National College of Ireland

Hybrid models Cloud-based Enhancements for Air Quality Prediction Systems

MSc Research Project
MSc CLoud Computingh

Chinmay Dhanawade
Student ID: X23200197

School of Computing
National College of Ireland

Supervisor: Prof. Sudarshan Deshmukh

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Chinmay Dhanawade
Student ID:	X23200197
Programme:	MSc in Cloud Computing
Year:	2024
Module:	MSc Research Project
Supervisor:	Prof. Sudarshan Deshmukh
Submission Due Date:	20/12/2024
Project Title:	Title
Word Count:	5952
Page Count:	22

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	CHINMAY
Date:	27th January 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Hybrid models Cloud-based Enhancements for Air Quality Prediction Systems

Chinmay Dhanawade
X23200197

Abstract

This paper analyzes the creation and implementation of a hybrid model for air quality forecasting that incorporates statistical and machine learning techniques within the cloud computing paradigm. A major goal was to use the SARIMA (Seasonal AutoRegressive Integrated Moving Average) and Random Forest models with the aim of increasing predictive accuracy and reliability. First, LSTM could not be introduced to the initial trials in Cloud9 as the tool didn't have TensorFlow -a prerequisite for running and integrating LSTMs into one model. Key results thus conclude that the SARIMA combined with Random Forest has indeed been deployed in the cloud, as the program developed in Google Colab proved successful. The deployment process involved linking the application to GitHub, which triggered a pipeline that facilitated the deployment to Elastic Beanstalk. Such a process underscores the role of cloud computing in dealing with large datasets and enabling real-time data processing for air quality predictions. The study points out the critical role of efficient model integration and cloud infrastructure in dealing with environmental challenges. This research contributes to the field of environmental monitoring by demonstrating that traditional statistical models can be effectively combined with machine learning techniques. The results suggest that such hybrid approaches can significantly improve decision-making related to public health and environmental policy, ultimately fostering more sustainable urban development practices.

1 Introduction

Air pollution is one of the greatest global challenges, affecting not only public health and environmental quality, but also quality of life. Damages caused by polluted air include respiratory diseases, cardiovascular disease, and higher mortality rates. In addition, air pollution leads to climatic change, acid rains, and ecosystem degradation. This is the reason why proper air quality forecasts are vitally important since these negative effects increase with urbanization and industrialization. Traditional techniques of air quality forecasting involve statistical analyses of past records, which might lack the accuracy and responsiveness inherent in their nature.

1.1 Background and Motivation

The inspiration for this study comes from the immediate need to integrate new technologies into methodologies to improve the predictability of air quality. Traditional statistical

models are very useful but lack the capacity to describe complexities and nonlinearities within air quality data. In this respect, such a limitation has been a motivating factor in looking towards hybrid approaches that blend traditional statistical techniques with machine learning algorithms. This paper focuses on the utilization of cloud computing capabilities to develop a strong framework for real-time data processing and accurate predictions. The biggest advantage in scalability, flexibility, and resource management comes from cloud computing. Many data will need to be processed quickly to maintain accurate and timely updates in a changing environment. By bringing machine learning models into the mix, this system could achieve an increase in prediction accuracy based on the discovery of patterns within the data that the standard methods cannot determine.

1.2 Objectives of the Report

- To develop of a hybrid model, merging statistical methods (SARIMA) with machine learning techniques (Random Forest) to predict higher accuracy in air quality.
- To be deployed on a cloud computing site to realize scalability and real-time processing of data.
- To compare the performance of the hybrid model against the regular forecasting methods based on its effectiveness.
- To enable proper analysis of the challenges encountered during model integration and deployment in the cloud environment.

1.3 Research Problem

Though considerable advancements in the field of cloud computing as well as machine learning have been achieved, its suitable application for an air quality prediction system is still missing. Current methods fail to truly utilize the potential of hybrid models, relying on statistical modeling combined with ML techniques.

The main research question that guides this study is: How can a hybrid model that incorporates statistical methods and machine learning algorithms be designed and tested to improve the predictability of air quality and its reliability with the help of cloud computing systems? This research addresses this very question by developing a scalable solution with improvements in the predictive accuracy and reliability of real-time environments.

1.4 Importance of the Research

The significance of the study is that it can provide enlightenments on better mechanisms of government, organizations and people in making better decision on air quality management issues. This could facilitate timely intervention, and in consequence minimize the health and environmental degradation risks associated with poor air quality. Cloud computing provides better computing power to enhance air quality prediction and very little work has been done on integrating the cloud with efficient hybrid models for air quality prediction. The objective of this paper is to fill this gap by proposing a solution that combines traditional statistical models and modern machine learning techniques simultaneously optimized in a cloud environment. Advances in cloud technologies over the past few years is an optimal base for handling massive data about air quality. Yet

there has been very little research into capturing these advances for use in environmental applications. These include ARIMA, Random Forest and LSTMs and propose to combine these powers collectively under a cloud environment with better precision and more reliability in forecasting air quality.

1.5 Structure of the Document

The document structure includes:

