

Dissertation on

**Monitoring the concentration of air pollutants and its
health hazards using Machine Learning models**

Submitted in partial fulfilment of the requirements for the award of degree of

**Bachelor of Technology
in
Computer Science & Engineering**

UE20CS461A – Capstone Project Phase - 1

Submitted by:

Aditi Jain	PES2UG20CS021
Aditya R Shenoy	PES2UG20CS025
Ananya Adiga	PES2UG20CS043
Anirudha Anekal	PES2UG20CS051

Under the guidance of

Prof. Saritha
Assistant Professor
PES University

June - Nov 2023

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF ENGINEERING
PES UNIVERSITY**

(Established under Karnataka Act No. 16 of 2013)
Electronic City, Hosur Road, Bengaluru – 560 100, Karnataka, India

PES UNIVERSITY
(Established under Karnataka Act No. 16 of 2013)
Electronic City, Hosur Road, Bengaluru – 560 100, Karnataka, India

FACULTY OF ENGINEERING CERTIFICATE

This is to certify that the dissertation entitled

Monitoring the concentration of air pollutants and its health hazards using Machine Learning models

is a bonafide work carried out by

Aditi Jain	PES2UG20CS021
Aditya R Shenoy	PES2UG20CS025
Ananya Adiga	PES2UG20CS043
Anirudha Anekal	PES2UG20CS051

In partial fulfilment for the completion of seventh semester Capstone Project Phase - 2 (UE20CS461A) in the Program of Study -Bachelor of Technology in Computer Science and Engineering under rules and regulations of PES University, Bengaluru during the period June 2023 – Nov. 2023. It is certified that all corrections / suggestions indicated for internal assessment have been incorporated in the report. The dissertation has been approved as it satisfies the 7th semester academic requirements in respect of project work.

Signature
Prof. Saritha
Assistant Professor

Signature
Dr. Sandesh B J
Chairperson

Signature
Dr. B K Keshavan
Dean of Faculty

External Viva

Name of the Examiners

1. Dr. Sarasvathi V
2. Prof.Komal Baheti

Signature with Date

DECLARATION

We hereby declare that the Capstone Project Phase - 2 entitled **Monitoring the concentration of air pollutants and its health hazards using Machine Learning models** has been carried out by us under the guidance of Prof. Saritha, Assistant Professor, PES University and submitted in partial fulfilment of the course requirements for the award of degree of **Bachelor of Technology in Computer Science and Engineering** of PES University, Bengaluru during the academic semester June – Nov. 2023. The matter embodied in this report has not been submitted to any other university or institution for the award of any degree.

Name	SRN	Signature
Aditi Jain	PES2UG20CS021	
Aditya R Shenoy	PES2UG20CS025	
Ananya Adiga	PES2UG20CS043	
Anirudha Anekal	PES2UG20CS051	

ACKNOWLEDGEMENT

I would like to express my gratitude to Prof. Saritha, Department of Computer Science and Engineering, PES University, for her continuous guidance, assistance, and encouragement throughout the development of this UE20CS461A - Capstone Project Phase – 2.

I am grateful to the Capstone Project Coordinator, Dr. Sarasvathi V, Professor and Dr. Sudeepa Roy Dey, Associate Professor, for organizing, managing, and helping with the entire process.

I take this opportunity to thank Dr. Sandesh B J, Chairperson, Department of Computer Science and Engineering, PES University, for all the knowledge and support I have received from the department. I would like to thank Dr. B.K. Keshavan, Dean of Faculty, PES University for his help.

I am deeply grateful to Dr. M. R. Doreswamy, Chancellor, PES University, Prof. Jawahar Doreswamy, Pro Chancellor – PES University, Dr. Suryaprasad J, Vice-Chancellor, PES University and Prof. Nagarjuna Sadineni, Pro-Vice Chancellor - PES University, for providing to me various opportunities and enlightenment every step of the way. Finally, this project could not have been completed without the continual support and encouragement I have received from my family and friends.

ABSTRACT

Industrialization stands as a predominant contributor to contemporary pollution, and a concerning lack of awareness envelops individuals, rendering them oblivious to the omnipresent threat lingering with each breath they take. Among the plethora of pollutants, PM2.5, PM10 and CO emerge as formidable adversaries to human health, particularly the lungs, serving as the frontline when individuals draw their breath. As a result, it becomes imperative to raise awareness among the masses regarding these pervasive health hazards.

The proposed system assumes the critical role of incessantly monitoring the air quality in the user's vicinity, subsequently alerting them to the potential risk of developing lung cancer. This research-driven initiative primarily aims to augment the efficiency of existing systems, ensuring their adaptability to the present era and the capacity to handle the burgeoning data influx. The system adopts a hybrid model, fusing Adaptive LSTM and ARIMA models, known for their proficiency in dealing with time series data. Their innate adaptability and reliability in the face of dynamic changes bolster the system's capabilities. Following the training phase, the model is deployed on a secure cloud platform to guarantee scalability and accessibility for users, rendering it aptly suited for contemporary demands. The project also incorporates Arduino boards and high-quality sensors to collect real-time air quality data from specific environments, which subsequently informs predictions and enhances the model's continuous learning, ultimately refining its predictive

accuracy. The overarching objective of this undertaking lies in fostering public awareness, refining predictive accuracy, and, in the process, contributing to the potential preservation of lives, as well as fostering a sense of environmental responsibility within governmental and industrial realms for the benefit of future generations.

TABLE OF CONTENTS

Chap No.	Title	Pg No.
1.	INTRODUCTION 1.1 Motivation 1.2 Solution	1 1 1
2.	PROBLEM STATEMENT 2.1 Description	3 3
3.	LITERATURE REVIEW 3.1 Determination of Air Quality Life Index (AQLI) in Medinipur City of West Bengal(India) During 2019 To 2020 : A contextual Study 3.2 The nexus between COVID-19 deaths, air pollution and economic growth in New York state: Evidence from Deep Machine Learning. 3.3 An Application of IoT and Machine Learning to Air Pollution Monitoring in Smart Cities 3.4 Assessment of indoor air quality in academic buildings using IOT and deep learnings 3.5 Household Ventilation May Reduce Effects of Indoor Air Pollutants for Prevention of Lung Cancer: A Case-Control Study in a Chinese Population. 3.6 How Is the Lung Cancer Incidence Rate Associated with Environmental Risks? Machine-Learning-Based Modeling and Benchmarking 3.7 Air pollution and skin diseases: Adverse effects of airborne particulate matter on various skin diseases	4 4 5 6 7 8 9 10

	3.8 The spatial association between environmental pollution and long-term cancer mortality in Italy 3.9 Conclusion to the literature survey	11 12
4.	DATA 4.1 Overview 4.2 AQI Dataset 4.3 Lung cancer Dataset	14 14 15 16
5.	PROJECT REQUIREMENTS SPECIFICATION 5.1 Introduction 5.2 System Requirements 5.2.1 Features 5.2.2 User classes and characteristics 5.2.3 Operating Environment 5.2.4 Software requirements 5.2.5 Hardware requirements 5.2.6 User Interface 5.2.7 Communication interface 5.3 Functional Requirements 5.4 Non-Functional Requirements 5.5 Data Requirements	17 17 17 18 18 19 20 21 21 22 22 23
6.	SYSTEM DESIGN 6.1 Current System 6.2 Design Considerations 6.2.1 Architecture 6.2.2 Detailed flow 6.2.3 Pros and Cons of Using the Hybrid Network	25 25 26 26 28 29

	6.3 Design Description	30
	6.3.1 Use case diagram	30
	6.3.2 Class Diagram	32
	6.3.3 ER Diagram	33
	6.3.4 Deployment Diagram	34
	6.4 Design Details	34
	6.4.1 Novelty	34
	6.4.2 Innovativeness	35
	6.4.3 Reliability	35
	6.4.4 Privacy and Security	36
	6.4.5 Performance	36
	6.4.6. Interoperability	37
	6.4.7 Maintainability	37
	6.5 Constraints, dependencies, and assumptions	38
7.	IMPLEMENTATION AND PSEUDOCODE	40
	7.1. Data Preparation	40
	7.2 Data Visualisation	41
	7.3 ARIMA model implementation	48
	7.4 LSTM model implementation	53
	7.5 IoT station implementation	57
	7.6 Cloud platform	61
8.	RESULTS AND DISCUSSION	63
9.	CONCLUSION AND FUTURE WORK	65
REFERENCES/BIBLIOGRAPHY		67
APPENDIX A DEFINITIONS, ACRONYMS AND ABBREVIATIONS		69
APPENDIX B SUPPORTING DATA		70

LIST OF FIGURES

Figure No.	Title	Page No.
3.7.1	Retention of pollutants into the skin	10
3.8.1	Methodology the paper uses	11
4.2.1	AQI dataset description	15
4.3.1	Lung cancer dataset description	16
6.2.1	Architecture diagram	26
6.2.2	Circuit Diagram	27
6.3.1	Use case diagram	30
6.3.2	Class Diagram	32
6.3.3	ER Diagram	33
6.3.4	Deployment Diagram	34
7.1.1	Data preparation of the AQI dataset obtained from CPCB	40
7.1.2	Dropping null values from AQI dataset obtained from CPCB	41
7.1.3	Data preparation of the lung cancer dataset obtained from data.world	41
7.2.1	Count plot for AQI column in AQI dataset obtained from CPCB	42
7.2.2	AQI vs. PM 2.5 and checking outliers in AQI dataset obtained from CPCB	42
7.2.3	Correlation matrix in AQI dataset obtained from CPCB	43
7.2.4	Visualization of various attributes to check if scaling is necessary in AQI dataset obtained from CPCB	44
7.2.5	Skew plot for PM2.5 in AQI dataset obtained from CPCB	45
7.2.6	AQI vs. PM 10 and checking outliers in AQI dataset obtained from CPCB	45
7.2.7	Skew plot for PM10 in AQI dataset obtained from CPCB	46
7.2.8	Relation of AQI and passive smokers of lung cancer dataset obtained from data.world	46

7.2.9	For checking outliers of lung cancer dataset obtained from data.world	47
7.2.10	Scatter plot to check the risk level based on gender and age of lung cancer dataset obtained from data.world	47
7.3.1	PACF & ACF plots for AQI dataset	48
7.3.2	ADF test for AQI dataset	49
7.3.3	Training, testing split for AQI dataset	49
7.3.4	p,q,d value selection	50
7.3.5	Fitting the model and generating residuals	50
7.3.6	Residuals generated from ARIMA modelling of AQI dataset	51
7.3.7	Error calculation	51
7.3.8	Plot of predicted values	52
7.4.1	Training, testing split of data for lung cancer dataset	52
7.4.2	Scaling of lung cancer dataset to fit the model	53
7.4.3	LSTM model initialisation	54
7.4.4	Reshaping test data	55
7.4.5	Predictions of LSTM model on the lung cancer dataset	55
7.4.6	Error analysis	56
7.5.1	Circuit diagram	58
7.5.2	Real life circuit diagram	58
7.5.3	Arduino IDE code for collecting readings from the sensors and connection to ThingSpeak cloud platform	59-60
7.6.1	Working application on Streamlit	62
10.1.1	Lung Cancer death attributable to Long-Term Ambient Particulate Matter (PM2.5)	70
10.1.2	Tobacco Attributable Cancer trends in Males and Females	71

LIST OF TABLES

Figure No.	Title	Page No.
4.2.1	AQI attributes & descriptions	15
6.2.3	Pros and Cons of Using the Hybrid Network	29

CHAPTER I

INTRODUCTION

1.1 Motivation

The industrialization process has been a significant contributor to the current pollution crisis facing the world. However, many people remain unaware of the dangers associated with poor air quality. PM2.5, PM10, NO₂, SO₂, and CO are examples of pollutants which are particularly harmful to human health, with the lungs being the first point of contact. Particulate matter of a small diameter enters the lungs easily. As such, it is crucial to increase public awareness of these hazards and encourage individuals to take protective measures.

