ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25jul758

AQIP: Air Quality Index Prediction Using Supervised ML Classifiers

Nayan Adhikari[1]; Pallabi Ghosh[2]; Abhinaba Bhattacharyya[3]; Siddhartha Chatterjee[4*]

[1]Department of Computer Science and Engineering, College of Engineering and Management, Kolaghat, KTPP Township, Purba Medinipur – 721171, West Bengal, India.
[2]Department of Electronics and Communication Engineering, College of Engineering and Management, Kolaghat, KTPP Township, Purba Medinipur – 721171, West Bengal, India.
[3]Department of Computer Science and Engineering, College of Engineering and Management, Kolaghat, KTPP Township, Purba Medinipur - 721171, West Bengal, India.
[4]Department of Computer Science and Engineering, College of Engineering and Management, Kolaghat, KTPP Township, Purba Medinipur - 721171, West Bengal, India.

Corresponding Author: Siddhartha Chatterjee[4*]

Publication Date: 2025/07/16

Abstract: In current years, Air pollution has emerged as a significant environmental concern. Accuracy modeling the complex relationships between air quality variables using advanced machine learning techniques is a promising area of research. The study aims to evaluate and compare the performance of supervised machine learning methods including Support Vector Regressor (SVR), Random Forest (RF), XGBoost, LightGBM for the prediction of air quality index. For the research, we collect a dataset from Kaggle. To assess the model performance, metrices such as root-mean-square-error (RMSE), Mean Absolute Error (MAE) and coefficient of determination (R^2) were used. Experimental result showed how LightGBM model outperformed the others in AQI prediction (RMSE = 1.4704, R2 = 0.9987 and MAE = 0.1824). Furthermore, all models were evaluated using these metrices, offering a clear comparison that highlighted the factors contributing to the improved accuracy.

Keywords: *Air Quality; Air Pollutant; Support Vector Regressor; Random Forest Regressor; XGBoost; LightGBM; Root-Mean-Squared-Error; Mean Absolute Error; Coefficient of Determination; Supervised Methos.*

How to Cite: Nayan Adhikari; Pallabi Ghosh; Abhinaba Bhattacharyya; Siddhartha Chatterjee (2025) AQIP: Air Quality Index Prediction Using Supervised ML Classifiers. *International Journal of Innovative Science and Research Technology*, 10(7), 835-842. https://doi.org/10.38124/ijisrt/25jul758

I. INTRODUCTION

Air pollution took huge amount of lives every year, report from WHO. It has also contributed to environmental issues such as acid rain, global warming, aerosol build-up and the formation of photo-chemical smog.

In earlier times, fossils fuels such as coal and petroleum were predominantly relied upon as the primary sources of energy. These fossil fuels were consumed extensively for various purposes without much regulation. However, the unchecked use of these fuels led to significant air pollution, posing serious health hazards to humans. The combustion fossil fuels release harmful gases like nitrogen oxides, carbon dioxides, sulfur dioxide and others, which have been increasing steadily. This contributes to acid rain and intensifies the greenhouse effect in impacted areas. In villages, the burning of materials like cow dung and dry leaves as fuel also deteriorates air quality. Additionally, burning waste in the name of cleanliness further escalates air pollution levels. Urban vehicles are another major contributor to this issue. Consequently, the full adoption of electric vehicles in India has yet to be realized. The excessive pollution and declining air quality have affected people across the country in various ways. For instance, in December 2017, Delhi was temporarily shut down due to severe air pollution. The worsening air quality makes managing pollution more challenging. The AQI is a standard used to determine how clean or polluted the air is. This index evaluates air pollution levels based on various pollutants. According to the United States Environmental Protection Agency, the AQI is divided into six categories, ranging from 'good' to 'hazardous'. The method for calculating the AQI score involves a specific mathematical formula.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25jul758

$$AQI = \frac{(I_{high} - I_{low})}{(C_{high} - C_{low})}(C - I_{low}) + I_{low} \quad (1)$$

Where:

- C is pollutant concentration.
- C_{low}, the value less than normal pollutant concentration.
- C_{high}, the value above the pollutant concentration.
- I_{low}, index breakpoint respect to C_{low}.
- I_{high}, index breakpoint respect to C_{high}.

The Environmental Protection Agency (EPA) monitors well-known criteria pollutants, including carbon monoxide (CO), particulates matter (PM10 and PM2.5) and ground-level ozone (O3). The Air Quality Index (AQI) is calculated based on the concentrations of these pollutants and serves as an indicator of how clean or polluted the air is at a given time. A rising AQI value generally signals worsening air quality, which can pose health concerns.

