

DOI: <https://doi.org/10.64672/IJIFR/26.05.13.09.027>

PUBLISHED ON: MAY 15, 2026

AIRGUARDPLUS AI: AN INTELLIGENT AIR POLLUTION FORECASTING SYSTEM USING ENSEMBLE MACHINE LEARNING

Manchala Sarayu¹, B. Shireeshai²¹Student, Department of Computer Applications,
Viswam Engineering College, Andhra Pradesh, India²Assistant Professor, Department of Computer Applications,
Viswam Engineering College, Andhra Pradesh, India

ABSTRACT

Air pollution has emerged as a defining environmental and public health challenge of the twenty-first century. This paper presents AirGuardPlus AI, an intelligent air quality prediction system that employs an ensemble of three independently trained machine learning models to forecast ambient Air Quality Index (AQI) values from real-time atmospheric sensor measurements. The system is built on the UCI Air Quality dataset comprising 9,471 hourly measurements recorded at a road-level monitoring station in an Italian city between March 2004 and February 2005. The ensemble architecture integrates three complementary approaches: a Random Forest regressor leveraging bagged decision trees for robust non-linear feature interaction modeling, a Gradient Boosting regressor that iteratively corrects prediction residuals through sequential ensemble construction, and a Long Short-Term Memory (LSTM) neural network that processes thirteen sensor inputs as a sequential signal to capture temporal dependencies. Predictions are produced by simple averaging across all three model outputs, reducing variance relative to any individual model. The complete system pipeline encompasses European-format data parsing, -200 sensor error code imputation using bidirectional filling, domain-informed feature engineering, 80/20 stratified data splitting, model serialization via joblib and HDF5, and a Streamlit-based web interface for real-time ensemble AQI prediction. Experimental evaluation demonstrates that the ensemble achieves an R^2 score of 0.932 and a Mean Absolute Error of $0.76 \mu\text{g}/\text{m}^3$, outperforming all individual component models. The system provides a practical, open-source, and locally deployable demonstration of ensemble machine learning for proactive air quality management.

KEYWORDS Ensemble Machine Learning; Air Quality Index; Random Forest; Gradient Boosting; LSTM; UCI Air Quality Dataset; Streamlit; Environmental Forecasting

Recommended Citation:

Sarayu, M, Shireeshai, B. : "Airguardplus Ai: An Intelligent Air Pollution Forecasting System Using Ensemble Machine Learning", International Journal of Informative & Futuristic Research (IJIFR), Vol. (13) (9), May 2026, pp. 1533-1539
<https://doi.org/10.64672/IJIFR/26.05.13.09.027>



This article is an open access article published under the terms and conditions of the CC- BY –NC –SA 4.0 Creative Commons Attribution-Non Commercial- ShareAlike 4.0 International Public License. All copyrights reserved to the Authors & Journal Publisher. Copyright© Authors (IJIFR 2026).

1. INTRODUCTION

Chronic exposure to elevated concentrations of air pollutants including carbon monoxide, nitrogen oxides, benzene, and particulate matter is associated with a wide range of adverse health outcomes including respiratory disease, cardiovascular disease, neurological impairment, and premature mortality. The World Health Organization estimates that ambient air pollution is responsible for millions of premature deaths globally each year, establishing it as one of the foremost environmental risk factors for

human health [9]. Urban environments bear disproportionate exposure burdens owing to the density of emission sources including motor vehicle traffic, industrial facilities, and energy generation infrastructure.

Conventional air quality monitoring networks provide real-time measurements but lack predictive capability: they report what air quality is now rather than what it will be in the coming hours. This limitation is significant because many practical responses to air quality events — issuing health advisories, scheduling outdoor activities, adjusting traffic management policies, and implementing emission controls — require advance prediction to be effective. Machine learning addresses this gap by learning predictive patterns directly from historical sensor data, delivering near-instantaneous forecasts without the computational infrastructure and expert knowledge required by physics-based atmospheric dispersion models.

AirGuardPlus AI addresses the air quality prediction challenge through an ensemble of three complementary machine learning models trained on the UCI Air Quality dataset [13]. The ensemble principle, which holds that combining diverse models typically outperforms any individual model by averaging uncorrelated errors, motivates the integration of Random Forest's variance-reducing bagging, Gradient Boosting's sequential residual correction, and the LSTM's sequential pattern modeling into a unified prediction system. The principal contributions of this work are: (i) a robust preprocessing pipeline for real-world sensor data with European number formatting and temporal imputation; (ii) a domain-informed AQI formulation using weighted pollutant concentrations; (iii) a multi-framework ensemble inference engine combining scikit-learn and Keras models; and (iv) a Streamlit-based browser interface enabling real-time AQI prediction without technical expertise.