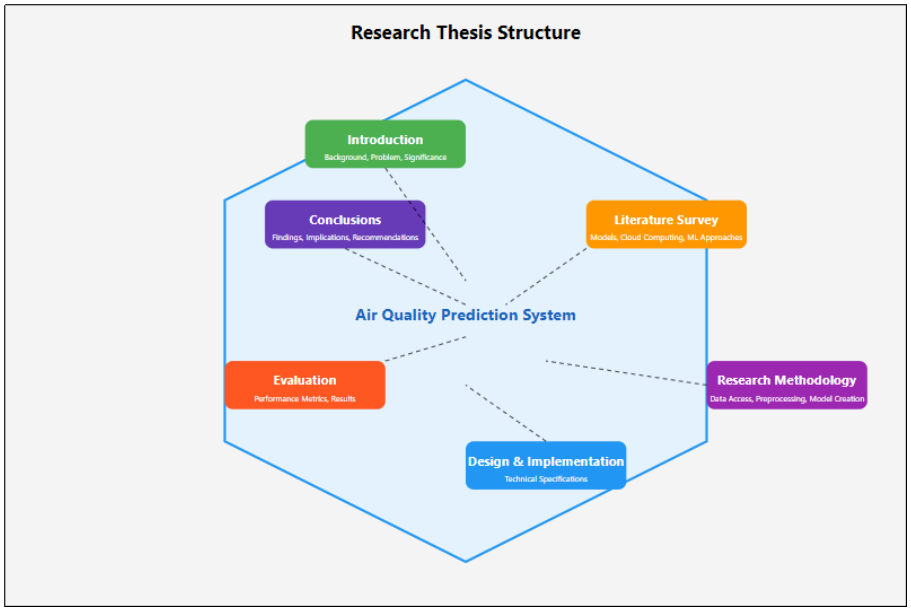

Figure 1: Structure of the Document

- **Introduction:** Background, problem of research, significance, research questions, proposed solution, contributions, and outline of the thesis.
- ***Literature Survey:*** Air quality prediction models, literature, and relevance of cloud computing in data analytics, integrating machine learning with traditional statistical approaches.
- ***Research Methodology:*** Research approach, data access strategies, preprocessing of data, process of model creation, and integration with the cloud ecosystem.
- ***Design and Implementation Specification:*** This section will outline the technical specifications for the development and deployment of models.
- ***Evaluation:*** Presents the performance evaluation metrics with the results of the hybrid model testing.
- ***Conclusions and Discussion:*** Concludes the findings and implications along with recommendations for future studies.

This structured Figure 1 approach ensures coverage of all aspects related to the development and implementation of a hybrid model for air quality prediction within a cloud computing framework.

2 Related Work

Air quality forecasting research is important because it affects human health and environmental sustainability directly. Many ways have been applied to forecast air quality: from classical statistical methods to modern machine learning approaches. To show us how, this section reviews the popular approaches of air quality forecasting, discusses the significance of cloud computing in data analytics, followed by present research gap and expected contributions, and ultimately, literature gap and conceptual framework of this study.

2.1 Machine Learning Approaches

Machine learning approaches have recently been more prominent for air quality prediction due to the ability to model complex interactions and patterns within data. Better performance has been observed when using Random Forest and Gradient Boosting algorithms over conventional methods in estimating air pollutants, such as PM2.5 Liu et al. (2024). These models use massive datasets and can capture non-linear relationships. However, they are not an exception to problems; things like overfitting and massive computational needs may limit their practice Subramaniam et al. (2022). Last but not least, for sequential data analysis, LSTMs, which actually are a subset of RNNs, have been found to serve as a very powerful tool. LSTMs capture well long-time dependencies in the time-series data, so they seem suitable for forecasting of air quality wherein the temporal dependencies are relatively important)Mampitiya et al. (2023). Introducing an LSTM with statistical models leads to a strong predictive system, since these two have the strengths from each method combined Cheng et al. (2014)

2.2 Hybrid Models: Cloud-based Enhancements for Air Quality Prediction Systems

2.2.1 Cloud 9 and Pipeline Integration

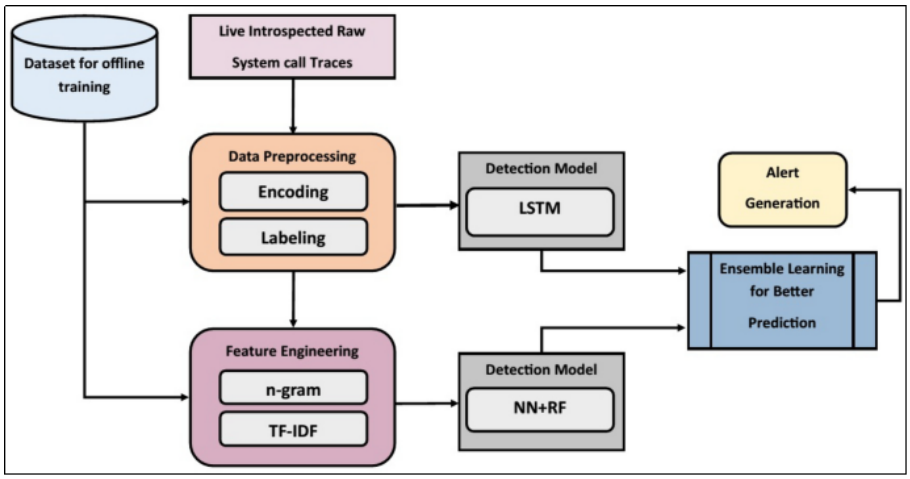

Figure 2: **Cloud 9 and Pipeline Integration on ML (Source: Ansari, and Alam, 2024)**

Together with AWS Elastic Beanstalk, the Cloud 9 can be used in order to enhance the deployment of hybrid models for air quality prediction. A strong IDE for application development in a vast collection of programming languages, including Python, Cloud 9. The development of these applications would then allow users to interact with predictive models Rahman et al. (2024), with interfaces to Flask. Cloud 9 makes coding and debugging simpler too. This therefore allows a developer to easily build and test the application before it is deployed for use. When the Flask application is built it can be merged with GitHub for version control and a collaborative version control. Through this integration one off workflow streamlines, and any change to the codebase can be easily tracked and managed. Once the GitHub repository was set up, a CI/CD pipeline can be

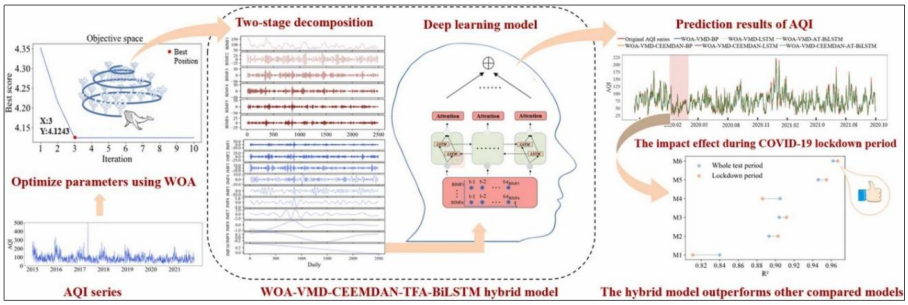

Figure 3: **A hybrid model for daily air quality index prediction** (Source: Ansari, and Alam, 2024)

setup Ansari and Alam (2024). This pipeline typically consists of two stages: It's two stages, the source stage that pulls from GitHub, the deploy stage that pushes the application to AWS Elastic Beanstalk. AWS Elastic Beanstalk is a highly scalable, easy to use service to deploy web applications and services of any size. It deploys automatically, starts with capacity provisioning, load balancing, auto scaling to maintaining application health. In addition, model artifacts are stored in background services like Amazon S3, or Simple Storage Service. Necessary scalable computing resources to run complex models is provided by EC2, Elastic Compute Cloud Uppal et al. (2022). Not only does this architecture facilitate more efficient hybrid model deployment, it also retains real time data processing ability in order to get more accurate air quality predictions.

