

**Project Report**

**On**

**Air Quality Analysis and Forecasting**



Submitted in partial fulfillment for the award of

Post Graduate Diploma in Big Data Analytics (PGDBDA)

From Know IT(Pune)

**Guided by:**

Trupti Joshi Ma’am & Prasad Deshmukh Sir

**Submitted By:**

Himanshu Samal (240343025018)

Pritam Nimbalkar (240343025038)

Ritul Tayade (240343025042)

Shalu Sharma (240343025045)



**CERTIFICATE**

TO WHOMSOEVER IT MAY CONCERN

This is to certify that

Himanshu Samal (240343025018)

Pritam Nimbalkar (240343025038)

Ritul Tayade (240343025042)

Shalu Sharma (240343025045)

Have successfully completed their project on

Analysis of Air Quality and Forecasting

Under the guidance of Mrs. Trupti Joshi Ma’am and Prasad Deshmukh sir



**ACKNOWLEDGEMENT**

This project Analysis of Air Quality and Forecasting using time series Analysis was a great learning experience for us and we are submitting this work to CDAC Know IT (Pune).

We all are very glad to mention the name Mrs. Trupti Joshi and Mr. Prasad Deshmukh for their valuable guidance to work on this project. His guidance and support helped us to overcome various obstacles and intricacies during the course of project work.

We are highly grateful to Mr. Vaibhav Inamdar Manager (Know IT), CDAC, for his guidance and support whenever necessary while doing this course Post Graduate Diploma in Big Data Analytics (PGDBDA) through CDAC ACTS, Pune.

Our most heartfelt thanks go to Mr. Prasad Deshmukh (Course Coordinator, PGDBDA) who gave all the required support and kind coordination to provide all the necessities like required hardware, internet facility and extra Lab hours to complete the project and throughout the course up to the last day here in CDAC Know IT, Pune.



**TABLE OF CONTENTS**

1. ABSTRACT
2. INTRODUCTION
3. DATA COLLECTION AND FEATURES
4. SYSTEM REQUIREMENTS
   1. Software Requirements
   2. Hardware Requirements
5. FUNCTIONAL REQUIREMENTS
6. ARCHITECTURE
7. MACHINE LEARNING ALGORITHMS
8. DATA VISUALIZATION AND REPRESENTATION
9. CONCLUSION AND FUTURE SCOPE

10. REFERENCES



**Abstract**

This project focuses on developing a machine learning model to predict important factors effecting air pollution levels across Indian states, leveraging a comprehensive dataset spanning from 2010 to 2023 and covering 453 cities. We employ PySpark for efficient Extract, Transform, and Load (ETL) processes, enabling scalable data handling and processing. Advanced machine learning techniques are then applied to uncover trends and patterns in air pollution, assess the effectiveness of pollution control measures, and generate actionable insights. These insights, visualized using Tableau, aim to support policymakers and researchers in formulating strategies to enhance air quality and public health in India.

Our project addresses a critical need for precise and reliable air pollution forecasting, providing valuable tools to tackle environmental challenges in the country. In the era where technology plays a pivotal role in understanding complex systems, leveraging vast datasets through advanced analytics becomes crucial. We aim to harness the power of Big Data and employ cutting-edge machine learning algorithms to process extensive meteorological data, enabling us to unveil precise and reliable air quality patterns.



**Introduction**

As the largest growing industrial nation, India is producing record amount of pollutants specifically Co2, pm2.5, etc and other harmful aerial contaminants. Air quality of a particular state or a country is a measure on the effect of pollutants on the respected regions, as per the Indian air quality standard pollutants are indexed in terms of their scale, these air quality indexes indicate the levels of major pollutants on the atmosphere. There are various atmospheric gases which causes pollution on our environment.

Each pollution has individual index and scales at different levels. The major pollutants Such as (no2, so2, rspm, spm) indexes AQI is acquired, with this individual AQI, the data can be categorized based on the limits. We collected the data from the Indian government database, which contains pollutant concentration occurring at various places across India.

We have designed a model to predict the air quality of every available data points in the dataset, our model is capable of forecasting the air quality of India in any given area. By predicting the air quality, we can backtrack the major pollution causing pollutant and the location affected seriously by the pollutant across India. With this forecasting model, various knowledge about the data are extracted using various techniques to obtain heavily affected regions on a particular region(cluster). This give more information and knowledge about the cause and seniority of the pollutants.

By uniting machine learning models with cutting-edge big data technologies, our overarching goal is to provide accurate analysis and forecasting of air quality for selected states within our extensive dataset. This project not only contributes to the advancement of air quality forecasting methodologies but also underscores the significance of integrating data science approaches to enhance prediction accuracy in a field crucial to numerous industries.



**Dataset Collection and Features**

The dataset contains air quality data for Indian cities from the year 2010 to 2023.