To address this issue, it is necessary to develop and implement air quality monitoring systems. Such systems would provide individuals with real-time information on air quality levels in their surroundings and alert them to potential health risks. By increasing awareness and providing early warnings, that can help individuals take preventative measures and protect their health.

1.2 Solution

Therefore, the proposed system is a continuous air quality monitoring system that will keep track of the air quality in the user's vicinity and alert them to the likelihood of developing lung cancer. This research-based project aims to improve the effectiveness of existing systems to ensure maximum efficiency, taking into account the current volume of data.

The proposed system is designed to monitor the levels of PM2.5, PM10 and CO in the air continuously, providing the user with real-time information about the quality of the air they are breathing. The system will then use this data to assess the user's risk of developing lung cancer and

alert them accordingly. By keeping track of the air quality and providing early warnings, this system can help people take appropriate measures to safeguard their health.

In conclusion, the proposed air quality monitoring system is an essential tool for ensuring people's health and well-being in today's polluted environment. With its ability to continuously monitor the air quality and provide real-time alerts, this system will enable people to take preventive measures to protect their health from the harmful effects of air pollution.

CHAPTER II

PROBLEM STATEMENT

2.1 Description

To address the issue of increasing mortality rate due to air pollution , the model proposed is a hybrid model of Adaptive LSTM and ARIMA deep learning models that can predict the probability of a person being affected by the air quality in his surroundings.

The model will be deployed on a cloud platform which ensures scalability and accessibility to store huge amounts of real time data of air quality collected using arduino boards and high quality IOT sensors. Continuously monitoring air quality, predicting and alerting individuals to potential health risks. These deep learning models have proven to be effective in handling time series data and are easy to implement. Moreover, they have high reliability and can dynamically adjust to changes, making them ideal for real-time monitoring of air quality.

By constantly training on real-time and static data, the model will become increasingly accurate, allowing it to provide more reliable and relevant predictions. This will enable individuals to take preventive measures to protect themselves from the harmful effects of air pollution and make informed decisions about their daily activities.

CHAPTER III

LITERATURE REVIEW

3.1 Determination of Air Quality Life Index (AQLI) in Medinipur City of West Bengal(India) During 2019 To 2020 : A contextual Study

3.1.1 Approach & Results:

- They do not use any ML model. They use the predefined method provided by various organizations like WHO, INAAQS etc.
- They show a provable reduction in life expectancy and the correlation to various air pollutants.
- They also speak about the direct effects of PM2.5 on the lungs.

3.1.2 Advantages:

- They use an air pollutant rather than AQI.
- Speaks about the direct effects of PM on various parts of the body.
- Uses an expensive but very accurate, and lab tested/ approved sensor for PM 2.5
- (Plantower particulate matter sensor PMS3003)

3.1.3 Limitations:

- No ML model used. They use existing methods to determine everything.
- Only looks at one pollutant.
- Not clear if all the effects (of the pollutant) are considered .

3.2 The nexus between COVID-19 deaths, air pollution and economic growth in New York state: Evidence from Deep Machine Learning.

3.2.1 Approach & Results:

- They use an ensemble of ML models (ANN, Deep Learning, Decision Tree)
- The association between COVID-19-related mortality, economic growth, PM10, PM2.5, and NO₂ concentrations in New York state is examined in this article.

3.2.2 Advantages:

- They consider individual pollutants.
- Very well thought out data gathering and cleaning.
- The use methods used are well suited for the type of predictions and the type of data being used.

3.2.3 Limitations:

- It has been done mainly to covid related deaths.
- This has many inherent issues especially considering how little we truly know about covid.
- They attribute over 50% of deaths due to covid as deaths facilitated by these pollutants.
- This cannot be proven due to the lack of research time on the effects of covid.
- Does not consider other underlying issues

3.3 An Application of IoT and Machine Learning to Air Pollution Monitoring in Smart Cities

3.3.1 Approach & Results:

- They use ANN models for predicting the concentrations of air contaminants and Pearson's coefficient for the correlation with weather conditions.
- The objective of this paper is to monitor pollutant concentrations in smart cities and find a correlation between pollutants and weather parameters and prediction of the pollutants to prevent diseases like lung cancer.

3.3.2 Advantages:

- The paper takes into account the weather conditions as well. As air pollutants mixed with other factors like water/high winds cause differing effects
- The model created has achieved a Root Mean Square Error of only 0.0128 for SO₂ prediction and 0.0001 for PM2.5
- They have used two different methods (Pearson Correlation and ANN) for the weather correlation and prediction.

3.3.3 Limitations:

- Does not consider all the environmental weather conditions (Only wind speeds, temperature and humidity)
- The model only contains a single hidden layer and they have not run it only for 500 epochs
- The IoT infrastructure they have used is very crowded and complex.

3.4 Mohammad Marzouk and Mohammad Atef "Assessment of indoor air quality in academic buildings using IOT and deep learnings": Mdpi(June 2022)

3.4.1 Approach & Results:

To find the correlation between outdoor pollution and indoor air quality, by monitoring real time IAQ using IOT sensors and sending the collected data to the cloud via wireless connections. CNN and LSTM are used to train the collected data.

3.4.2 Advantages:

- Uses real time data.
- Finds the correlation between outdoor and indoor environments.
- Concentrates on rural areas too, which might be affected by burning wood, construction particles and unpaved roads.

3.4.3 Limitations:

- Does not consider the family history with lung cancers.
- Does not consider the occupational exposures to carcinogens.
- It is biased as the readings were collected only during summer when the humidity will be relatively high.
- The study was restricted to just a few primary pollutants and did not consider volatile organic compounds, NH₃ and O₃

3.5 Jin Z-Y, Wu M, Han R-Q, Zhang X-F, Wang X-S, et al. (2014)

"Household Ventilation May Reduce Effects of Indoor Air Pollutants for Prevention of Lung Cancer: A Case-Control Study in a Chinese Population."

3.5.1 Approach & Results:

The purpose of this research is to investigate the link between home ventilation and lung cancer. A standardized questionnaire was used to obtain epidemiologic and home ventilation data. The adjusted odds ratios (ORadj) and 95% confidence intervals (CI) were calculated using unconstrained logistic regression.

3.5.2 Advantages:

- The information gathered takes into account a person's family history of lung cancer, basic demographic characteristics, socioeconomic status, tobacco smoking history, alcohol intake, dietary history, and physical activity.
- Active smokers, second hand smokers, carcinogens, tobacco smoking and high temperature cooking oil flames were considered for the study thereby covering a wide scope.

3.5.3 Limitations:

- Because the data is limited to the local community, selection and memory bias may present in the study.
- The data about family history, age, gender is considered via a standardized questionnaire by interviewers, using quantitative data might have been effective instead.
- Have not considered occupational exposure
- The study is limited to IAQ and does not consider how outdoor pollutants are affecting IAQ.

3.6 How Is the Lung Cancer Incidence Rate Associated with Environmental Risks? Machine-Learning-Based Modeling and Benchmarking

3.6.1 Approach & Results:

- They use an ensemble of models like Logistic Regression, Random Forest, SVM and Gradient Boosting and perform benchmarking on them.
- The goal of this research is to use machine-learning-based modeling and benchmarking to examine the association between lung cancer incidence rate and air pollution.
- The study's goal is to create a prediction model to better understand the influence of air pollution on illness.

3.6.2 Advantages:

- They perform benchmarking, i.e, comparing the performance of different ML models and provides insights on which the best one is .
- The dataset they have used contains both environmental risk factors (air pollutants) and the lung cancer incidence rates from various countries and hence increases generalizability.

3.6.3 Limitations:

- Does not consider all the environmental weather conditions (Only wind speeds, temperature and humidity)
- The model only contains a single hidden layer and they have not run it only for 500 epochs
- The IoT infrastructure they have used is very crowded and complex.

3.7 Kim Kyung Eun, Cho Daeho, Park Hyun Jeong, Air pollution and skin diseases: Adverse effects of airborne particulate matter on various skin diseases, Life Sciences (2016), doi: 10.1016/j.lfs.2016.03.039

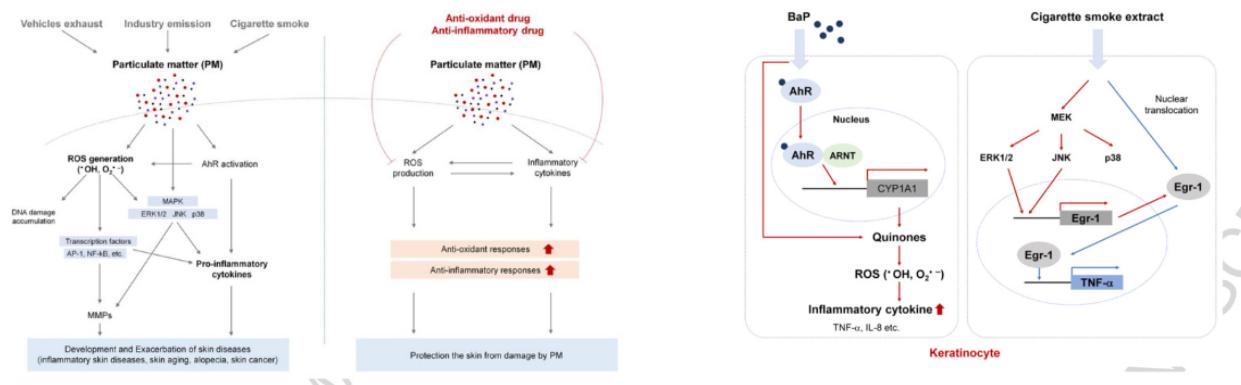


Fig 3.7.1- Retention of pollutants into the skin

3.7.1 Approach & Results:

- This page discusses the relationship between PM and skin illnesses, as well as the underlying immunological pathways.
- Increased PM levels have been linked to the development of a variety of skin illnesses through the control of oxidative stress and inflammatory cytokines.
- As a result, antioxidant and anti-inflammatory medicines may be beneficial in the treatment of PMS-induced skin disorders.

3.7.2 Advantages:

- This paper has listed in detail which air pollutants are known to cause which disease.
- It lists various diseases like Atopic dermatitis, Acne, Psoriasis, and skin cancer.

- It also discusses the conditions of skin aging, alopecia, and oxidative stress.

3.7.3 Limitations:

- The amount of statistics is less than is used to prove its point.
- It doesn't discuss the correlation between these diseases, and how they affect the probability of another happening.

3.8 Roberto Cazzolla Gatti, Arianna Di Paola, Alfonso Monaco, Alena Velichevskaya, Nicola Amoroso, Roberto Bellotti, The spatial association between environmental pollution and long-term cancer mortality in Italy, Science of The Total Environment, <https://doi.org/10.1016/j.scitotenv.2022.158439>

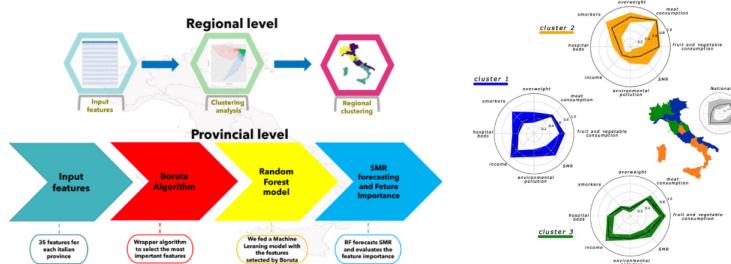


Fig 3.8.1:- Methodology and cluster analysis output

3.8.1 Approach & Results:

- Using an artificial intelligence method, this article examined the relationships between cancer mortality, socioeconomic characteristics, and sources of environmental pollution in Italy at both broad regional and narrower provincial scales.

- Random Forest (RF) regression in combination with a Boruta feature importance analysis, K means clustering.
- SMR forecasting and Feature importance, Regional cluster analysis.

3.8.2 Advantages:

- It has taken into consideration all the different types of body parts that can get affected due to air pollution in extensive detail.
- The data sources and the algorithms used have been discussed.
- Investigated the possible geographical relationship between socioeconomic and lifestyle characteristics.

3.8.3 Limitations:

- Some individual sources of pollution have no link to any form of cancer. This study is limited to Italy, but it should be expanded to other nations as well, because air pollution levels vary by region.