As given in Fig. 1, An AQI score between 0 and 50 falls under level one, indicating good air quality with minimal pollution. Level Two covers scores from 51 to 100, where the air quality is satisfactory. For level three, the AQI range is 101 to 200, which signifies moderate pollution levels. Level four corresponds to scores between 201 and 300, reflecting poor air quality. When the AQI ranges from 301 to 400, it is classified as very poor. Finally, an AQI score between 401 and 500 denotes a severe level of air pollution in the area [1,2].

Table 1 AQI Pollution for Different Categories

AQI Category, Pollutants and Health Breakpoints				
AQI Category (Range)	PM_{10} 24-hr	$PM_{2.5}$ 24-hr	NO_2 24-hr	O_3 8-hr
Good (0-50)	0-50	0-30	0-40	0-50
Satisfactory (51-100)	51-100	31-60	41-80	51-100
Moderately polluted (101-200)	10-250	61-90	81-180	101-168
Poor (201-300)	251-350	91-120	181-280	169-208
Very poor (301 – 400)	351-430	121-250	281-400	209-748*
Severe (401 – 500)	430+	250+	400+	748+*

Artificial Intelligence-based AQI prediction models are generally divided into two categories: regression models and time-series models. [3,24].

This study aims to comparison of supervised Machine learning classifier models for pred, using some most powerful existing machine learning (ML) approaches, Light Gradient-Boosting Machine (LightGBM), Random Forest, Support vector Regressor (SVR) and Extreme Gradient Boosting (XGBoost).

II. LITERATURE REVIEW

In 2011, Anikender Kumar and Pramila Goyal Conducted a study that predicted daily AQI levels in Delhi, India, Utilizing historical AQI data and weather parameters through principal component regression and multiple linear regression analysis. They forecasted daily AQI levels for 2006 using historical data from 2000 to 2005 and various statistical models. They then compared the predicted AQI values for 2006 with the actual observed values for different seasons, namely summer, monsoon, post-monsoon, and winter, based on the multiple linear regression model. Principle component analysis is utilized to identify multicollinearity among independent variables. By using principal components in multiple linear regression, we addressed collinearity issues among the predictors and reduced the number of variables in the model. The results showed that principal component regression performed better in predicting AQI during winter than in other seasons. However, the study's predictive model was limited to meteorological factors and did not consider the potential health effects of ambient air pollutants.

Huixiang Liu (et al.2019) selected two cities, Beijing and an Italian city, for a comparative study. They used two distinct datasets to forecast the air quality index in Beijing and predict NOx concentrations in the Italian city. The Beijing dataset, spanning from December 2013 to August 2018, comprised 1738 instances and was sourced from the Beijing Municipal Environment Center. This dataset included hourly averages of AQI and pollutant concentrations, such as PM2.5, O3, SO2, PM10, and NO2. A separate dataset from an Italian city, covering March 2004 to February 2005, included 9358 hourly data points on CO, non-methane hydrocarbons, benzene, NOx, and NO2.With a focus on NOx prediction due to its importance in air quality assessment, the study applied SVR and RFR techniques to predict AQI and NOx levels. The results showed SVR excelled in AQI prediction, while RFR performed better for NOx. In related work, Yyang et al. designed a mobile AQI monitoring system using a neural network-based Gaussian plume model at 2018.

Nearest neighbor classification is a technique where an unclassified sample is assigned to the class of its nearest neighbor among a set of pre-classified points. Hastie and Tibshirani's Discriminant Adaptive Nearest neighbor (DANN) method adapts to local data structures by estimating a subspace for dimensionality reduction. This approach enables the use of customized distance measures for different classification problems, making it a versatile and effective method.

