

# AI-Based Smart Air Quality Monitoring and AQI Prediction System Using IoT and Hybrid Machine Learning Models

Disha Garge<sup>1</sup>, Akshita Dey<sup>2</sup>, Sachi Jangid<sup>3</sup>, Rishi Kumar Yadav<sup>4</sup>

<sup>1234</sup>Department of Computer Science and Engineering, M S Ramaiah University of Applied Sciences, Bengaluru, India

\*\*\*

**Abstract**-Air pollution poses a significant threat to human health, contributing to respiratory diseases and environmental degradation. Conventional air quality monitoring systems are expensive, stationary, and lack predictive capabilities. This paper presents a smart air quality monitoring and AQI prediction system integrating Internet of Things (IoT) with hybrid Machine Learning (ML) models. The system employs an ESP32 microcontroller interfaced with an MQ135 gas sensor, DHT11 temperature and humidity sensor, and a GPS module for real-time data acquisition. Sensor data is transmitted over WiFi to a hosted cloud server (aislyntech.com) and processed using a hybrid model combining Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and K-Nearest Neighbors (KNN). A Flask-based web application and Blynk IoT mobile app provide real-time visualisation and predictive analytics. Experimental results demonstrate reliable monitoring and prediction performance with an MAE of approximately 0.75, making the system well-suited for smart city and environmental health awareness applications.

**Keywords:** IoT, Air Quality Monitoring, ESP32, CNN, LSTM, KNN, AQI Prediction, Flask, MQ135, Blynk, Smart Systems

## 1. INTRODUCTION

Air pollution has emerged as a critical global challenge with far-reaching consequences for human health and environmental sustainability. Toxic pollutants including carbon monoxide, nitrogen oxides, sulphur dioxide, and fine particulate matter (PM<sub>2.5</sub>, PM<sub>10</sub>) are closely linked to chronic respiratory conditions such as asthma, bronchitis, and Chronic Obstructive Pulmonary Disease (COPD). The World Health Organization estimates that ambient air pollution causes over 4.2 million premature deaths annually worldwide.

Traditional air quality monitoring infrastructure relies on large, expensive fixed-location stations operated by governmental agencies. These systems suffer from three primary limitations: high capital and operational costs, limited spatial coverage, and the complete absence of predictive intelligence. Citizens and urban planners are left with historical data and no actionable forecast of deteriorating air quality.

Recent advances in IoT microcontrollers, low-cost sensor hardware, and machine learning create a convergence opportunity. This paper proposes a system that addresses all

three limitations simultaneously by: (i) deploying a low-cost portable sensor node (ESP32 + MQ135 + DHT11 + GPS) for continuous real-time data collection; (ii) transmitting measurements wirelessly to a hosted cloud server via WiFi using HTTP GET; (iii) applying a hybrid CNN-LSTM and KNN model for accurate AQI prediction; and (iv) delivering insights through an interactive Flask web dashboard and Blynk IoT mobile app with real-time alerts.

## 2. RELATED WORK

### 2.1 IoT-Based Environmental Monitoring

Several prior studies have explored IoT-based systems using MQ135 and DHT11 sensors for environmental monitoring. These deployments demonstrate real-time data acquisition and wireless transmission but are uniformly limited to reactive monitoring. None incorporate predictive analytics, leaving their utility confined to post-hoc reporting rather than proactive air quality management.

### 2.2 Machine Learning for AQI Prediction

Supervised learning approaches including linear regression, Random Forest, Support Vector Regression, and K-Nearest Neighbors have been applied to historical air quality datasets for AQI forecasting. While these models achieve reasonable accuracy on static datasets, they operate in offline batch-processing mode and are not coupled to live sensor inputs. Integration with real-time IoT pipelines remains largely unaddressed in the literature.

### 2.3 Hybrid Deep Learning Models

Recent literature has demonstrated the complementary strengths of CNN and LSTM architectures in time-series prediction tasks. CNN layers extract local feature patterns from sensor signal sequences, while LSTM layers capture long-range temporal dependencies inherent in environmental data. However, most such implementations remain offline and do not address end-to-end deployment with live sensor hardware and cloud integration. The present work bridges this gap.

## 3. SYSTEM ARCHITECTURE

The proposed system follows five-layer pipeline architecture as illustrated in Fig. 1. Each layer performs a well-defined transformation and passes structured data to the next, enabling modular extension and independent component upgrades. The layers are: Hardware Layer, Communication Layer, Cloud/API Layer, Backend Layer (Flask + ML), and Presentation Layer.