It provides information about the air quality conditions in 453 cities across India. The data has been collected from the Central Control Room for Air Quality Management, which indicates that it is a reliable and authoritative source.

The data used in this project has been sourced from the Central Pollution Control Board (CPCB), which is the official portal of the Government of India. The CPCB has made the data publicly available and can be accessed at their website: [https://cpcb.nic.in](https://cpcb.nic.in/) The data was compiled from the CPCB website, which serves as the official body of the Government of India for monitoring and controlling pollution. Selenium is used for data collection. It is a powerful tool for web automation, which was instrumental in extracting and processing the data from the CPCB website. The utilization of Selenium greatly facilitated the completion of this task.

This dataset is valuable for understanding and analyzing the air quality trends and patterns in Indian cities over a span of 13 years. It can help researchers, policymakers, and the general public gain insights into the air pollution levels, identify areas with high pollution, assess the impact of air quality control measures, and develop strategies for improving air quality and public health.

It can be used for conducting statistical analyses, creating visualizations, developing predictive models, and generating actionable insights to address the air pollution challenges in India.



* **Dataset Sources:**

Air Quality Monitoring Stations: Data collected from government or independent air quality monitoring stations.

Satellite Data: Remote sensing data providing information on atmospheric composition and pollution levels.

Weather Data: Meteorological data from weather stations or services.

* **Dataset Components:**

Historical Data: Long-term historical air quality data for trend analysis.

Real-Time Data: Current data for immediate analysis and forecasting.

Forecast Data: Predictions from models or services for future air quality levels.

* **Features for Air Quality Analysis**
* Air Quality Metrics:

Concentration of Pollutants: Levels of specific pollutants measured in µg/m³ or ppb (parts per billion). Common pollutants include:

* + **PM2.5:** Particulate matter with a diameter of 2.5 micrometers or less.
  + **PM10:** Particulate matter with a diameter of 10 micrometers or less.
  + **NO2:** Nitrogen dioxide.
  + **SO2:** Sulfur dioxide.
  + **O3:** Ozone.
  + **CO:** Carbon monoxide.
  + **VOC:** Volatile organic compounds.
  + **NH3:** Ammonia.



* Meteorological Data

Temperature:

Air temperature, typically measured in °C or °F.

Humidity: Relative humidity percentage.

Wind Speed and Direction: Wind speed (km/h or mph) and direction (degrees or cardinal directions).

Precipitation: Amount of rainfall or snowfall (mm or inches).

Atmospheric Pressure: Pressure measurements (hPa or mmHg).

* Temporal Features

Date and Time: Timestamp for each data point to analyze temporal patterns and trends.

Day of the Week: To account for variations in air quality related to daily activities.

Season: Seasonal patterns in air quality related to weather and pollution sources.

* **Features for Forecasting:**

Historical Air Quality Data:

Lagged Values: Previous time steps of air quality metrics used for time series forecasting.

Moving Averages: Average values over specific time windows to smooth out fluctuations.

Weather Forecast Data:

Forecasted Meteorological Variables: Predicted values for temperature, humidity, wind speed, etc.

Emission Data:

Predicted Emissions: Forecasts of emissions from sources such as vehicles, industries, or agriculture.



Additional Contextual Data:

Traffic Volume: Forecasted traffic data to estimate future vehicular emissions.

Industrial Activity: Predictions of industrial activities that could affect air quality.

Special Events: Information on upcoming events (e.g., festivals, industrial operations) that might impact air quality.

* **Data Preprocessing and Feature Engineering**

Data Cleaning:

Handling Missing Values: Techniques for filling in or interpolating missing data.

Outlier Detection: Identifying and managing anomalies in the data.

Feature Scaling:

Normalization or Standardization: Scaling features to ensure they contribute equally to model performance.

Data Transformation:

Temporal Aggregation: Aggregating data into daily, weekly, or monthly summaries.

Feature Engineering: Creating new features based on existing data, such as pollution indices or combined meteorological factors.

Data Splitting

Training and Testing Sets: Dividing the data into training and testing sets for model evaluation.



**System Requirement**

**Hardware Requirements:**

* Computer: Ensure your computer has sufficient processing power and memory to execute data processing and analysis tasks effectively. A modern multicore processor and at least 25 GB of RAM are recommended for handling the complexities of our project.
* Storage: Allocate ample storage space for storing the generated dataset and any additional datasets if required. Utilizing an SSD (Solid State Drive) is advisable to facilitate faster data access, particularly when managing large volumes of structured and unstructured data.
* Internet Connection: A stable internet connection is critical for downloading and installing necessary software packages, libraries, and accessing online resources essential throughout our project.