Y Yang et al. introduced ImageSensingNet, a vision-based aerial – ground sensing system that leverages UAV-captured images for AQI monitoring and forecasting, in 2019. In 2018, Y Yang et al. presented an aerial-ground Wireless Sensor network (WSN) for real time PM2.5 monitoring in urban areas using UAVs. Z Zheng et al. developed a 3D real-time AQI monitoring system in 2017, Utilizing an Adaptive Gaussian Plume Model (AGPM) with UAVs. Z hu et al. proposed a real-time, fine-grained, and power-efficient air quality sensing system for smart cities in 2019, integrating ground and aerial sensing data to enhance data accuracy. In 2011, a study compared the performance of model-based and artificial neural network approaches for forecasting future values using various datasets. Neural networks can be difficult to work with due to their intricate nature, and they often struggle to adapt to real-time data changes over short periods, as evident from the literature review. In response, leaving et al. introduced a novel approach called Timeless Competition, which enables efficient study of multi-point replenishment problems. Jie Deng et al. (2020) introduced a stochastic model that optimized ordering, holding, lost sales, and transportation costs for joint replenishment and distribution problems, analyzing the impact of stochastic factors on total cost. The same team presented a novel algorithm, Bare Bones Differential Evolutionary Algorithm, to mitigate uncertainty in joint replenishment problems. Additionally, they explored optimal ordering strategies for stakeholders utilizing RFID technology. Meanwhile, a study in Thailand by John Joseph (2019) proposed a unified approach combining IoT and data analytics to predict particulate matter pollution. The same author used support vector regression to predict future PM2.5 concentrations based on weather data. Another project implemented a weather monitoring system using IoT applications in environmental monitoring highlighted various subdomains and research challenges.

The XGBoost algorithm was employed in 2018 for forecasting hourly PM2.5 concentrations in Tianjin's air quality monitoring data. Its performance was evaluated using three major forecasting metrices, and the results showed strong predictive capabilities. When compared to other models like random forest, multiple linear regression, decision tree regression, and support vector machines, XGBoost demonstrated superior performance in predicting PM2.5 levels. [4-7,25-28].

III. METHODOLOGY

➢ *This Project can Work on any Operating System, Ranging from Windows to Ubuntu. its Requirements are:*

- Programming Language: Python
- Software: Jupiter Notebook or Google Collab (hosted version of Jupiter Notebook)

- *Data Preprocessing:*

Pre-processing transforms raw input into a structured format which is best for model training. It handles cleaning, missing values, normalization, outlier removal etc.

- *Feature Selection:*

The various pollutant indices PM10, PM2.5, CO and O_3 are used for measuring air quality index in India. [8,9].

➢ *The Algorithm below Shows the Basic Implementation of the Program:*

- *Algorithm:*

➢ *Algorithm Steps*

- *Dataset Preparation:*

✓ First, we collect data from open source.
✓ Clean and process the dataset using python libraries.
✓ Get relevant features and split the dataset into training (80%) and testing (20%).

- *Model Selection:*

✓ Now we train the dataset using different supervised machine learning models
✓ LightGBM
✓ XGBoost
✓ Random Forest
✓ SVR.

- *Hyper Tuning the Model:*

✓ Use optimization techniques such as grid search to fine-tune model parameters for better predictive performance.

- *Train the Model:*

✓ Train the model and test it on the test dataset.

- *Predict and Analyze;*

✓ Generate predictions for the test set and evaluate model performance.

In below we discussed about those models:

➢ *Support Vector Regressor (SVR):*

Support Vector Regression (SVR) is an extension of Support Vector Machines (SVM) adapted for regression tasks. Unlike classification SVM that aims to find the optimal hyperplane to separate classes, SVR seeks to find a function that best predicts continuous output values for given input data while maintaining a flat hyperplane with minimal complexity.

An approximate relationship between input and output variables is defined as function which operate SVR. It doesn't affect the margin of tolerance for error. The optimization process involves minimizing the norm of the coefficients while allowing for some errors outside the ε-insensitive zone. The structural risk minimization is a principal which aims to balance model complexity and training error. SRM principle underlies SVR. Overfitting is a common problem in traditional regression approaches that use empirical risk management. SRM helps prevent overfitting.

SVR is nonparametric technique. It uses kernel functions to map input data into higher dimensional feature

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25jul758

spaces, making it possible to model complex, nonlinear relationships. Linear Kernel, Polynomial Kernal, Radial Basis Function Kernel, Sigmoid Kernal are the common Kernels [10,11].

