International Journal of Engineering and Information Systems (IJEAIS)
ISSN: 2643-640X
Vol. 9 Issue 10 October - 2025, Pages: 128-136

Development of an Air Quality Forecasting System Based on Random Forest and XGBoost Algorithms

1 Thanh Nguyen Cong 2 Truong Tran Quang

[1] International Leadership of Texas Garland Highschool, US Email: congthanh74598@gmail.com

[2] AI Engineer, Viet Nam Email: truongtq.tekverse@gmai.com

Abstract*: Air pollution is becoming a serious problem in large cities, in which PM2.5 fine dust is a factor that directly affects human health. This study introduces an integrated system that allows assessment, forecasting and warning of air quality based on real-time data collected from environmental sensors. The goal of the study is to support the monitoring and management of urban environments in a smarter and more effective way. Machine learning algorithms are used by the System to analyze and forecast PM2.5 concentrations. Specifically, Random Forest and XGBoost models are applied to the forecasting problem, while unsupervised learning methods such as PCA (Principal Component Analysis) and K-Means clustering are used to explore data structure and group pollution samples with similar characteristics. Experimental results on real data show that the Random Forest model achieves high forecasting performance with RMSE = 4.612, MAE = 2.417, and R^2 = 0.916, outperforming baseline models in both accuracy and stability. In terms of architecture, the system is built on Flask (backend) to process data and run machine learning models, along with React (frontend) to display forecast results and visualize multidimensional data. This design makes the system flexible, scalable, and deployable in many different environments. Overall, the study shows that the combination of machine learning and modern web technology is a potential approach for real-time air quality monitoring and management systems in the future.*

Keywords: Air quality, PM2.5, Machine Learning, Random Forest, Anomaly Detection.

1. Introduction

Air pollution is one of the most critical environmental challenges of the 21st century [1]. According to the World Health Organization (WHO), more than 90% of the global population lives in areas where the concentration of fine particulate matter (PM2.5) exceeds the recommended safety threshold, Each year, approximately seven million deaths are attributed to stroke, heart disease, lung cancer, chronic obstructive pulmonary disease (COPD), and respiratory infections [2][3]. In Vietnam, at least 70,000 deaths annually are linked to air pollution, and this trend has been increasing in recent years [4]. Major cities such as Hanoi, Da Nang, and Ho Chi Minh City frequently record Air Quality Index (AQI) levels classified as “unhealthy” on international scales [5]. The primary causes include emissions from urban traffic and personal vehicles, dust from construction activities, industrial and agricultural production, and pollution from domestic activities and waste management processes [6][7][8].

The impacts of air pollution extend beyond public health, affecting economic growth and sustainable development. [9] Numerous studies have shown that air pollution reduces labor productivity, increases healthcare costs, and results in billions of dollars in GDP losses each year [10]. Therefore, air quality monitoring, forecasting, and early warning are essential to protect human health, raise social awareness, and support environmental policy-making. The application of machine learning and real-time data analysis systems offers an effective approach to address this issue [11]. Recent research on air quality forecasting can be categorized into three main approaches. The first approach uses traditional statistical models such as ARIMA and linear regression [12][13]. These models are simple, easy to implement, and provide good interpretability. However, they struggle to capture complex nonlinear relationships between meteorological factors and pollutant concentrations, and their performance declines significantly when data are noisy or incomplete. The second approach focuses on deep learning models such as LSTM, GRU, and CNN [14][15][16]. These models are capable of learning long-term temporal dependencies and effectively processing sequential data. Nevertheless, they require large training datasets to avoid overfitting, consume considerable computational resources, and are often difficult to interpret due to their “black-box” nature. Moreover, many studies in this group overlook the analysis of anomalies or outliers, emphasizing prediction accuracy instead. The third approach employs ensemble decision tree models such as Random Forest, XGBoost, and LightGBM [17][18]. These methods handle missing and noisy data well and do not require strict data normalization. However, they are less effective in capturing long-term temporal dependencies and typically rely on lag features to improve time-series forecasting performance.

Based on the above analysis, this study adopts an ensemble tree-based approach, specifically using Random Forest and XGBoost [19]. These models achieve high accuracy in PM2.5 concentration prediction while maintaining interpretability and computational efficiency suitable for real-time sensor data [20]. Additionally, their application enhances system stability, scalability, and adaptability for integration into urban air quality monitoring applications [21]. To enhance reliability and practical applicability, this report proposes a multi-objective pipeline that integrates forecasting (regression), classification, and anomaly detection [22].