2.3 Role of cloud computing in DATA analysis

2.3.1 Scalability and Flexibility

Cloud computing is having an impact on the data analytics , which has made an ideal scalable and flexible data infrastructure for processing large data. Services of AWS, Microsoft Azure and Google Cloud include auto scaling features that scale resources according to the demand Kaginalkar et al. (2021). For example, if the data comes from IoT sensors and meteorological data, you need to process the data in real time to predict air quality, and this is very useful. As opposed to physical infrastructure, resources can be scaled horizontally or vertically by researchers to assure availability of computation resources at the time they are needed Shah et al. (2020) . Correspondingly, it makes it possible to predict when to do something to achieve good air quality management, at the right time.

2.3.2 Real-Time Data Processing

Real time data ingestion and analysis are key towards accurate air quality forecasting. Platfoms that provide cloud computing have powerful solutions to quickly process any real time changes in the environment, so immediate reaction with those changes is possible. Real time data flow is managed by services such as AWS Lambda and Azure Stream Analytics, both of which play a key role in monitoring sudden spikes in pollution levels Singh et al. (2021). Real time air quality (such as at a traffic intersection) is updated Iskandaryan et al. (2020) which enables increased effectiveness of predictive models, as we can study high speed data streams.

2.4 Research Niche and Expected Contribution

Although there are already advanced stage hybrid modeling approaches combining statistical methods with machine learning techniques, the actual use of these in the design of air quality prediction systems is still a significant gap. Based on this, this research will build a hybrid cloud model that combines both the statistical model such as ARIMA and the more advanced machine learning algorithms like Random Forest and LSTM to increase the accuracy and reliability of air quality forecasts Chang et al. (2020). Through providing scalability and real time processing of data using cloud computing infrastructure, the research addresses. The expected contributions are:

- Development of a hybrid modeling approach that improves predictive effectiveness.
- Establish a scalable infrastructure that can scale with data from multiple sources.
- Creation of a replicable framework applicable to various environmental monitoring applications.

Natarajan et al. (2024) Sokhi et al. (2021)

2.5 Literature Gap

While many studies have focused on single methodologies for air quality prediction, relatively few have successfully pursued models that bridge the gap between cloud computing with hybrid modeling approaches statistically and machine learning techniques. There still lacks a comprehensive framework to discuss technical challenges in integrating the model and their practical implications on real-time air quality monitoring Mahbub et al. (2020).

This research aims at filling this gap by putting forward a systematic approach that utilizes cloud technologies in combination with hybrid strategies for modeling. It thus aims at providing actionable insights into improving air-quality predictions, while keeping in mind that scalable solutions shall be very key to environmental monitoring

2.6 Conceptual Framework

The conceptual framework guiding this research is illustrated in Figure 4
This framework highlights the interconnectedness of each component in achieving accurate and reliable air quality forecasts while taking advantage of the capabilities of cloud computing.

Author(s)	Year	Research Method	Used Models
Liu, Q., Cui, B., Liu, Z.	2024	Machine Learning and Secondary Modeling	LSTM, AQI Prediction
Subramaniam, S., Raju, N., et al.	2022	Narrative Review	Various AI Technologies
Natarajan, S.K., Shanmurthy, P., et al.	2024	Optimized Machine Learning	Grey Wolf Optimization (GWO), Decision Tree
Sokhi, R.S., Moussiopoulos, N., et al.	2021	Review of Current Challenges in Air Quality	Not applicable
Mampitiya, L., Rathnayake, N., et al.	2023	Machine Learning Techniques	Various ML Techniques
Rahman, M.M., Nayeem, M.E.H., et al.	2024	Predictive Machine Learning Model	AirNet (ML Framework)
Ansari, M., Alam, M.	2024	IoT-Cloud-Based Time-Series Analysis	Univariate Time-Series Analysis
Zhang, Q., Han, Y., et al.	2022	Hybrid Deep Learning Framework	CNN-LSTM
Cheng, Y., Li, X., et al.	2014	Cloud-Based Air Quality Monitoring System	Not specified
Uppal, M., Gupta, D., et al.	2022	Cloud-Based Fault Prediction	Machine Learning
Kaginalkar, A., Kumar, S., et al.	2021	Review of Urban Computing in Air Quality	IoT, AI and Cloud Technologies
Singh, D., Dahiya, M., et al.	2021	Review of Sensors and Systems	Not applicable
Chang, Y.S., Abimannan, S., et al.	2020	Ensemble Learning Based Hybrid Model	Ensemble Learning
Iskandaryan, D., Ramos, F., Trilles, S.	2020	Review of Machine Learning Technologies	Not applicable
Goh, C.C., Kamarudin, L.M., et al.	2021	Real-Time In-Vehicle Air Quality Monitoring	Machine Learning Prediction Algorithm
Shah, S.K., Tariq, Z., et al.	2020	Real-Time Machine Learning for Environmental Detection	Not specified

Table 1: Related work

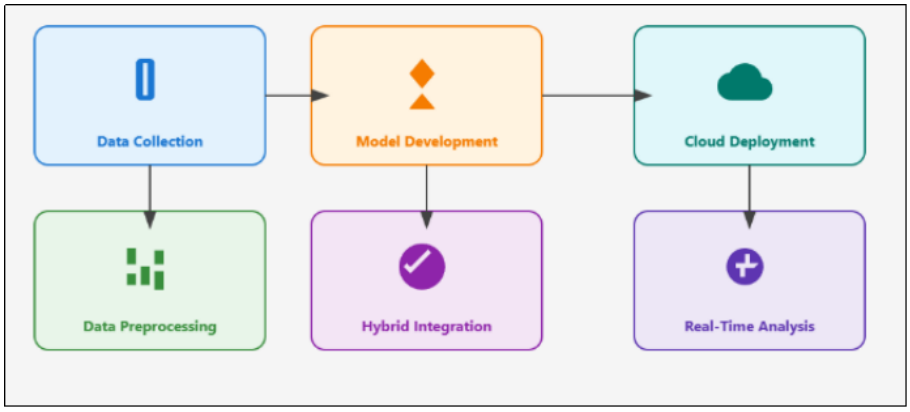

Figure 4: **Conceptual Framework**(self created)

2.7 Summary

In summary, it would present the evolution of methodologies in air quality forecasting from traditional statistical approaches to advanced machine learning techniques that make use of cloud computing: Due to the need to enhance the scalability of the system so that real time data processing required for effective environmental monitoring can be achieved. As a way of filling those gaps left by existing literature and making a major contribution to the discipline of air quality prediction, a hybrid cloud-based development of model is developed.