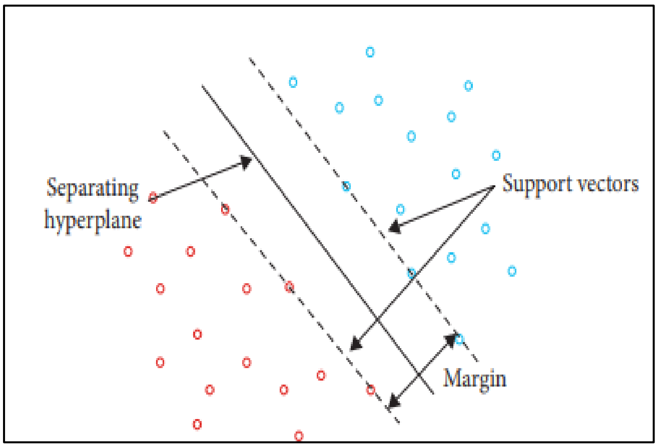

Fig 1 Linearly Separable Problem

The core principle of SVR is based on the ε-insensitive loss function, which creates a margin of tolerance where no penalty is imposed on prediction errors. The mathematical formulation begins with finding a linear function:

$$f(x) = w^T x + b \tag{2}$$

The optimization problem aims to minimize:

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\left(\xi_i + \xi_i^*\right) \tag{3}$$

Subject to the constraints:

- $Y_i - w^T x_i - b \leq \varepsilon + \xi_i$
- $w^T x_i + b - Y_i \leq \varepsilon + \xi_{i*}$
- $\xi_i, \xi_{i*} \geq 0$

Where:

- C is the regularization parameter controlling the trade-off between model complexity and tolerance for errors.
- ε (epsilon) defines the width of the insensitive tube where no penalty is applied.
- ξ_i *and* ξ_i* are slack variables allowing for constraint violations.

SVR demonstrates excellent robustness to outliers and noise due to its ε-insensitive loss function. The algorithm focuses only on support vectors (data points outside the ε-tube), making it less sensitive to the majority of training data points.

Unlike many regression methods, SVR produces sparse solutions where only support vectors contribute to the final model, leading to more interpretable and computationally efficient predictions.

- *Light Gradient-Boosting Machine (LightGBM):*

LightGBM is a gradient boosting framework developed by Microsoft. It mainly uses tree-based learning algorithms which is optimized for distributed and efficient training. With high accuracy and unbelievable speed, it shows a significant improvement in the area of gradient boosting algorithmic techniques. LightGBM selects training data by keeping all samples that have large gradient values. This method bundles mutually exclusive features to reduce the number of features without losing information. It can be 4-10 times faster than traditional gradient boosting implementations, especially on large databases.

Given a labeled dataset X, LightGBM aims to approximate an unknown target function f*(x) by minimizing the expected value of a loss function L(y, f(x)). The optimal solution is:

$$f = argmin_f E_{y,X} L(y, f(x)) \tag{4}$$

LightGBM constructs its predictive model as an ensemble of T regression trees. The cumulative prediction after T iterations is represented as:

$$f_T(X) = \sum_{t=1}^{T} f_T(X) \tag{5}$$

Each individual regression tree is defined by a structure q(x), which assigns each input to a specific leaf and a set of leaf weights w. the number of leaves in a tree is denoted by J.

At each boosting iteration t, LightGBM updates the models by adding a new tree $f_t(x)$ to minimize the loss over all training samples:

$$\Gamma_t = \sum_{i=1}^{n} L(y_i, F_{t-1}(x_i) + f_t(x_i)) \tag{6}$$

To efficiently optimize this objective, LightGBM applies a second-order Taylor expansion of the loss function around the current prediction, omitting constant terms:

$$\Gamma_t \cong \sum_{i=1}^{n} (g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)) \tag{7}$$

Where g_i and h_i are the first and second derivatives (gradient and Hessian) of the loss with respect to the model's prediction for sample i.

Grouping samples by their assigned leaf j, the objective becomes,

$$\Gamma_t = \sum_{j=1}^{J}((\sum_{i\in I_j} g_i)w_j + \frac{1}{2}(\sum_{i\in I_j} h_i + \lambda)w_j^2) \quad (8)$$

The optimal value for each leaf weight w^*_j is derived by minimizing the above expression:

$$w_j^* = \frac{\sum_{i\in I_j} g_i}{\sum_{i\in I_j} h_i + \lambda} \quad (9)$$

The corresponding minimized value of the objective, which serves as a measure of tree quality, is:

$$\Gamma_t^* = -\frac{1}{2}\sum_{j=1}^{J}\frac{(\sum_{i\in I_j} g_i)^2}{\sum_{i\in I_j} h_i + \lambda} \quad (10)$$

Γ^*_T can be thought of as a score function structure q that measures the quality of the regression tree.