3.9 Conclusion to the literature survey:

Most studies only include a small number of risk variables when examining the influence of air quality on human health. Some initiatives, for example, consider just indoor air quality (IAQ) or outdoor air quality (OAQ), but not both, or they exclude environmental variables, family history, and occupational exposure. This method can lead to biased and untrustworthy results that do not completely reflect the complexities of the topic at hand.

Furthermore, despite the fact that air pollution is a major public health problem, little study has been conducted on its real health impacts. While numerous research have looked at the association

between air quality and health outcomes, few have looked at the causative relationship. Due to a dearth of study, properly predicting the influence of air quality on health can be difficult.

Another problem is that projections based on static and out-of-date data are uncommon. This can be an issue since air quality can fluctuate fast owing to a number of variables such as weather, industrial activity, and transportation. Predictions based on old data might result in incorrect conclusions and perhaps dangerous choices.

To address these difficulties, more extensive and sophisticated models that reflect a larger variety of risk variables and the real health impacts of air pollution are required. These models must also be based on real-time data and capable of fast adapting to changes in air quality levels. This allows us to create more precise forecasts regarding the impact of air pollution on human health, which may guide policy decisions and assist individuals in making educated decisions about their health and well-being.

CHAPTER IV

DATA

4.1 Overview

The aim is to use a static dataset containing correlation between lung cancer and air pollutants, for training the model. Two datasets have been used for solving this approach.

The lung cancer patient dataset has been acquired from [data.world](#) which is a data catalog platform built to provide open access to generative AI ready datasets.

Link to dataset: <https://data.world/cancerdatahp/lung-cancer-data>

The Air Quality Data in India (2015 - 2020) dataset which has hourly data across stations and cities in India has been taken from kaggle.com, where it had been collected and compiled from publicly available data given by the Central Pollution Control Board, an official portal by the Govt. of India.

Link to dataset: <https://www.kaggle.com/datasets/rohanrao/air-quality-data-in-india>

4.2 AQI dataset

	StationId	Date	PM2.5	PM10	NO	NO2	NOx	NH3	CO	SO2	O3	Benzene	Toluene	Xylene	AQI	AQI_Bucket
0	AP001	2017-11-24	71.36	115.75	1.75	20.65	12.40	12.19	0.10	10.76	109.26	0.17	5.92	0.10	NaN	NaN
1	AP001	2017-11-25	81.40	124.50	1.44	20.50	12.08	10.72	0.12	15.24	127.09	0.20	6.50	0.06	184.0	Moderate
2	AP001	2017-11-26	78.32	129.06	1.26	26.00	14.85	10.28	0.14	26.96	117.44	0.22	7.95	0.08	197.0	Moderate
3	AP001	2017-11-27	88.76	135.32	6.60	30.85	21.77	12.91	0.11	33.59	111.81	0.29	7.63	0.12	198.0	Moderate
4	AP001	2017-11-28	64.18	104.09	2.56	28.07	17.01	11.42	0.09	19.00	138.18	0.17	5.02	0.07	188.0	Moderate

Fig 4.2.1: AQI dataset description

The dataset contains 108035 rows and 16 columns.

Table 4.2.1- The below table mentions the attributes in the dataset and its description.-

Attributes	Attribute description
StationId	Id of station
Date	Date the record was recorded on
PM 2.5	Particulate Matter 2.5-micrometer in ug / m3
PM 10	Particulate Matter 10-micrometer in ug / m3
NO	Nitric Oxide in ug / m3
NO2	Nitric Dioxide in ug / m3
NOx	Any Nitric x-oxide in ppb
NH3	Ammonia in ug / m3
CO	Carbon Monoxide in ug / m3
SO2	Sulphur Dioxide in ug / m3

O3	Ozone in ug / m3
Benzene	Benzene in ug / m3
Toluene	Toluene in ug / m3
Xylene	Xylene in ug / m3
AQI	Air Quality Index
AQI_Bucket	Air Quality Index bucket

4.3 Lung cancer dataset

Patient Id	Age	Gender	AQI	Dust Allergy	Occupational Hazards	Genetic Risk	Chronic Lung Disease	Smoking	Passive Smoker	Clubbing of Finger Nails	Frequent Cold	Level
0	P1	33	1	2	5	4	3	2	3	2	1	2 Low
1	P10	17	1	3	5	3	4	2	2	4	2	1 Medium
2	P100	35	1	4	6	5	5	4	2	3	4	6 High
3	P1000	37	1	7	7	7	6	7	7	7	5	6 High
4	P101	46	1	6	7	7	7	6	8	7	2	4 High

Fig 4.3.1: Lung cancer dataset description

The dataset contains 1001 rows and 25 columns.

We have only used 12 columns for our project as they are the ones that are valid for our research.

Columns such as Age, Gender, AQI, Dust Allergy, Occupational hazards, etc. have been used.

Columns like Alcohol use, Obesity, etc. have been dropped.

CHAPTER V

PROJECT REQUIREMENTS SPECIFICATION

5.1 Introduction

In response to the growing concern over the dangers of air pollution on human health, the proposed model is a hybrid model of Adaptive LSTM and ARIMA models to monitor air quality continuously. These models have proven to be highly reliable and adaptable to changes in time series data. To ensure scalability, the model will be deployed on a secure cloud platform, making it easily accessible to users. Real-time data will be collected using Arduino boards and high-quality sensors, allowing the model to learn continuously and improve its predictions over time. By increasing accuracy in predictions and spreading awareness, the aim is to encourage environmentally conscious decision-making and potentially save lives.

5.2 System Requirements

5.2.1 Features

The product features entail:

- The user will be able to check the current levels of pollution in his surroundings and the probability of contracting lung cancer based on the current exposure to pollutants.
- The user can check the potential probability of contracting lung cancer. This will be derived from results from the machine learning model.

5.2.2 User classes and characteristics

- **Smart home companies:**

Smart home companies currently incorporate various sensors like temperature sensors, heat sensors and motion sensors. They can also incorporate the pollutant sensors as an additional layer of detection on the current infrastructure.

- **Government:**

Governments can use our product to identify pollution hotspots and create policies and regulations to reduce the harmful emissions and issue public health warnings in high risk areas.

- **City planning:**

City planners and private companies can use this information to guide urban planning and development and create communities in a healthier living space.

- **General Public:**

The general should be informed about the levels of pollution and the risks associated with it to take preventive measures

5.2.3 Operating Environment

- **Arduino:**

The default operating system present on the board will be used. A custom code will be added to it. Any OS that supports the Arduino IDE (or the web version) can be used to interface with the Arduino (this is only to add code to the board).

- **Coding Platform:**

Any OS and any editor can be used that you prefer as long as it supports git or some other way to have a shared codebase. It should also have inbuilt support for languages that will be used for coding and a way to integrate databases.

5.2.4 Software requirements

- **Data collection and preprocessing tools:**

Tools to collect air pollutant data from various sources, such as air quality sensors and software to clean, normalize, and prepare the data for analysis will be used.

- **Machine Learning frameworks:**

Machine learning frameworks such as TensorFlow, Keras, will be used to build and train the models. These frameworks provide a range of tools for data processing, model building, and evaluation.

- **Statistical analysis tools:**

Statistical analysis tools such as Python packages like NumPy ,Pandas and Statsmodels will be used for data analysis and exploratory data analysis (EDA).

- **Visualization tools:**

The project will need visualization tools like Matplotlib, Seaborn to create visualizations and graphs that help you understand the data and communicate results.

- **Cloud computing platforms:** To store and analyse massive amounts of data and deploy our machine learning models, the project will also utilize a cloud computing platform such as Streamlit and Thingspeak.

5.2.5 Hardware requirements

- ESP8266 wifi module to collect data from sensors and store it in the ThingSpeak cloud platform.
- MQ7 to monitor the concentration of CO in the atmosphere.
- GP2Y1010AUF Dust sensor to monitor particulate matter of different aerodynamic diameter (PM2.5 and PM10)

5.2.6 User Interface

- Basic information can be displayed on an LCD connected to the stations. More data can be viewed on a computer with basic internet connection (either through a custom made website/application or the actual cloud platform directly).
- In the provided interface there will be basic instructions on how to interpret the data received.
- The stations can measure pollutants every 20 seconds. It can send each measurement to the database, preferably immediately. The model can rerun with the new received data every time the user requests.
- All error messages will be sent to the cloud platform. Some errors can be displayed on the LCD.

5.2.7 Communication interface

- **Arduino:**

The default operating system present on the board will be used. The project will use custom code written. Any OS that supports the arduino IDE (or the web version) can be used to interface with the arduino (this is only to add code to the board) .

- **Cloud:**

The project will use a few cloud platforms for this. This will evolve as time progresses and pros and cons are found out for each platform. Thinkspeak to save the data from the sensor stations, AWS Glue, Google Cloud Dataflow or Streamlit to process the data and to host the model.

5.3 Functional Requirements

- **Machine Learning Algorithms:** Implement ML algorithms to analyze the air quality data and find the correlations between pollutants and lung cancer rates.
- **Cloud Infrastructure:** Use a cloud infrastructure such as Streamlit, AWS, Google Cloud Platform, or Microsoft Azure to host the ML models and the database.
- **Deployment:** Deploy the ML models to the cloud infrastructure and provide a user friendly interface to allow users to access the models and the data.
- **Real-time Monitoring:** Continuously monitor the air quality using IoT devices and update the database in real-time. The ML models should also be updated periodically to ensure that they remain accurate. .
- **Disease Prediction:** Based on the quantity of contaminants in the air, the system should be able to assess the collected data and compute the likelihood of lung cancer and other associated diseases.

5.4 Non-Functional Requirements

- **Performance:** The system should be able to manage enormous volumes of data and process it in real time.

- **Maintainability:** The system should be simple to maintain and upgrade, with as little downtime as possible.
- **Security:** Because IoT devices are frequently attacked, the system must be secure and preserve data privacy.
- **Scalability:** The system should be scalable and capable of supporting an increasing number of devices and data points.

5.5 Data Requirements

- **Data volume:** Sufficient data volume is required to train the machine learning model effectively. Generally, more data leads to better accuracy and more robust models.
- **Data quality:** The data used in machine learning projects must be accurate and representative of the real-world scenarios that the model will encounter. Inaccurate or inconsistent data can lead to incorrect predictions and unreliable models.
- **Data diversity:** It is essential for ensuring that the model is not biased toward a certain sample of data. To generalize properly, the model needs to be exposed to a variety of data samples.
- **Data labeling:** Labeled data is required to train supervised learning models effectively. The labeling process involves adding tags or labels to the data that indicate the input/output relationship.

- **Data preprocessing:** Preprocessing involves cleaning and transforming the raw data to prepare it for machine learning algorithms. Data preprocessing can include tasks such as normalization, feature engineering, and data augmentation.
- **Data storage and management:** to handle massive amounts of data and make it available to machine learning algorithms.

CHAPTER VI

SYSTEM DESIGN

6.1 Current System

Current systems have many shortcomings that are listed as follows:

- They monitor air quality but do not predict the health effects of the monitored air on the human body.

This is extremely harmful as the uninformed user will not be able to generate any useful inferences from it making the whole system redundant.

- They have modeled LSTM or Random Forest algorithms. This does not provide proper accuracy or efficiency.

These models are not scalable for real time data or upcoming environmental factors.

- They focus on only one factor that affects lung cancer.

They do not consider that there are multiple air pollutants that lead to lung cancer or lung cancer related diseases. Our research has shown that PM 2.5, PM 10 and CO are all pollutants of major interest as far as lung cancer is concerned.

- Furthermore, they have not used any cloud based technologies to test the project's feasibility.

The system will use these platforms to make the system more scalable and reliable.

