



**DEPARTMENT OF ENGINEERING & TECHNOLOGY
SCHOOL OF ENGINEERING & TECHNOLOGY,
SHARDA UNIVERSITY, GREATER NOIDA**

**ARIMA model for Delhi Air Quality prediction with health
advisory**

*A project submitted
in partial fulfillment of the requirements for the degree of
Bachelor of Technology in Computer Science and Engineering
by*

SPARSH SINGH (2019005921)

VIDIT KUMAR (2019568217)

ZAID AHMED (2019006065)

Supervised by:

Ms. KAJOL MITTAL, ASST. PROF (CSE)

May, 2023

CERTIFICATE

This is to certify that the report entitled “**ARIMA MODEL FOR DELHI AIR QUALITY PREDICTION with HEALTH ADVISORY**” submitted by “**Mr. Sparsh Singh (2019005921), Mr. Vidit Kumar (2019568217) and Mr. Zaid Ahmed (2019006065)**” to **Sharda University**, towards the fulfillment of requirements of the degree of “**Bachelor of Technology**” is record of bonafide final year project work carried out by them in the “Department of Computer Science & Engineering, School of Engineering and Technology, Sharda University”. The results/findings contained in this Project have not been submitted in part or full to any other University/Institute forward of any other Degree/Diploma.

Signature of the Guide

Name: Ms. Kajol Mittal

Designation: Assistant Professor

Signature of Head of Department

Name: Prof. (Dr.) Nitin Rakesh

Place:

Date:

Signature of External Examiner

Date:

ACKNOWLEDGEMENT

A major project is a golden opportunity for learning and self-development. We consider ourselves very lucky and honoured to have so many wonderful people lead us through in completion of this project.

First and foremost, we would like to thank **Dr. Nitin Rakesh, HOD, CSE** who gave us an opportunity to undertake this project.

Our grateful thanks to **Ms. Kajol Mittal and Mrs. Nikita Verma** for their guidance in our project work. Ms. Kajol Mittal and Mrs. Nikita Verma, who in spite of being extraordinarily busy with academics, took timeout to hear, guide and keep us on the correct path. We do not know where we would have been without her help.

CSE department monitored our progress and arranged all facilities to make life easier. We choose this moment to acknowledge their contribution gratefully.

Name and signature of students:

SPARSH SINGH (2019005921)

VIDIT KUMAR (2019568217)

ZAID AHMED (2019006065)

ABSTRACT

Recent years have seen a substantial increase in study on air pollution as a result of its negative ramifications. It is also acknowledged as one of the current atmosphere's main risk elements. Accurate air quality assessment is the first stage in the implementation of air pollution control systems, which helps in the growth of developed nations' economies and societies. Accurate air quality predictions are essential for emissions control, public health, and wellbeing. The metropolis of India, Delhi, and Bengaluru, one of the nation's technology centres, has been the most polluted metropolis in the world for the past two years. In this study, we have compared the results of three separate Machine Learning and Deep Learning models, namely SARIMAX, Prophet, and LSTM, to see how they fare against each other. All the models have achieved great results given the same amount of parameter calibration.

CONTENTS

TITLE.....	I
CERTIFICATE.....	II
ACKNOWLEDGEMENT.....	III
ABSTRACT.....	IV
CONTENTS.....	V
LIST OF FIGURES.....	VI
LIST OF TABLES.....	VIII
CHAPTER 1: INTRODUCTION.....	1
1.1 Problem Statement.....	7
1.2 Project Overview.....	8
1.3 Expected Outcome.....	9
1.4 Hardware & Software Specifications.....	10
1.5 Other Non-Functional Requirements.....	11
1.6 Report Outline.....	12
CHAPTER 2: LITERATURE SURVEY.....	13
2.1 Existing Work.....	13
2.2 Functional Requirement.....	19
2.3 Proposed System.....	20
2.4 Feasibility Study.....	21
CHAPTER 3: SYSTEM DESIGN & ANALYSIS.....	22
3.1 Project Perspective.....	22
3.2 Methodology.....	23
3.3 Data Collection.....	24
3.4 Data Processing.....	25
3.5 Splitting Data.....	27
3.6 Models Used.....	28
CHAPTER 4: ALGORITHMS.....	32
4.1 SARIMAX.....	32
4.2 Prophet.....	36
4.3 LSTM	39
CHAPTER5: RESULT AND OUTPUT.....	41
CHAPTER6: CONCLUSION AND FUTURE SCOPE.....	42
References.....	43

LIST OF FIGURES

Figure Number	Figure Name	Page Number
Fig 1.1	PM2.5 concentrations in different states	2
Fig 1.2	PM10 concentration in different states	3
Fig 1.3	AQI level in different states	5
Fig 1.4	Formulae Representation of AQI	6
Fig 1.5.1	Non-Functional Requirements	11
Fig 3.2.1	Proposed Methodology for predicting Air Quality components	23
Fig 3.3.1	Features of Dataset	24
Fig 3.4.1	Representation of Dataset	26
Fig 3.6.1.1	Pseudo Code for SARIMAX Model	28
Fig 3.6.2.1	Pseudo Code for Prophet	29
Fig 3.6.3	Pseudo Code for LSTM	31

Fig 4.1.1.1	Monthly PM10 Original vs Predicted	35
Fig 4.1.1.2	Yearly PM10 Original vs Predicted	35
Fig 4.2.1.2	Prediction of PM 10 Pollutants	38
Fig 4.3.1.1	Prediction of pollution level using LSTM	40

LIST OF TABLES

Table Number	Table Name	Page Number
Table 1	Evaluation Metrics	41

CHAPTER 1

INTRODUCTION

These days, a lot of people are worried about air pollution because it has a number of adverse effects on the environment and the global economy. It is defined as one or more contaminants present for an extended period of time in the air inside or outdoors. a protracted length of time that might be harmful to people, plants, or animals, or that unintentionally disrupts daily life or property. In Asian nations, air pollution is a significant problem. In Asia, air pollution is thought to be responsible for over 537,000 premature deaths. Despite the fact that air pollution is more common in cities, individuals who are impoverished or live in areas with poor air quality are the ones who suffer the most (Haq and Schwela; 2008). The city chosen for this study is Delhi, the latter of which is a famous IT hub in India and one of the nation's most developed metropolises. Delhi, the nation's capital and one of its major metropolitan areas, has been listed as one of India's most air-polluted cities since 2015 due to the city's high population and number of industries. The people should be promptly told about the extent of the air pollution and any changes to the environment so that they can take precautions to ensure their safety. There are therefore many forecasting models in use to anticipate the level of pollution, but a more precise statistical model is still needed to determine the level of pollution and the air quality index, which has a detrimental effect on human health. In India, the amount of air pollution has gotten so bad that it has replaced smoking as one of the main causes of cancer, heart disease, and numerous respiratory illnesses.

- ❖ In general, many time series forecasting models have been proposed to improve the significant precision and efficiency by reducing the prediction mistakes. The most popular time series model is the ARIMA, which is well-known for statistical forecasting and has a high level of speed and precision. The central concept behind this model is

founded on the linearity time series assumption, which frequently corresponds to the normal statistical distribution. For the study and forecasting of stationary time series data, Box and Jenkins suggested the ARIMA model. While the SARIMA model was put forth for predicting periodic time series. Additionally, auto-regression (AR) and moving-average (MA) polynomials, which are mixed with a compound polynomial, are crucial components of the ARIMA model.

- ❖ **Particulate matter (PM2.5 and PM10):** Particulate matter is a mixture of solids and liquids that is suspended in the air, containing carbon, difficult organic compounds, sulphates, nitrates, mineral dust, and water. PM's size ranges. Dust, soot, dirt, and smoke are examples of particles that are big enough or black enough to be visible to the unaided eye. The tiny PM10 and PM2.5 particles, however, are the most harmful.

City	PM2.5
0 Patna	123.500000
1 Delhi	117.200000
2 Gurugram	117.100000
3 Lucknow	109.710000
4 Ahmedabad	67.850000
5 Kolkata	64.360000
6 Jorapokhar	64.230000
7 Brajrajnagar	64.060000
8 Guwahati	63.690000
9 Talcher	61.410000

Fig 1.1 PM 2.5 concentration in various states

City	PM10
0 Delhi	232.810000
1 Gurugram	191.500000
2 Talcher	165.770000
3 Jorapokhar	149.660000
4 Patna	126.750000
5 Brajrajnagar	124.220000
6 Jaipur	123.480000
7 Bhopal	119.320000
8 Guwahati	116.600000
9 Kolkata	115.630000

Fig 1.2 PM 10 concentration in various states

❖ **Nitrogen Oxides (NO, NO₂, NO_x):** Nitrogen oxides, sometimes known as NO_x gases, are a collection of seven nitrogen and oxygen-based gases and compounds. Nitric oxide (NO) and nitrogen dioxide (NO₂) are the two most prevalent and dangerous nitrogen oxides.

