regression Integrating Moving Average Model (ARIMA) are integrated here to forecast air quality and the performance metrics of the proposed model are also discussed in this section.

In ARIMA modeling, the crucial step involves transforming the data to achieve stationary conditions, a prerequisite for effective model application. Stationary condition plays a crucial role in time series analysis and is defined by stable mean, standard deviation, and auto-correlation structures. To attain the stationary conditions, the power transformations are applied, effectively eliminating any underlying trends in the data. Once the ARIMA model is identified, model parameters are thoroughly assessed, and the selected model is then employed for accurate forecasting. This methodology ensures that the time series data conforms to the necessary stationary conditions, enhancing the reliability and predictive capabilities of the ARIMA model as shown in Figure 3.

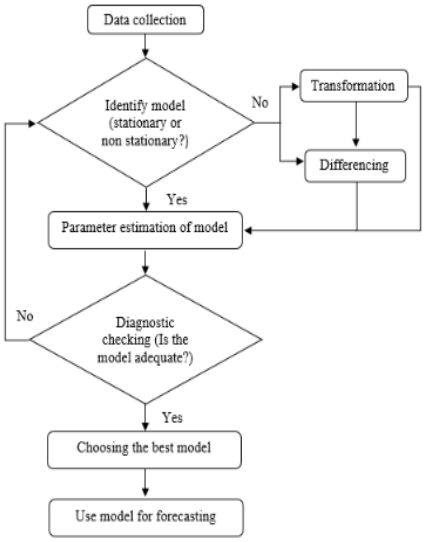

Figure 3. Working Process of ARIMA Model

Naive Bayes is typically used for classification tasks rather than optimization. However, if you're interested in a probabilistic approach to estimating the likelihood of different AQI levels given certain features, you can adapt Naive Bayes for this purpose. In this example, we'll create a simplified function that estimates the probability of different AQI levels based on the concentrations of NH3, O3, and CO. The developed Python program as shown in Figure 4

defines a function named *"naive_bayes_aqi_estimation"* that uses a Naive Bayes approach to estimate the Air Quality Index (AQI) level based on concentrations of different air pollutants: NH3 (Ammonia), O3 (Ozone), and CO (Carbon Monoxide). The function takes these concentrations along with an array of prior probabilities for different AQI levels. It calculates the likelihood of the given concentrations for each AQI level using a Gaussian probability density function and computes the unnormalized posterior probabilities. These posterior probabilities are then normalized, and the function returns the estimated AQI level with the maximum probability. The program includes an example of prior probabilities for different AQI levels, sets concentrations of NH3, O3, and CO, and prints the estimated AQI level using the defined function.

```
def naive_bayes_aqi_estimation(pm25, pm10, o3, co, prior_probs):

    Naive Bayes-based estimation of AQI levels.

    Parameters:
    nh3 (float): Concentration of nh3.
    o3 (float): Concentration of O3 (Ozone).
    co (float): Concentration of CO (Carbon Monoxide).
    prior_probs (dict): Prior probabilities for different AQI levels.

    Returns:
    str: Estimated AQI level.

    features = np.array([nh3, o3, co])

    # P(AQI Level | Features) = P(Features | AQI Level) * P(AQI Level) / P(Features)

    P(Features | AQI Level)
    likelihoods = {}
    for aqi_level in prior_probs:
        likelihood = np.prod(np.exp(-(features - prior_probs[aqi_level]['mean'])**2 /
                                    (2 * prior_probs[aqi_level]['variance']**2)) /
                             (np.sqrt(2 * np.pi) * prior_probs[aqi_level]['variance']))
        likelihoods[aqi_level] = likelihood

    P(AQI Level | Features) * P(AQI Level)
    unnormalized_posteriors = {aqi_level: likelihoods[aqi_level] * prior_probs[aqi_level]['prior']
                               for aqi_level in prior_probs}

    normalized_posteriors = {aqi_level: unnormalized_posteriors[aqi_level] / sum(unnormalized_posteriors.values())
                             for aqi_level in prior_probs}

    estimated_aqi_level = max(normalized_posteriors, key=normalized_posteriors.get)

    return estimated_aqi_level

prior_probabilities = {
    'Good': {'prior': 0.2, 'mean': 30, 'variance': 5},
    'Moderate': {'prior': 0.4, 'mean': 60, 'variance': 10},
    'Unhealthy': {'prior': 0.3, 'mean': 100, 'variance': 15},
    'Hazardous': {'prior': 0.1, 'mean': 150, 'variance': 20}
}

nh3_concentration = 40
o3_concentration = 50
co_concentration = 20

estimated_aqi = naive_bayes_aqi_estimation(nh3_concentration, o3_concentration, co_concentration, prior_probabilities)

print(f"Estimated AQI Level: {estimated_aqi}")
```