The system exploits time-series specific feature engineering such as lag features, rolling statistics, and temporal features to model both short-term variations and long-term trends of air pollution [23]. A Flask backend and a React visual interface implement the real-time system architecture, ensuring a closed loop from data acquisition to inference. Finally, the models are evaluated comprehensively using RMSE, MAE, R^2, and MAPE, together with time-aware validation methods, demonstrating the system's effectiveness and robustness across diverse environmental conditions [24].

2. Methodology

2.1. Pipeline overview

The paper builds the system according to the data processing pipeline process consisting of six main stages, as shown in **Fig 1**.

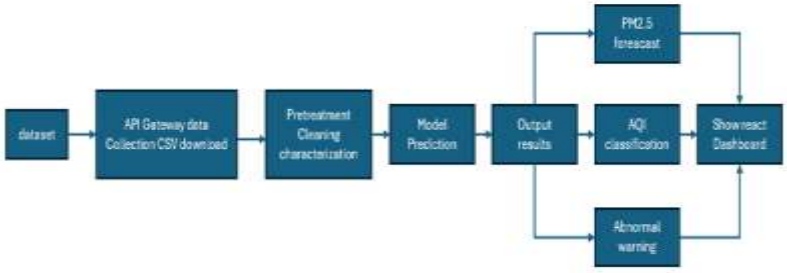

Figure 1. *System pipeline overview*

The Fig 1. shows the data processing chain from input to displaying forecast results, ensuring a closed process from collection, analysis to warning and visualization. Specifically, the process includes the steps Dataset - input data collected from measurement sources or environmental sensors. API Gateway data collection / CSV download - data downloaded via API or CSV file. Pretreatment, Cleaning, Characterization: data is preprocessed, cleaned and extracted appropriate features. Model Prediction - machine learning model trained to predict PM2.5 or AQI index. Output Results - output prediction results. PM2.5 Forecast / AQI Classification - the system performs forecasting of PM2.5 concentration and classifies air quality. Abnormal Warning: warning when detecting abnormal values. Show React Dashboard: displaying results and warnings on a visual dashboard built with React.

Phase 1: Environmental data collection. Sensors are installed in urban areas to record real-time data. Parameters include PM2.5 concentration, temperature, humidity and wind speed. These data reflect the air quality at each point in time and are used as input for subsequent analysis and forecasting.

Phase 2: Data transmission and storage. The collected data is transmitted to the server via API Gateway or uploaded periodically as CSV files. This phase ensures data synchronization from different sources while maintaining the integrity and continuity of the time series.

Phase 3: Preprocessing and feature extraction. Raw data often contains missing values, noise or outliers. Preprocessing involves cleaning the data, removing invalid values and standardizing the format. Then, feature extraction techniques are applied to create statistically and temporally significant factors, including lag features (such as PM2.5_lag1, lag24, lag168), sliding statistics (mean, standard deviation, maximum, minimum) and temporal features (hour, day of the week, season). This step helps the model to recognize both short-term fluctuations and long-term trends in air pollution concentrations.

Phase 4: Model prediction. A Random Forest model is trained and stored as a .joblib file for inference. This model is selected for its robustness to noisy data, minimal normalization requirements, and interpretability regarding the contribution of each feature. In this stage, the system predicts PM2.5 concentrations based on the processed input features.

2.2. Data preprocessing

a. Standardize time

Time data plays an important role in modeling the PM2.5 fine dust concentration time series. Therefore, the system is designed to automatically extract time features from the original time column of the input data [25].

The system performs the extraction process through an automatic processing function, which is responsible for converting the time series to a standard format (datetime) and generating the following features:

- Hour in a day (0-23): Short-term fluctuations by hour
- Day in a week (0-6): Models the difference between working days and weekends
- Month in a year (1-12): Reflects seasonal factors and meteorological conditions
- Weekend flag (0/1): Binary feature to separate holidays

From these features, the machine learning model can identify the time-varying patterns of dust concentrations, such as hourly changes during the day due to traffic or industrial activities; differences between working days and holidays during the week; or seasonal fluctuations associated with climate conditions and agricultural activities. Automating the process of normalization and

feature extraction over time helps ensure data consistency, while increasing the scalability and application of the system across different data sources.