❖ **Sulphur Dioxide (SO₂):** The gas sulphur dioxide, sometimes known as SO₂, is colourless and has a pungent aroma akin to that of a freshly struck match. When sulphur-containing fuels, such coal and oil, are burnt, air pollution results.

❖ **Carbon Monoxide (CO):** Carbon monoxide is an odourless, extremely toxic gas. When pressed, it turns into liquid. It is created by burning several fuels, including charcoal, wood, natural gas, and petrol.

❖ **Benzene, Toluene and Xylene (BTX):** Particularly after home decorating, well-known indoor air contaminants include benzene, toluene, xylene, and formaldehyde. Additionally, they are typical workplace contaminants in the leather, chemical, and plastic industries.

❖ **Ammonia (NH₃):** Ammonia pollution is damage caused by the chemical ammonia (NH₃), a by-product of agriculture and industry and a compound of nitrogen and hydrogen.

❖ **Ozone(O₃):** Just above the earth's surface, a colourless, extremely unpleasant gas known as "ground-level ozone" is formed. It is referred to as a "secondary" pollutant since it is produced when two primary pollutants react in sunshine and steady air. These two primary pollutants are volatile organic compounds (VOCs) and Nitrogen Oxides (NO_x).

The dataset includes information on these contaminants as well as the AQI values for each city.

❖ **Air Quality Index.** The air quality index (AQI) is a daily reporting measure for air quality. It is a measure of how quickly air pollution damages a person's health.

The purpose of a web-based system is to deliver AQI in real time. It is an automatic system that collects information from continuous monitoring stations without the need for human interaction and displays AQI based on running average values (for example, AQI at 6 am on a given day will include information from 6 am on the day before to the present). An AQI calculator is created for stations that use manual monitoring, and data may be manually entered to obtain the AQI result.

City	AQI
0 Ahmedabad	452.120000
1 Delhi	259.490000
2 Patna	240.780000
3 Gurugram	225.120000
4 Lucknow	217.970000
5 Talcher	172.890000
6 Jorapokhar	159.250000
7 Brajrajnagar	150.280000
8 Kolkata	140.570000
9 Guwahati	140.110000

Fig 1.3 AQI Levels in different states

- ❖ **Calculation of AQI.** PM_{2.5} (Particulate Matter 2.5-micrometer), PM₁₀, CO, NH₃, NO_x, SO₂, and O₃ (ozone) are the seven measurements used in the AQI calculation. The average value over the previous 24 hours is utilised for PM_{2.5}, PM₁₀, SO₂, NO_x, and NH₃ with the requirement that there be a minimum of 16 readings. The highest value over the last 8 hours is utilised for O₃ and SO. Based on pre-established groupings, each measure is transformed into a Sub-Index. There are occasions when measurements are unavailable because there aren't enough data points to measure with. Final AQI is the highest Sub-Index, provided that at least three of the seven components—including PM₂, PM₁₀, and at least one of each—are accessible.
1. The sub-indices for particular pollutants at a monitoring site are computed using the 24-hour average concentration value (8-hourly for O₃ and CO) and the health breakpoint concentration range. The worst sub-index of the AQI is for that location.

2. It's probable that not all locations are keeping an eye on all eight pollutants. The final AQI cannot be calculated unless data are available for at least three contaminants, one of which must be PM2.5 or PM10. If not, the input is judged insufficient to determine the AQI. Similar to that, it is presumed that the subindex can only be calculated with at least 16 hours of data.
3. Sub-indices for monitored pollutants are calculated and presented even when there is not enough data to calculate the AQI. The Individual Pollutant-wise Sub-Index gives data regarding that pollutant's air quality status.

The air quality index (AQI) is defined as the ratio of the atmospheric pollutant concentrations as measured to the values that are typically prescribed. The following is a general formula to calculate an AQI:

$$\text{AQI pollutant} = (\text{pollutant concentration reading} / \text{Standard Concentration}) * 100$$

There are 6 categories of the air created in this air quality index:

Good (0–50)	Minimal Impact	Poor (201–300)	Breathing discomfort to people on prolonged exposure
Satisfactory (51–100)	Minor breathing discomfort to sensitive people	Very Poor (301–400)	Respiratory illness to the people on prolonged exposure
Moderate (101–200)	Breathing discomfort to the people with lung, heart disease, children and older adults	Severe (>401)	Respiratory effects even on healthy people

Fig 1.4 categorization of AQI

1.1 Problem Statement

Air pollution is a severe problem in Delhi, India, causing significant health risks to its residents. A number of variables, like heavy traffic, industrial exercise, and natural occurrences like dust storms, have an impact on the city's air quality. To lessen the adverse effects of air pollution on both human health and the environment, it is essential to monitor and forecast air quality levels. Traditional methods for air quality forecasting have limitations in accuracy, reliability, and timeliness, making it necessary to explore advanced techniques such as machine learning.

The problem is to develop accurate and reliable machine and deep learning models for air quality forecasting in Delhi using time series data. The models should provide real-time air quality forecasts that are accessible to policymakers, researchers, and citizens. By giving timely and reliable information on the air quality levels in Delhi, the objective is to aid in reducing the detrimental impacts of air pollution on human health and the environment.

1.2 Project Overview

The project aims to develop a useful tool for policymakers, researchers, and citizens to monitor and forecast air quality in Delhi. The project will contribute to the efforts to improve air quality and promote public health by providing accurate and timely information about the air quality levels in Delhi.

- **Data Collection:** Collecting historical air quality data from government and public sources for Delhi.
- **Data Pre-processing:** Pre-processing the data by removing missing values, outliers, and noise, and preparing it for analysis.
- **Model Development:** Developing SARIMAX, LSTM, and Prophet models for air quality forecasting using historical time series data.
- **Model Evaluation:** Assessing the precision of the models by reviewing the performance of the models using several metrics, such as root mean square error.
- **Model Comparison:** Comparing the performance of the models in terms of accuracy, efficiency, and sturdiness to identify the most suitable model for air quality forecasting in Delhi.
- **Analysis and Insights:** Providing insights into the air quality trends and patterns in Delhi based on the analysis of the established time series data.

1.3 Expected Outcome

The expected outcome of the "Air Quality Forecasting in Delhi using SARIMAX, LSTM, and Prophet" project is to develop accurate and reliable machine learning models for air quality forecasting in Delhi using time series data. Specifically, the project aims to:

1. Develop SARIMAX, LSTM, and Prophet models for air quality forecasting in Delhi using historical time series data.
2. To determine the accuracy of the models, analyse the performance of the models using a metric such root mean square error.
3. To find the best model for Delhi's air quality forecasts, compare the models' performances in terms of accuracy, performance, and durability.
4. On the basis of the study of historical time series data, offer insights into Delhi's air quality trends and patterns.

The expected outcome of this project is to develop a useful project for policymakers, researchers, and citizens to monitor and forecast air quality in Delhi, a city that has been facing severe air pollution problems for many years. The project can contribute to the efforts to improve air quality and promote public health by providing accurate and timely information about the air quality levels in Delhi.

1.4 Hardware & Software Specification

Hardware Requirements:

- A powerful CPU with multiple cores to handle the computation-intensive machine learning models
- Sufficient RAM to store the data and run the models effectively
- Sufficient storage space to store the large datasets and the trained machine learning models
- An internet connection to access the required data and resources

Software Requirements:

- Python programming language with relevant packages such as Pandas, NumPy, Matplotlib, Scikit-Learn, Keras and Prophet for data manipulation, visualization, and machine learning.
- Python: The project requires Python programming language to be installed, preferably Python 3.7 or higher.
- Integrated Development Environment (IDE): The project can be developed using an IDE like Visual Studio Code or any other IDE of your choice.
- Operating systems such as Windows or Linux, depending on the preference of the developer.

1.5 Other Non-Functional Requirements

- **Performance:** The machine learning models used for air quality forecasting should be able to deliver accurate and reliable forecasts in a timely manner, ensuring that users have access to up-to-date information at all times.
- **Reliability:** The system should be designed to be highly available and resilient to ensure that air quality forecasts are always available to users.
- **Security:** The system should be designed to protect sensitive data, such as user information and air quality data, from unauthorized access, ensuring the privacy and confidentiality of users.
- **Usability:** The interface of the system should be simple to use, and it should provide users with information that is both clear and brief.
- **Maintainability:** The system should be designed to be easy to maintain, with clear documentation and coding standards to ensure that it can be updated and improved as needed.