3 Research Methodology

In this section, this study discusses the methodology used in employing a hybrid model to estimate air quality by merging statistical method and machine learning techniques. This methodology is subdivided into five major components, which are: data collection methods, research approach, analytical technique, cloud integration and ethical considerations.

3.1 Research Approach/Design

The study is statistically oriented toward the analysis of numerical data, and hence increases the accuracy and reliability of its air quality prediction. This type of research approach helps to critically evaluate all connections of these factors based on the impact it has on the AQI. The model designed for this type of statistical study would take into account both the historical datasets as well as the real-time data stream and be further validated in terms of statistical metrics. It allows for systematic experimentation with different modelling techniques, which otherwise would be hard to do objectively.

3.2 Data Collection Methods

Data collection is incorporated into this research. This study used various sources in gathering comprehensive data regarding the air quality. There are two primary methods:

- Govt databases: Historical air quality data can be found in national environmental agencies, which provide reliable and standardized datasets.

- API : The different IoT sensors data in different urban areas is received via API, and this API gathers real-time data of pollutants such as CO, NO2, O3, SO2, PM2.5, and PM10.

Then, data collected undergoes rigorous preprocessing concerning missing values, outliers, and inconsistencies to ensure integrity and quality of the dataset prior to fitting a model to it.

3.3 Analytical Techniques

3.3.1 ARIMA Model Implementation

The first analytical method applied being the ARIMA model which stands for AutoRegressive Integrated Moving Average. It is one of the most commonly used classical statistics statistical time series forecasting methods. The data have linear trends and stationary characteristics so it's appropriate. Our preconditioning must include log transformations and differencing, to induce stationarity, and thus, the model becomes more powerful at predicting actual air pollution levels.

3.3.2 LSTM Model Implementation

The other type of analytical technique is that of LSTM models. It is a recurrent neural network that is specifically fit for sequential inputs with long-term dependencies since it captures non-linear relationships between the variables. Such models make it possible to give accurate predictions about the indicators of air quality. In LSTM run experiments to find out what effect different variables would have on the prediction of AQI. The accuracy and reliability of the models in air quality level prediction is verified with common error metrics, MSE and RMSE

3.3.3 Random Forest Model Implementation

The second analytical approach uses the Random Forest model, but constraints using LSTM due to the lack of requirements for TensorFlow in Cloud 9. Random Forest is an ensemble method for learning, training multiple decision trees and outputting the vote of these prediction through classification or average on regression. Because it can highly capture complex interactions of variables and is able to predict non linear relationships in the air quality data, this model is very effective. Finally, two models are tested on common error metrics that measure their performance - Mean Squared Error (MSE) and Root Mean Square Error (RMSE) which approximate their accuracy and reliability in forecasting air quality levels.

3.3.4 Cloud 9 and Pipeline Integration

Cloud 9 is an integrated development environment that allows you to code either in the language of your choice or develop predictive models. After training the Jupyter Notebook files (.ipynb), we can then save the models out to pickle files, making it easy for us to retrieve and deploy the models. That said, I train the models and develop a Flask application in Cloud 9 to serve as the user interface for the predictive models. Finally, we link this application to GitHub, to be able to version control and also collaborate. A CI/CD pipeline with two main stages :

Source: Pulls code from GitHub.
Deploy: It will deploy the application into AWS Elastic Beanstalk.
But this pipeline completely streamlines it in a way that any codebase changes you'd make will be reflected in the actual deployed application pretty effortlessly. Scaling and performance is also enabled during model execution by using background applications such as Amazon S3 as storage and EC2 instances as computing power.

3.3.5 Online Forecasting Capabilities

Evaluation Metrics Online forecasting and deployment of these models for real time air quality prediction is a crucial part of this work. Then after training the models, save them as a pickle file (. pkl), This develop a Flask application on Cloud 9 to provide a user interface to the prediction models.

This practical implementation using Random Forest provides a hands-on approach to learn how machine learning can be used to analyse the air quality data, predict and finally visualize the results for users! Not only this, it also helps the users in understanding the current air quality conditions better.

3.3.6 Evaluation Metrics

To assess the performance of the model's hybrid developed in this study, several evaluation metrics are used:

- Mean Squared Error (MSE) represents the average squared difference between predicted values and how actual values are. The smaller MSE, the better the model will fit.
- Root Mean Square Error (RMSE): It defines the RMSE as the square root of MSE and it is the measure of how predicted values fit the observed values. This is useful for interpreting errors as units of the predicted variable.
- r-squared is one of the statistical measurement of how well data points can be fitted to a statistical model, how much of the variance of the dependent variable can be explained by the independent variables. Higher the value, the better the value of model.

As the metrics above will be important in comparison of performance of ARIMA and LSTM models individually and hybrid combination of the two.

3.3.7 Ethical Considerations

For this research the ethics about data privacy and transparency would be key considerations. The ethics underlying the present study are guided by the following:

- Data Privacy: The data captured from IoT sensors and government databases will be anonymized, all data accumulated, to protect individual privacy.
- Transparency: The research process will be documented in such a way that replicability and accountability can be achieved.
- Compliance: It complies with the study of the legal regulations about practices concerned the environmental monitoring and data usage.

This research discusses these ethical considerations with the hope that their values are maintained while still producing valuable insight in methodologies that predict air quality. In a nutshell, this is a well structured research method in articulating the development of such a hybrid model for air quality prediction. It suggests different Statistical methods combined with Machine learning techniques in cloud computing framework. Data will be gathered systematically for predictive accuracy and reliability for monitoring air quality at correct levels by strict analytical techniques, efficient use of the cloud integration and ethically.

4 Design and Implementation Specifications

This section provides design and implementation requirements for an air quality prediction system that hybridizes statistical methods with approaches of machine learning techniques. This section focuses on architectural design, tools, technological use, implementation details of the system, and the key challenges that arose in actual development.

4.1 System Architecture or Conceptual Model

The overall system design is constructed to optimize data processing efficiency, along with model training, and high-performance real-time predictions concerning air quality levels. Therefore, this architecture can be generally divided into the following sections:

4.1.1 Data Ingestion Layer

The data ingestion layer is responsible for collecting and preprocessing air quality data from multiple sources. These include:

- Government Databases: The historical datasets are fetched from environmental agencies.
- API: Real-time data is fetched from different API.