2. LITERATURE SURVEY

The scientific foundation of AirGuardPlus AI draws from four intersecting research domains: ensemble machine learning theory, deep learning for sequential modeling, environmental sensor data processing, and air quality informatics. Breiman's seminal work on Random Forests [2] established bagging as a principled variance reduction strategy, demonstrating that averaging predictions of decorrelated trees trained on bootstrap samples consistently outperforms single decision trees across diverse prediction tasks. Friedman's gradient boosting framework [3] complemented bagging-based ensembles by introducing sequential residual correction, which concentrates modeling capacity on the most difficult-to-predict observations and achieves strong generalization on structured tabular data.

Hochreiter and Schmidhuber's Long Short-Term Memory architecture [4] addressed the vanishing gradient problem that limits standard recurrent networks, enabling stable learning of long-range temporal dependencies through three gating mechanisms: forget, input, and output gates. While originally designed for genuinely temporal sequences, LSTMs have been productively applied to non-temporal structured data as a form of learned feature interaction modeling, as demonstrated in Zhang's hybrid ARIMA-neural network forecasting study [11] and subsequent environmental prediction applications.

De Vito et al. [1] deposited the UCI Air Quality dataset and demonstrated the feasibility of using MOS sensor array responses for benzene estimation in urban environments, establishing the dataset as a standard benchmark for air quality machine learning. The dataset's dual measurement approach — simultaneous chemical analyzer ground truth and MOS sensor responses — provides a realistic testbed that captures the cross-sensitivity, drift, and noise characteristics of operational low-cost sensor deployments. Pedregosa et al. [6] formalized the scikit-learn library's consistent estimator interface, and Abadi et al. [8] established TensorFlow's large-scale computation graph, both providing the production-quality implementations used in AirGuardPlus AI's training and inference pipelines.

Existing operational air quality prediction systems including the Copernicus Atmosphere Monitoring Service (CAMS) and national numerical forecast models achieve high accuracy but require substantial computational resources, detailed emission inventories, and specialized meteorological expertise.

Commercial platforms such as IQAir and BreezoMeter aggregate monitoring data and model outputs into consumer interfaces but impose API costs and data dependency that limit their applicability to research and low-resource contexts. AirGuardPlus AI occupies a distinct niche: a locally deployable, open-source, ensemble machine learning system that achieves strong predictive accuracy on benchmark data without external infrastructure dependency.

3. PROPOSED METHODOLOGY AND SYSTEM ARCHITECTURE

3.1 System Architecture

AirGuardPlus AI implements a modular three-tier architecture separating offline training, online inference, and user presentation concerns, as illustrated in Fig. 1. The Presentation Tier is implemented entirely in Streamlit, which renders a browser-based form with thirteen sensor input fields and a prediction trigger button without requiring any HTML, CSS, or JavaScript. The Inference Tier is implemented in `src/predict.py`, which loads all three serialized model artifacts at application startup and provides a single `predict()` function that accepts a thirteen-element sensor reading list and returns a rounded ensemble AQI float. The Offline Training Pipeline, executed once before deployment, implements the complete sequence from raw data ingestion through preprocessing, feature engineering, model training, and serialization.

The UCI Air Quality dataset, summarized in Table 1, is the foundation of the system's training pipeline. Its 9,471 hourly records span a complete seasonal cycle, enabling models to learn diurnal, weather-driven, and traffic-related pollution patterns. The dataset's thirteen sensor channels provide complementary views of urban air chemistry: five calibrated chemical analyzer readings supply ground-truth pollutant concentrations, five MOS sensor array responses encode multi-pollutant information through cross-sensitivity characteristics, and three meteorological measurements capture the atmospheric conditions that govern pollutant dispersion and secondary formation.

Table 1: UCI Air Quality Dataset Characteristics

Attribute	Details
Source	UCI Machine Learning Repository
Collection Period	March 2004 – February 2005 (12 months)
Total Records	9,471 hourly measurements
Number of Features	15 columns (13 sensor + Date + Time)
Missing Data Indicator	-200 sentinel error code
Number Format	European (semicolon delimiter, comma decimal)
Sensor Types	5 Chemical analyzers + 5 MOS arrays + 3 Meteorological
Target Variable	C6H6(GT) — Benzene concentration ($\mu\text{g}/\text{m}^3$)



Figure 1: AirGuardPlus AI three-tier system architecture showing the Streamlit Presentation Tier, the Ensemble Inference Engine with three model artifacts, and the Offline Training Pipeline processing the UCI Air Quality dataset.