Finally, the objective function can be expressed as:

$$G = \frac{1}{2}(\frac{(\sum_{i\in I_L} g_i)^2}{\sum_{i\in I_L} h_i + \lambda} + \frac{(\sum_{i\in I_R} g_i)^2}{\sum_{i\in I_R} h_i + \lambda} - \frac{(\sum_{i\in I} g_i)^2}{\sum_{i\in I} h_i + \lambda}) \quad (11)$$

Where I_L and I_R are the sample sets that fall into left and right branches, respectively.

Unlike traditional gradient boosting methods (such as XGBoost), which expand trees level by level (horizontal or depth-wise growth), LightGBM grows trees by repeatedly splitting the leaf with the highest potential loss reduction (vertical or leaf-wise growth). This approach allows LightGBM to achieve greater accuracy with fewer splits but can also increase the risk of overfitting if not properly regularized [12,21-23].

➢ *Random Forest*

Random Forest is a machine learning technique that construct an ensemble of decision trees and in widely applied to both classification and regression problems in this approach, each tree is trained using a randomly sampled portion of the dataset and at each node, a random selection of features is evaluated for splitting. For regression tasks, the final output is computed by averaging the prediction from all trees, whereas for classification, the most frequent class predicted by the tress is chosen as the final result.

The algorithm employs Classification and Regression Trees (CARTs), with each tree built using randomly sampled data and feature subsets. Two key parameters govern the model's performance, and the number of features (NF) randomly chosen at each split. Adding more trees typically enhances the model's accuracy and stability but comes at the cost of higher computational demand. The number of features per split is often set using the rule: $NF = \sqrt{M}$, where M is the total number of input features.

Random Forest can be implemented to both classification and regression tasks, depending on whether the trees are built for classification or regression trees. The structure of regression model is shown in Figure 3. If the model consists T regression trees (learners), the final prediction is obtained by averaging the outputs from all these trees.

$$H(x) = \frac{1}{T}\sum_{i=1}^{T} h_i(x) \quad (12)$$

Where:

- T is the number of regression trees
- h_i (x) is the output of the i-th regression tree, h_i(x) on sample x.

Therefore, the prediction of the RF is the average of the predicted values of all the trees [13-15].

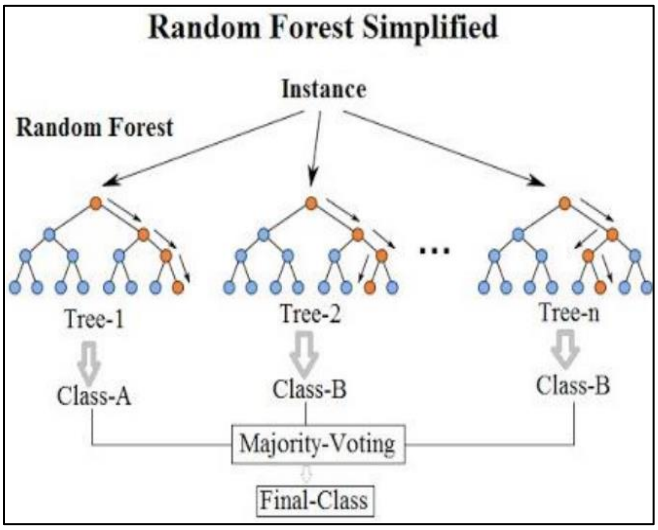

Fig 2 The Schematic Representation of Random Forest Regression (RFR) model [16].

➢ *Extreme Gradient-Boosting Algorithm (XGBoost):*

Boost is an effective technique for constructing supervised regression models. Chen and Guestrin took the XGBoost algorithm in front of everyone, which builds upon the Gradient Boosted Decision Trees (GBDT). It gained significantly popularity due to its consistently delivering results in various Kaggle machine learning challenges. Unlike traditional GBDT models, XGBoost includes a regularization term in its objective function, which help minimize the risk of overfitting. The main objective function is described as follows:

$$O = \sum_{i=1}^{n} L(y_i F(x_i)) + \sum_{k=1}^{t} R(f_k) + C \quad (13)$$

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25jul758

Where:

- $R(f_k)$ is the regularization term at iteration k
- C being a constant that can be removed selectively.

Regularization term $R(f_k)$ written as,

$$R(f_k) = \alpha H + \frac{1}{2}\eta\sum_{j=1}^{H} w_j^2 \tag{14}$$

Where:

- α is the complexity of leaves
- H denotes the number of leaves
- η signifies penalty variable
- ω_j represents output results in each leaf node

[17,29-34].