6.2 Design Considerations

6.2.1 Architecture:

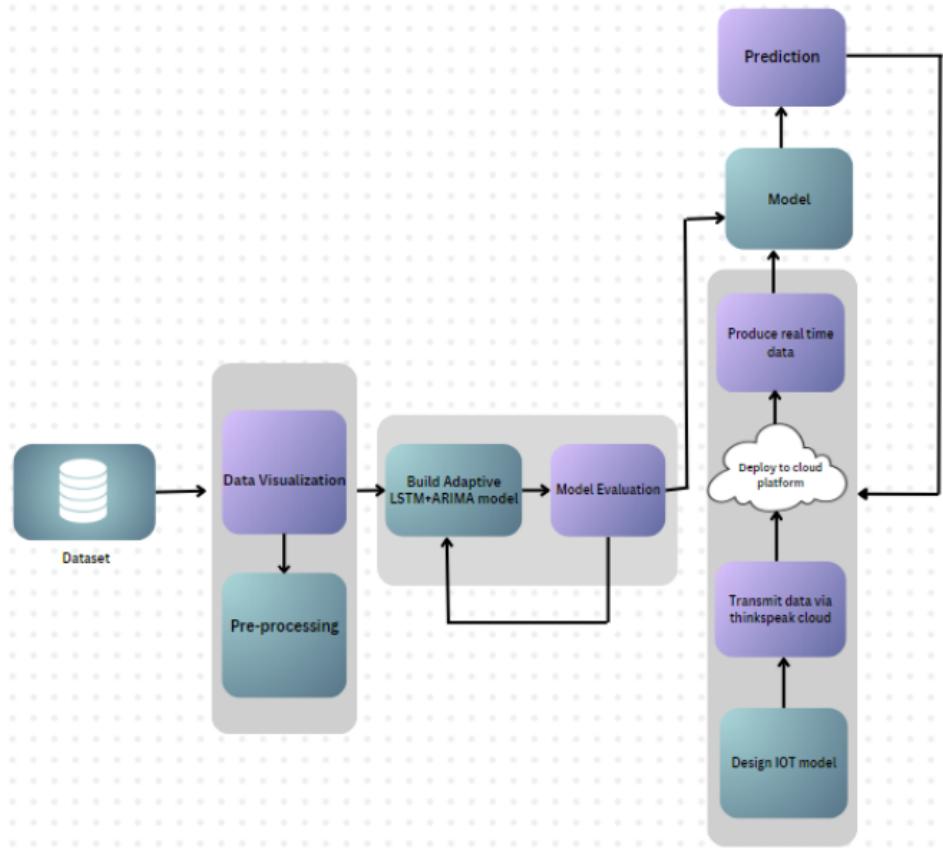


Fig 6.2.1: Architecture diagram for proposed system - Monitoring the concentration of air pollutants and its health hazards using Machine Learning models

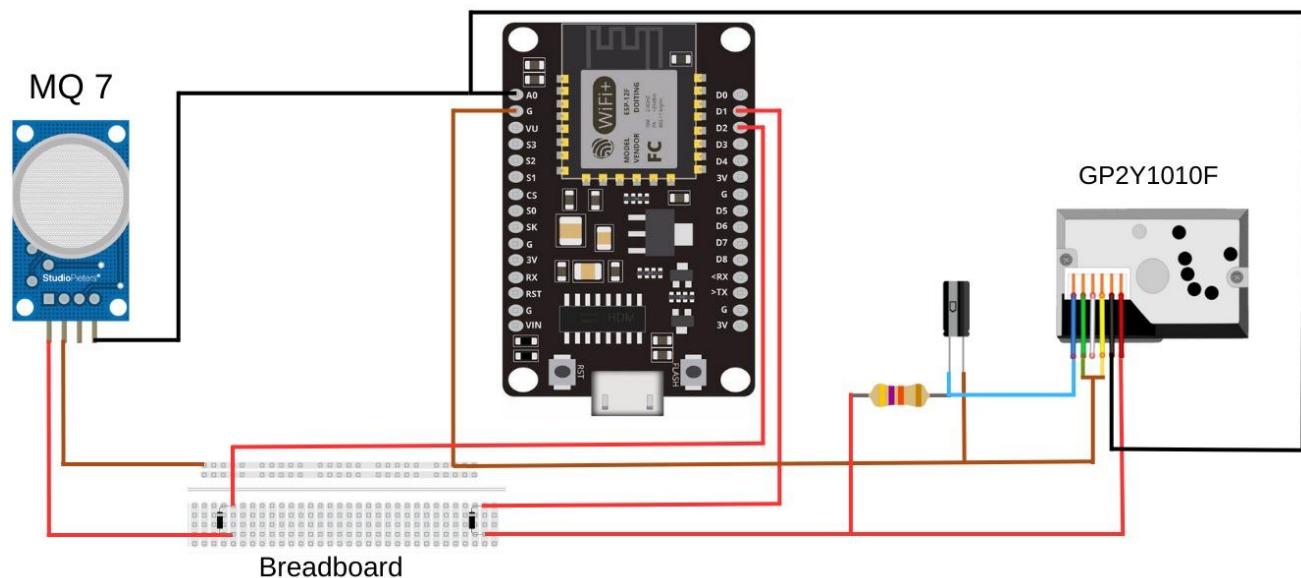


Fig 6.2.2: Circuit Architecture diagram for proposed system - Monitoring the concentration of air pollutants and its health hazards using Machine Learning models

The IoT station consists of the GP2Y1010AUF sensor for measuring the concentration of PM2.5 dust particles as well as the MQ-7 gas sensor for measuring the concentration of CO gas. The IoT station will consist of the ESP8266 WiFi module which will collect the data from the sensors and upload it to ThingSpeak cloud platform.

The system has 2 components wired up to the board.

- **MQ-7:** It receives power from the power line from ESP to the breadboard.
- **GP2Y1010AUF:** It receives power from the power line from ESP to the breadboard. Ground is set up in the same way.

6.2.2 Detailed flow

- **Pre-process the data:** Clean and preprocess the time series data, including handling missing values, transforming variables if necessary, and splitting the data into training and testing sets.
- **Train ARIMA model:** Fit an ARIMA model to the training data of the AQI dataset to capture the seasonal and trend components of the time series. Tune the ARIMA hyperparameters (e.g., order of differencing, autoregressive and moving average orders) using techniques like grid search or time series cross-validation.
- **Generate ARIMA forecasts:** Use the trained ARIMA model to generate short-term forecasts on the testing data.
- **Train Adaptive LSTM model:** Use the lung cancer dataset as input to train an Adaptive LSTM model on the training data, predicting the possibility of a person having cancer. This is done by converting the already existing values of “high”, “moderate”, “low” to 1, 2 and 3 using one-hot encoding.
- **Generate predictions:** Generate predictions using the trained Adaptive LSTM model to predict the possibility of a person having lung cancer and use the forecasts from the ARIMA model of future AQI. This will make sure the predictions of the LSTM model are not only accurate but also more localised to the area that is being considered in the dataset.
- **Evaluate and optimize:** Evaluate the hybrid ARIMA-Adaptive LSTM model's performance on the testing data using appropriate evaluation metrics, and fine-tune the model as needed by altering hyperparameters or model architecture.
- **Use sensors to get real time data:** Real time pollutant concentration data will be collected from an IoT station. The IoT station consists of the MQ-7 gas sensor for measuring the concentration of CO gas and the GP2Y1010AUF dust sensor to measure the concentration of PM2.5 (dust). The IoT station will be using the ESP8266 WiFi module to read values from the sensors and upload it to the ThingSpeak cloud platform.

- **Deploy the model and sensor station to cloud platforms:** The model will be uploaded to the cloud and use live reading to make predictions.

Table 6.2.3- Pros and Cons of Using the Hybrid Network:

PROS	CONS
Both ARIMA and adaptive LSTM work best with time series data.	Adaptive LSTM limits the accuracy in cases where there is limited data.
ARIMA is easy to interpret and reliable for small and stationary datasets.	Adaptive LSTM models can be highly complex, making them difficult to interpret.
While adaptive LSTM works best for large, real time, varying length, non-stationary data and learns complex patterns.	Selecting the appropriate ARIMA model parameters (such as the order of differencing and the number of autoregressive and moving average terms) can be challenging and require a good understanding of the underlying data.
Adaptive LSTM is capable of dynamically adjusting to changes.	
LSTMs can capture long-term dependencies and temporal patterns in the data, which can be useful for predicting lung cancer risk based on changing air quality conditions.	
ARIMA captures the short-term forecasts, and adaptive LSTM captures the long-term forecasts.	

6.3 DESIGN DESCRIPTION:

6.3.1 Use case diagram:

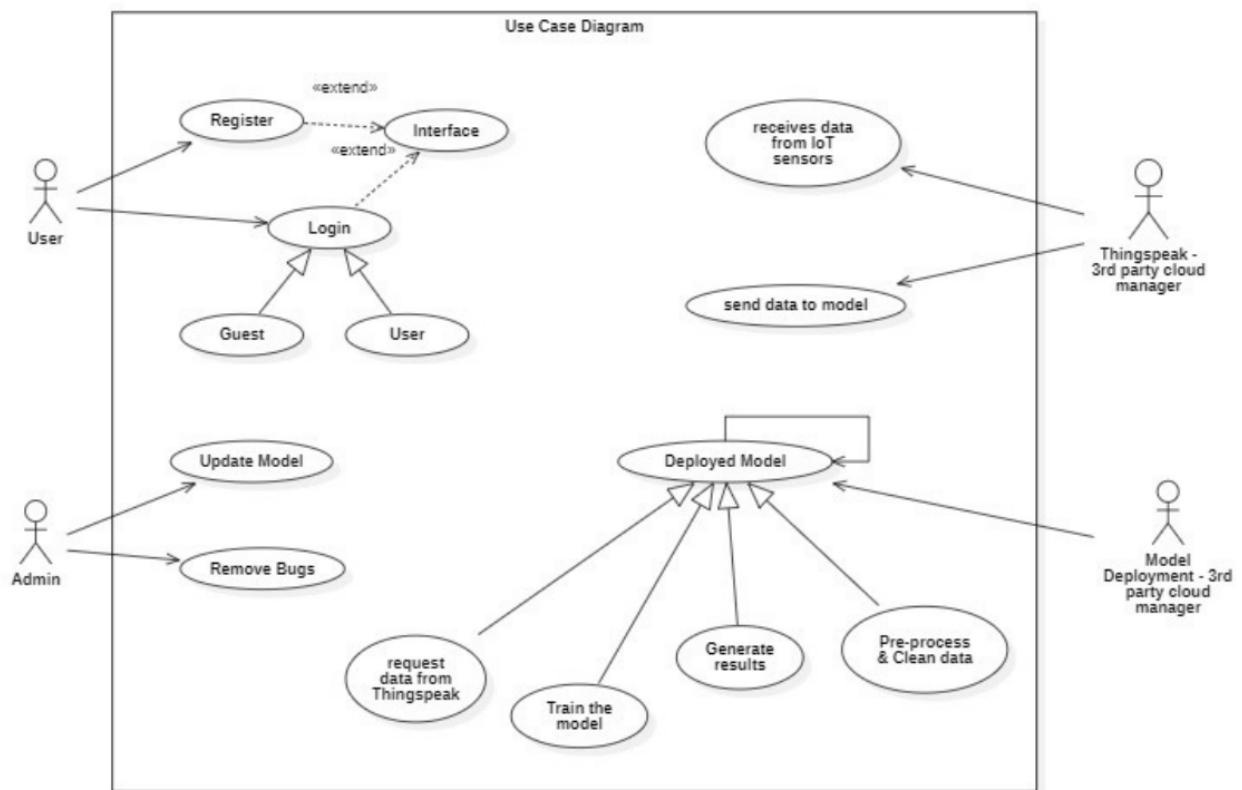


Fig 6.3.1: Use Case diagram for proposed system - Monitoring the concentration of air pollutants and its health hazards using Machine Learning models

Use cases that have been identified throughout the research:

- **User:** User will be an individual or entity that interacts with a system to see the predictions done on the air quality around them. Users will register and/or login in the system and will be the source of data from which the model will be further trained.
- **Admin:** Admin will be responsible for managing and maintaining a system or application. Admin will configure settings, manage user accounts, monitor system performance, and resolve technical issues. They will have the authority to access user details and patient details obtained from hospitals as well.
- **Cloud platforms:** Cloud platform refers to a set of cloud-based tools and services that will be used to deploy, manage, and scale applications and services. It will perform tasks such as deploying applications, scaling resources, managing data, and monitoring performance.