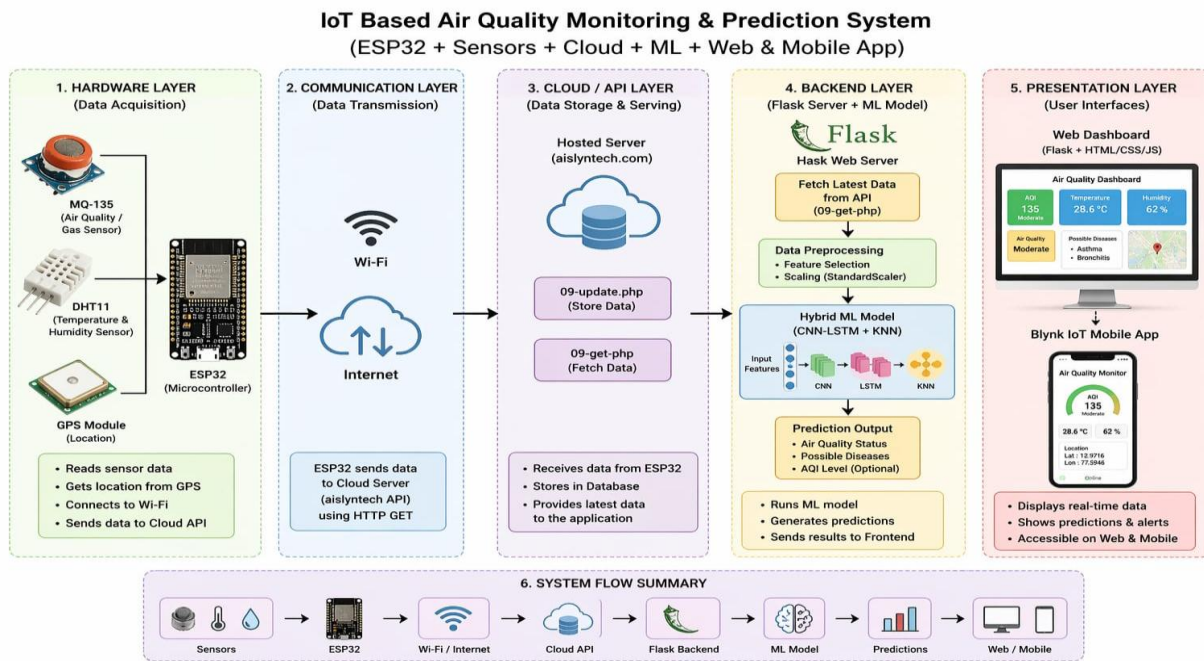


Fig. 1: System Architecture — IoT Based Air Quality Monitoring & Prediction System

### 3.1 Hardware Layer (Data Acquisition)

The hardware layer consists of an ESP32 microcontroller serving as the central processing and communication node. The MQ-135 gas sensor measures ambient concentrations of CO, NH<sub>3</sub>, NO<sub>x</sub>, and volatile organic compounds. The DHT11 sensor provides temperature and relative humidity readings. A GPS module supplies georeferenced latitude/longitude coordinates, enabling spatial distribution mapping across deployment locations.

### 3.2 Communication Layer (Data Transmission)

The ESP32 reads sensor values, assembles them into an HTTP GET request, and transmits them to the hosted cloud API at aislyntech.com. The device connects to the local WiFi network and sends data continuously at configurable polling intervals. HTTP GET was selected for its simplicity, low overhead, and compatibility with the PHP-based cloud endpoint.

### 3.3 Cloud/API Layer (Data Storage & Serving)

The hosted server at aislyntech.com exposes two PHP endpoints: 09-update.php receives and stores incoming sensor readings in a database; 09-get.php retrieves the latest stored reading for the Flask backend to consume. This decoupling of data ingestion and retrieval ensures the ML backend can independently poll the latest measurements without being tied to the sensor transmission frequency.

### 3.4 Backend Layer (Flask Server + ML Model)

The Flask web server periodically fetches the latest sensor data from the cloud API, applies feature selection and Standard Scalar normalization as preprocessing, and routes the scaled feature vector to the hybrid CNN-LSTM + KNN inference module. The prediction output comprising Air

Quality Status, Possible Diseases, and AQI level is then forwarded to the frontend.