Data preprocessing such as cleaning, normalization, and transformation is implemented with data quality is ensured before feeding it into the model.

4.1.2 Model Development Layer

In this layer, the following two models are developed:

- SARIMA Model is the class of classical statistical time-series forecasting models, which model linear trends in stationary datasets.
- Random Forest Model: An ensemble learning method that efficiently accommodates non-linear relationships and interactions amongst the variables.

Both models are trained on historical data to predict future air quality levels, and then the trained models are serialized into pickle files for deployment. ALong with these two models other models were also tried which are as follows:

1. ARIMA
2. BI-LSTM
3. LSTM

4.1.3 Application Layer

The application layer comprises a Flask application developed on Cloud 9. The application acts as the user interface to interact with the predictive models. Users can input real-time environmental data and get immediate predictions regarding the air quality levels.

4.1.4 Cloud Deployment Layer

The cloud deployment layer uses AWS Elastic Beanstalk to host the Flask application. It consists of:

- AWS S3: Storage for model artifacts and datasets.
- AWS EC2 Instances supply on-demand scalable computing resources for running complex models.
- CI/CD Pipeline: A pipeline in continuous integration and deployment, which automates the process of upgrading the application any time there is a change in the GitHub repository.

4.1.5 Architectural Diagram

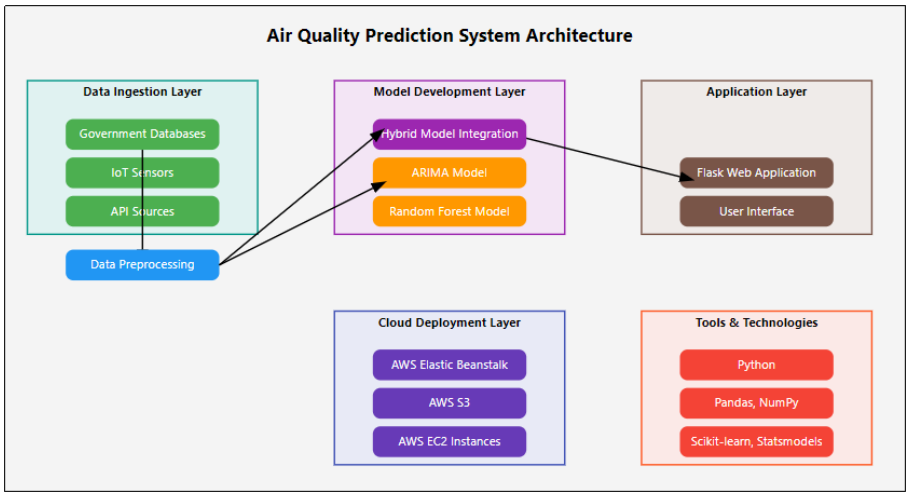

Figure 5: **Architectural Diagram of AQ Prediction(self-created)**

4.1.6 Tools and Technologies Used

Following is the reason for choosing appropriate tools and technology for developing a robust air quality prediction system:

- Cloud Platform: The choice here was AWS since it gives a comprehensive set of services that support scalable data storage, processing, and deployment.
- Programming Language: Python would be used due to large libraries for data analysis, machine learning, and web development.

Libraries:

1. Pandas
2. NumPy.
3. Scikit-learn: To implement some basic machine learning algorithms, specifically Random Forest.

- Statsmodels: For ARIMA models implementation.
- Flask: Building for the web application's graphical user interface.
- Data Visualization Tool: Matplotlib and Seaborn were used in terms of visualizing output by analyzing trends in air quality.

5 Implementation Details

The implementation process took the following steps to have a systematic approach to the development of the hybrid model:

5.1 Data Collection

Data was sourced from various sources, which included government databases, and API. This diverse dataset provided an overall view of air quality indicators over time.

5.2 Data Preprocessing

The data was gathered and then pre processed for missing values, outliers, and inconsistencies. To ensure that all features contributed equally to training the model i used normalization techniques.

5.3 Model Development

SARIMA Model Development

- The SARIMA model was developed using historical air quality data with seasonality. This included:
- Checking for stationarity using statistical tests (e.g., ADF test) and seasonal decomposition.
- Applying transformations (e.g., seasonal differencing) to enforce stationarity.
- Implementing SARIMA using optimal parameters `(p, d, q) × (P, D, Q, s)` achieved via grid search.

ARIMA Model Development

- The ARIMA model was developed using historical air quality data. This included:

- Checking for stationarity with statistical tests (e.g., ADF test).
- Transformations (e.g., differencing) to enforce stationarity when necessary.
- Implementing the ARIMA model using the best-fit parameters that have been achieved via grid search.

Development of Random Forest Model

- Implementation of the Random Forest model as follows:
- Split the dataset into a training and test set.
- Train the model on the training data by cross-validation of the hyperparameters.
- Evaluation of the model in terms of metrics such as MSE and RMSE.

5.4 Hybrid Model Integration

```
def predict():
    # Get user input
    future_days = int(request.form['future_days'])

    # Fetch historical data (replace with API call or preloaded data)
    data = {
        'Date': pd.date_range(start='2023-01-01', periods=100, freq='D'),
        'AQI': np.random.randint(50, 150, size=100)
    }
    historical_df = pd.DataFrame(data)
    historical_df.set_index('Date', inplace=True)

    # SARIMA Predictions
    sarima_forecast = sarima_model.get_forecast(steps=future_days)
    sarima_predictions = sarima_forecast.predicted_mean

    # Regression Predictions
    regression_predictions = []
    time_steps = 10
    last_sequence = historical_df['AQI'].values[-time_steps:]

    for _ in range(future_days):
        next_prediction = regressor.predict(last_sequence.reshape(1, -1))
        regression_predictions.append(next_prediction[0])
        last_sequence = np.append(last_sequence[1:], next_prediction)

    # Hybrid Predictions
    hybrid_predictions = (sarima_predictions.values + np.array(regression_predictions)) / 2

    # Future Dates
    future_dates = pd.date_range(start=historical_df.index[-1] + pd.Timedelta(days=1), periods=future_days)
```

Figure 6: Models prediction

A hybrid approach was adopted combining predictions from both models. Final prediction was determined by averaging or weighting predictions based on model performance during validation.