The system will make use of two different cloud platforms: Thingspeak to receive data from the sensor stations and to send data to the cloud platform that hosts the model. Streamlit, Spaces or AWS to host the machine learning model.

6.3.2 Class Diagram:

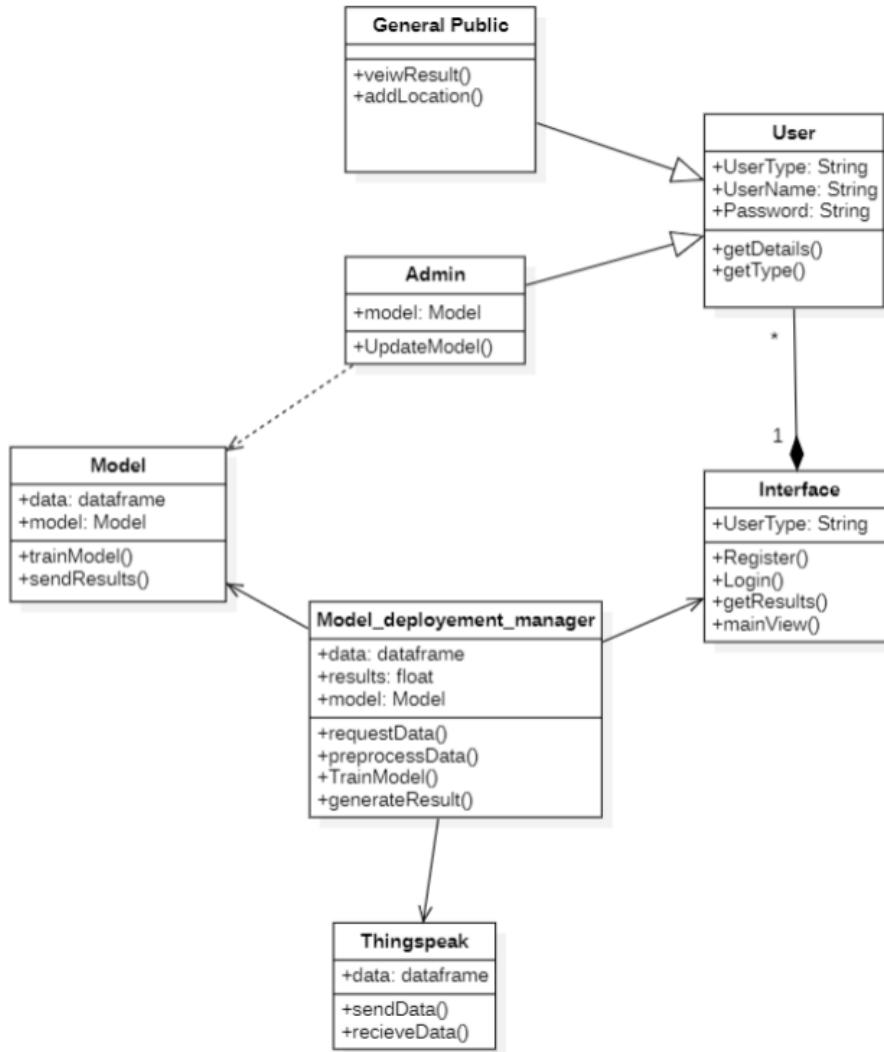


Fig 6.3.2: Class diagram for proposed system - Monitoring the concentration of air pollutants and its health hazards using Machine Learning models

6.3.3 ER DIAGRAM:

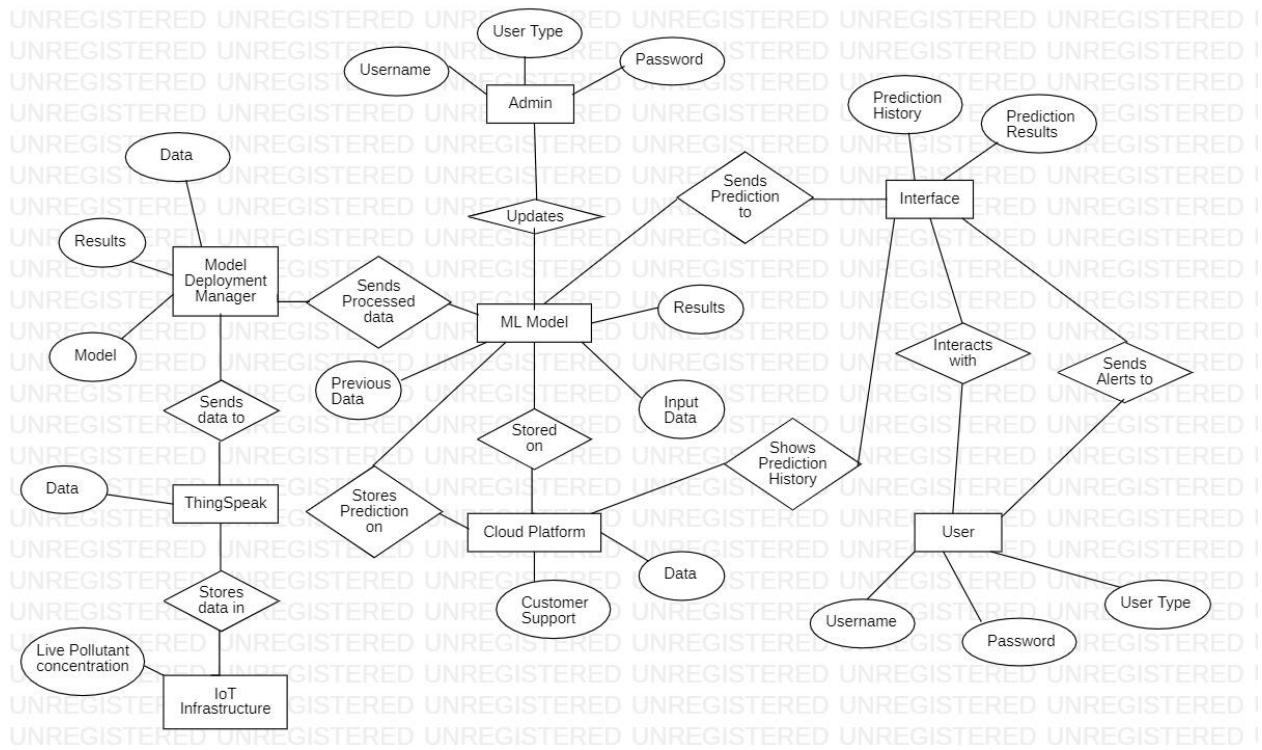


Fig 6.3.3: ER diagram for proposed system - Monitoring the concentration of air pollutants and its health hazards using Machine Learning models

6.3.4 Deployment Diagram:

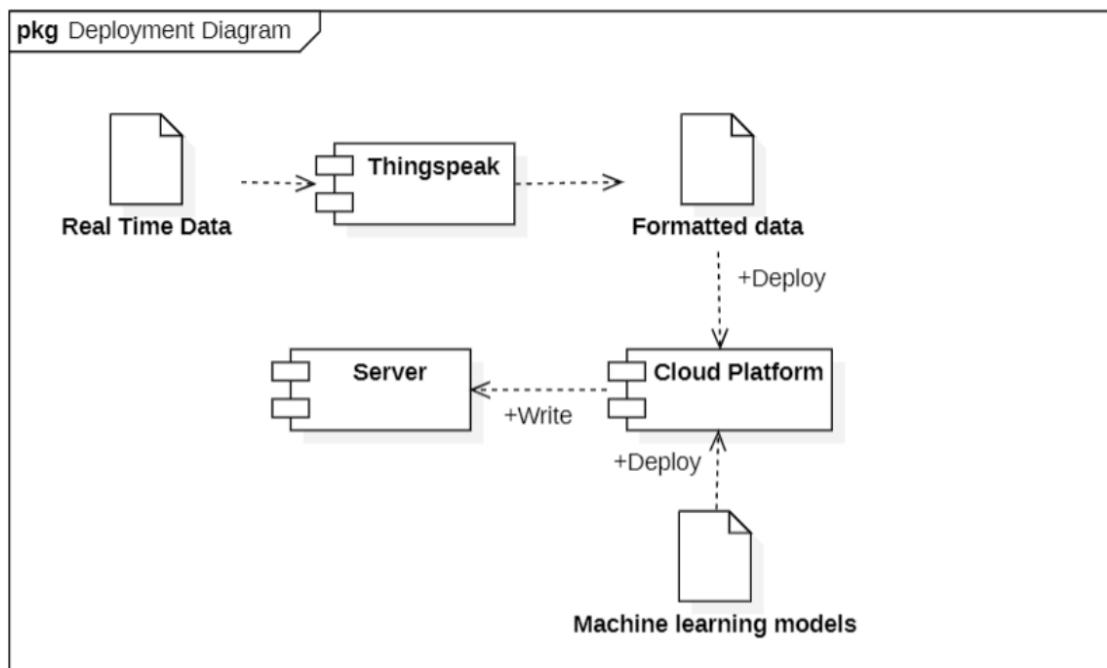


Fig 6.3.4: Deployment diagram for proposed system - Monitoring the concentration of air pollutants and its health hazards using Machine Learning models

6.4 Design Details

6.4.1 Novelty

- The projects on this subject to date have barely researched the consequences of the observed air quality on health, and the predictions are based on outdated and static data.
- The system aims to use real time data to make predictions instead of static and outdated data.
- Most of the projects use just the LSTM model, The system will use an hybrid ARIMA + adaptive LSTM that is dynamically capable of adapting to changes.

- Optimization of already existing models for relevant data to produce practical and useful results.

6.4.2 Innovation

- Combining three domains- machine learning, cloud computing, and IoT
- Using a hybrid network of forecasting models- ARIMA+Adaptive LSTM
- Increased efficiency to produce an advantageous output.
- By using a cloud platform for all the major functions, the entire system should be scalable, both horizontally and vertically.
- The system can be integrated into many devices due to the use of a cloud platform.
- The design will be durable, and long-lasting. It will also be relevant because it is malleable enough to accommodate multiple types of parameters.

6.4.3 Reliability

- The project involves complex technical components, such as machine learning algorithms, cloud computing, and IoT devices.
- Any technical issues with these components could affect the accuracy and reliability of the study result.
- The system will be using tested and industry trusted cloud platforms for the deployment of the system.
- Ensuring consistent performance and functionality across different contexts and use cases.
- Incorporating redundancies and fail-safe mechanisms to prevent system failures.
- System modules will be designed that can help identify and replace any failed modules, thus improving overall reliability.
- Extensive testing will be conducted during the design and development phases to help identify potential problems and improve reliability.

- E-monitoring and feedback systems will be incorporated that can help detect and diagnose any issues that arise during use, allowing for timely repair or replacement of any faulty components.

6.4.4 Privacy and Security

- The dataset obtained does not contain the patient's personal details, thereby supporting privacy.
- The system should be secure and protect data privacy, as IoT devices are often vulnerable to attacks.
- A cloud platform will be used to store all the data as well as the model. Cloud platforms offer several security features to ensure that data stored on them is secure.
- Admin access will be given only to trusted individuals, thereby protecting the personal details of users and patients.
- The system will only collect and store the minimum amount of data necessary to perform the service. This reduces the amount of data that can be compromised in the event of a security breach.
- The security of the system will be checked regularly and updates to the security measures as needed. This can help prevent vulnerabilities from being exploited.
- The system will use strong encryption methods to protect user data.

6.4.5 Performance

- The system should be able to manage enormous volumes of data and process it in real time.
- The ML model should give an accuracy of 90%+ with a minimum error.
- The use of sensors that are known to give high amounts of accuracy in reasonable time periods will improve the performance of the whole system.