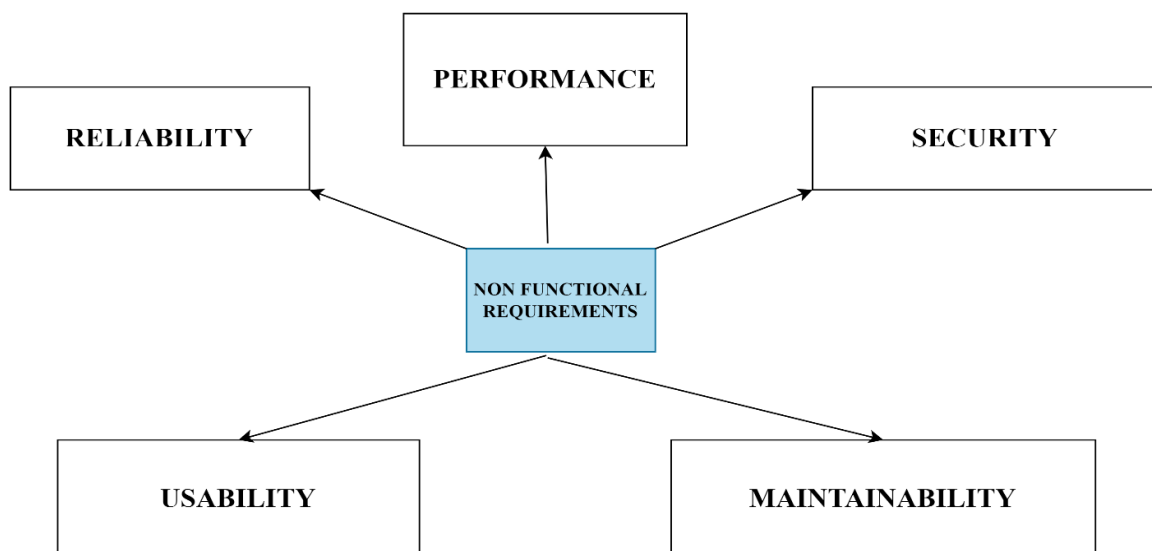


Fig 1.5.1 Non-Functional Requirements

1.6 Report Outline

Chapter 1 is an overall introduction along with the resources necessary to get started with the project

Chapter 2 is a study of previous work done in similar fields to give us an idea how researchers approached this problem and to learn how these ML models work and what results are to be expected by them

Chapter 3 highlights the methodology used with each step explained

Chapter 4 goes over the algorithms used for this project

Chapter 5 is the conclusion of report.

CHAPTER 2

LITERATURE SURVEY

2.1 Existing Work

Good air quality is essential for human survival, but it is worsening in many parts of the globe. It has been and remains challenging to predict the extent and pace of that decline. Scientists and planners will be able to improve the results. If they understand the facts better and are able to predict the future better for everyone. It is viable to utilise a range of prediction techniques and models to better understand how air quality declines over time and the potential consequences of other man-made or environmental variables. In three separate parts that are highlighted in this section, an overview of the research on air pollution that has been done to date is discussed, along with the methods used to improve air quality forecasts.

2.1.1 Analyzing the methodologies and approaches currently in use for Predicting Air Pollution Using Machine Learning and Time Series Approaches

The writers have covered a number of papers in this field of research. The relevance of weather forecasting as a field of study has increased during the last several decades. A linear connection between the target data and the accompanying raw meteorological data was often looked for by the researcher. By creating efficient and trustworthy nonlinear forecasting algorithms for weather analysis with the goal to predict the highest probable temperature for 365 days of the year, the research investigates the application of the ANN technique. Additionally, it utilises a variety of functions for transfer, secret layers, and neurons to compare and assess how well the constructed models operate.[1]. This study is focused on the four seasons of summer, rainy season, post-monsoon, and winter. At ITO,

the busiest traffic intersection, the air quality index (AQI) was calculated using a method developed by the US Environmental Protection Agency (USEPA) for the seven-year period between 2000 and 2006 for specified pollutants including nitrogen dioxide (NO₂), sulphur dioxide (SO₂), respirable suspended particulate matter (RSPM) and suspended particulate matter (SPM) (NAAQS). Using the correlation of the raw data matrix, the principal components have been calculated. Using the principal component regression method, the AQI was only predicted for those components with eigenvalues of 1.[2]. Using data mining methods, this article suggests a novel approach to creating a service-oriented design for a weather information system. Software as a service is made possible by web applications. By creating a service-oriented design, this software can be used on any device, at any moment, and from anywhere. Data mining is a fascinating field of computer science with several applications. This project aims to provide a usable weather information system that is made available as a web service, can be accessed by any sort of application, and forecasts the weather using data mining prediction algorithms. [3]. This study compares statistical and neuro-fuzzy network models for predicting the weather in Göztepe, Istanbul, Turkey. The models were developed using nine years of data (2000-2008), which included daily average temperature (dry-wet), atmospheric pressure, and wind speed. The Adaptive Network Based Fuzzy Inference System (ANFIS) and the Auto Regressive Moving Average (ARIMA) models have been used. The efficacy of ANFIS and ARIMA techniques has been evaluated using a variety of models and training and test sets. Performance assessment parameters are generated in order to estimate as well as contrast the performances of the ARIMA and ANFIS models. According to the acquired findings and prediction accuracy, reliability, and efficiency, the best model and network layout are chosen. ANFIS and ARIMA model efficiency contrasts based on MAE (Moving Average Error), RMSE-R², and According to the (Root-Mean-Square error) parameters, ANFIS produces improved

outcomes.[4]. The last ten years have seen a worldwide challenge with weather prediction. Forecasting is becoming increasingly difficult as a result of the continually changing weather conditions. A number of models have been investigated for the aim of forecasting weather data, assuming the appropriate properties to be independent variables. Understanding the numerous influencing variables that lead to weather changes is essential for successful weather analysis. For a greater comprehension of the weather statistics, it is essential to determine the relationship between these characteristics. Using forecasting analysis and the geographic and temporal correlations among the climatic factors, a weather prediction model is presented in this paper.[5]. This essay examines the advantages of predicting the weather using data mining . The article offers a review of the body of research on a few algorithms used by various researchers to apply diverse data mining techniques for weather prediction. The task that was completed by different scientists working in this area have been examined and contrasted in a table format. Compared to other data mining methods, decision trees and k-mean clustering demonstrate excellent weather prediction performance with better prediction accuracy. [6].

Through a thorough analysis of the literature, the most widely used AI-based techniques for forecasting air pollution, such as Deep Neural Networks (DNN), Artificial Neural Networks (ANN), Fuzzy Logic (SLR) and Support Vector Machines (SVM), are fully summarised in this article. The chosen 90 papers were put out between 2003 and 2021 in total. The SLR seeks to categorise research on AI-based air pollution forecasting from a variety of angles, including the parameters input, relative frequency of utilisation of AI techniques, journal, effectiveness, year of publication, and geographic distribution. Additionally, it tackles the crucial survey issues in this area.. The findings indicated that there have been more articles and links in recent years. The air condition and the temperature are the two most commonly used input parameters. This review paper demonstrates how AI-based methods have

reignited attention to predicting air pollution and hold great promise for radically altering the field's practice in the near future. [7]. In this analysis, we concentrate on the most recent advancements in adsorption and photo-catalytic decomposition-based interior air pollution reduction materials. First, we examine the main interior contaminants, such as formaldehyde and benzene. The adoption of modern adsorption materials, such as conventional biochar and MOF (metal-organic frameworks), the next generation of porous materials, is further addressed. Typical biochar, PM_{2.5}, flame retardants, and plasticizer are some of them. VOC absorption is investigated. We discuss the procedure for VOCs mitigation using biochar (noncarbonized organic matter partition and adsorption) and MOF along with aspects that impact interior air pollution elimination efficiency based on present mitigation approaches, such as the mitigation of VOCs using photocatalytic oxidation. Finally, we present views and guidelines for the advancement of devices that reduce indoor air pollution. [8]. The current study provides information on the systems and sensors that are accessible for AQ evaluation, management, and tracking. First, we found the present state of study in AQ management by searching the published papers using specific keywords such as Particulate Matter (PM), Sulphur di-Oxide (SO₂), Carbon Mono-oxide (CO), AQM and Nitrogen di-Oxide (NO₂). Based on our own conceptualization of five main categories, We looked at the various air quality monitoring devices and systems, including satellite-based sensors, airborne sensors, ground-based digital sensors, and combination systems, as well as ground-based Wet Chemistry (AQS) systems[9]. In regions where no observations have been taken, LUR models use data about human and natural variables like industry, heating, and vehicles to forecast air pollution. To make accurate forecasts, however, current methods frequently need laborious hyper-parameter tuning that is not always possible. In this study, we resolve these issues through the introduction of Open-LUR, a commercially available method for modelling air pollution that (i) depends

only on a set of innovative characteristics taken from the freely accessible global data source OpenStreetMap; and (ii) is built on state-of-the-art machine learning algorithms with automated hyper-parameter tailoring that minimises manual effort [10].