b. Detect and handle outliers

In addition to missing values, outliers often appear during the process of collecting environmental data due to sensor noise, measurement errors or unusual meteorological fluctuations. To ensure the reliability of the forecasting model, this study deploys two outlier detection methods, the IQR statistical method and the Isolation Forest algorithm. In the first step, the IQR (Interquartile Range) method is applied to identify data points that are outside the reasonable range. Specifically, the system calculates the first (Q1) and third (Q3) percentile values, then determines the IQR interval = Q3 – Q1. An observation is considered an outlier if it is outside the interval [Q1−1.5×IQR, Q3+1.5×IQR]. This method is simple, effective and especially suitable for features with a near-normal distribution, helping to eliminate abnormal measurement values due to sensor errors. In parallel, Isolation Forest is incorporated to detect complex anomalies in multidimensional space. This algorithm models the data using a binary tree, in which points that are easily "isolated" are considered anomalies. The contamination rate is set at 5%, to balance the detection ability and avoid discarding valid samples. It is worth noting that instead of completely removing outliers, the system simply labels them for analysis and follow-up in subsequent stages. This approach helps maintain the integrity of the time series while ensuring the accuracy of the statistical features used in the machine learning model.

c. StandardScaler

Data normalization is an essential step in currency processing, which helps to bring specific symbols to the same scale, thereby ensuring stable operation of machine learning algorithms, avoiding numerical bias and achieving optimal forecasting performance. Normalization helps features have the same scale, avoiding the situation where some variables with large values dominate the training results.

In this study, the StandardScaler method is applied to convert features to the same distribution with a mean of 0 and a standard deviation of 1, according to the formula:

$$z = \frac{(x - \mu)}{\sigma} \qquad (1)$$

where: z is value after normalization, x is original value of the feature, μ is mean of the feature and σ is standard deviation of the feature.

This method is implemented in many different contexts of the pipeline:

Preprocessing with PCA (Principal Component Analysis): ensure that features contribute in a balanced way to the data dimensionality reduction process, avoiding the phenomenon of features with dominant magnitude dominating the results.

Meanwhile, K-Means clustering is based on Euclidean distance, so data normalization helps this distance reflect more accurately the similarity between samples, improving the quality and reliability of clustering results.Random Forest and XGBoost model training: helps improve convergence speed and stability when handling features of different scales.

The application of normalization not only increases the uniformity between features but also helps improve the overall accuracy of the PM2.5 concentration prediction model and AQI classification in the study.

2.3. Feature Engineering

a. Lag Features

In time series data, the current value of PM2.5 concentration often has a close relationship with past values. In order for the machine learning model to capture this autoregressive relationship, lag features are created from the target variable and related meteorological variables. Specifically, the create_lag_feature() function is built to automatically generate lagged versions of each feature according to predetermined time intervals, including 1, 3, 6, and 24 hours. This study includes the following lag features: PM2.5(t−1), representing the value immediately before the current time, reflecting the instantaneous fluctuations; PM2.5(t−3): modeling the short-term trend within the last 3 hours; PM2.5(t−6): representing the medium-term fluctuations, helping to detect stable pollution phases; PM2.5(t−24): describes the daily recurring cycle, associated with traffic, industrial and meteorological activities.

The construction of lag features is scientifically important because they help machine learning models identify autoregressive relationships in air quality data. Specifically, short-term lag features (from 1 to 6 hours) describe the delay and diffusion of air pollution over time. Meanwhile, 24-hour lag features capture the typical day-night cycle in urban areas, often affected by traffic and industrial activities. When these features are combined, the accuracy and short-term forecasting ability of machine learning models such as Random Forest and XGBoost are significantly improved.

b. Rolling Statistics

Sliding statistical features are used to reduce noise and identify trends in the time series of air quality data. Specifically, with time windows of 3, 12 and 24 hours, statistics including mean, standard deviation, minimum and maximum values are calculated for each consecutive time period. The sliding average plays the role of smoothing the data and removing short-term noise, while the sliding standard deviation reflects the degree of fluctuation of PM2.5 concentrations in each period. In addition, extreme values (min/max) help identify unusual pollution phenomena. The addition of sliding statistical features helps the machine learning model better exploit the dynamic structure of the data, thereby improving the stability and accuracy of the forecasting process.

c. Temporal Features

Temporal features are extracted to model the natural cycles of air quality variability over time. Specifically, the data are encoded according to the hour of the day, day of the week, month of the year, and weekend or weekday classification. This extraction helps the machine learning model to recognize recurring patterns in the data such as: PM2.5 concentrations tend to increase during peak traffic hours, decrease at night or on weekends due to reduced traffic and industrial activities, and tend to be higher in the winter months due to temperature inversions and heating fuel use. The addition of temporal features allows the model to better capture hourly, daily, and seasonal cycles, thereby improving the ability to forecast real-time fluctuations in PM2.5 concentrations.