- A hybrid model will make sure to get the best of both worlds, providing us with the best features of both.
- The system will efficiently use space, which will lead to a more compact design, which can improve portability, reduce weight, and improve performance.
- The system will use the most relevant and informative features to avoid overfitting or underfitting the model.
- Tuning hyperparameters has a significant influence on model performance. hyperparameters will be tuned the best to fit the model.

6.4.6. Interoperability

- To allow data exchange and analysis, the system should be able to interact with other systems or platforms, such as electronic health records or public health databases.
- The data is collected and monitored on a cloud platform that can be accessed from any device.
- Adopting open standards for data formats, communication protocols, and APIs.
- Clearly defining the interfaces between systems, including the format of data exchanged and the methods of communication.
- Clear and comprehensive documentation.
- Rigorously testing the system for compatibility with other systems can help identify and address interoperability issues before they become a problem.
- Following industry standards for interoperability, such as those defined by standards bodies or industry consortia, can help ensure that the system can work effectively with others in the industry or ecosystem.

6.4.7 Maintainability

- The database and computer maintenance will be handled by the cloud platforms. The workstations must be maintained by the owners.
- The system should be easy to maintain and update, with minimal downtime.
- The system will implement a modular architecture that separates different components of the model, such as data processing, feature extraction, model training, and prediction. This allows for easier modification of each component without affecting the entire model.
- Version control tools will be used, such as Git, to track changes to the ML model and its codebase. This allows for easy collaboration among team members and helps ensure that changes can be reverted if necessary.
- The system will be monitored regularly to detect any performance degradation or changes in the data distribution. This helps ensure that the model remains accurate and up-to-date.

6.5 Constraints, dependencies, and assumptions:

- **Interoperability requirements-**

Basic information can be displayed on an LCD connected to the stations. More data can be viewed on a computer with a basic internet connection. All error messages will be sent to the cloud platform. Some errors can be displayed on the LCD.

- **Interface/protocol requirements-**

The accuracy of the data collected through IoT devices depends on the quality and reliability of the sensors used. There may also be biases in the data collected, which could affect the validity of the study results. Therefore, maintaining the performance of the IOT sensors is necessary.

- **Discuss the performance related issues as relevant-**

The adaptive LSTM model might not have good accuracy if the amount of data fed is limited. The hybrid model might not give good accuracy if the data is irregular and contains anomalies that should be removed.

- **End-user environment-**

Should provide real-time monitoring, continuously monitor the air quality using IoT devices, and update the database in real-time. The ML models should also be updated periodically to ensure that they remain accurate.

- **Availability of Resources-**

Finding a skin cancer dataset and a global lung cancer dataset.

- **Hardware or software environment-**

The amount of energy used by IoT technology is significant, and it needs to be running constantly.

- **Issues related to deployment in target environment, maintainability, scalability, and availability-**

The data produced by IoT devices is often unstructured and provides a limited perspective due to its massive size.

CHAPTER VII

IMPLEMENTATION AND PSEUDOCODE

7.1 DATA PREPARATION:

```
[ ] df = pd.read_csv("/content/station_day.csv")
[ ] df.head(5)
```

	StationId	Date	PM2.5	PM10	NO	NO2	NOx	NH3	CO	S02	O3	Benzene	Toluene	Xylene	AQI	AQI_Bucket
0	AP001	2017-11-24	71.36	115.75	1.75	20.65	12.40	12.19	0.10	10.76	109.26	0.17	5.92	0.10	NaN	NaN
1	AP001	2017-11-25	81.40	124.50	1.44	20.50	12.08	10.72	0.12	15.24	127.09	0.20	6.50	0.06	184.0	Moderate
2	AP001	2017-11-26	78.32	129.06	1.26	26.00	14.85	10.28	0.14	26.96	117.44	0.22	7.95	0.08	197.0	Moderate
3	AP001	2017-11-27	88.76	135.32	6.60	30.85	21.77	12.91	0.11	33.59	111.81	0.29	7.63	0.12	198.0	Moderate
4	AP001	2017-11-28	64.18	104.09	2.56	28.07	17.01	11.42	0.09	19.00	138.18	0.17	5.02	0.07	188.0	Moderate

Fig 7.1.1: Data preparation of the AQI dataset obtained from CPCB

Checking for null values:

```
df1= df.dropna()
print(df1.head())

      Date  PM2.5    PM10     NO    NO2    NOx   NH3     CO    SO2     O3  \
1 2017-11-25  81.40  124.50  1.44  20.50  12.08  10.72  0.12  15.24  127.09
2 2017-11-26  78.32  129.06  1.26  26.00  14.85  10.28  0.14  26.96  117.44
3 2017-11-27  88.76  135.32  6.60  30.85  21.77  12.91  0.11  33.59  111.81
4 2017-11-28  64.18  104.09  2.56  28.07  17.01  11.42  0.09  19.00  138.18
5 2017-11-29  72.47  114.84  5.23  23.20  16.59  12.25  0.16  10.55  109.74

  Benzene  Toluene  Xylene     AQI
1    0.20     6.50   0.06  184.0
2    0.22     7.95   0.08  197.0
3    0.29     7.63   0.12  198.0
4    0.17     5.02   0.07  188.0
5    0.21     4.71   0.08  173.0
```

Fig 7.1.2: Dropping null values from AQI dataset obtained from CPCB

	lstm_df.head()													
	Patient Id	Age	Gender	AQI	Dust Allergy	Occupational Hazards	Genetic Risk	Chronic Lung Disease	Lung Smoking	Passive Smoker	Clubbing of Finger Nails	Frequent Cold	Level	
0	P1	33	1	2	5	4	3	2	3	2	1	2	Low	
1	P10	17	1	3	5	3	4	2	2	4	2	1	Medium	
2	P100	35	1	4	6	5	5	4	2	3	4	6	High	
3	P1000	37	1	7	7	7	6	7	7	7	5	6	High	
4	P101	46	1	6	7	7	7	6	8	7	2	4	High	

Fig 7.1.3: Data preparation of the lung cancer dataset obtained from data.world

7.2 DATA VISUALIZATION:

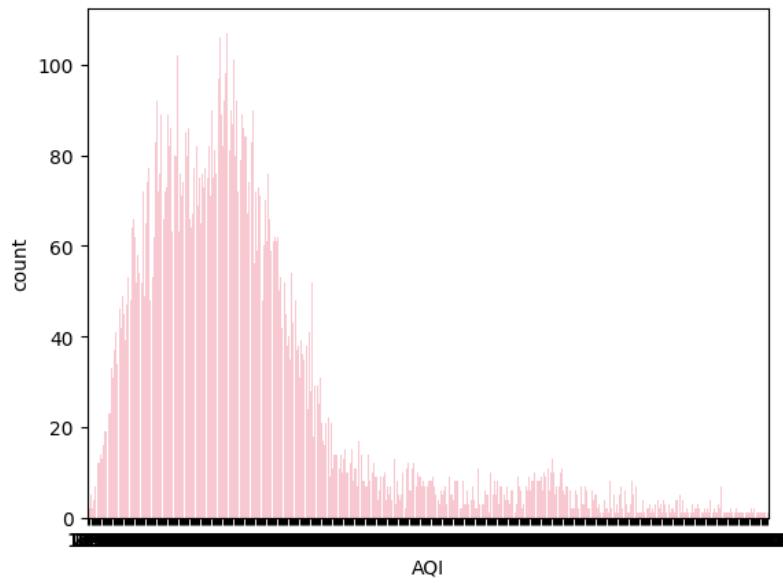


Fig 7.2.1: Count plot for AQI column in AQI dataset obtained from CPCB

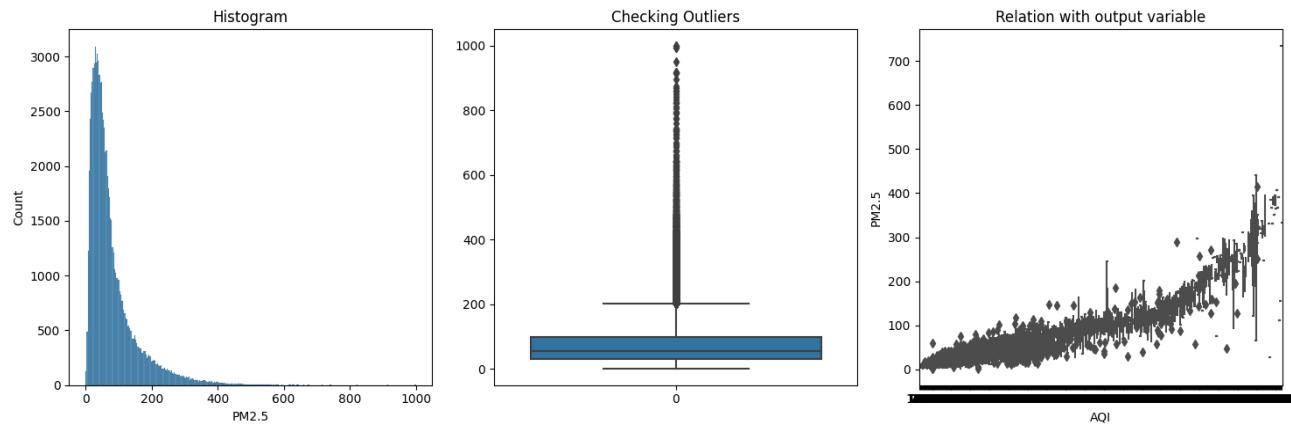


Fig 7.2.2: AQI vs. PM 2.5 and checking outliers in AQI dataset obtained from CPCB