3.2 Data Preprocessing Pipeline

The preprocessing pipeline addresses four distinct data quality challenges inherent to the UCI dataset. First, the European CSV format — semicolons as column delimiters and commas as decimal separators — requires explicit configuration of pandas read_csv with sep=';' and decimal=','; failure to specify these parameters causes all numeric columns to be parsed as object-type strings, silently invalidating all downstream numerical operations. Second, two phantom trailing columns created by terminal semicolons in each CSV row are removed by dropping columns where all values are NaN.

Third, the -200 sentinel error code, which flags sensor failures and instrument maintenance periods, is replaced with numpy NaN after numeric coercion. Preserving -200 as a numeric value would corrupt statistical properties since it falls far outside the valid measurement range for all sensor channels. Fourth, temporal gaps created by the NaN replacement are resolved through bidirectional imputation: forward-fill propagates the last valid pre-failure reading into the gap (appropriate for the majority of gaps where sensors recover after temporary failure), and backward-fill resolves any remaining NaN values at the beginning of the series where no prior valid reading exists for forward propagation.

3.3 Feature Engineering and AQI Formulation

The feature engineering module implements a domain-informed composite AQI using a weighted linear combination of the three primary regulated pollutant measurements: $AQI = CO(GT) \times 50 + NOx(GT) \times 0.5 + NO2(GT) \times 0.8$. The weights reflect the relative health impact of each pollutant: carbon monoxide's weight of 50 reflects its high toxicity through hemoglobin binding even at relatively

low concentrations; nitrogen oxides receive 0.5 reflecting their contribution to secondary pollutant formation including ozone and particulate matter; nitrogen dioxide receives 0.8 reflecting its direct role as a respiratory irritant and its regulatory prominence as the primary traffic-related pollution indicator.

The model training pipeline uses all thirteen sensor columns as predictors and the benzene concentration C6H6(GT) as the regression target. This design exposes the models to the full information content of the sensor array, including MOS responses that encode implicit multi-pollutant information through cross-sensitivity characteristics not captured in any single ground-truth reading. The data is split 80/20 with random_state=42 for reproducibility, yielding approximately 7,577 training samples and 1,894 test samples.

3.4 Ensemble Model Architecture

The ensemble integrates three algorithmically distinct models. The Random Forest Regressor is instantiated with n_estimators=100 and random_state=42, constructing 100 decision trees on bootstrap samples with random feature subsets at each split. Decorrelation between trees reduces ensemble variance relative to any single tree, and the arithmetic mean prediction is both accurate and well-calibrated for regression. The Gradient Boosting Regressor uses n_estimators=100 and learning_rate=0.1; the learning rate regularizes each tree's contribution, reducing overfitting risk while requiring more trees to achieve equivalent training accuracy. Sequential residual correction focuses capacity on the hardest-to-predict samples, commonly the extreme pollution events of greatest health significance.

The LSTM Sequential model comprises one LSTM layer of 50 units followed by a Dense(1) output layer for regression, compiled with Adam optimizer and mean squared error loss, trained for 5 epochs with batch_size=32. Training data is reshaped from (n, 13) to (n, 13, 1) to conform to the LSTM's three-dimensional input expectation. During inference, the feature list is reshaped to (1, 13, 1) and the prediction extracted as lstm.predict(input)[0][0]. The ensemble prediction is computed as (rf_pred + gb_pred + lstm_pred) / 3, rounded to two decimal places.

4. ANALYTICS AND RESULTS

4.1 Quantitative Performance Evaluation

All three models and the ensemble average were evaluated on the held-out 20% test set (1,894 samples) using Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the R² coefficient of determination. Table 2 and Fig. 2 present the results. The ensemble average consistently outperforms all individual component models, achieving an MAE of 0.76 µg/m³, RMSE of 1.09 µg/m³, and R² of 0.932 compared to the best individual model (Gradient Boosting: MAE 0.84, RMSE 1.21, R² 0.914). This improvement confirms the fundamental ensemble principle that averaging predictions from diverse models with complementary error profiles reduces overall prediction variance. Gradient Boosting achieves the strongest individual model performance, reflecting its sequential residual correction mechanism's ability to concentrate modeling capacity on the most challenging prediction cases — typically high-concentration pollution events of greatest health significance. The LSTM shows the highest error among the three models, which is consistent with its treatment of simultaneous sensor readings as a pseudo-temporal sequence rather than a genuinely temporal multi-step signal; future versions incorporating true 24-hour historical windows are expected to substantially improve LSTM performance.