Figure 4. Python Programming Code for the Proposed Naïve Bayes Model

5. Results And Discussion

Case Study on Coimbatore Region

Coimbatore, located in the western part of Tamil Nadu, India, is positioned amidst the picturesque Western Ghats and the serene Noyyal River basin. Renowned as the "Manchester of South India," Coimbatore is characterized by its undulating terrain and fertile plains. The city has earned acclaim for its thriving textile industry, hosting iconic mills like Lakshmi Mills and PSG Industrial Institute, contributing significantly to India's textile heritage. Beyond textiles, Coimbatore boasts a diverse industrial landscape encompassing engineering, automobile, and software sectors. However, this industrial prosperity has brought forth environmental challenges, with the city experiencing air contamination due to emissions from factories and vehicular traffic. The need for sustainable development and air quality management becomes imperative as Coimbatore navigates its dual role as an industrial hub and an eco-friendly city.

In the process of real-time data logging, the Arduino program facilitates the collection of analogue data from sensors at one-hour intervals, with the Raspberry Pi storing this data in the Firebase. Live data collection is performed without any external references. Subsequently, the collected data is employed for forecasting future values using Naïve Bayes Python code and ARIMA Time Series model. The forecasted values are stored in the Firebase database. Online Monitoring of Air Pollutants is achieved by displaying real-time and forecasted pollutant values for next 24 hours on a dedicated website as shown in Figures 5, 6, 7.

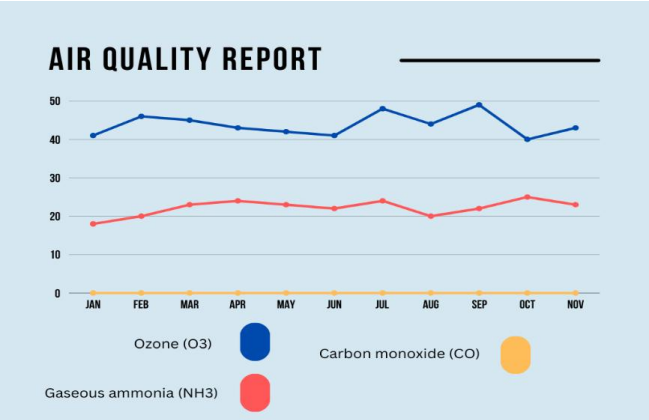

Figure 5. Cumulative Month-wise Air Quality Report

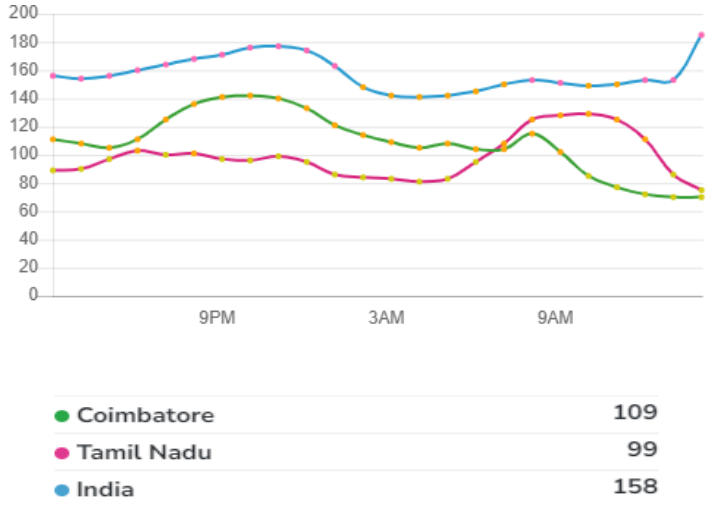

Figure 6. Illustration of 24-Hours AQI Prediction

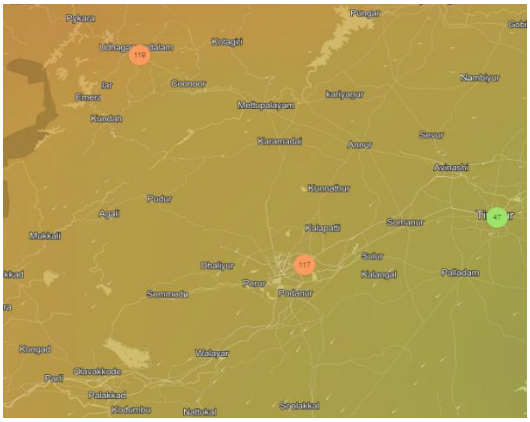

Figure 7. Location-wise AQI Indication

Performance Metrics

The performance evaluation relies on statistical measures, specifically the Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE).