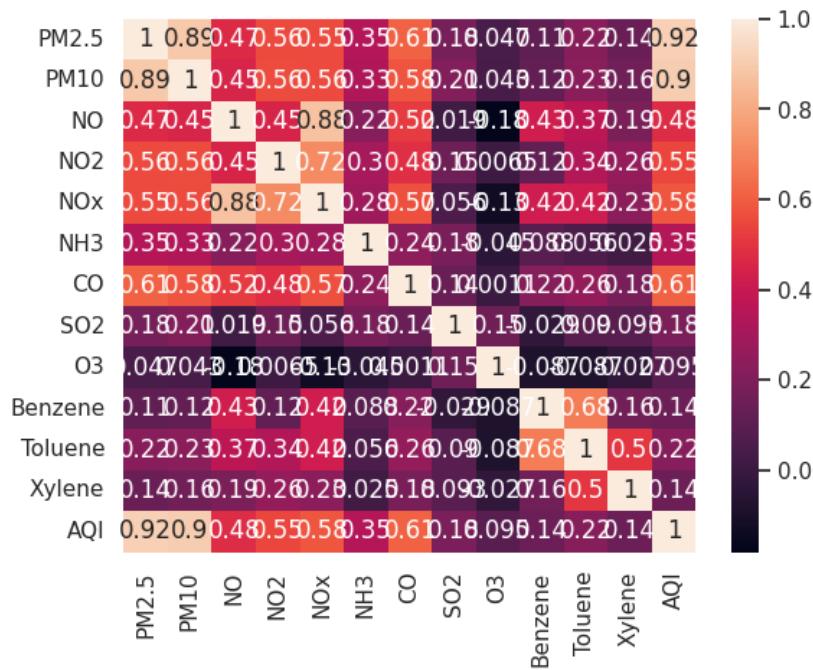


Fig. 7.2.3: Correlation matrix in AQI dataset obtained from CPCB

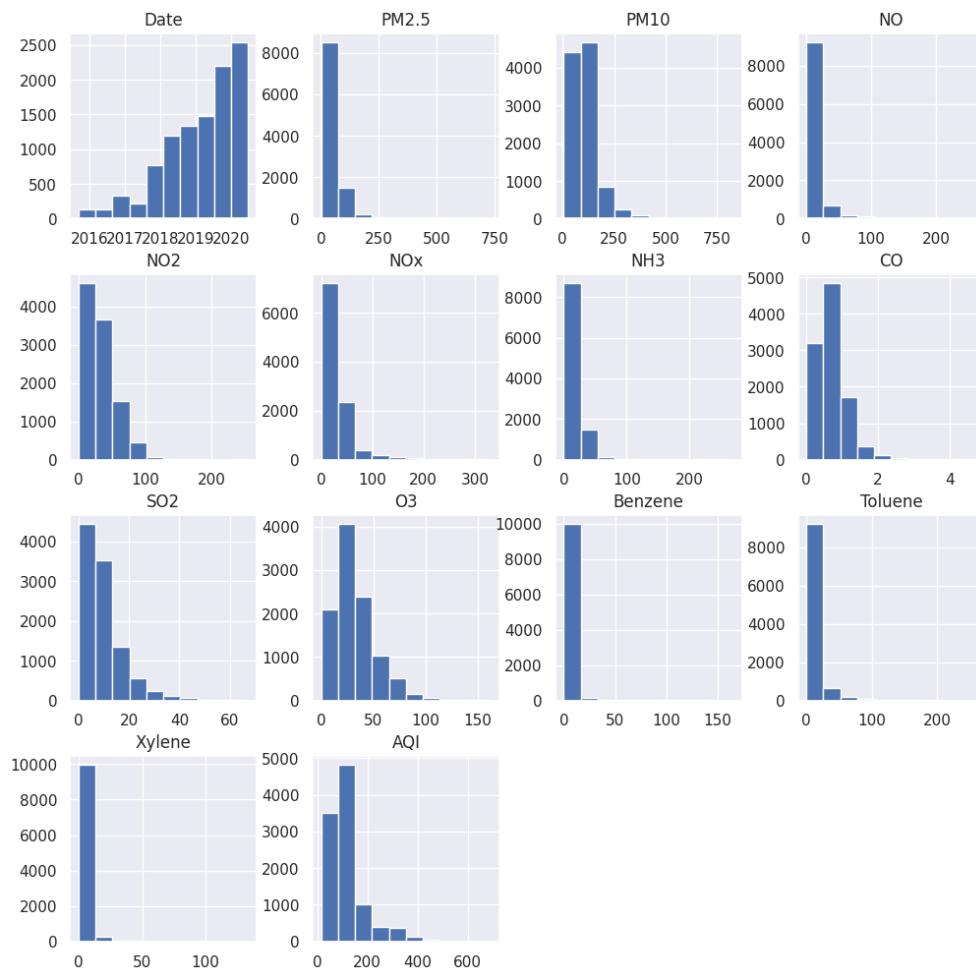


Fig 7.2.4: Visualization of various attributes to check if scaling is necessary in AQI dataset obtained from CPCB

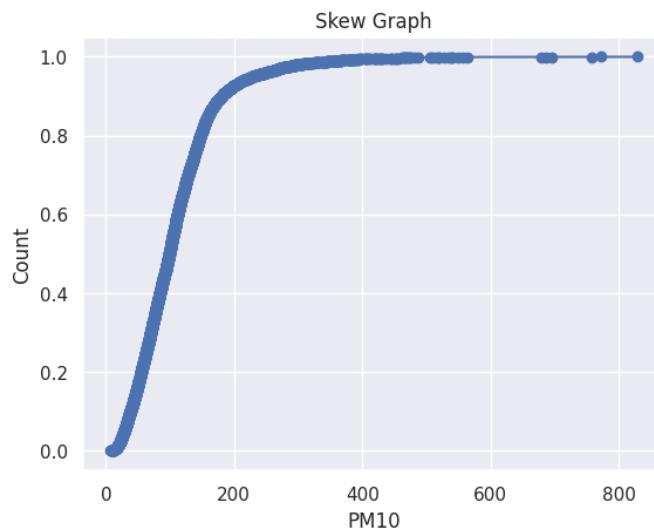


Fig 7.2.5: Skew plot for PM2.5 in AQI dataset obtained from CPCB



Fig 7.2.6: AQI vs. PM 10 and checking outliers in AQI dataset obtained from CPCB

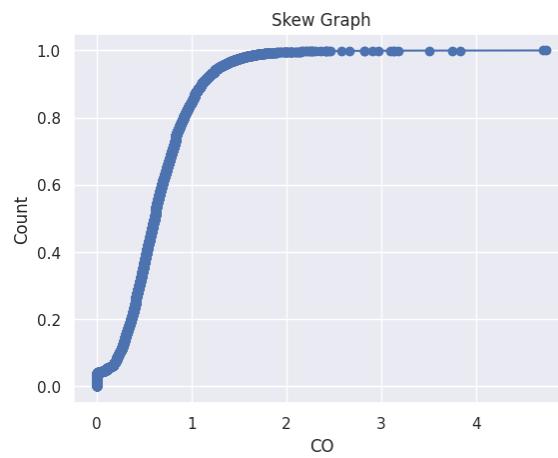


Fig 7.2.7: Skew plot for PM10 in AQI dataset obtained from CPCB

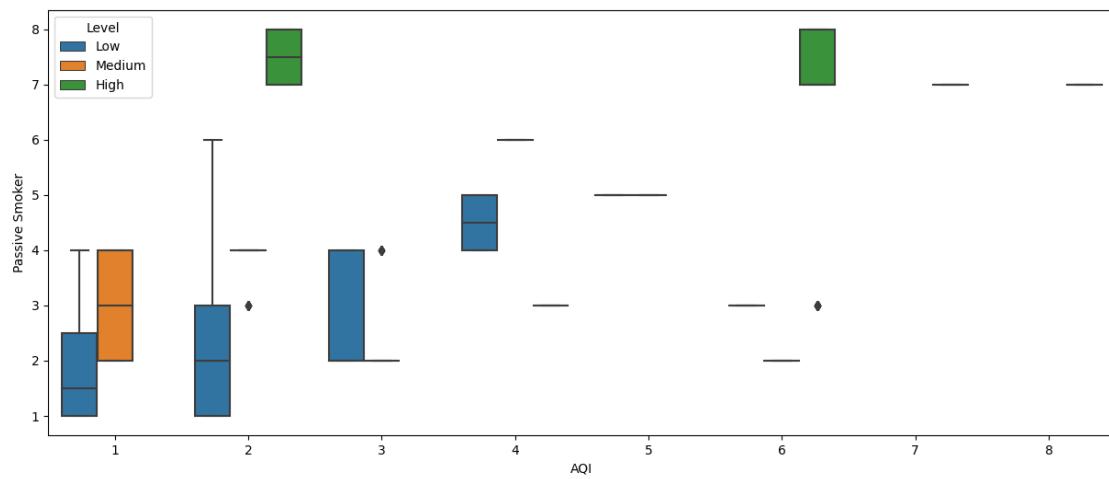


Fig 7.2.8: Relation of AQI and passive smokers of lung cancer dataset obtained from data.world

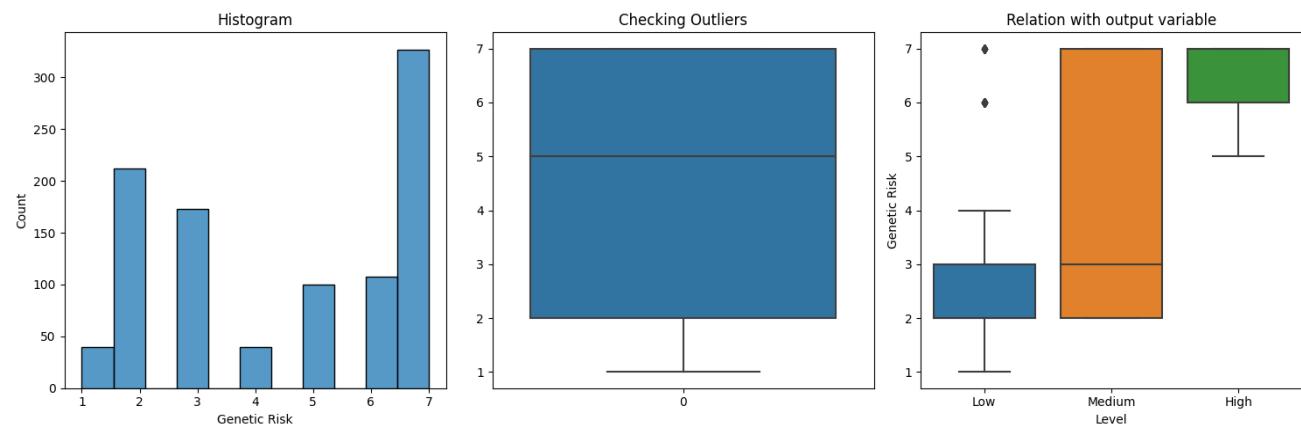


Fig 7.2.9: For checking outliers of lung cancer dataset obtained from data.world

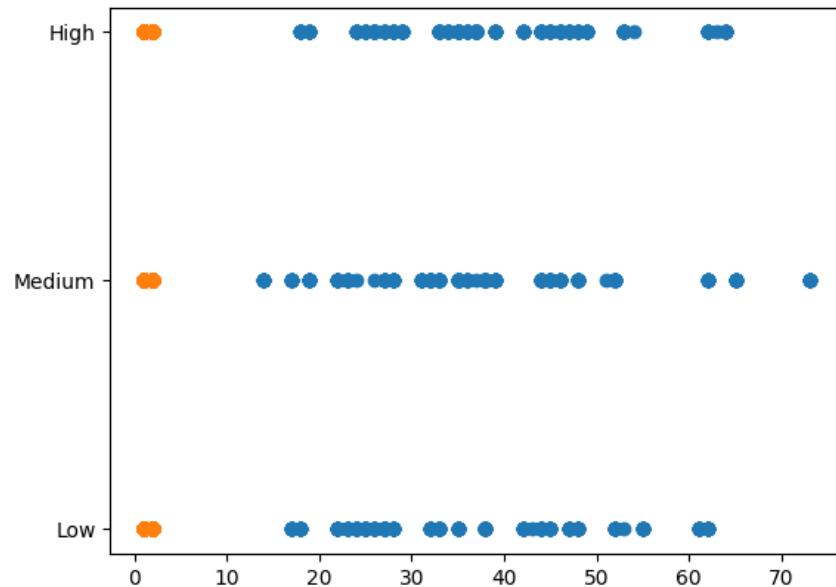


Fig 7.2.10: Scatter plot to check the risk level based on gender and age of lung cancer dataset

obtained from data.world

7.3 ARIMA MODEL IMPLEMENTATION:

Autocorrelation plots:

ARIMA model works on stationary data, to check if the data is stationary, autocorrelation plots- ACF and PACF are unstructured.

ACF plots represent the correlation of time series with its lags. It measures the linear relationship between lagged values of the time series. PACF plots represent partial correlation of time series with its lags, after removing the effects of lower-order-lags between them.