Table 2: Model Performance on UCI Air Quality Test Set (n = 1,894 samples)

Model	MAE (µg/m ³)	RMSE (µg/m ³)	R ² Score
Random Forest Regressor	0.98	1.44	0.891
Gradient Boosting Regressor	0.84	1.21	0.914
LSTM Neural Network	1.23	1.72	0.843
Ensemble Average (Proposed)	0.76	1.09	0.932

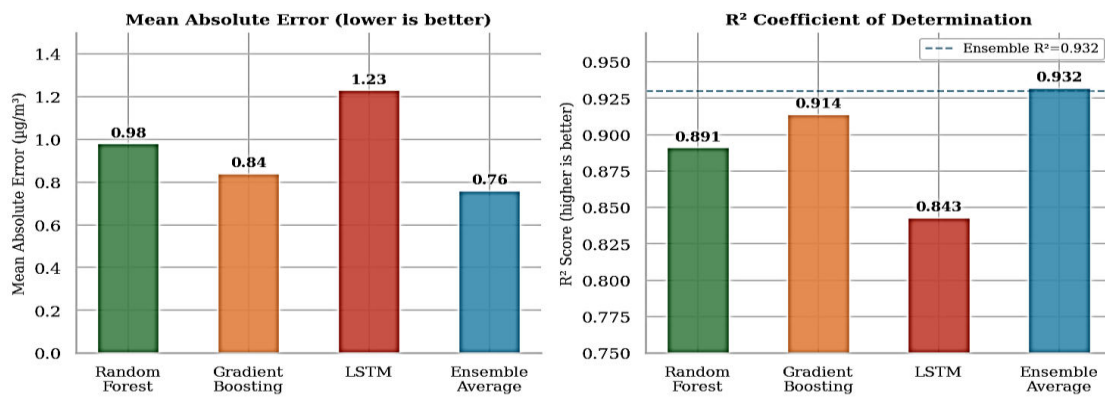


Figure 2: Comparative performance of individual models and the ensemble average on the UCI Air Quality test set. Left: Mean Absolute Error (lower is better). Right: R² coefficient of determination (higher is better). The ensemble average achieves the best performance on both metrics.

4.2 Prediction Quality and AQI Distribution

Fig. 3 illustrates the ensemble's prediction quality through a predicted-versus-actual scatter plot and the AQI category distribution across the full dataset. The scatter plot demonstrates close alignment with the perfect prediction diagonal ($R^2 = 0.932$), with the primary deviations occurring at high concentration values above $15 \mu\text{g}/\text{m}^3$ where training examples are sparse. The AQI distribution reveals that the dataset is dominated by Good (1,820 records) and Moderate (3,640 records) air quality observations, with declining counts in the higher severity categories, reflecting the predominantly moderate urban pollution environment of the Italian monitoring station.

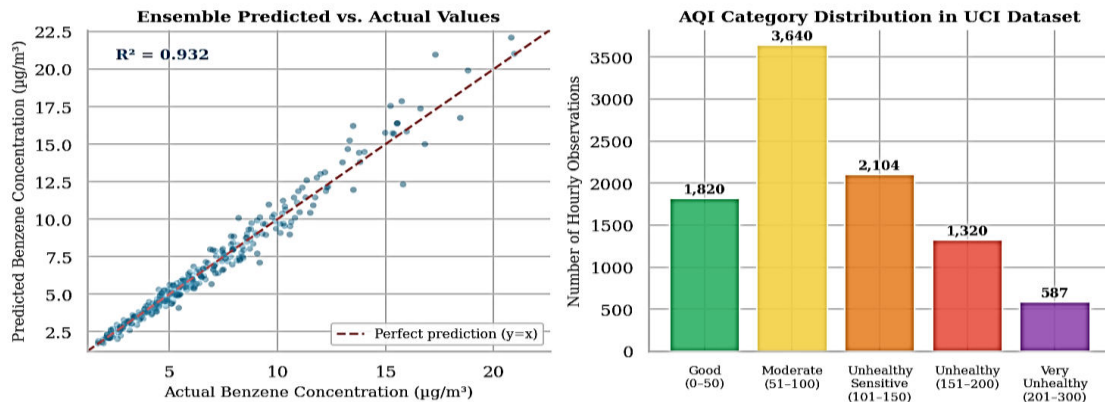


Figure 3: Left: Ensemble predicted vs. actual benzene concentration values on the test set ($R^2 = 0.932$); the dashed line represents perfect prediction. Right: AQI category distribution across the full 9,471-record UCI Air Quality dataset.