d. Interaction Features

The interaction features are built to help the machine learning model capture the complex nonlinear relationships and interactions between meteorological factors. Specifically, two main types of interactions are used: temperature × humidity (temp humidity) – which acts as a thermo-humidity index reflecting the combined effect of temperature and humidity on the ability to diffuse pollutants, and wind speed squared (wind_speed²) – to model the nonlinear effect of wind on the process of dilution and dispersion of fine dust in the atmosphere. Scientifically, meteorological factors do not operate independently but interact with each other: high temperature combined with low humidity can promote the formation of fine dust, while high wind speed helps increase the ability to diffuse pollutants. Adding these interaction features helps the Random Forest and XGBoost models improve their ability to identify nonlinear relationships, thereby improving the accuracy of forecasting PM2.5 concentrations.

2.4. Main Algorithms

a. Random Forest Regressor

The study chose to use the Random Forest algorithm as the baseline model. This is a machine learning method belonging to the ensemble learning group, which works by combining multiple decision trees to reduce variance and improve forecast accuracy. The model is set up with fixed hyperparameters to ensure stability. Specifically, the model consists of 200 decision trees, each tree has a maximum depth of 20 levels. Each split requires a minimum of 5 samples and each leaf contains at least 2 samples. During training, the model only considers the square root of the total number of features at each split. The random number generator is set to 42 to ensure the reproducibility of the results. The training process is performed in parallel on all processing cores to increase the calculation speed and optimize system resources. Random Forest has the ability to effectively prevent overfitting thanks to the bagging mechanism. This method allows independent decision trees to be trained on randomly selected data samples, which helps to reduce bias and improve model reliability.

In addition, the model is capable of describing nonlinear relationships in the data well. This is an important factor in the problem of air quality forecasting, where input variables are often nonlinear and interdependent over time. Random Forest also provides information on the importance of features, allowing researchers to assess the influence of each factor on the forecast results. Finally, the model shows high stability and is easy to adjust, suitable for short-term forecasting applications in the environmental field.

b. XGBoost

XGBoost (Extreme Gradient Boosting) is used as a boosting model in research to improve forecasting accuracy. This algorithm belongs to the gradient boosting group, which works on the principle of correcting the errors of previous models in a sequential manner. The model is set up with 200 boosting loops and the maximum depth of each tree is 6. The learning rate is set at 0.1 to balance the convergence ability and the risk of overfitting. The training process uses 80% of row data samples and 80% of column feature samples in each loop, which helps to increase the generalization ability of the model. The random number generator is set to 42 to ensure the reproducibility of the results. The early stopping mechanism is activated with a threshold of 10 loops, allowing early training to stop when the model no longer improves on the validation set. The main evaluation index used is RMSE (Root Mean Squared Error), which reflects the average squared error between the forecasted value and the actual value. XGBoost is equipped with L1 and L2 regularization mechanisms to control the complexity of the model. Adding a penalty condition to the objective function helps reduce overfitting and increases stability when processing data with many correlated features.

The algorithm has high performance thanks to gradient optimization and parallel processing support, which significantly shortens the training time compared to traditional boosting methods. In addition, XGBoost has the ability to automatically select the best model based on the continuous evaluation process, ensuring that the final result achieves optimal accuracy.

c. PCA (Principal Component Analysis)

PCA (Principal Component Analysis) is applied to reduce data dimensionality and visualize the feature space. This method helps to convert the original data set with many variables into a new set of principal components, retaining most of the variance of the original data. Before performing PCA, the data is normalized using StandardScaler to ensure that the features have the same units of measurement and equivalent influence during the analysis process. Normalization helps PCA focus on the relationship between features instead of their absolute magnitude.

In this study, PCA is configured with two principal components (n_components = 2), allowing visualization of data in two-dimensional space. These two components usually explain more than 90% of the total variance, ensuring that most of the important information of the original data is retained. The application of PCA helps to significantly reduce the number of features, thereby

reducing computational costs and limiting multicollinearity in machine learning models. In addition, representing data in 2D space supports the detection of similar data clusters, the identification of latent structures, and the removal of noise in the original data set.

d. KMeans Clustering

KMeans is used to cluster data samples based on the similarity between normalized features. The algorithm works by dividing the dataset into K separate clusters, where each cluster contains data points with the most similar features according to the Euclidean distance. In this study, the Elbow method is applied to determine the optimal number of clusters (K optimal) through the KElbowVisualizer tool from the Yellowbrick library. This method evaluates the change in the total squared error within the cluster (inertia) when increasing the value of K from 1 to 10, thereby selecting the "elbow" point - where increasing the cluster no longer brings a significant improvement in accuracy.