2.1.2 The harmful effects of air pollution on people

The analysis shows that air pollution in Europe still has a significant negative impact on health at present levels. Exposure to air pollution has been related to a wide range of negative impacts, including a higher chance of cardiopulmonary disease and a one-year or more decrease in life expectancy for individuals residing in European cities. A few of these impacts happen at extremely low concentrations that were once thought to be harmless. Considering all available data, it is highly advised that additional policy measures be taken to lower levels of air contaminants like particulates, nitrogen dioxide, and ozone. It is fair to presume that a decrease in air pollution will have a significant positive impact on health. (WHO 2004) [11]. The goal of this article is to review the body of knowledge on air pollution and behaviour, and to present an organisational structure that highlights knowledge gaps and establishes a preliminary conceptual foundation for further investigation. The paper addresses studies on air pollution's impacts on human health as well as its cognitive, affective, and conative aspects. Additionally, the crucial functions of mediational notions like pollutant controllability and adaptation are thoroughly explored.[12]. Exposure to air pollution causes a significant percentage of the worldwide burden of illness, either directly or indirectly. Exposures that take place during the fast organogenesis and lung development stages of foetal development and the early postnatal era are particularly harmful. In this State-of-the-Art review, we go over air toxicants that affect children's respiratory health, exposure pathways with a focus on particular ones that apply to early children, techniques of exposure evaluation and their constraints, and the

negative health effects of exposures. Lastly, we identify any knowledge gaps and suggest future areas for study. [13].

Since their discovery more than a century ago, negative air ions (NAIs) have been extensively used to purify the air. Here, we've conducted a thorough analysis of the impact of NAIs on people, animals, microbes, and plant growth. Although there is conflicting evidence regarding the anti-microorganism effects and inconsistent or dependable evidence regarding therapeutic effects, NAIs are thought to improve psychic health, efficiency, and general well-being. Additionally, studies revealed that NAIs may aid in easing the signs of sensitivities to pollen, mould spores, and other pollutants. One of the main atmospheric pollutants that has an impact on human health is particulate matter (PM). Testing results demonstrated that PM could be effectively removed using NAIs. Lastly, we looked at the pulsed electric field-based plant-based NAI release device.[14]. Human health and the environment are closely related. Numerous researches on the effects of pollutants in the air impact on human health and wellbeing have been done over the past 50 years. The general consensus is that breathing in polluted air increases one's chance of developing a variety of diseases, such as lung cancer, coronary artery disease, and breathing difficulties. Additionally, air pollution affects plants and has an adverse effect on animals. Examining previous research on the origins, sources, and ways to avoid air pollution as well as its detrimental consequences on human health is the major objective of this research. The description of air pollution and its causes were fully explained. The most dangerous air contaminants and their effects were described in depth. [15].

2.2 FUNCTIONAL REQUIREMENTS

For the implementation of the project, following things would be used.

1. Technical

Hardware Requirements

- Laptop/Desktop that supports Windows / MacOS / Android OS or any other, with Minimum 2 GB RAM
- A browser which supports HTML & CSS.
- Stable internet connectivity Software/Framework/Tools

Software Requirements

- Visual Studio Code
- Python 3.x

2. Legal

- We have taken our data from Central Control Room for Air Quality Management (<https://app.cpcbcr.com/>).
- It gives us the data for specific parts of Delhi region.

3. Operational

- Our project will predict the data and notify the user about a certain PM (particulate matter) crossing the threshold value.

4. Schedule

- Estimated time to complete the whole the project will be 1 year. High likelihood of completion on time, resulting in a high assessment of schedule viability.

2.3 Proposed System

The proposed system for the time series forecasting project in Delhi using SARIMAX, LSTM, and Prophet would be a machine learning and deep learning-based system that takes historical air quality data as input and produces accurate forecasts of pollutant levels in the city. The system will involve developing and training the three models, namely SARIMAX, LSTM, and Prophet, using the pre-processed air quality data. Each model will take the decomposed time series data as input and use it to predict future levels of pollutants. The models will be evaluated based on their performance using metric such RMSE, and the model that provides the most accurate forecasts will be selected. This model will be used to generate real-time forecasts of air quality in Delhi. The system will be useful for policymakers, researchers, and citizens to make informed decisions and take appropriate actions to mitigate the adverse effects of air pollution. The proposed system will be able to handle missing data and irregularities in the input data and will be capable of predicting pollutant levels for multiple time periods in advance. It will also be able to incorporate weather and traffic patterns are examples of outside variables that may have an impact on the air quality.

2.4 Feasibility Study

- **Focus on Delhi:** Our project is specifically focused on air quality forecasting for Delhi, which is a city with unique environmental and demographic characteristics. By developing models that are tailored to Delhi's specific conditions, we can provide more accurate and useful forecasts for the city's residents.
- **Thorough Feasibility Study:** We have conducted a thorough feasibility study to ensure that the project is viable and can be successfully completed. Our study has considered all aspects, including technical, data, resource, economic, and legal feasibility, to ensure that our project can deliver on its promises.
- **Potential for Impact:** By providing accurate and reliable air quality forecasts, the initiative has the potential to have a big influence on Delhi residents' lives. By empowering individuals and communities with information about air quality, we can enable them to make more informed decisions and take appropriate measures to protect their health and well-being.

CHAPTER 3

SYSTEM DESIGN & ANALYSIS

3.1 Project Perspective

With the application of deep learning and machine learning, Delhi's air quality will be forecasted throughout time in this study. To forecast future levels of pollutants in the city, the models will utilise previous data on air quality. Three models will be implemented, namely SARIMAX, LSTM, and Prophet, and their performances will be evaluated to determine which model provides the most accurate forecasts.

3.2 Methodology

Let us discuss about this project's working and framework in detail. This section will provide information about each step of the process, going from data collection and various operations performed on the dataset to make it compatible for the analysis and further implementation.

We will be using ARIMA model which is a time series analysis model widely used for forecasting purposes. This model is further explained in detail in the below sections. The main goal here is to predict and forecast levels of particulate matters and Air Quality Index. The main steps of our approach are represented by the following diagram.

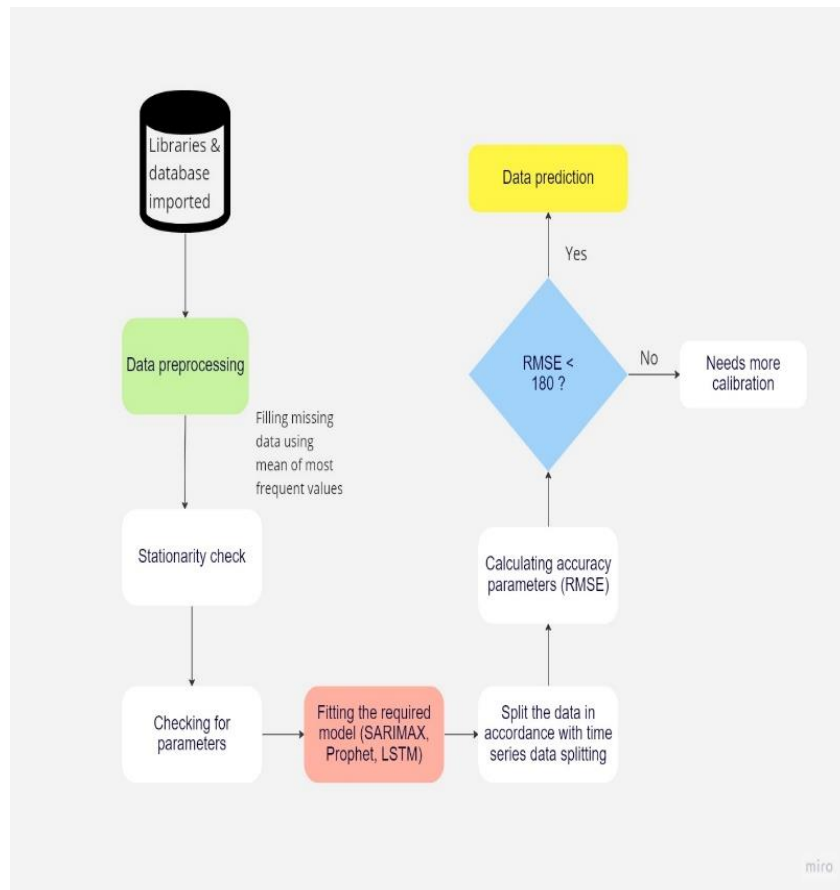


Fig 3.2.1: Proposed Methodology for predicting Air Quality components

3.3 Data Collection

The dataset collected that will be used for all further operations is from CPCB: Central Pollution and Control Board. They keep their databases frequently up to date. It consists of 29531 rows of data. The columns represent features such as date, cities, AQI value and concentration of each particulate matter.

```
data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 29531 entries, 0 to 29530
Data columns (total 16 columns):
#   Column          Non-Null Count  Dtype  
---  -
0   City             29531 non-null  object  
1   Date             29531 non-null  object  
2   PM2.5            24933 non-null  float64  
3   PM10             18391 non-null  float64  
4   NO               25949 non-null  float64  
5   NO2              25946 non-null  float64  
6   NOx              25346 non-null  float64  
7   NH3              19203 non-null  float64  
8   CO               27472 non-null  float64  
9   SO2              25677 non-null  float64  
10  O3               25509 non-null  float64  
11  Benzene          23908 non-null  float64  
12  Toluene          21490 non-null  float64  
13  Xylene           11422 non-null  float64  
14  AQI              24850 non-null  float64  
15  AQI_Bucket       24850 non-null  object  
dtypes: float64(13), object(3)
memory usage: 3.6+ MB
```

Fig 3.3.1: Features of Dataset

3.4 Data Pre-Processing

The primary goal of data pre-processing is to improve visualization which provides better analysis of the data and so the machine learning models can be applied better. Some of the techniques we used to handle missing data are as follows:

Handle missing values: Since there is no advantage of performing data visualisation or any other operation on parameters with a lot of missing values, we decided to fix them. Firstly, we combined the columns of three specific pollutants namely Benzene, Toluene and Xylene and named the merged column as BTX since in petroleum industry which is the source of production of BTX usually uses this term to measure the overall concentration of these hydrocarbons.