5.5 Deployment in Cloud

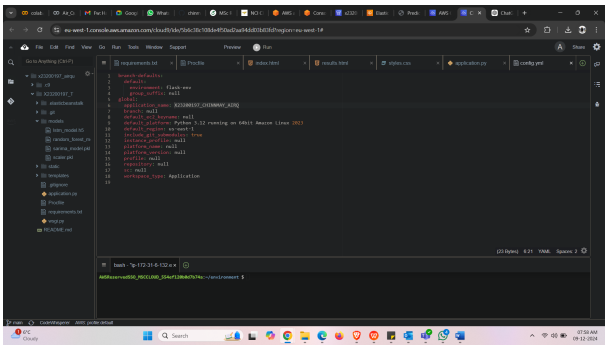

Figure 7: Integration with git

The trained models were serialized to pickle files (.pkl). Using Cloud 9, a Flask application was built for serving input from users containing real-time data and retrieving the output predictions. It integrates into GitHub for the purpose of version control. The created pipeline has two stages namely:

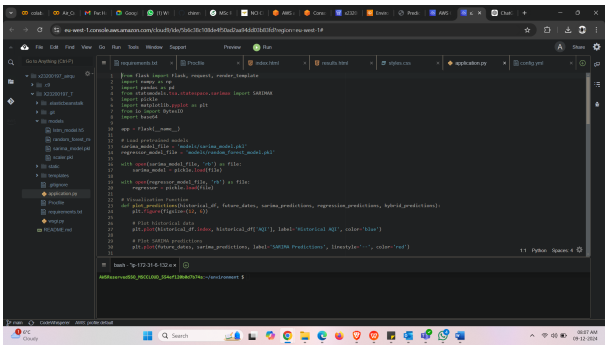

Figure 8: Flask application Deploy on cloud9

- Source and Deploy; Source: pull from GitHub code and Deploy.
- deploys the application into AWS Elastic Beanstalk.
- Testing and Validation: Deploying the model, historical and real-time inputs from the sensors of IoT were fed in to test the correctness of predictions generated by the hybrid model.

5.6 Challenges and Limitations

Issues Faced During the Implementation Phase

- Resource limitation: Attempting to deploy the LSTM alongside SARIMA is challenging when the resource needed by TensorFlow is more than what can be accommodated by Cloud 9, thus turning towards only deploying SARIMA and Random Forest models.
- Data quality problems: Consistent or missing data points were found demanding significant efforts for preprocessing.

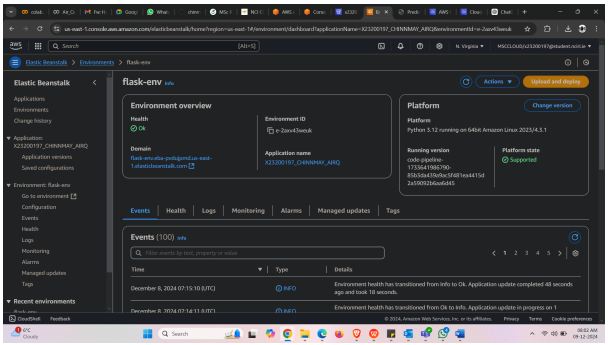

Figure 9: Domain it redirects to the application

- Integration Complexity: This integration of various components like data ingestion, model development, and cloud deployment proved complex in terms of smooth flow of data and interaction among layers.

Alternative modeling techniques were considered if resource constraint was the issue that didn't allow for initial planning.

- Rigorous protocols of preprocessing ensured the data integrity.
- Continuous testing in all integration phases made sure the issues are detected at early stages of the process.

In summary, this chapter gives a comprehensive design and implementation specification of how a hybrid air quality prediction system can be developed with the help of cloud computing technologies. The present research is trying to increase the accuracy of prediction with regard to real-world problems that are faced in air quality monitoring by making use of statistical methods combined with machine learning techniques within a strong cloud infrastructure.

6 Evaluation

The validation of the air quality prediction models developed in this study are presented in this section. Metrics for describing model performance are evaluated, the model results are displayed and contrasted with available solutions in this domain.

6.1 Criterias and Evaluation Metrics

A set of key metrics were used to measure the success of the air quality prediction models. These are the quantitative terms for judging models performance and reliability.

- Root Mean Square Error: One of the most commonly used metrics in this study is the Root Mean Square Error. RMSE is a measure of the average magnitude of error between actual observations and predictions.

It is calculated as:

$$RMSE = 1ni = 1n(yiyî)2$$

yî are the predicted values, yi are actual values, n is a number of observations. The value of RMSE lower means that the model has performed better.

- Mean Absolute Error (MAE)

Another important measure expressing average absolute difference between predicted and actual in Mean Absolute Error (MAE). It can be calculated by the formula:

$$MAE = 1ni = 1nyiyî$$

RMSE and MAE are both excellent supplements, each giving the mean error magnitude without considering the sign, so they can be used together to quantify completeness.

- R-squared Value

The level of R-squared tells us how good the independent variables are in explaining the dependent variable. It varies between 0 and 1, the higher the value, the better the fit. R-squared is calculated as:

$$R2 = 1i = 1n(yiyî)2i = 1n(yiy)2$$

y, this is the average of actual values. This is used as a metric for evaluating to what extent models explain AQI variance

6.2 Results and Analysis

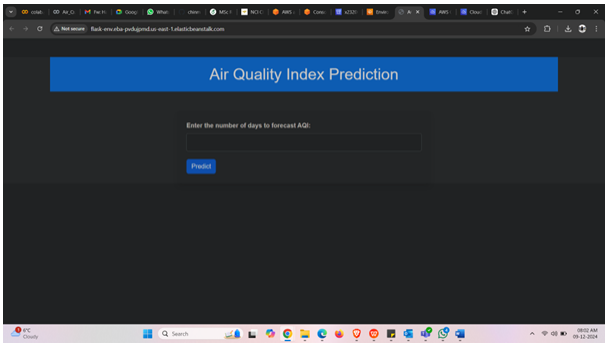

Figure 10: Web Application Interface

The web application called "Air Quality Index Prediction" contains an input field wherein the user will type the number of days forward in which they would want to forecast AQI levels. The user inputs their preferred forecasting horizon into a text box. A "Predict" button serves as the act to invoke the prediction functionality based on user input. This functionality probably employs historical AQI data with other relevant environmental factors in building its forecast.

A graphical representation of historical AQI data with model predictions gives a visual sense of model performance:

- Historical AQI (Blue Line): Observed air quality index over time.
- SARIMA Predictions (Red Dashed Line): Prediction using SARIMA.