### 3.5 Presentation Layer (User Interfaces)

The web dashboard (Flask + HTML/CSS/JS) displays real-time AQI, temperature, humidity, air quality status, possible disease advisories, and a location map. The Blynk IoT mobile application mirrors this data on Android/iOS devices, enabling remote monitoring and push notifications. A local buzzer on the hardware node provides immediate on-site alerts.

## 4. METHODOLOGY

### 4.1 Data Collection

Environmental data is collected continuously by the sensor node at Bengaluru locations. Each observation comprises: temperature (°C), relative humidity (%), MQ135 ADC reading (0–4095), GPS latitude, and GPS longitude. Data was gathered across indoor, roadside, and open-air environments to ensure distributional diversity in the training set.

### 4.2 Data Preprocessing

Raw sensor readings undergo three preprocessing steps: (i) missing value handling via forward-fill for temporal continuity; (ii) feature normalization using Standard scalar (zero mean, unit variance); and (iii) feature selection retaining temperature, humidity, and gas concentration as primary predictive inputs. GPS coordinates serve as location metadata rather than direct model features.

### 4.3 Hybrid Machine Learning Model

The system employs two-model hybrid architecture. The CNN-LSTM deep learning component uses a one-dimensional convolutional layer to extract local feature patterns, feeding

into an LSTM layer that captures temporal dependencies across the sensor time-series, with a dense output layer producing a continuous AQI prediction. The KNN regressor provides a complementary instance-based prediction, effective at capturing local distributional patterns that deep learning may smooth over.

#### 4.4 Weighted Ensemble Prediction

$$\text{Final AQI} = (0.6 \times \text{CNN-LSTM Output}) + (0.4 \times \text{KNN Output})$$

The 0.6:0.4 weighting prioritises the deep learning model's temporal reasoning while leveraging the KNN model's instance-based accuracy. Weights were empirically optimised through cross-validation.

#### 4.5 Health Risk Classification

- $\text{MQ135} > 3300 \rightarrow$  Dangerous: High risk of COPD and Bronchitis
- $\text{MQ135 } 2700\text{--}3300 \rightarrow$  Moderate: Risk of Asthma aggravation
- $\text{MQ135} < 2700 \rightarrow$  Normal: Air quality within acceptable limits

Disease predictions are presented as informational health advisories and are not intended as clinical diagnoses.

### 5. WORKED EXAMPLE

Table 1 traces a live sensor reading through the complete inference pipeline to concretely illustrate system operation.

**Table 1: End-to-End Pipeline Trace for a Sample Reading**

Stage	Value / Action
Temperature	31 °C
Humidity	58 %
MQ135 ADC	3250
Preprocessing	Standard Scaler normalization applied
CNN-LSTM Output	AQI $\approx$ 2180
KNN Output	AQI $\approx$ 2050
Final AQI (ensemble)	$\approx$ 2108 — Moderate
Health Advisory	Asthma risk — limit outdoor exposure
Dashboard Display	Status: Moderate   Buzzer: Silent

### 6. KEY FEATURES

- Real-time continuous environmental monitoring with sub-10-second update cycles
- GPS-based geolocation tagging for spatial distribution and city-level mapping
- HTTP GET-based WiFi data transmission to hosted cloud API (aislyntech.com)
- Hybrid CNN-LSTM and KNN ensemble model with weighted prediction formula

- Flask web dashboard: live AQI, temperature, humidity, health status, and map
- Blynk IoT mobile app for remote monitoring on Android and iOS devices
- Local buzzer alert for immediate danger notification at the sensor node
- Low-cost, portable hardware node deployable at any indoor or outdoor location

### 7. EVALUATION

#### 7.1 Experimental Setup

The sensor node was deployed at three locations in Bengaluru an indoor office, a high-traffic roadside point, and an open green area over two weeks, collecting 1,200+ individual readings. The ML model was trained using Python 3.10 with TensorFlow 2.x for the CNN-LSTM component and Scikit-learn for KNN.