Once the optimal value of K is determined, the KMeans model is initialized with the K-means++ algorithm, which helps to select the initial cluster centers efficiently and reduces the risk of convergence at local solutions. Training the model on normalized data allows for accurate segmentation of regions with similar characteristics in a high-dimensional space. Applications of KMeans in research include classifying similar air quality conditions, identifying characteristic meteorological patterns, and detecting different pollution states over time. This clustering results not only help analyze environmental trends but also support decision-making and early warning in urban air quality management.

2.5. Validation Strategy

a. TimeSeriesSplit

To ensure the validity of the model when working with time series data, the study applies the method of dividing the data according to the time sequence instead of randomly selecting. This approach helps to preserve the temporal structure of the data and accurately reflects the serial dependence characteristics of the phenomenon to be forecasted. Specifically, the first 80% of the data is used for the training process, while the remaining 20% of the data is reserved for the testing phase. This division is done based on the time index, ensuring that the model is only trained with past data and tested with future data.

Unlike the random sampling method, the division according to time helps to prevent data leakage, which can distort the results of model evaluation. At the same time, keeping the time sequence helps to accurately simulate the forecasting situation in reality, where future data is unknown. This division also helps to avoid optimal bias, ensuring that the model performance indicators accurately reflect the forecasting ability. As a result, the model can be evaluated more objectively and applied more effectively in real-world air quality forecasting problems.

b. Evaluation Metrics

Model performance is evaluated by three main indexes: RMSE, MAE and R^2. RMSE calculation formula (unit µg/m³):

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

In there

This is an index commonly used in machine learning and statistical programs. This index is used to measure the accuracy of a forecasting model. It calculates the square root of the average square of the errors between the actual value and the forecast value. The more accurate the model, the smaller the RMSE and vice versa. MAE is also used as RMSE to measure the accuracy of a forecasting model. It calculates the average of the absolute value of the errors between the actual value and the forecast value. The smaller the MAE, the more accurate the model. But unlike RMSE, MAE does not square the error so it is less affected by large errors.

MAE calculation formula (unit µg/m³):

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \qquad (3)$$

In there:

R^2 (Coefficient of determination) represents the proportion of data variance explained by the model. where R^2 close to 1 indicates that the model explains the data variation well. Calculation formula:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \qquad (4)$$

In there

R^2 on the test set is the criterion for model selection, and a good model must also demonstrate reliability by the RMSE and MAE indices; the model with the best performance will be used for the actual forecasting task and a direct comparison between Random Forest and XGBoost is conducted using the same index.

Additional evaluation is performed using time-split techniques such as TimeSeriesSplit and walk-forward validation to properly reflect temporal dependencies and avoid data leakage. Feature importance analysis is used to evaluate the interpretability of the model and to identify the variables that contribute the most to the forecast. Residual analysis involves plotting residuals against

predicted values to test assumptions such as random distribution around zero and detect heteroscedasticity or lack of nonlinear structure in the model. Anomaly detection is integrated into the evaluation process with a default contamination rate of 5% to identify suspicious data points, supporting the testing of the model's sensitivity to outliers.

3. Results

The study presented a system that was trained and evaluated on three different datasets, using multiple machine learning algorithms. The system used XGBoost and Random Forest as the two main models for comparison. The overall results were compiled and automatically reported by the system, showing that:

- Random Forest: $R^2 \approx 0.895$
- XGBoost: $R^2 \approx 0.915$
- Best performance (aggregated on 3 datasets): $R^2 \approx 0.916$

These values show a high agreement between the predicted and actual values, especially with the XGBoost model, which shows better generalization ability than Random Forest.

Table 1. *Comparing results between machine learning models*

Model	Key performance indicators
Random Forest	$R^2 \approx 0.895$
XGBoost	$R^2 \approx 0.915$
Best (across datasets)	$R^2 \approx 0.916$

Table 1. shows the predictive performance of regression models in the PM2.5 index forecasting problem. The study compares two models, Random Forest Regressor and XGBoost Regressor, with the main evaluation index being the coefficient of determination R^2 – showing the degree of agreement between the predicted value and the actual value.