RMSE (Root Mean Squared Error)

RMSE serves as the square root of the mean of the squared errors. This metric provides insight into how closely the predicted values align with the actual values. A lower RMSE value signifies superior model performance. The calculation of RMSE follows equation 1:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}} \qquad \text{----(1)}$$

where n represents the number of observations, y_i denotes the actual values, and signifies the predicted values.

MAE (Mean Absolute Error)

MAE, the mean or average of the absolute values of errors (Predicted – Actual), is a measure of the average magnitude of errors. Equation 2 outlines the calculation of MAE:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Here, n again denotes the number of observations, y_i represents the actual values, and $\hat{y}_i$ indicates the predicted values. These metrics offer a comprehensive assessment of model accuracy, with lower values indicating better performance.

Tables 1, 2 and 3 shows the performance indicators of NH3, O3, and CO data.

Table 1. Performance Comparison of NH3 Values

Model	**RMSE Value**	**MAE Value**
Naïve Bayes	2.9423	2.8725
ARIMA	1.2532	0.963

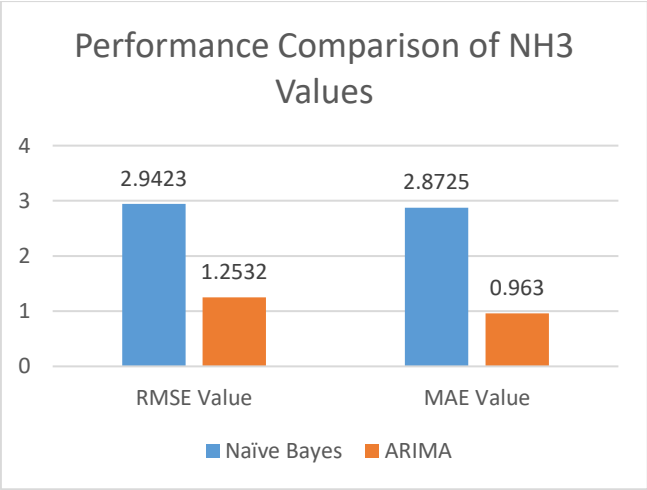

Figure 8. Comparative Illustration of NH3 Values

Table 2. Performance Comparison of O3 Values

Model	**RMSE Value**	**MAE Value**
Naïve Bayes	1.6402	1.2832
ARIMA	1.2631	1.1025

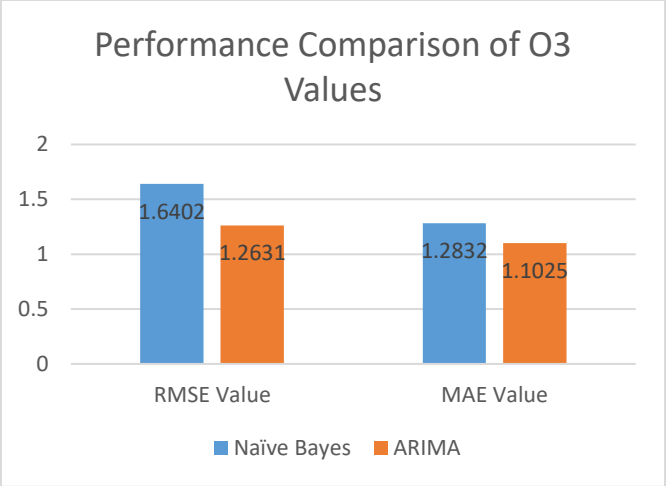

Figure 9. Comparative Illustration of O3 Values

Table 3. Performance Comparison of CO Values

Model	**RMSE Value**	**MAE Value**
Naïve Bayes	0.2672	0.3252
ARIMA	0.3203	0.2864

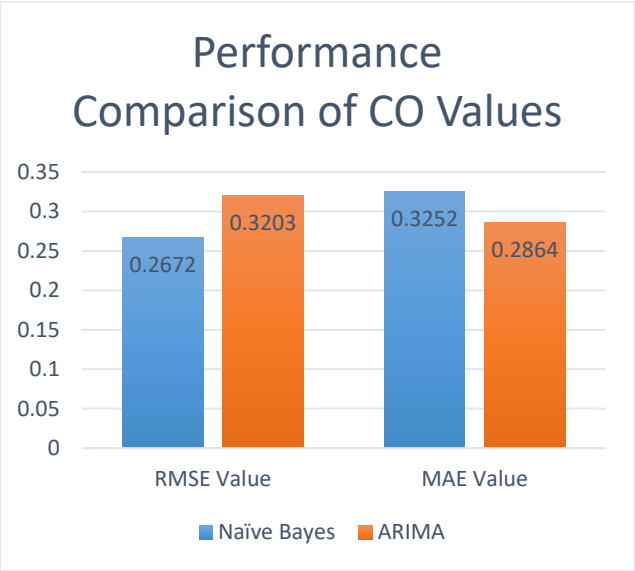

Figure 10. Comparative Illustration of CO Values

6. Conclusion

In conclusion, this study focused on continuous monitoring and forecasting of key pollutants, Ozone (O3), Ammonia (NH3), and Carbon Monoxide (CO). The Naive Bayes (NB) has successfully classified the air quality as “Good”, “Moderate” “Unhealthy” and “Hazardous” and Auto-Regression Integrating Moving Average (ARIMA) time-series model is used to optimize the data. The optimized model parameters are then validated using the performance metrics such as Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). Implemented on a Raspberry Pi-3, the proposed system ensured an efficient 24/7 monitoring and forecasting of air pollutants. The deployment of a machine learning algorithm and the availability of an online open-source dashboard enhanced the accessibility and contribute to a comprehensive solution for addressing air quality challenges in the region.

References