IV. RESULT

To find the accuracy of these models, performance evaluation metrics are used. In our paper we include MAE, R2-score, and RMSE. These metrics provide valuable information about different facets of a machine learning model's performance:

- MAE measures the average size of prediction errors, providing a straightforward indication of how accurate the model is. A smaller MAE value reflects better model performance.
- R^2 indicates how much of the variation in the dependent variable is accounted for by the model. Values approaching 1 suggest the model has strong explanatory capability.
- RMSE assesses the standard deviation of prediction errors, with lower values signifying a closer match between the predicted and actual outcomes.

The formula of RMSE and MAE are as follows:

$$RMSE = \left(\frac{1}{N}\sum_{I=1}^{N}(E_i - A_i)^2\right)^{\frac{1}{2}} \tag{15}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|E_i - A_i| \tag{16}$$

Where:

- n is instances count
- E_i is the estimated values.
- A_i is the actual value.

The lower value of these two metrics corresponds to a better prediction [18-20,35-36].

Table 2 Evaluation Metrices for all Research Models

Performance	RMSE	MAE	R^2 Score
LightGBM	1.4704	0.1824	0.9987
Random Forest	2.5527	1.2711	0.9960
Support Vector Regressor	0.5233	0.2790	0.7012
XGBoost	5.4680	0.5356	0.9819

As per Fig 4, we can see that LightGBM is the best model for predicting AQI. SVR has a lower RMSE value than LightGBM but the difference of r-squared is huge. That's why LightGBM is the most accurate model.

V. CONCLUSION

This research presents an effective model for predicting the Air Quality Index (AQI) using a publicly available dataset from Kaggle. A series of preprocessing steps— outlier detection, feature selection and handling of missing values—were applied to the input data for increasing its quality. Four prominent supervised machine learning models—LightGBM, Support Vector Regressor (SVR), Random Forest and XGBoost—were implemented and fine-tuned to develop accurate predictive models. Each model was examined by the standard regression performance metrics: RMSE, MAE, and R^2 score. Among the tested models, LightGBM predict with the best accuracy, achieving the best balance between low error and strong model fit. While SVR had a slightly lower RMSE, its significantly lower R^2 score indicated limited ability to capture the underlying data variance, making LightGBM the superior choice overall. This study confirms that ensemble methods, particularly gradient boosting approaches like LightGBM, are highly effective for AQI prediction tasks. The outcomes of this research can support quality of air and inform policy measures for environmental management.

REFERENCES

[1]. Liang, Y.-C., Maimury, Y., Chen, A. H.-L., & Juarez, J. R. C. (2020). Machine Learning-Based Prediction of Air Quality. Applied Sciences, 10(24), 9151.
[2]. Natarajan, S. K., Shanmurthy, P., Arockiam, D., Balusamy, B., & Selvarajan, S. (2024). Optimized machine learning model for air quality index prediction in major cities in India. Scientific Reports, 14(1), 6795.
[3]. Guo, Z., Jing, X., Ling, Y., Yang, Y., Jing, N., Yuan, R., & Liu, Y. (2024). Optimized air quality management based on air quality index prediction and air pollutants identification in representative cities in China. Scientific Reports, 14(1), 17923.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25jul758

[4]. Patil, R. M., Dinde, H. T., Powar, S. K., & Ganeshkhind, P. M. (2020). A literature review on prediction of air quality index and forecasting ambient air pollutants using machine learning algorithms. International Journal of Innovative Science and Research Technology, 5(8), 1148-1152.

[5]. Dragomir, E. G. (2010). Air quality index prediction using K-nearest neighbor technique. Bulletin of PG University of Ploiesti, Series Mathematics, Informatics, Physics, LXII, 1(2010), 103-108.

[6]. Mani, G., & Viswanadhapalli, J. K. (2022). Prediction and forecasting of air quality index in Chennai using regression and ARIMA time series models. Journal of Engineering Research, 10(2A), 179-194.

[7]. Umoh, M. D., Evans, U. F., & Utting, C. (2024). Air Quality Index Prediction Using Machine Learning Algorithms for Certain Locations in Nigeria. Journal of Environmental Science and Management, 25(2), 98-112.

[8]. Pant, A., Sharma, S., & Pant, K. (2023). Evaluation of machine learning algorithms for air quality index (AQI) Prediction. Journal of Reliability and Statistical Studies, 229-242.