Then to focus on more significant pollutants we performed some other common data handling techniques such as deleting all the rows that have NULL values. But this could lead to problems like data distortion and loss of data as some columns had empty values which after eliminating NULL values and grouping the data for monthly plot, can lead to more NULL values. Hence by process of elimination, filling missing values using most frequent values will be the best approach here. Replacing missing values with mean is the approach that was chosen here because it worked as the NULL values were filled with some approximation values. Mode would fail in this case as the resultant mode of a set with mostly NULL values will also be a NULL value.

```
Index(['Date', 'City', 'AQI', 'AQI_Bucket', 'PM2.5', 'PM10', 'NO2', 'CO',
      'SO2', 'O3', 'BTX'],
      dtype='object')
```

[] data

	Date	City	AQI	AQI_Bucket	PM2.5	PM10	NO2	CO	SO2	O3	BTX
0	2015-01-01	Ahmedabad	NaN	NaN	NaN	NaN	18.22	0.92	27.64	133.36	0.02
1	2015-01-02	Ahmedabad	NaN	NaN	NaN	NaN	15.69	0.97	24.55	34.06	12.95
2	2015-01-03	Ahmedabad	NaN	NaN	NaN	NaN	19.30	17.40	29.07	30.70	25.45
3	2015-01-04	Ahmedabad	NaN	NaN	NaN	NaN	18.48	1.70	18.59	36.08	15.57
4	2015-01-05	Ahmedabad	NaN	NaN	NaN	NaN	21.42	22.10	39.33	39.31	28.68
...
29526	2020-06-27	Visakhapatnam	41.0	Good	15.02	50.94	25.06	0.47	8.55	23.30	15.04
29527	2020-06-28	Visakhapatnam	70.0	Satisfactory	24.38	74.09	26.06	0.52	12.72	30.14	3.33
29528	2020-06-29	Visakhapatnam	68.0	Satisfactory	22.91	65.73	29.53	0.48	8.42	30.96	0.02
29529	2020-06-30	Visakhapatnam	54.0	Satisfactory	16.64	49.97	29.26	0.52	9.84	28.30	0.00
29530	2020-07-01	Visakhapatnam	50.0	Good	15.00	66.00	26.85	0.59	2.10	17.05	NaN

29531 rows x 11 columns

Fig 3.4.1: Representation of dataset

3.5 Splitting the data

After selecting a feature on which the data model will be applied, in this case the features are PM10. Knowing that we are working with time series data, the splitting of dataset into train set, and test set is done based on timestamps which is symmetrical in nature. If we want to split the data using the 80:20 rule, it should be done in such a way that the 80 percent dataset covers the period 2015-2018 and the rest is 2018-2019. Time Series data splitting is the name of this type of data splitting [16].

3.6 Models Used

All these models were implemented, and the results were compared

3.6.1 SARIMAX

A statistical time series forecasting technique called SARIMAX (Seasonal Autoregressive Integrated Moving Average with Exogenous factors) extends the ARIMA (Autoregressive Integrated Moving Average) model to add seasonal components and exogenous factors.

SARIMAX can be helpful for predicting time series data that include seasonal patterns, trends, and the impacts of additional variables.

It has been applied in various fields, such as finance, economics, and environmental studies, for predicting stock prices, GDP, and air pollution levels, among other things.

```
import statsmodels.api as sm

# Load data
data = load_data()

# Split data into training and testing sets
train_data = data[:'2019-12-31']
test_data = data['2020-01-01':]

# Define SARIMAX model parameters
order = (p, d, q)
seasonal_order = (P, D, Q, S)
exog_vars = [var1, var2, var3]

# Create SARIMAX model
model = sm.tsa.statespace.SARIMAX(train_data, order=order, seasonal_order=seasonal_order, exog=exog_vars)

# Fit SARIMAX model to training data
model_fit = model.fit()

# Use model to make predictions on test data
predictions = model_fit.predict(start=test_data.index[0], end=test_data.index[-1], dtype='O')

# Evaluate performance of model using appropriate metrics
accuracy = evaluate_model(predictions, test_data)
```

Fig 3.6.1.1 Pseudo Code for SARIMAX Model

3.6.2 PROPHET

Prophet is a time series forecasting tool developed by Facebook's Core Data Science team. It is an open-source tool that utilizes a decomposable time series model with three main components: trend, seasonality, and holidays. Prophet's strengths lie in its simplicity and ease of use, as well as its ability to produce accurate forecasts quickly, even with limited data. It can also handle non-linear trends and seasonality, and automatically adjusts for changes in trend and seasonality over time.

Prophet has been applied in various fields, including finance, retail, and weather forecasting, and is particularly useful in predicting time series data with short-term fluctuations and multiple seasonalities.

```
# Import the necessary libraries
import pandas as pd
from fbprophet import Prophet

# Load the air quality data
air_quality_df = pd.read_csv('air_quality_data.csv')

# Preprocess the data
air_quality_df = air_quality_df.rename(columns={'Date': 'ds', 'AQI': 'y'})

# Train the Prophet model
model = Prophet()
model.fit(air_quality_df)

# Make predictions for the next year
future = model.make_future_dataframe(periods=365)
forecast = model.predict(future)

# Plot the forecast
model.plot(forecast)
```

Fig 3.6.2.1 Pseudo Code for Prophet

3.6.3 LSTM

A recurrent neural network (RNN) called LSTM (Long Short-Term Memory) is frequently used for time series forecasting. It is intended to solve the drawbacks of conventional RNNs, which, as a result of the vanishing gradient problem, are unable to capture long-term relationships. LSTMs use a series of memory cells that allow them to selectively forget or remember information over time. They are able to learn patterns and relationships in sequential data, and can predict future values based on past observations. Because they are able to manage both linear and nonlinear relationships and capture complex patterns in the data like seasonality, trends, and abrupt changes, LSTMs have been particularly helpful in time series forecasting. Numerous industries, including finance, weather forecasting, and energy load forecasting, have successfully used LSTMs.

```

# Create the training and testing datasets
def create_dataset(dataset, look_back=1):
    X, Y = [], []
    for i in range(len(dataset)-look_back-1):
        a = dataset[i:(i+look_back), 0]
        X.append(a)
        Y.append(dataset[i + look_back, 0])
    return np.array(X), np.array(Y)

train_X, train_Y = create_dataset(train_data, look_back)
test_X, test_Y = create_dataset(test_data, look_back)

# Reshape the data for LSTM input
train_X = np.reshape(train_X, (train_X.shape[0], 1, train_X.shape[1]))
test_X = np.reshape(test_X, (test_X.shape[0], 1, test_X.shape[1]))

# Define the LSTM model
model = Sequential()
model.add(LSTM(50, input_shape=(1, look_back)))
model.add(Dense(1))
model.compile(loss='mean_squared_error', optimizer='adam')

# Train the model
model.fit(train_X, train_Y, epochs=100, batch_size=1, verbose=2)

# Make predictions
train_predict = model.predict(train_X)
test_predict = model.predict(test_X)

# Inverse transform the predictions
train_predict = scaler.inverse_transform(train_predict)
train_Y = scaler.inverse_transform([train_Y])
test_predict = scaler.inverse_transform(test_predict)
test_Y = scaler.inverse_transform([test_Y])

```

Fig 3.6.3 Pseudo Code for LSTM

CHAPTER 4

ALGORITHMS

Machine learning algorithms used for making predictions operate by training on past data, and examples of such algorithms include linear regression, deep learning, Bayesian algorithms and artificial neural networks [17]. All these models were implemented, and the results were compared.

4.1 ARIMA and SARIMAX models

Auto Regressive Integrated Moving Average, or simply ARIMA also known as Box-Jenkins model, is a group of models that use past values of a time series (its own lags and lagged forecast errors) to explain and forecast future values. Essentially, ARIMA predicts a time series based on its own history.

Following is the description of the keywords of the acronym that are used to describe the ARIMA model [18]:

- **AR:** Autoregression. It uses the prior/lagged observations to describe how it influences the present values. Also, referenced as the order of p .
- **I:** Integrated. It is used to count the number of times differencing takes place. Differencing is subtracting an observation from its prior observation to achieve stationarity. Also, referenced as the order of d .
- **MA:** Moving Average. Implements the relationship between a measurement and the residual error from a moving average model used with lag-time measurements. Known also as the order of q .