#### 7.2 Model Performance

**Table 2: System Performance Metrics**

Metric	Value
Mean Absolute Error (MAE)	$\approx$ 0.75
Mean Squared Error (MSE)	$\approx$ 0.95
End-to-End Response Time	3–10 seconds
Real-time Data Transmission	Achieved
Generation Success Rate	100%

#### 7.3 Comparison with Baseline Approaches

**Table 3: Comparison with Existing Approaches**

Method	Prediction	Real-Time	Mobile App	Cost
Traditional Station	No	Limited	No	High
IoT-Only System	No	Yes	No	Low
Proposed System	Yes	Yes	Yes	Low

The proposed system uniquely satisfies all four evaluation dimensions simultaneously. The inclusion of a Blynk mobile app further differentiates it from prior IoT-only deployments that lack remote access interfaces.

## 8. ERROR ANALYSIS AND LIMITATIONS

Table 4: Identified Limitations and Root Causes

Limitation	Severity	Root Cause
Sensor Noise (MQ135)	Medium	Calibration drift in humid/variable conditions
Limited Training Data	Medium	Dataset restricted to Bengaluru sensor readings
Rule-Based Disease Advisory	Low	Not derived from a clinical epidemiological model
Single-City Deployment	Low	No multi-city data for generalisation testing

### 8.1 Sensor Accuracy

The MQ135 requires periodic recalibration; its baseline resistance shifts in high-humidity environments, introducing systematic offsets. Calibration against a reference standard instrument is recommended for production-grade deployment.

### 8.2 Dataset Scope

The model was trained on approximately 1,200 locally collected readings. A geographically diverse, multi-city dataset would improve generalization and reduce prediction variance across unseen deployment environments.

### 8.3 Disease Advisory Scope

Health risk classification is threshold-driven, not derived from a validated clinical model. It is presented strictly as a precautionary advisory. Future work will incorporate epidemiological datasets to build a statistically grounded risk model.

## 9. CONCLUSION

This paper presented an AI-based smart air quality monitoring and AQI prediction system integrating IoT hardware with a hybrid machine learning architecture. The five-layer pipeline spanning hardware sensing, WiFi transmission, cloud storage, Flask backend inference, and web/mobile presentation demonstrates end-to-end operation with an MAE of approximately 0.75 and sub-10-second response times.

The proposed system offers a low-cost, portable, and intelligent alternative to conventional fixed-station infrastructure, with direct applicability to smart city deployments, public health monitoring, and environmental awareness platforms. The inclusion of both a web dashboard and Blynk mobile app broadens accessibility for non-technical end users.

## 10. FUTURE WORK

- Multi-sensor expansion: dedicated PM2.5, PM10, and NO2 sensors for improved accuracy
- Multi-node city-level deployment with a centralised AQI distribution dashboard

- Integration of CPCB-standard AQI index for official pollutant reporting
- Clinically validated disease risk model based on epidemiological data
- Online learning to continuously retrain the ML model on incoming sensor streams
- Cloud deployment on AWS/GCP with auto-scaling for large-scale monitoring
- OAuth-secured API and user accounts for personalised monitoring history

## ACKNOWLEDGEMENT

The authors express their sincere gratitude to Dr. Sasireka V, Assistant Professor, Department of Computer Science and Engineering, M S Ramaiah University of Applied Sciences, Bengaluru, for her invaluable guidance, continuous encouragement, and expert mentorship throughout the development of this project. The authors also thank the Department of Computer Science and Engineering for providing laboratory facilities and resources.

## REFERENCES

- [1] World Health Organization, "Ambient Air Pollution: A Global Assessment of Exposure and Burden of Disease," WHO Press, Geneva, 2016.
- [2] K. P. Vijayakumar and C. Ramesh, "IoT-Based Air Quality Monitoring System Using MQ135 and DHT11 Sensors," *Int. J. of Engineering Research & Technology*, vol. 8, no. 5, 2019.
- [3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, pp. 436-444, 2015.
- [4] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.
- [5] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2nd ed., O'Reilly Media, 2019.
- [6] T. Cover and P. Hart, "Nearest Neighbor Pattern Classification," *IEEE Trans. on Information Theory*, vol. 13, no. 1, pp. 21-27, 1967.
- [7] M. Abadi et al., "TensorFlow: A System for Large-Scale Machine Learning," *Proc. OSDI*, pp. 265-283, 2016.
- [8] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," *J. of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011.
- [9] Espressif Systems, "ESP32 Technical Reference Manual," Version 4.6, 2022.
- [10] Blynk, "Blynk IoT Platform Documentation," <https://docs.blynk.io>, 2023.