[9]. Liang, Y.-C., Maimury, Y., Chen, A. H.-L., & Juarez, J. R. C. (2020). Machine Learning-Based Prediction of Air Quality. Applied Sciences, 10(24), 9151.

[10]. Castelli, M., Clemente, F. M., Popovič, A., Silva, S., & Vanneschi, L. (2020). A machine learning approach to predict air quality in California. Complexity, 2020(1), 8049504.

[11]. Ravindiran, G., Hayder, G., Kanagarathinam, K., Alagumalai, A., & Sonne, C. (2023). Air quality prediction by machine learning models: A predictive study on the indian coastal city of Visakhapatnam. Chemosphere, 338, 139518.

[12]. Gupta, N. S., Mohta, Y., Heda, K., Armaan, R., Valarmathi, B., & Arulkumaran, G. (2023). Prediction of air quality index using machine learning techniques: a comparative analysis. Journal of Environmental and Public Health, 2023(1), 4916267.

[13]. Avvari, P., Nacham, P., Sasanapuri, S., Mankena, S. R., Kudipudi, P., & Madapati, A. (2023). Air Quality Index Prediction. In E3S Web of Conferences (Vol. 391, p. 01103). EDP Sciences.

[14]. Liu, H., Li, Q., Yu, D., & Gu, Y. (2019). Air quality index and air pollutant concentration prediction based on machine learning algorithms. Applied sciences, 9(19), 4069.

[15]. Zhou, Y., Wang, W., Wang, K., & Song, J. (2022). Application of LightGBM algorithm in the initial Design of a Library in the cold area of China based on comprehensive performance. Buildings, 12(9), 1309.

[16]. Kang, G. K., Gao, J. Z., Chiao, S., Lu, S., & Xie, G. (2018). Air quality prediction: big data and machine learning approaches. Int. J. Environ. Sci. Dev, 9(1), 8-16.

[17]. Sharma, R., Shilimkar, G., & Pisal, S. (2021). Air quality prediction by machine learning. Int. J. Sci. Res. Sci. Technol, 8, 486-492.

[18]. Amjad, M., Ahmad, I., Ahmad, M., Wróblewski, P., Kamiński, P., & Amjad, U. (2022). Prediction of pile bearing capacity using XGBoost algorithm: modeling and performance evaluation. Applied Sciences, 12(4), 2126.

[19]. Singh, M. P., Bisht, N., Choudhary, M., Goswami, A., & Tagore, N. K. (2025). A Web-Based Supervised Machine Learning Model for Air Quality Index and Respiratory Care Prediction. Procedia Computer Science, 258, 1747-1756.

[20]. Iskandaryan, D., Ramos, F., & Trilles, S. (2020). Air quality prediction in smart cities using machine learning technologies based on sensor data: a review. Applied Sciences, 10(7), 2401.

[21]. Ghosh, P., Hazra, S., & Chatterjee, S. Future Prospects Analysis in Healthcare Management Using Machine Learning Algorithms. *the International Journal of Engineering and Science Invention (IJESI), ISSN (online)*, 2319-6734.

[22]. Hazra, S., Mahapatra, S., Chatterjee, S., & Pal, D. (2023). Automated Risk Prediction of Liver Disorders Using Machine Learning. In *the proceedings of 1st International conference on Latest Trends on Applied Science, Management, Humanities and Information Technology (SAICON-IC-LTASMHIT-2023) on 19th June* (pp. 301-306).

[23]. Gon, A., Hazra, S., Chatterjee, S., & Ghosh, A. K. (2023). Application of machine learning algorithms for automatic detection of risk in heart disease. In *Cognitive cardiac rehabilitation using IoT and AI tools* (pp. 166-188). IGI Global.

[24]. Das, S., Chatterjee, S., Sarkar, D., & Dutta, S. (2022). Comparison Based Analysis and Prediction for Earlier Detection of Breast Cancer Using Different Supervised ML Approach. In *Emerging Technologies in Data Mining and Information Security: Proceedings of IEMIS 2022, Volume 3* (pp. 255-267). Singapore: Springer Nature Singapore.

[25]. Das, S., Chatterjee, S., Karani, A. I., & Ghosh, A. K. (2023, November). Stress Detection While Doing Exam Using EEG with Machine Learning Techniques. In *International Conference on Innovations in Data Analytics* (pp. 177-187). Singapore: Springer Nature Singapore.