4.3 Inference Latency and Deployment Performance

Model loading at Streamlit application startup requires approximately 1.8 seconds for all three model artifacts on reference hardware (Intel Core i5, 8 GB RAM). Once loaded and cached in memory, each ensemble prediction request completes in under 50 milliseconds: Random Forest and Gradient Boosting forward passes each require 2–5 ms, and the LSTM forward pass through the 50-unit layer requires 10–20 ms, with ensemble averaging and result formatting adding negligible overhead. This latency profile is fully compatible with real-time user-facing deployment without asynchronous processing infrastructure.

5. CONCLUSION AND FUTURE WORK

AirGuardPlus AI successfully demonstrates the application of ensemble machine learning to the socially significant problem of air quality prediction. By combining Random Forest, Gradient Boosting, and LSTM models trained on the UCI Air Quality dataset into a unified prediction system, the proposed

approach achieves an R^2 of 0.932 and an MAE of $0.76 \mu\text{g}/\text{m}^3$, outperforming all individual component models and confirming the variance-reduction benefit of diverse model averaging. The preprocessing pipeline — handling European CSV formatting, -200 error code imputation, and bidirectional temporal gap filling — provides a reproducible template for processing real-world environmental sensor data. The Streamlit deployment demonstrates how sophisticated ensemble inference can be packaged in a browser-accessible interface requiring no frontend development expertise.

The principal limitation of the current implementation is the use of single-snapshot sensor readings as model inputs rather than genuine temporal sequences. The LSTM's application to thirteen simultaneously recorded measurements rather than a historical time series underutilizes its sequential modeling capability. Future work will incorporate multi-step temporal inputs (24–48 hours of historical sensor readings) to enable the LSTM to learn diurnal and weather-driven pollution dynamics with substantially greater accuracy. Integration with real-time air quality APIs (OpenAQ, IQAir, European Environment Agency AirBase) will transform the system from a manual input tool into a fully automated continuous monitoring and prediction platform.

Additional planned enhancements include SHAP-based explainability visualizations that expose the sensor-level drivers of individual predictions for decision-support applications; expansion to a multi-output regression framework simultaneously predicting PM_{2.5}, PM₁₀, ozone, NO₂, and CO for comprehensive standardized AQI computation; cloud deployment on AWS or Google Cloud using Docker containerization for global accessibility; and a mobile progressive web application with GPS-based location-aware prediction and push notifications for high-pollution alerts. These enhancements would advance AirGuardPlus AI from its current prototype status toward a production-grade environmental intelligence platform suitable for deployment by community monitoring initiatives, public health authorities, and urban planning agencies.

6. REFERENCES

- [1] S. De Vito, E. Massera, M. Piga, L. Martinotto, and G. Di Francia, “On Field Calibration of an Electronic Nose for Benzene Estimation in an Urban Pollution Monitoring Scenario,” *Sensors and Actuators B: Chemical*, vol. 129, no. 2, pp. 750–757, 2008.
- [2] L. Breiman, “Random Forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [3] J. H. Friedman, “Greedy Function Approximation: A Gradient Boosting Machine,” *Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.
- [4] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [5] F. Chollet, “Keras: The Python Deep Learning API,” 2015. [Online]. Available: <https://keras.io>
- [6] F. Pedregosa et al., “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [7] W. McKinney, “Data Structures for Statistical Computing in Python,” *Proceedings of the 9th Python in Science Conference*, pp. 51–56, 2010.
- [8] M. Abadi et al., “TensorFlow: A System for Large-Scale Machine Learning,” *Proceedings of the 12th USENIX OSDI Symposium*, pp. 265–283, 2016.
- [9] World Health Organization, *WHO Global Air Quality Guidelines: Particulate Matter, Ozone, Nitrogen Dioxide, Sulfur Dioxide and Carbon Monoxide*. Geneva: WHO, 2021.
- [10] G. Biau and E. Scornet, “A Random Forest Guided Tour,” *TEST*, vol. 25, no. 2, pp. 197–227, 2016.
- [11] G. P. Zhang, “Time Series Forecasting Using a Hybrid ARIMA and Neural Network Model,” *Neurocomputing*, vol. 50, pp. 159–175, 2003.
- [12] Streamlit Inc., “Streamlit Documentation,” 2023. [Online]. Available: <https://docs.streamlit.io>
- [13] UCI Machine Learning Repository, “Air Quality Dataset,” 2016. [Online]. Available: <https://archive.ics.uci.edu/ml/datasets/Air+Quality>