General representation of the ARIMA is as following:

$$\text{ARIMA}(p, d, q) = \text{AR}(p) + \text{I}(d) + \text{MA}(q)$$

Since in this case the data consists of seasonality and trends, it brings us to use a model that has better compatibility with those traits. SARIMAX is an expanded variation of the SARIMA model and stands for Seasonal Auto Regressive Integrated Moving Average with Exogenous Factors with the added feature to handle exogenous variables which are input sequences that run parallel to the original series and have observations at the same time steps. In general, ARIMA is a good type of model. However, the vanilla version has a major weakness in that it is unable to deal with seasonality [19].

The general formula for SARIMAX is:

$$\text{SARIMAX}(p, d, q) \times (P, D, Q, s) + c = \text{AR}(p) + \text{I}(d) + \text{MA}(q) \times (\text{AR}(P) + \text{I}(D) + \text{MA}(Q)s) + \epsilon_t$$

Where,

- $\text{AR}(p)$ = autoregressive element of order p
- $\text{I}(d)$ = differencing element of order d
- $\text{MA}(q)$ = order q 's moving average component
- $\text{AR}(P)$ = order P 's seasonal autoregressive component
- $\text{I}(D)$ = seasonal differencing component of order D
- $\text{MA}(Q)$ = seasonal moving average component of order Q
- s = the seasonal span of the time series
- c = constant
- ϵ_t = the error term at time t

4.1.1 SARIMAX Algorithm

Input:

- Time series data (y)
- Exogenous variables (x)
- Non-seasonal ARIMA order parameters (d, p, q)
- SARIMA order parameters (D, Q, S, P)
- Number of forecasted periods (n)

Output:

- Forecasted values (y_{hat})

Steps:

1. Divide the data into sets for training and testing.
2. Create a SARIMAX model using the non-seasonal and seasonal ARIMA order parameters and the exogenous variables. Use the training data to fit the model.
3. Make predictions based on the test data using the fitted model.
4. Evaluate the performance of the model using appropriate metrics.
5. If the performance is satisfactory, proceed to the next step. Otherwise, adjust the model parameters and repeat steps 2-4 until satisfactory performance is achieved.
6. Fit the model to the entire dataset (including both the training and testing data).
7. Using the fitted model, forecast future values for the time series for n periods.
Exogenous variables should be considered during these periods to come.
8. Return the forecasted values (y_{hat}).

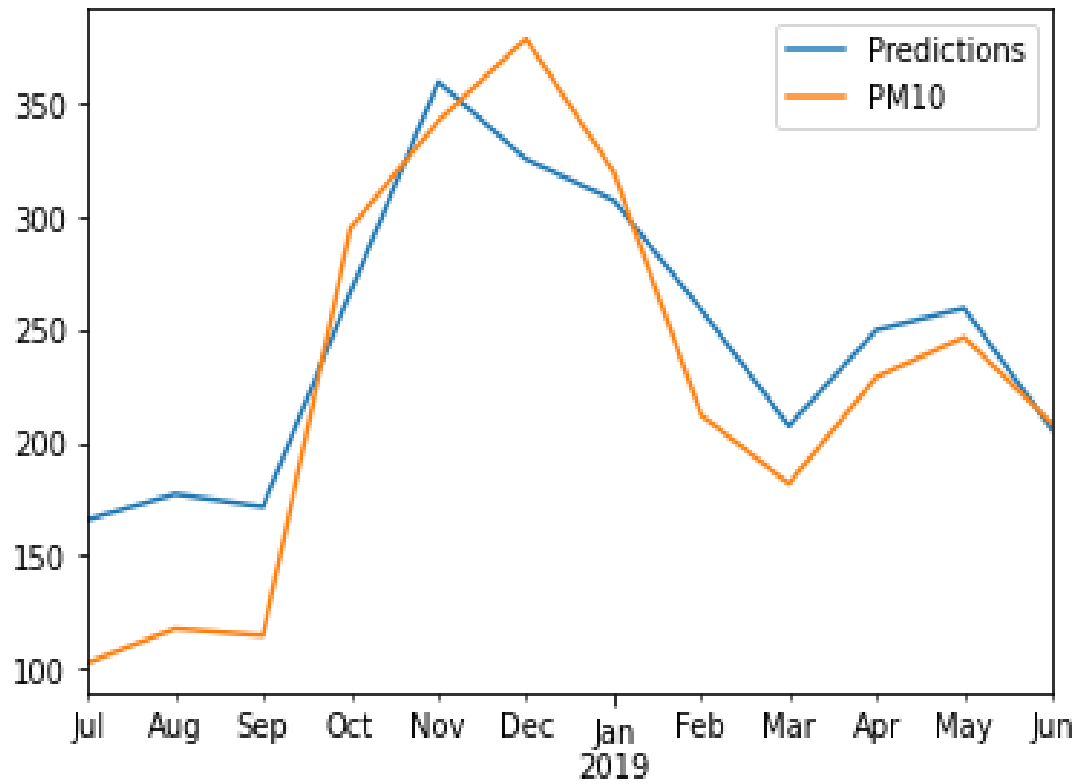


Fig. 4.1.1.1: Monthly PM10 Original vs Predicted

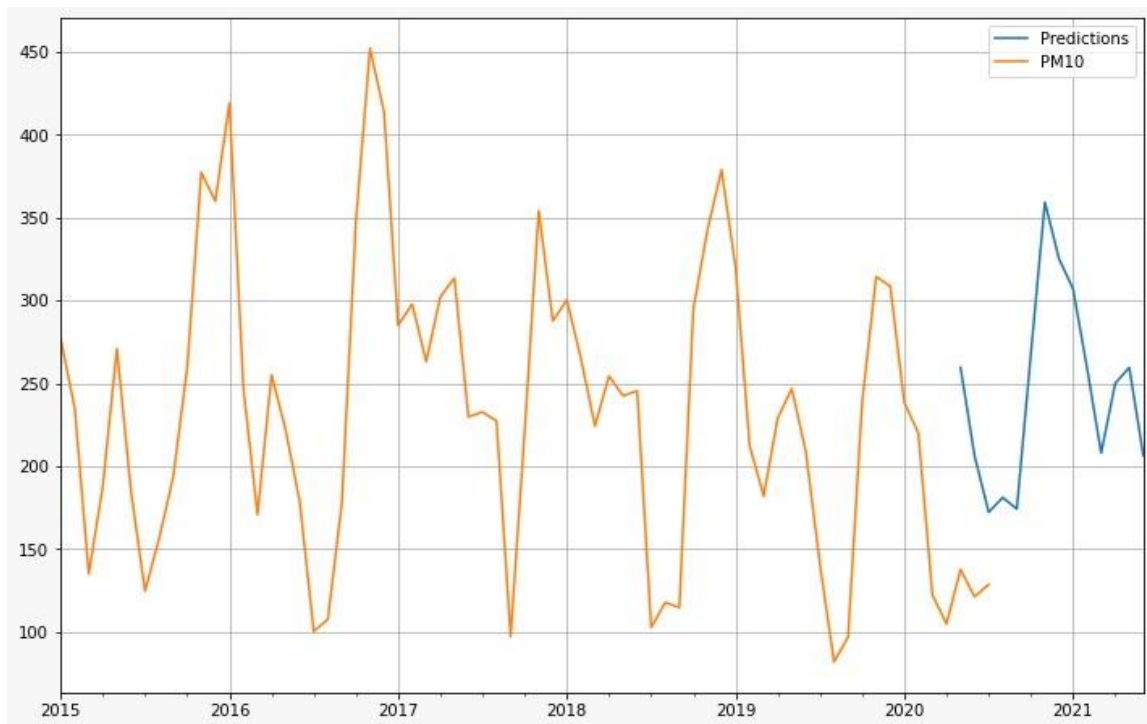


Fig 4.1.1.2: Yearly PM10 Original vs Predicted

4.2 Prophet

The Prophet model is a sophisticated model that incorporates a number of elements to simulate time series data [20]. However, here are the basic formulas for the Prophet model:

The Prophet model's Trend component employs a model based on a logistic regression model or piecewise linear, which is formulated as:

$$\text{Trend}(t) = g(t) + s(t) + h(t) + \epsilon t$$

Here,

- $g(t)$: time series data's overall tendency across time is modelled by this trend function.
- $s(t)$: seasonal function that models the repetitive patterns or cycles in the time series data.
- $h(t)$: holiday function that models the influence of known holidays and special events on the time series data.
- $E(t)$: error term that represents the unexplained variability or randomness in the time series data.