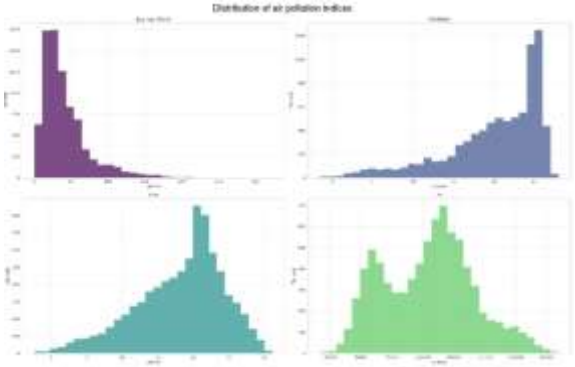

Figure 2. *Index distribution*

The Fig 2. clearly shows the right-skewed trend of PM2.5 data, showing that the majority of values are concentrated in the low to medium concentration range, while only a small proportion of data points show abnormally high pollution levels (outliers).

This characteristic indicates an imbalanced distribution, causing machine learning models such as XGBoost and Random Forest to tend to predict well in the low value range but lose accuracy at high pollution peaks — where training data is scarce. Therefore, to improve forecasting performance in the future, techniques to deal with imbalanced data such as data augmentation, re-sampling, or weighting loss should be considered. These measures can help reduce model bias and increase the ability to identify severe pollution periods.

The residual plot of the XGBoost model shows that most of the errors are concentrated around 0 for small prediction samples, but the

variance of the residual increases significantly when the prediction value is large, a phenomenon called heteroscedasticity. Some outliers with large positive or negative errors are observed at the PM2.5 peak points, proving that the model does not predict accurately in high pollution situations. Specifically, at low values, the model gives an average error close to 0, with small dispersion. At high values: the error tends to be negative, the model predicts lower than reality (underprediction). There is a time lag between the actual and predicted values in fast fluctuations.

International Journal of Engineering and Information Systems (IJEAIS)
ISSN: 2643-640X
Vol. 9 Issue 10 October - 2025, Pages: 128-136

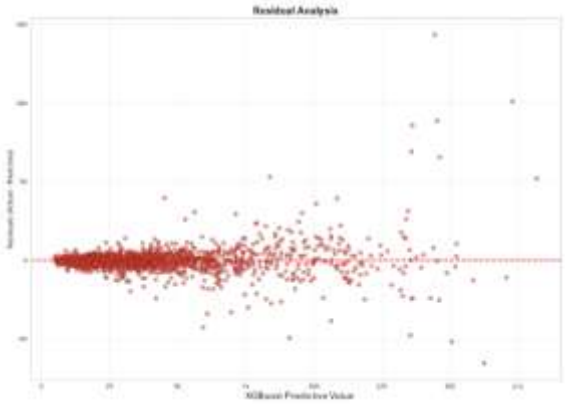

Figure 3. *Residual plot of XGBoost model*

The Fig 3. showed the relationship between meteorological variables (temperature, humidity, wind speed – WS, absolute humidity – QV10M) and temporal characteristics of PM2.5 (rolling mean, rolling min, rolling max in the last 3 hours). The results showed that rolling variables of PM2.5 had the highest correlation coefficient with the target value, with |r| ranging from 0.7–0.9. This demonstrates the key role of temporal statistical characteristics in modeling and predicting short-term trends of fine dust concentrations. In contrast, meteorological variables such as wind speed and absolute humidity showed weak correlations ($|r| < 0.3$), indicating that their direct influence on PM2.5 is limited. Overall, the analytical results reinforce the importance of exploiting time series information in air quality forecasting and suggest that dimensionality reduction or feature selection techniques (such as PCA) can be applied to optimize performance and eliminate data duplication.

Figure 4. *Correlation matrix*

The Fig 4. illustrates the correlation matrix between environmental pollution indicators, in which each cell represents the Pearson correlation coefficient between two variables. Red areas indicate strong positive correlations, while blue areas indicate negative correlations, and intermediate colors indicate weak relationships. The results show that PM2.5 has a significant positive correlation with temperature (T2M) and atmospheric pressure (PS), indicating that the concentration of fine dust tends to increase as these factors increase. In contrast, PM2.5 has a significant negative correlation with wind speed (WS) and total humidity (TQV), meaning that the dust concentration decreases when the wind is strong or the humidity is high. Meteorological variables such as T2M, T2M_MAX and T2M_MIN have very strong positive correlations with each other, reflecting the co-variation with temperature conditions. Overall, this figure shows that temperature, pressure and wind speed are the main factors influencing the variation of PM2.5 concentration, and also supports the selection of input features for air quality forecasting models.