Figure 11: Obtained graph and score as prediction

- Random Forest Predictions (Orange Dashed Line): Prediction using Random Forest.
- Hybrid Predictions (Green Line): A combination of insights from multiple models for refined predictions.

This visualization provides a way of intuitively understanding how well each model fits actual AQI trends over time.

Tabular Display of Predictions

The following table compares specific date predictions across different models:

Date	SARIMA Predictions	Regression Predictions	Hybrid Predictions
2023-04-11	83.87	103.24	93.55
2023-04-12	94.93	89.15	92.04
2023-04-13	102.77	98.96	100.87
2023-04-14	97.99	107.48	102.73
2023-04-15	94.92	100.13	97.52
2023-04-16	106.99	105.97	106.48

This table illustrates how the predictions vary over specific dates in each model, drawing out the strengths and weaknesses of each model in forecasting AQI levels. In a

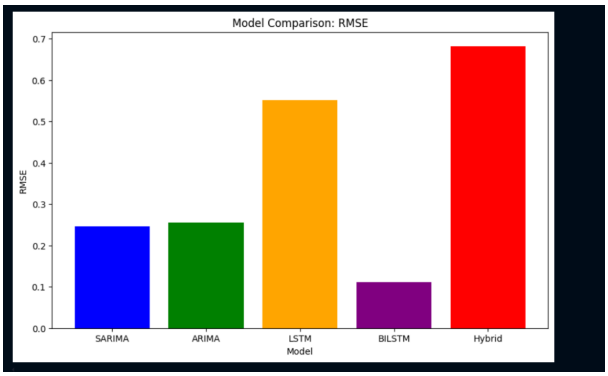

Figure 12: **Model Comparison of RMSE**

bar graph, "Model Comparison: RMSE", it shows the different models in comparison to their respective RMSE:

· Hybrid model is very RMSE, so the predictions of these models are comparatively least in accuracy.

· The nearest to this is LSTM in comparison to the prediction errors of the above models.

· The performance is also enhanced compared to that of LSTM from the above BILSTM model and still comparable to the ARIMA one.

· In conclusion, ARIMA is highly efficient compared to the mentioned models. · Lastly, due to the lowest RMSE value, SARIMA remains as the most accurate model.

· SARIMA demonstrated the lowest RMSE, which implies it outperformed all other models in providing good forecasts of AQI.

· BILSTM showed remarkably low RMSE, which indicates that it might have captured complex patterns quite accurately, because of the infrequent usage of it in this context.

· Hybrid model showed higher value of RMSE, which indicates that models combined predictions with noise, or rather needs more tuning to improve its performance. · The

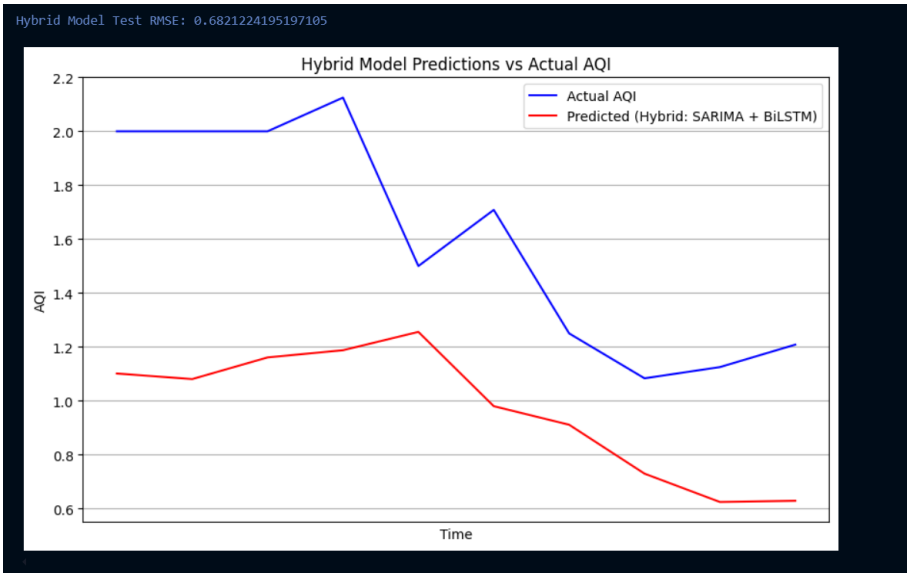

Figure 13: **Hybrid Model Predictions vs Actual AQI**

"Hybrid Model Predictions vs Actual AQI" line graph compares actual AQI with the hybrid model's predictions:

· The blue line shows the time-series actual measured values of the AQI.

· The hybrid model combining the approaches of SARIMA and BILSTM predicts the values, and the red line is visualized.

Although there are instances where predicted values are not the same as measured values, trends are that the hybrid model will be able to pick up essential patterns in the change of AQI. The value of RMSE for this hybrid model prediction is acceptable but shows that there is still room for improvement for future versions of this research. Conclusion This evaluation section gives much detail about model performance but places these findings in context with other solutions that could be used in air quality prediction systems.

6.3 Comparison with Existing Solutions

When comparing this implementation with other existing solutions in air quality prediction, several key points come up:

· Performance Improvement: Hybrid approach promises that it would leverage both statistical methods and machine learning algorithms to better improve predictive accuracy compared to traditional methods based solely on ARIMA or other statistical techniques.

· Real-Time Capabilities: Due to the real time data processing and prediction which are critical for timely decision on air quality management, this solution is deployed on cloud infrastructure like AWS Elastic Beanstalk.

· Scalability: Scalability that traditional implementations lack is achieved via use of cloud computing resources, and by using cloud computing resources one can accommodate varying data loads without making large investments in infrastructure.

· Typically, some form of modeling where one can combine insights from different modeling approaches will likely result in more robust predictions, occurring at the expense of additional tuning to optimize hybrid results.

In a nutshell, this has shown that though single models such as SARIMA are doing

very well on their own, the air quality prediction abilities can be greatly improved when diverse approaches are put together in the cloud, which is beneficial to public health outcomes and environmental management strategies. Hybrid models introduced in this work will need further research and optimization to achieve greater accuracy and reliability in real world application.

7 Conclusion and Future Work

This section summarizes the conclusions and discussion of this research on hybrid models of air quality prediction through cloud computing. It is a summary of the main insights, resulting implications of the research, and concludes with possible future work directions.