**Software Requirements:**

* Operating System: Our project relies on Python (3.10.12) as the primary programming language. It's also essential to mention that the operating system used is Linux.
* Python: Python is the backbone of our project, serving as the main programming language for data generation, analysis, and machine learning tasks.
* PySpark: PySpark (3.4.0) plays a crucial role in preprocessing our downloaded data, initially in a complex JSON unstructured format. We use PySpark to convert this data into a structured format by defining a schema. The processed data is then stored in MongoDB.
* MongoDB: MongoDB (6.0.13) is employed as a database to store the structured data resulting from the preprocessing phase. MongoDB's flexibility with JSON-like documents aligns well with the nature of our project's data.
* Machine Learning Libraries (stats models 0.14.1): Machine learning components in our project, such as ARIMA and SARIMAX, are implemented using the stats model library (0.14.1). SARIMAX is particularly chosen for training due to the seasonal nature of our data.



* Kaggle: Our machine learning models are trained on Kaggle, leveraging its platform to make efficient use of resources.
* Tableau Public: Tableau Public remains part of our project's software stack, likely used for data visualization and creating interactive dashboards to convey insights derived from our analysis.

This comprehensive set of hardware and software requirements showcases the robust infrastructure and tools employed in our project, emphasizing efficiency and scalability in handling large datasets and complex data processing tasks. The adjustment to 25 GB of RAM reflects the optimization made during the project, maintaining effective performance while conserving resources.



**Architecture**

Data Cleaning

PySpark

Dataset

Data Visualization

Hive

Pre-processing

Machine Learning



The architectural workflow for our system is designed to efficiently handle data retrieval, preprocessing, and analysis, ensuring a seamless flow of information from user input to meaningful insights. The various components collaborate to create a robust and dynamic system:

* User Input and Initial API Request:

The process begins with the API receiving a request containing user input, typically a city name. This request is sent to the data source.

* Data Retrieval and Initial File System Storage:

The data source retrieves relevant information based on the user input and saves it in the initial file system. This serves as the foundational dataset for subsequent analysis.

* Data Size: The dataset comprises approximately **16 gigabytes of data** for over 200 cities, with each city recording hourly data for over 20 years. This substantial data volume poses challenges in terms of storage, processing, and analysis.
* Data Complexity: The data is structured in a complex nested JSON format, necessitating careful preprocessing to extract relevant information and transform it into a usable format. Dealing with nested structures and hierarchical data requires specialized processing techniques and schema manipulation.
* Schema-Based Filtering: An initial schema is defined to filter out irrelevant data and extract essential attributes. This schema-based approach helps streamline the preprocessing workflow and optimize resource utilization.
* Machine Learning Model Training and File System Storage:

The initial dataset is utilized to train a machine learning model, and the trained model is saved in the file system. This model acts as the baseline for generating predictions.

to ensure it aligns with the required structure for further analysis.

* Model Loading and Prediction Generation:

The trained model is loaded, and predictions are generated on the preprocessed data. These predictions form the basis of the insights provided to the user.

* Structured Data Storage in MongoDB:

The structured data, including both input and output, is saved in MongoDB. This database provides a scalable and flexible solution for storing data in a JSON-like format.



* Daily Data Processing and File System Update:

On a daily basis, new data is processed and saved in the file system. This ensures that the system is constantly updated with the latest information for ongoing analysis and model training.

* Model Retraining and File System Update:

The saved data is used to retrain the machine learning model, and the retrained model is stored in the file system. This iterative process ensures that the model stays current and relevant.

* Structured Data Storage in File System:

The structured data resulting from the daily processing is saved in the file system, facilitating easy access and future analysis.

* Trained Model Storage in File System:

The trained model, now updated through regular retraining, is stored in the file system, ensuring its availability for predictions and analysis.



This comprehensive workflow underscores the synergy of APIs, data sources, file systems, machine learning models, and user interfaces in creating a system that efficiently processes and analyzes vast amounts of data to provide meaningful and up-to-date insights to the user. The integration of MongoDB and file systems ensures robust data storage and accessibility for ongoing analysis and model training. The daily updates and retraining processes contribute to the system's adaptability and responsiveness to the ever-changing data landscape.



**Algorithms**

**AutoRegressive Integrated Moving Average (ARIMA):**

The AutoRegressive Integrated Moving Average (ARIMA) model is a widely used statistical tool for time-series forecasting, including weather temperature prediction. The model combines autoregression, differencing, and moving average components to analyze and predict future values based on historical data.

However, the ARIMA model does have certain limitations. One major disadvantage is its restricted ability to capture long-term trends and seasonal variations, particularly in complex and nonlinear weather systems. Air quality can change over time due to various factors, which may not be adequately accounted for in the ARIMA model.