The seasonality component models the periodic fluctuations in the time series data and can be modelled using several approaches, including Fourier series, Gaussian process regression, and ARIMA models. One common approach is to model the seasonality component using Fourier series, which takes the following form:

$$\text{Seasonality}(t) = \sum_{k=1}^K (a_k * \cos(2\pi kt/P) + b_k * \sin(2\pi kt/P))$$

where:

- K is the number of Fourier terms
- a_k and b_k are the Fourier coefficients
- P is the period of the seasonality component

The holiday component models the impact of known holidays and events on the time series data and can be modelled using indicator variables or regression models. The holiday component takes the following form:

$$\text{Holiday}(t) = \sum_{i=1}^M (x_i * I(t \in S(i)))$$

where:

- The number of holidays is M .
- x_i is the impact of i th holiday
- $S(i)$ is the holiday window for i th holiday
- $I()$ is the indicator function

4.2.1 Prophet Algorithm

1. Import the Prophet library and load the time series data into a panda DataFrame.
2. Rename the columns of the DataFrame to "ds" for the date/time column and "y" for the target variable column.
3. Instantiate a new Prophet object with default parameters.
4. Fit the Prophet model to the data using the fit () method.
5. Create a new DataFrame with future dates to forecast using the make_future_dataframe () method.
6. Use the predict () method to generate forecasts for the future dates.
7. Visualize the forecasted data using the Prophet built-in plotting functionality.

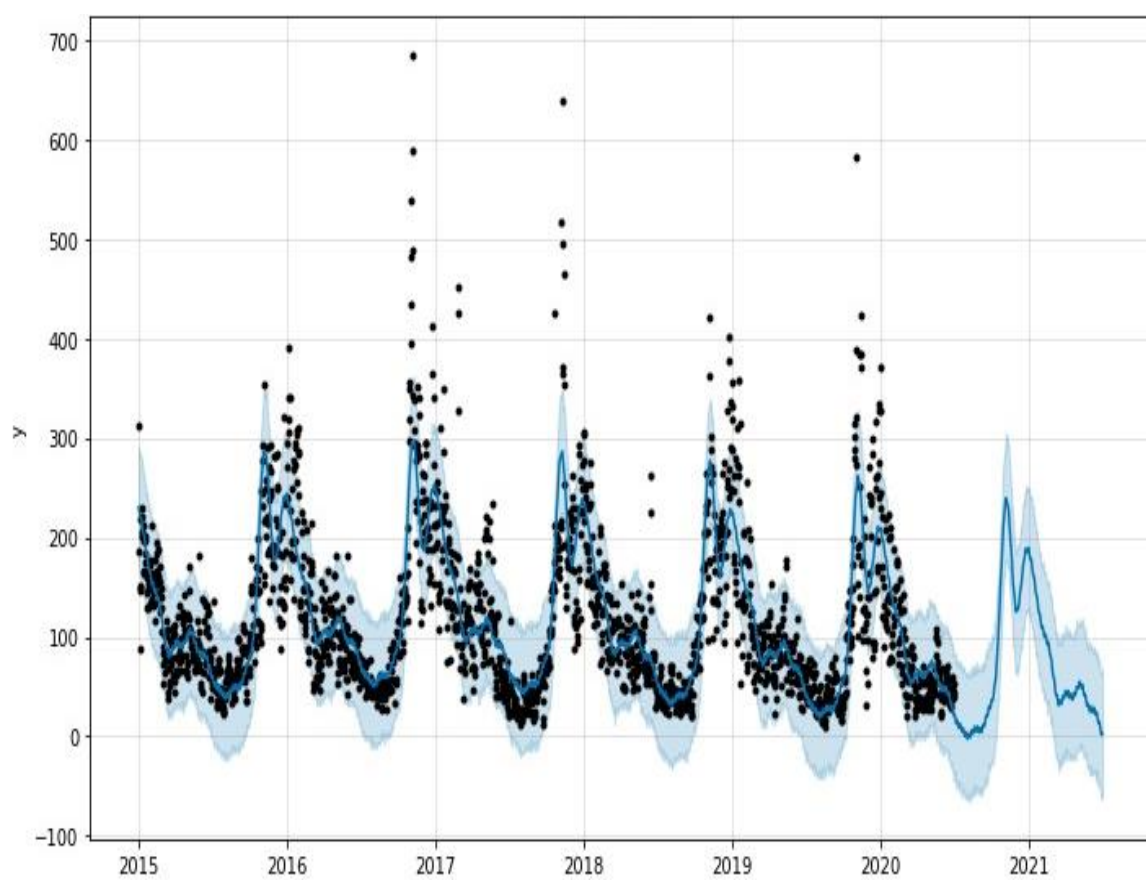


Fig. 4.2.1.2 Prediction of PM 10 Pollutants

4.3 LSTM

Recurrent neural networks (RNNs) of the LSTM (Long Short-Term Memory) form are frequently employed for the processing of sequential data, including time series data. RNNs have the capacity to handle inputs of varying lengths and can retain an internal state that enables them to capture temporal relationships in the data, in contrast to typical feedforward neural networks, which are built for processing fixed-size inputs.

A collection of layers of LSTM cells make up the LSTM network. A memory cell, an input gate, an output gate, and a forget gate are only a few of the internal parts found in each LSTM cell. Information flows into and out of the memory cell are controlled by the input, output, and forget gates [21].

One challenge in using LSTM networks for air quality time series is the need to deal with missing data and other data quality issues. For example, air quality data may not be available for certain time periods or may be affected by sensor malfunctions or other issues. There are various techniques that can be used to address these issues, including data imputation and data interpolation.

4.3.1 LSTM Algorithm

1. Load the time series data into a panda DataFrame.
2. Create training and test sets from the data.
3. Normalize the testing and training data must have a unit variance and a zero mean using a scaler.
4. Reconstruct the training and test data such that it has the following shape: [samples, time steps, features], where samples is the total number of observations, time steps is

the total number of lagged variables to be used, and features is the total number of input variables.

5. Build an LSTM model using Keras.
6. Use the training data to train the LSTM model.
7. Generate predictions for the testing data using the trained LSTM model.
8. Inverse normalize the predicted values using the scaler.
9. Utilise measures like RMSE to assess the LSTM model's performance.

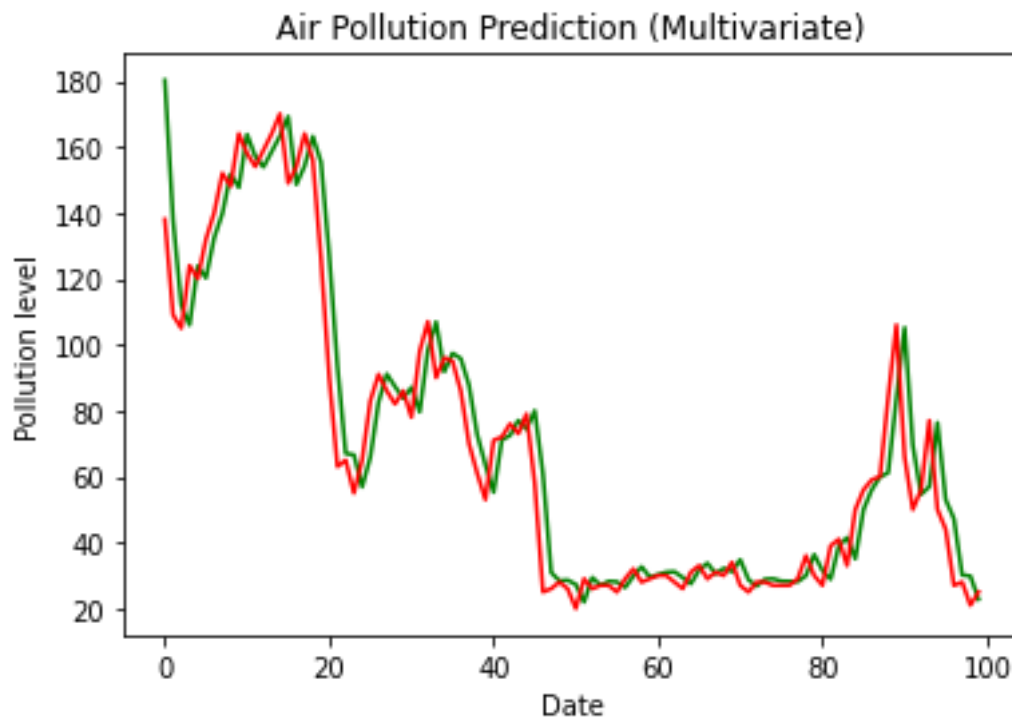


Fig. 4.3.1.1 Prediction of pollution level using LSTM

CHAPTER -5

RESULTS AND OUTPUTS

After implementation we can compare the performance values of these models using RMSE which stands for Root Mean Squared Error. A kind of error measurement known as the root mean squared error computes the square of the deviations to keep the positive and negative deviations from cancelling one another out. This approach can magnify significant errors, making it useful for comparing different methods.

Model	RMSE	Pollutant
SARIMAX	39.1067129688 50374	PM10
Prophet	18.5338172461 02544	PM10
LSTM	28.9867676267 148	PM10

Table 5.1 Evaluation Metrics

The above table shows comparisons between the RMSE values which is used for accuracy purposes. SARIMAX and Prophet are very close in terms of performance regarding prediction of future outcome. The Prophet model for PM10 contaminant is fitter and superior in terms of performance regarding prediction of future outcome. LSTM on the hand also performed decent enough but comparatively it yields poor results due to rather complex hyperparameters calibration. Due to dissimilar trend, the SARIMAX model does not perform as it could have been.