In practice, although $R^2 \approx 0.91$ is achieved, large errors at the PM2.5 peak can directly affect the reliability of the air pollution warning system. Therefore, it is necessary to further optimize the loss function to penalize errors at the extreme regions more severely. Add data augmentation for rare events. Combine an ensemble of specialized models for peak detection to improve the overall accuracy. Analysis of the feature importance of the XGBoost model shows that the rolling variables of PM2.5 play a key role in the forecasting ability.

The three features PM2.5_rolling_mean_3, PM2.5_rolling_min_3 and PM2.5_rolling_max_3 are ranked in the top 15 most important features. These features show a superior level of influence compared to other meteorological factors such as wind speed (WS) or humidity (QV10M). This result shows that the temporal statistical features (rolling mean, rolling min, rolling max) help the

model capture the short-term trend and fluctuation of PM2.5 concentration. As a result, the model can significantly improve the accuracy in short-term forecasting.

4. Discussion

The study successfully built an air quality forecasting and analysis system based on environmental data. The implemented machine learning model includes Random Forest and XGBoost, which shows stable forecasting ability and high accuracy for PM2.5 concentration. PCA and K-Means methods support visualization and identification of characteristic data groups, contributing to improving the model interpretability.

Experimental results demonstrate that combining traditional machine learning algorithms with data preprocessing and dimensionality reduction techniques provides good forecasting performance in the context of urban data. The user interface system helps visualize the results and facilitates non-expert users to approach environmental data analysis.

However, the system still has certain limitations. The forecast quality strongly depends on the reliability of input data and the frequency of data collection. The current model does not support automatic training, dynamic parameter update mechanism or real-time data processing. In addition, the system still uses a file-based storage structure, which is not suitable for large data scale and production environment applications. In the future, the development direction focuses on model optimization through AutoML, expansion to deep learning architectures such as LSTM and Transformer, and implementation of real-time forecasting. In addition, the integration of databases and cloud platforms will improve the scalability and performance of the system. Long-term research can aim at spatial modeling using GIS, causal analysis and policy recommendations based on environmental data.

5. Conclusions

The methodology presented in this study provides a comprehensive and scientific analytical framework for air quality analysis and forecasting. By combining advanced data processing techniques, powerful machine learning models, multidimensional analysis, and AI-based intelligent visualization, the proposed system has demonstrated high accuracy and clear interpretability.

Specifically, the study has integrated:

- Advanced feature extraction techniques, including lag features, rolling statistics, and interaction features, to improve the ability to identify temporal relationships between environmental variables.
- Powerful machine learning algorithms such as Random Forest and XGBoost, combined with automatic model selection mechanisms to optimize forecasting performance.
- Multidimensional data analysis, using PCA, clustering and anomaly detection to identify hidden patterns and outliers in the dataset.
- Rich visualizations, with more than 15 types of statistical charts and machine learning, make the analysis intuitive and easy to understand.
- AI-based automated analysis through Gemini AI, allowing the system to automatically interpret results and provide meaningful insights.
- Modern system architecture, including Flask (backend), Next.js (frontend) and real-time streaming, ensures high performance and scalability.

The system not only provides accurate forecasts of air quality, but also helps to deeply understand the factors affecting the environment. From there, the research results can support managers and policymakers in making decisions and building environmental protection strategies.

In addition, this method has the potential to expand its application to other environmental fields, creating a foundation for further research in the field of environmental quality monitoring and forecasting based on machine learning and artificial intelligence technology.

References

1. Šarčević-Todosijević, L., Malivuk, A., Perić, M., Popović, V., Golijan, J., & Živanović, L. (2023). Environmental pollution-the leading challenge of the 21st century. In Proceedings, 27th International Eco-Conference and 15th Environmental Protection of Urban and Suburban Settlements, 27-29.09. 2023, Novi Sad (pp. 387-394). Novi Sad: Ecological Movement of Novi Sad.
2. Viegi, G., Maio, S., Fasola, S., & Baldacci, S. (2020). Global burden of chronic respiratory diseases. Journal of aerosol medicine and pulmonary drug delivery, 33(4), 171-177.
3. Labaki, W. W., & Han, M. K. (2020). Chronic respiratory diseases: a global view. The Lancet Respiratory Medicine, 8(6), 531-533.
4. Vu, H. N. K., Ha, Q. P., Nguyen, D. H., Nguyen, T. T. T., Nguyen, T. T., Nguyen, T. T. H., ... & Ho, B. Q. (2020). Poor air quality and its association with mortality in Ho Chi Minh City: case study. Atmosphere, 11(7), 750.