Additionally, ARIMA models assume linearity in the relationships between variables, while weather systems can be highly nonlinear and dynamic. Another challenge with ARIMA models is the need for a stationary time series, which means that the statistical properties, such as mean and variance, should remain constant over time. In practice, data can be non-stationary, requiring differencing or other transformations to stabilize the series. These transformations can introduce additional complexity and potential errors in the model.

graph



The Root Mean Square Error (RMSE) is a crucial metric for evaluating the accuracy of a forecasting model, including the ARIMA model for air quality prediction. An RMSE of \_\_\_\_\_\_\_\_ indicates that, on average, the model's predictions are approximately \_\_\_\_\_\_\_\_\_ units away from the actual observed values.

In the context of air quality prediction, an RMSE of \_\_\_\_\_\_\_\_ implies that the model's forecasts may deviate from the true temperatures by around 8 degrees Celsius. While this might be acceptable for some applications, it is essential to aim for lower RMSE values to improve the model's accuracy.

**Seasonal Autoregressive Integrated Moving Average (SARIMAX):**

SARIMAX (Seasonal Autoregressive Integrated Moving Average with exogenous variables) is a powerful time series forecasting model that builds upon the ARIMA model by incorporating seasonal components and exogenous variables. This extension makes SARIMAX particularly effective for capturing complex seasonal patterns and accounting for external factors that may influence the variable of interest, such as air quality.

In the context of air quality prediction, SARIMAX can account for various patterns, such as daily, weekly, or annual cycles, and incorporate external factors like humidity, pressure, or wind speed. By considering these additional components, SARIMAX can provide more accurate and reliable forecasts compared to the basic ARIMA model.

graphs

The Root Mean Square Error (RMSE) of \_\_\_\_\_\_ for the SARIMAX model indicates that, on average, the model's predictions are approximately \_\_\_\_\_\_\_ units away from the actual observed values. This RMSE value is relatively low, suggesting



that the SARIMAX model's forecasts are quite accurate in predicting

Achieving an RMSE of \_\_\_\_\_\_\_ demonstrates the model's ability to effectively capture both seasonal patterns and the influence of external factors on air quality. This improvement in accuracy can significantly benefit various applications, such as energy management, agriculture, or transportation planning, by providing more precise and reliable air quality.

Here's how it works:



**Data Visualization:**



**Conclusion And Future Scope**

Our project represents a sophisticated and comprehensive system for predicting the current data with 95% accuracy. It will successfully predict the upcoming air quality of any particular data within a given region. With this model we can forecast the air quality and alert the respected region of the country also it a progressive learning model it is capable of tracing back to the particular location needed attention provided the time series data of every possible region needed attention. The air quality information utilized in air quality checking and investigation stage, and incorporates the normal every day fine particulate issue (PM2.5), inhalable particulate issue (PM10), ozone (O3), CO, SO2, NO2 fixation and air quality record(AQI).The essential perspectives that should be viewed as with regards to gauging of the poison focus are its different sources alongside the components that impact its fixation

In conclusion, our project represents a sophisticated and comprehensive system for predicting and analyzing data across 400+ cities in India. The integration of Python, PySpark, MongoDB, Kaggle and Tableau facilitated a robust and scalable infrastructure for data retrieval, preprocessing, machine learning modeling, and user interaction. The combination of PySpark and Kaggle played a pivotal role in efficiently handling the complex and unstructured initial data, enabling us to derive valuable insights through machine learning models. The use of MongoDB ensured structured data storage and retrieval, while Tableau enhanced the visualization of our findings.



**Future Scope:**

The project lays a solid foundation for future enhancements and expansions. Here are some potential avenues for future development:

* **Enhanced Data Collection**

**Advanced Sensors:** Development of more accurate, cost-effective, and compact sensors for detecting various pollutants.

**IoT Integration:** Increased use of Internet of Things (IoT) devices for real-time air quality monitoring.

**Satellite Technology:** Improved satellite systems for broader and more detailed air quality data collection.

* **Policy and Decision Support**

**Regulatory Frameworks:** Development of more effective air quality regulations based on advanced forecasting models and data analysis.

**Urban Planning:** Integration of air quality forecasts into urban planning and development to create healthier living environments.

* Emerging Pollutants

Novel Pollutants: Identification and analysis of emerging pollutants (e.g., microplastics, new chemicals) and their impact on air quality.

Innovative Detection Methods: Creation of new methods for detecting and measuring these pollutants.

* International Collaboration

Global Networks: Enhanced international cooperation for sharing air quality data and forecasting models across borders.

Standardization: Development of global standards for air quality monitoring and reporting.

**References**



* Apache Spark. [https://spark.apache.org/]
* MongoDB. [https://[www.mongodb.com/]](http://www.mongodb.com/)
* Python. [[https://ww](http://www.python.org/)w.py[thon.](http://www.python.org/)org/]
* scikit-learn. [https://scikit-learn.org/]
* [<https://www.ripublication.com/ijaerspl2019/ijaerv14n11spl_34.pdf>]
* [https://cpcb.nic.in]