7.1 Summary of Findings

The main objective of this research was to create a hybrid model by combining traditional statistical methods like ARIMA with machine learning techniques, specifically Random Forest and LSTM, to enhance the accuracy and reliability of air quality predictions. This research utilized cloud computing technologies for the real-time processing of data and deployment of the model. Key findings from the research are:

· This demonstrates the model with the smallest RMSE to be the SARIMA at 0.246193, demonstrating how good the model has been at identifying linear patterns within air quality data; while the RMSE associated with the hybrid model was comparatively high at 0.682122, thereby indicating a model which though useful provides additional complexity to be calibrated upon.

· Real-Time Prediction Capabilities: The Flask application on Cloud 9 was implemented in such a way that users could input data and receive immediate AQI predictions. This feature shows the practical applicability of the developed models in real-world scenarios, enabling timely decision-making for public health and environmental management.

· Cloud Integration: Elastic Beanstalk in AWS offered deployability with scalable and flexible aspects, where the system accommodated variable loads of data efficiently. Integrating applications in the background like S3 for storage and EC2 instances for computation-based resources helped in adding extra performance to the overall output.

· Data Quality and Preprocessing: The importance of rigorous data preprocessing was highlighted, as it greatly impacted the performance of the model. Techniques such as normalization and handling missing values were critical in preparing the dataset for analysis.

These findings cumulatively establish that merging cloud computing with hybrid modeling frameworks significantly enhances air quality prediction capabilities and offers scalable real-time monitoring.

7.2 Implication of the Research

Aside from the scholarly contributions that this research is making, the practical implication is actually more important because it feeds into public health and environment policy.

1. Public Health: Precise air quality forecasts can provide individuals and communities with the information necessary to take precautionary measures against the adverse

health effects of poor air quality. The government can issue alerts in case of a high pollution event, which will help to protect vulnerable populations.

2. Environmental Policy: Policymaking will learn from the predictive models to make regulations effective in regulating emissions and improve the quality of air. The trend of pollution studied enables authorities to create specific intervention measures against key sources of air pollution.

3. Research Advancements: This study contributes to the ever-growing literature on hybrid modeling approaches by demonstrating their efficiency in environmental monitoring. It encourages further exploration of how advanced machine learning techniques could be integrated with more traditional statistical methods across other domains.

Scalability and Flexibility: The cloud-based framework established in this research allows for future scalability as new data sources become available or as monitoring needs change. This ensures that over time, the system remains relevant and effective.

7.3 Future Work Directions

This study has shown some promise toward enhancing the air quality prediction capability using hybrid modeling, and the following are potential future research avenues:

1. Optimization of Models: The hybrid model needs to be optimized to have better predictions. Hyperparameter tuning, feature selection and ensemble methods may be useful techniques to accomplish this. In future studies the predictions can be refined further by including data from other sources such as meteorological data (temperature, humidity) or traffic patterns as future studies can take advantage of more data sources. This will more comprehensively understand factors that affects air quality.

2. Increased possibilities of the system in incorporating streaming data from real-time sensors to IoT devices: The sudden change in air conditions to improve responsiveness of handling. Moreover, Advanced Analytics can help improve real time insights into pollution events.

3. User-Centric Applications: Specifically, applications could be designed with user friendly interfaces allowing individuals or organisations to design predictions focused on a particular need. A feature like personalized alerts, or historical trend analysis, will provide some level of engagement / utility.

4. Cross-Regional Studies: If similar studies are found in different regions geographically, it can offer clues on if environmental conditions in different regions drive differences in model performance, thereby improving our understanding of global air quality dynamics.

5. Other Hybrid Models: Other hybrids of statistical and machine learning models perhaps might work better for prediction. An example would be looking into other deep learning architectures, or even other algorithms. The work lays the ground work for improving air quality predictability by hybrid modeling coupled with cloud computing. While this is important, with respect to public health as well as management, this afford a window of opportunity for field of study such an important one which brings in the space for more research work to be carried out in the same

LINK: http://flask-env.eba-pvdujpmd.us-east-1.elasticbeanstalk.com/

References

Ansari, M. and Alam, M. (2024). An intelligent iot-cloud-based air pollution forecasting model using univariate time-series analysis, *Arabian Journal for Science and Engineering* **49**(3): 3135–3162.

Chang, Y., Abimannan, S., Chiao, H., Lin, C. and Huang, Y. (2020). An ensemble learning based hybrid model and framework for air pollution forecasting, *Environmental Science and Pollution Research* **27**: 38155–38168.

Cheng, Y., Li, X., Li, Z., Jiang, S., Li, Y., Jia, J. and Jiang, X. (2014). Aircloud: A cloud-based air-quality monitoring system for everyone, *Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems*, pp. 251–265.

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

Kaginalkar, A., Kumar, S., Gargava, P. and Niyogi, D. (2021). Review of urban computing in air quality management as smart city service: An integrated iot, ai, and cloud technology perspective, *Urban Climate* **39**: 100972.

Liu, Q., Cui, B. and Liu, Z. (2024). Air quality class prediction using machine learning methods based on monitoring data and secondary modeling, *Atmosphere* **15**(5): 553.

Mahbub, M., Hossain, M. and Gazi, M. (2020). Iot-cognizant cloud-assisted energy efficient embedded system for indoor intelligent lighting, air quality monitoring, and ventilation, *Internet of Things* **11**: 100266.

Mampitiya, L. et al. (2023). Machine learning techniques to predict the air quality using meteorological data in two urban areas in sri lanka, *Environments* **10**(8): 141.

Natarajan, S. et al. (2024). Optimized machine learning model for air quality index prediction in major cities in india, *Scientific Reports* **14**(1): 6795.

Rahman, M. et al. (2024). Airnet: predictive machine learning model for air quality forecasting using web interface, *Environmental Systems Research* **13**(1): 44.

Shah, S., Tariq, Z., Lee, J. and Lee, Y. (2020). Real-time machine learning for air quality and environmental noise detection, *2020 IEEE International Conference on Big Data (Big Data)*, pp. 3506–3515.

Singh, D. et al. (2021). Sensors and systems for air quality assessment monitoring and management: A review, *Journal of Environmental Management* **289**: 112510.

Sokhi, R. et al. (2021). Advances in air quality research–current and emerging challenges, *Atmospheric Chemistry and Physics Discussions* **2021**: 1–133.

Subramaniam, S. et al. (2022). Artificial intelligence technologies for forecasting air pollution and human health: a narrative review, *Sustainability* **14**(16): 9951.

Uppal, M. et al. (2022). Cloud-based fault prediction for real-time monitoring of sensor data in hospital environment using machine learning, *Sustainability* **14**(18): 11667.