5. Kumar, P. G., Lekhana, P., Tejaswi, M., & Chandrakala, S. J. M. T. P. (2021). Effects of vehicular emissions on the urban environment-a state of the art. Materials Today: Proceedings, 45, 6314-6320.
6. Gupta, V. (2019). Vehicle-generated heavy metal pollution in an urban environment and its distribution into various environmental components. In Environmental Concerns and Sustainable Development: Volume 1: Air, Water and Energy Resources (pp. 113-127). Singapore: Springer Singapore.
7. Vlasov, D., Ramírez, O., & Luhar, A. (2022). Road dust in urban and industrial environments: Sources, pollutants, impacts, and management. Atmosphere, 13(4), 607.
8. Mujtaba, G., & Shahzad, S. J. H. (2021). Air pollutants, economic growth and public health: implications for sustainable development in OECD countries. Environmental Science and Pollution Research, 28(10), 12686-12698.
9. Taghizadeh-Hesary, F., & Taghizadeh-Hesary, F. (2020). The impacts of air pollution on health and economy in Southeast Asia. Energies, 13(7), 1812.
10. Essamlali, I., Nhaila, H., & El Khaili, M. (2024). Supervised machine learning approaches for predicting key pollutants and for the sustainable enhancement of urban air quality: A systematic review. Sustainability, 16(3), 976.
11. Naseem, F., Rashid, A., Izhar, T., Khawar, M. I., Bano, S., Ashraf, A., & Adnan, M. N. (2018). An integrated approach to air pollution modeling from climate change perspective using ARIMA forecasting. Journal of Applied Agriculture and Biotechnology, 2(2), 37-44.
12. Mani, G., Viswanadhapalli, J. K., & Stonier, A. A. (2022). Prediction and forecasting of air quality index in Chennai using regression and ARIMA time series models. Journal of Engineering Research, 10(2), 179-194.
13. Tao, Q., Liu, F., Li, Y., & Sidorov, D. (2019). Air pollution forecasting using a deep learning model based on 1D convnets and bidirectional GRU. IEEE access, 7, 76690-76698.
14. Guo, Z., Yang, C., Wang, D., & Liu, H. (2023). A novel deep learning model integrating CNN and GRU to predict particulate matter concentrations. Process Safety and Environmental Protection, 173, 604-613.
15. Zhang, Q., Han, Y., Li, V. O., & Lam, J. C. (2022). Deep-AIR: A hybrid CNN-LSTM framework for fine-grained air pollution estimation and forecast in metropolitan cities. IEEE access, 10, 55818-55841.
16. Tırınk, S. (2025). Machine learning-based forecasting of air quality index under long-term environmental patterns: A comparative approach with XGBoost, LightGBM, and SVM. PloS one, 20(10), e0334252.
17. Yu, T. K., Chang, I. C., Chen, S. D., Chen, H. L., & Yu, T. Y. (2025). Predicting potential soil and groundwater contamination risks from gas stations using three machine learning models (XGBoost, LightGBM, and Random Forest). Process Safety and Environmental Protection, 107249.
18. Özüpak, Y., Alpsalaz, F., & Aslan, E. (2025). Air Quality Forecasting Using Machine Learning: Comparative Analysis and Ensemble Strategies for Enhanced Prediction. Water, Air, & Soil Pollution, 236(7), 464.
19. Yan, X., Zang, Z., Luo, N., Jiang, Y., & Li, Z. (2020). New interpretable deep learning model to monitor real-time PM2. 5 concentrations from satellite data. Environment international, 144, 106060.
20. Kumari, S., Choudhury, A., Karki, P., Simon, M., Chowdhry, J., Nandra, A., ... & Garg, M. C. (2025). Next-Generation Air Quality Management: Unveiling Advanced Techniques for Monitoring and Controlling Pollution. Aerosol Science and Engineering, 1-22.
21. Espinosa Fernández, R. (2023). Multi-objective evolutionary feature selection with deep learning applied to air quality spatio-temporal forecasting. Proyecto de investigación.
22. Nath, P., Saha, P., Middya, A. I., & Roy, S. (2021). Long-term time-series pollution forecast using statistical and deep learning methods. Neural Computing and Applications, 33(19), 12551-12570.
23. Zhao, T., Chen, G., Pang, C., & Busababodhin, P. (2025). Application and performance optimization of SLHS-TCN-XGBoost model in power demand forecasting. Computer Modeling in Engineering & Sciences, 143(3), 2883.
24. Wu, C., Wang, R., Lu, S., Tian, J., Yin, L., Wang, L., & Zheng, W. (2025). Time-series data-driven pm2. 5 forecasting: From theoretical framework to empirical analysis. Atmosphere, 16(3), 292.