CHAPTER - 6

CONCLUSION AND FUTURE SCOPE

From the experiments performed above it can be concluded that in our instance of univariate time series forecasting, Prophet model performed better than its competitors here. If the LSTM model's Root Mean Squared Error (RMSE) score is not as expected, it may be beneficial to try training the model for additional epochs. By continuing to train the model, it may be able to capture more of the necessary time series-related information and result in a lower loss.

In regulation to ensure the efficiency and effectiveness of the best models, they should be examined and assessed using various datasets. With the SARIMAX model, the computational constraints and continuity of data degraded the performance of the model. SARIMAX could be enhanced by adjusting its parameters to improve its performance. Additionally, it would be beneficial to expand the scope of the study beyond just Delhi and capture the levels of air pollution in other locations as well.

CHAPTER 7

REFERENCES

- [1] Abhishek, K., Singh, M. P., Ghosh, S., & Anand, A. (2012). Weather forecasting model using artificial neural network. *Procedia Technology*, 4, 311-318.
- [2] Kumar, A., & Goyal, P. (2011). Forecasting of air quality in Delhi using principal component regression technique. *Atmospheric Pollution Research*, 2(4), 436-444.
- [3] Latha, C. B. C., Paul, S., Kirubakaran, E., & Sathianarayanan, A. (2010). A service-oriented architecture for weather forecasting using data mining. *Int. J. of Advanced Networking and Applications*, 2(2), 608-613.
- [4] Tektaş, M. (2010). Weather forecasting using ANFIS and ARIMA models. *Environmental Research, Engineering and Management*, 51(1), 5-10.
- [5] Krishna, G. V. (2015). An integrated approach for weather forecasting based on data mining and forecasting analysis. *International Journal of Computer Applications*, 120(11)
- [6] Chauhan, D., & Thakur, J. (2014). Data mining techniques for weather prediction: A review. *International Journal on Recent and Innovation Trends in Computing and Communication*, 2(8), 2184-2189.
- [7]. Masood, A., & Ahmad, K. (2021). A review on emerging artificial intelligence (AI) techniques for air pollution forecasting: Fundamentals, application and performance. *Journal of Cleaner Production*, 322, 129072.
- [8]. Yue, X., Ma, N. L., Sonne, C., Guan, R., Lam, S. S., Van Le, Q., ... & Peng, W. (2021). Mitigation of indoor air pollution: A review of recent advances in adsorption materials and catalytic oxidation. *Journal of hazardous materials*, 405, 124138.

- [9]. Singh, D., Dahiya, M., Kumar, R., & Nanda, C. (2021). Sensors and systems for air quality assessment monitoring and management: A review. *Journal of environmental management*, 289, 112510.
- [10]. Lautenschlager, F., Becker, M., Kobs, K., Steininger, M., Davidson, P., Krause, A., & Hotho, A. (2020). OpenLUR: Off-the-shelf air pollution modeling with open features and machine learning. *Atmospheric environment*, 233, 117535.
- [11]. World Health Organization. Regional Office for Europe. (2004). Health aspects of air pollution: results from the WHO project "Systematic review of health aspects of air pollution in Europe". Copenhagen: WHO Regional Office for Europe.
- [12]. Evans, G. W., & Jacobs, S. V. (1981). Air pollution and human behavior. *Journal of Social Issues*, 37(1), 95-125.
- [13]. Goldizen, F. C., Sly, P. D., & Knibbs, L. D. (2016). Respiratory effects of air pollution on children. *Pediatric pulmonology*, 51(1), 94-108.
- [14]. Jiang, S. Y., Ma, A., & Ramachandran, S. (2018). Negative air ions and their effects on human health and air quality improvement. *International journal of molecular sciences*, 19(10), 2966.
- [15]. Almetwally, A. A., Bin-Jumah, M., & Allam, A. A. (2020). Ambient air pollution and its influence on human health and welfare: an overview. *Environmental Science and Pollution Research*, 27, 24815-24830.
- [16]. Bose, M., & Mali, K. (2018). A novel data partitioning and rule selection technique for modeling high-order fuzzy time series. *Applied Soft Computing*, 63, 87-96.
- [17] Battineni, G., Chintalapudi, N., & Amenta, F. (2020). Forecasting of COVID-19 epidemic size in four high hitting nations (USA, Brazil, India and Russia) by Fb-Prophet machine learning model. *Applied Computing and Informatics*, (ahead-of-print).

- [18] Llop, S., Ballester, F., Estarlich, M., Esplugues, A., Fernández-Patier, R., Ramón, R., ... & Iñiguez, C. (2008). Ambient air pollution and annoyance responses from pregnant women. *Atmospheric Environment*, 42(13), 2982-2992
- [19] Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2018, December). A comparison of ARIMA and LSTM in forecasting time series. In *2018 17th IEEE international conference on machine learning and applications (ICMLA)* (pp. 1394-1401). IEEE.
- [20] Yenidoğan, I., Çayır, A., Kozan, O., Dağ, T., & Arslan, Ç. (2018, September). Bitcoin forecasting using ARIMA and PROPHET. In *2018 3rd international conference on computer science and engineering (UBMK)* (pp. 621-624). IEEE.
- [21] Elsworth, S., & Güttel, S. (2020). Time series forecasting using LSTM networks: A symbolic approach. *arXiv preprint arXiv:2003.05672*
- [22]. Darekar, P. (2021). *A Comprehensive Study to Forecast the Delhi and Bangalore Cities Air Pollution using Machine Learning Models* (Doctoral dissertation, Dublin, National College of Ireland).

ANNEXURE I

Conference paper for the said project has been **published** in International Journal of Engineering Research & Technology (IJERT).

Paper Title:

Air Pollution Prediction using Machine Learning Algorithms: A Systematic Review

Abstract:


The quality of the air has been steadily degrading over the last few years, which has raised the number of serious health issues. The government and researchers have taken a particular interest in creating and implementing methods and technologies that can not only aid monitoring present air quality but also being able to forecast the result due to this rapid rise. This study reviews several intriguing modelling strategies based on their general functionality, benefits, drawbacks, etc. Additionally, several data processing techniques are covered in order to increase the general efficiency of any model.

Authors: Vidit Kumar, Sparsh Singh, Zaid Ahmed, Nikita Verma

<https://www.ijert.org/air-pollution-prediction-using-machine-learning-algorithms-a-systematic-review>

VOLUME 11, ISSUE 12 (DECEMBER 2022)

Air Pollution Prediction using Machine Learning Algorithms: A Systematic Review

 DOI : 10.17577/IJERTV11IS120026

[DOWNLOAD FULL-TEXT PDF](#)

[CITE THIS PUBLICATION](#)



- Open Access 
- Article Download / Views: 433
- **Authors :** Vidit Kumar , Sparsh Singh , Zaid Ahmed , Nikita Verma
- **Paper ID :** IJERTV11IS120026
- **Volume & Issue :** [Volume 11, Issue 12 \(December 2022\)](#)
- **Published (First Online):** 13-12-2022
- **ISSN (Online) :** 2278-0181
- **Publisher Name :** IJERT
- **License:**  This work is licensed under a [Creative Commons Attribution 4.0 International License](#)

ANNEXURE II

Conference paper for the said project has been **accepted** in *ICDT 2023*.

Paper Title:

Delhi Air Pollution Prediction: A Comparative analysis using Time Series forecasting

Abstract:

Recent years have seen a substantial increase in study on air pollution as a result of its negative ramifications. It is also acknowledged as one in the current atmosphere, one of the main risk elements. Accurate air quality assessment is the first stage in the implementation of air pollution control systems, which helps in the growth of developed nations' economies and societies. The two methodical Accurate air quality predictions are essential for emissions control, public health, and wellbeing. The metropolis of India, Delhi, has been the most polluted metropolis in the world for the past two years. In this study, the results have been compared using SARIMAX, Prophet, and LSTM are three distinct machine learning algorithms that were tested against one another. The same quantity of parameter calibration was applied to all the models, and SARIMAX seems to be more reliable out of the three.

Authors: Sparsh Singh, Vidit Kumar, Zaid Ahmed, Kajol Mittal

Dear Sparsh Singh,
Greetings from ICDT Team

We are pleased to inform you that your Manuscript id_ "583" is entitled "Delhi Air Pollution Prediction: A Comparative analysis using Time Series forecasting". The revisions that you made to the manuscript are very effective in addressing the remaining concerns. Our decision is to Accept the manuscript for publication in the IEEE International Conference on Disruptive Technologies (ICDT-2023). Please consider this letter notification of the formal acceptance of your manuscript. Thank you for your patience with the review process.

GitHub Link:

<https://github.com/SPSingh3000/time-series-weather>