[26]. Hazra, S. (2024). Pervasive nature of AI in the health care industry: high-performance medicine.

[27]. Sima Das, Siddhartha Chatterjee, Sutapa Bhattacharya, Solanki Mitra, Arpan Adhikary and Nimai Chandra Giri "Movie's-Emotracker: Movie Induced Emotion Detection by using EEG and AI Tools", In the proceedings of the 4th International conference on Communication, Devices and Computing (ICCDC 2023), Springer-LNEE SCOPUS Indexed, DOI: 10.1007/978-981-99-2710-4_46, pp.583-595, vol. 1046 on 28th July, 2023.

[28]. Chatterjee, R., Chatterjee, S., Samanta, S., & Biswas, S. (2024, December). AI Approaches to Investigate EEG Signal Classification for Cognitive Performance Assessment. In *2024 6th International Conference on Computational Intelligence and Networks (CINE)* (pp. 1-7). IEEE.

[29]. Adhikary, A., Das, S., Mondal, R., & Chatterjee, S. (2024, February). Identification of Parkinson's

Disease Based on Machine Learning Classifiers. In *International Conference on Emerging Trends in Mathematical Sciences & Computing* (pp. 490-503). Cham: Springer Nature Switzerland.

[30]. Ghosh, P., Dutta, R., Agarwal, N., Chatterjee, S., & Mitra, S. (2023). Social media sentiment analysis on third booster dosage for COVID-19 vaccination: a holistic machine learning approach. *Intelligent Systems and Human Machine Collaboration: Select Proceedings of ICISHMC 2022*, 179-190.

[31]. Rupa Debnath; Rituparna Mondal; Arpita Chakraborty; Siddhartha Chatterjee (2025) Advances in Artificial Intelligence for Lung Cancer Detection and Diagnostic Accuracy: A Comprehensive Review. International Journal of Innovative Science and Research Technology, 10(5), 1579-1586. https://doi.org/10.38124/IJISRT/25may1339

[32]. Nitu Saha; Rituparna Mondal; Arunima Banerjee; Rupa Debnath; Siddhartha Chatterjee; (2025) Advanced Deep Lung Care Net: A Next Generation Framework for Lung Cancer Prediction. International Journal of Innovative Science and Research Technology, 10(6), 2312-2320. https://doi.org/10.38124/ijisrt/25jun1801

[33]. Poushali Das; Washim Akram; Arijita Ghosh; Suman Biswas; Siddhartha Chatterjee (2025) Enhancing Diagnostic Accuracy: Leveraging Continuous pH Surveillance for Immediate Health Evaluation. International Journal of Innovative Science and Research Technology, 10(7), 7-12. https://doi.org/10.38124/ijisrt/25jul123

[34]. Manali Sarkar; Aparajita Das; Sraddha Roy Choudhury; Siddhartha Chatterjee (2025). A* Based Optimized Travel Recommendation System for Smart Mobility. International Journal of Innovative Science and Research Technology, 10(5), 3185-3193. https://doi.org/10.38124/ijisrt/25may2352

[35]. Hazra, S., Chatterjee, S., Mandal, A., Sarkar, M., Mandal, B.K. (2023). An Analysis of Duckworth-Lewis-Stern Method in the Context of Interrupted Limited over Cricket Matches. In: Chaki, N., Roy, N.D., Debnath, P., Saeed, K. (eds) Proceedings of International Conference on Data Analytics and Insights, ICDAI 2023. ICDAI 2023. Lecture Notes in Networks and Systems, vol 727. Springer, Singapore. https://doi.org/10.1007/978-981-99-3878-0_46

[36]. Babli Kumari, Renu Dhir, Siddhartha Chatterjee, and Suchi Jain. 2025. Automated Identification of Traffic Accidents in Images and Videos Employing Advanced Deep Learning Methods. In 26th International Conference on Distributed Computing and Networking (ICDCN 2025), Janu ary 04–07, 2025, Hyderabad, India. ACM, New York, NY, USA,6 pages. https://doi.org/10.1145/3700838.370368

[37]. Madhuparna Das Hait; Priya Das; Washim Akram; Siddhartha Chatterjee (2025): A Comparative Analysis of Linear Regression Techniques: Evaluating Predictive Accuracy and Model Effectiveness. International Journal of Innovative Science and Research Technology, 10(7), 127-139. https://doi.org/10.38124/ijisrt/25jul34.