[1] Goossens, Janne, Anne-Charlotte Jonckheere, Lieven J. Dupont, and Dominique MA Bullens. "Air pollution and the airways: lessons from a century of human urbanization." *Atmosphere* 12, no. 7 (2021): 898.

[2] Kaginalkar, Akshara, Shamita Kumar, Prashant Gargava, Neelesh Kharkar, and Dev Niyogi. "SmartAirQ: A big data governance framework for urban air quality management in smart cities." *Frontiers in Environmental Science* 10 (2022): 785129.

[3] Pandya, Aum, Rudraksh Nanavaty, Kishan Pipariya, and Manan Shah. "A Comparative and Systematic Study of Machine Learning (ML) Approaches for Particulate Matter (PM) Prediction." *Archives of Computational Methods in Engineering* (2023): 1-20.

[4] Motlagh, Naser Hossein, Pranvera Kortoçi, Xiang Su, Lauri Lovén, Hans Kristian Hoel, Sindre Bjerkestrand Haugsvær, Varun Srivastava, Casper Fabian Gulbrandsen, Petteri Nurmi, and Sasu Tarkoma. "Unmanned aerial vehicles for air pollution monitoring: A survey." *IEEE Internet of Things Journal* (2023).

[5] Singh, Dharmendra, Meenakshi Dahiya, Rahul Kumar, and Chintan Nanda. "Sensors and systems for air quality assessment monitoring and management: A review." *Journal of environmental management* 289 (2021): 112510.

[6] Liu, Hui, Guangxi Yan, Zhu Duan, and Chao Chen. "Intelligent modeling strategies for forecasting air quality time series: A review." *Applied Soft Computing* 102 (2021): 106957.

[7] Ahmad, Shadab, and Tarique Ahmad. "AQI prediction using layer recurrent neural network model: a new approach." *Environmental Monitoring and Assessment* 195, no. 10 (2023): 1180.

[8] Sharma, Rohit, and Rajeev Arya. "UAV based long range environment monitoring system with Industry 5.0 perspectives for smart city infrastructure." *Computers & Industrial Engineering* 168 (2022): 108066.

[9] Elsaraiti, Meftah, and Adel Merabet. "A comparative analysis of the arima and lstm predictive models and their effectiveness for predicting wind speed." *Energies* 14, no. 20 (2021): 6782.

[10] Mathur, Vivek, Yashika Saini, Vipul Giri, Vikas Choudhary, Uday Bharadwaj, and Vishal Kumar. "Weather station using raspberry pi." In *2021 Sixth International Conference on Image Information Processing (ICIIP)*, vol. 6, pp. 279-283. IEEE, 2021.

[11] Kabiraj, Sabyasachi, and Nitin Vyankat Gavli. "Impact of SARS-CoV-2 pandemic lockdown on air quality using satellite imagery with ground station monitoring data in most polluted city Kolkata, India." *Aerosol Science and Engineering* 4 (2020): 320-330.

[12] Narayana, Mannam Veera, Devendra Jalihal, and SM Shiva Nagendra. "Establishing a sustainable low-cost air quality monitoring setup: A survey of the state-of-the-art." *Sensors* 22, no. 1 (2022): 394.