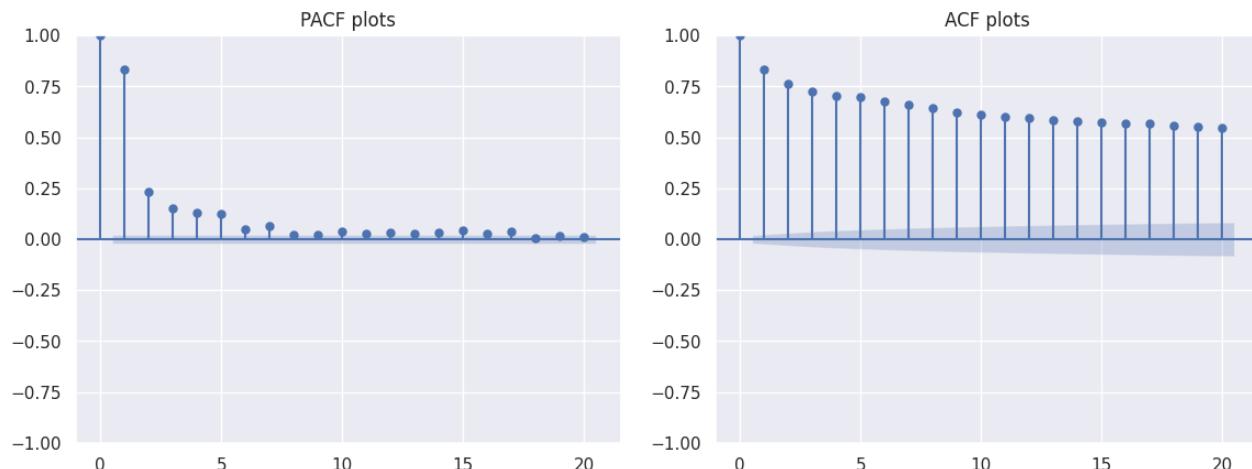


Fig 7.3.1 PACF & ACF plots for AQI dataset

The ACF plot shows the correlations with the lags are high and positive with very slow decay. While the PACF plot shows the partial autocorrelations have a single spike at lag 1. These are both signs of a trended time series. So the time series is not stationary.

```
from statsmodels.tsa.stattools import adfuller
from numpy import log
result = adfuller(ts)
print('ADF Statistic: %f' % result[0])
print('p-value: %f' % result[1])

ADF Statistic: -1.224505
p-value: 0.662967
```

Fig 7.3.2 ADF test for AQI dataset

In our analysis, we employed the Augmented Dickey Fuller (ADF) test as another method to assess the stationarity of our data. The resulting p-value of 0.66 from this test indicates a significant value. In this context, a large p-value signifies that the test doesn't provide sufficient evidence to reject the null hypothesis. Consequently, the ADF test reinforces the observation that our time series data is non-stationary.

```
] ts_train = ts[:50]
ts_test = ts[50:]
```

Fig 7.3.3 Training, testing split for AQI dataset

The dataset for AQI was split into a 50:50 ratio. This was done to ensure an equal amount of data for evaluating model performance. This balance allowed the team to assess how well the model generalizes to unseen data, which is crucial for model validation. Another reason for this choice was to ensure that the testing set covers a similar time period as the training set. This is crucial for time series models as they rely on historical data to make accurate forecasts.

```
for i in pqd_combination:  
    A_model = ARIMA(ts_train,order= i).fit()  
    predict = A_model.predict(len(ts_train),len(ts)-1)  
    e = np.sqrt(mean_squared_error(ts_test,predict))  
    pqd.append(i)  
    error.append(e)
```

Fig 7.3.4 p,q,d value selection

In ARIMA modeling, we focus on three vital parameters: p (autoregressive terms), d (non seasonal differences), and q (lagged forecast errors). We use PACF and ACF plots to guide our choices. A significant spike in PACF at 'p' without further spikes, along with a slowly decaying ACF, suggests an ARIMA(p, d, 0) model. Similarly, a substantial spike in ACF at 'q' without extending further, and a gradual PACF decay, implies an ARIMA(0, d, q) model. In our case, we choose 'p' as 2 and 'q' as 0, while 'd' is set to the minimum value to make the time series stationary. This approach ensures effective ARIMA parameter selection.

```
arima_predict = model_ts_fit.predict(start = len(ts_train),end = len(ts))  
  
import matplotlib.pyplot as plt  
residuals = model_ts_fit.resid[1:]  
fig, ax = plt.subplots(1,2)  
residuals.plot(title='Residuals', ax=ax[0])  
residuals.plot(title='Density', kind='kde', ax=ax[1])  
plt.show()
```

Fig 7.3.5 Fitting the model and generating residuals

Once the autoregressive and moving average components of the model have been taken into account, these residuals show the unexplained variation or noise in the time series data.

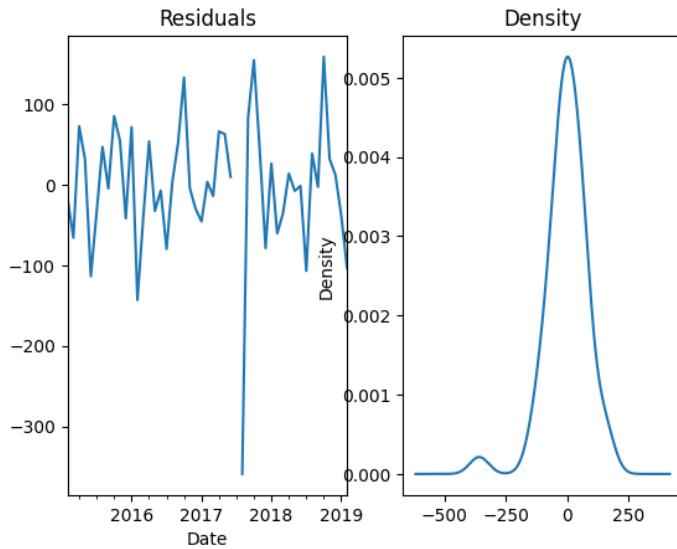


Fig 7.3.6 Residuals generated from ARIMA modelling of AQI dataset

The residuals look random in general, and their density looks normally distributed with a mean of around 0.

```
from sklearn.metrics import mean_absolute_error, mean_absolute_percentage_error, mean_squared_error

forecast_test = model_ts_fit.forecast(len(ts_test))

rmse = np.sqrt(mean_squared_error(ts_test, forecast_test))
mae = mean_absolute_error(ts_test, forecast_test)
mape = mean_absolute_percentage_error(ts_test, forecast_test)
print(f'mae -: {mae}')
print(f'mape -: {mape}')
print(f'rmse -: {rmse}')

mae -: 69.24219899616345
mape -: 0.4782969407191707
rmse -: 79.14471630337096
```

Fig 7.3.7 Error calculation

The MAPE is a percentage-based metric that measures the average of the absolute percentage differences between predicted and actual values.

The error between the actual values and the predicted values is 0.47% which is very minimal.

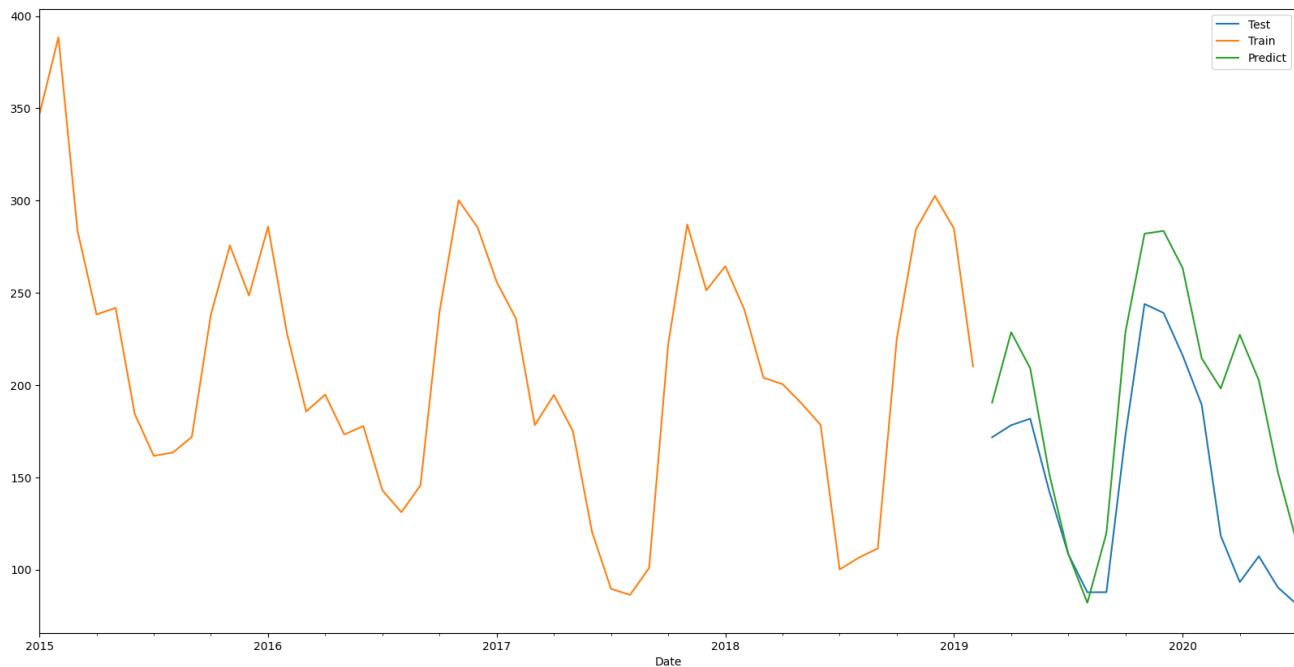


Fig 7.3.8 Plot of predicted values

7.4 LSTM MODEL IMPLEMENTATION:

```
import math
dataset = lstm_df.values
training_data_len = math.ceil(len(dataset)*.8)
training_data_len

800
```

Fig 7.4.1 Training, testing split of data for lung cancer dataset

```
from sklearn.preprocessing import MinMaxScaler

sc = MinMaxScaler(feature_range=(0,1))
scaled_data = sc.fit_transform(lstm_df[['Age', 'AQI']])

array([[0.3220339 , 0.14285714],
       [0.05084746, 0.28571429],
       [0.3559322 , 0.42857143],
       ...,
       [0.18644068, 0.42857143],
       [0.06779661, 0.71428571],
       [0.55932203, 0.71428571]])
```

Fig 7.4.2 Scaling of lung cancer dataset to fit the model

A preprocessing tool called MinMaxScaler is used to scale numerical characteristics in the range of (0, 1). The fit_transform method applies the transformation after fitting the scaler to the data (i.e., determining the minimum and maximum values). The variable scaled_data contains the returned scaled data. The fit_transform method applies the transformation after fitting the scaler to the data (i.e., determining the minimum and maximum values). The variable scaled_data contains the returned scaled data.

```
model_lstm = Sequential()
model_lstm.add(InputLayer((12,1)))
model_lstm.add(LSTM(50))
model_lstm.add(Dense(34 , 'relu'))
# model_lstm.add(Dropout(0.25))
model_lstm.add(Dense(15 , 'relu'))
model_lstm.add(Dense(1 , 'relu' ))

model = Sequential()
model.add(LSTM(50, return_sequences=True, input_shape=(x_train.shape[1],1)))
model.add(LSTM(50, return_sequences=False))
model.add(Dense(25))
model.add(Dense(1))
```

Fig 7.4.3 LSTM model initialisation

In this sequential neural network model, we've incorporated an LSTM layer dedicated to sequence processing, complemented by a Dense layer for the final output. To optimize the model, we've employed the Adam optimizer, known for its efficiency in training neural networks. The input shape specified is 12,1, indicating that the model anticipates input data characterized by 12 time steps and a single input feature.

Moving forward, we've included an LSTM layer with 50 units, a fundamental component for managing sequential data and capturing temporal dependencies. This layer plays a crucial role in understanding and learning from the sequential patterns within the data.

A fully connected dense layer with 34 units and “ReLU” activation function which processes the output from the LSTM layer.

Another fully connected dense layer with 15 units and ReLU activation is added.

A final dense layer with 1 unit and ReLU activation.

```
x_test = np.array(x_test)
x_test = x_test.reshape((x_test.shape[0], x_test.shape[1], 10))
```

Fig 7.4.4 Reshaping test data

It is essential to reshape the ARIMA residuals prior to supplying them as input to an LSTM (Long Short-Term Memory) neural network in order to guarantee that the data conforms to the input specifications of the LSTM layer. Typically, input data for LSTM layers takes the form of a 3D tensor with dimensions (batch_size, time_steps, features) in a predetermined shape.

```
low_threshold = 1.10
high_threshold = 1.1

thatarray = []
categories = []

# Categorize the predictions
for prediction in predictions:
    per_cent = prediction[0] / (prediction[0]+prediction[1])
    thatarray.append(per_cent)
x = thatarray
x_norm = (x-np.min(x))/(np.max(x)-np.min(x))
print(x_norm)
```

.. [0.5034645 0.20249954 0.76791406 0.80944556 0.5788333 0.6824281
0.6989642 0.8321171 0.54046166 0.8167146 0.865501 0.67782295
0.69182265 0.67732704 0.97840536 0.8031117 0.42357558 0.68966883
0.51739335 0. 0.21008034 0.4038655 0.5739447 0.7449166
0.21674012 0.56863105 0.7432021 0.5743273 0.48653167 0.22172786
0.44511357 0.5951993 0.5136384 0.48847294 0.24333669 0.3958596
0.7993142 0.79816645 0.9215706 0.6504612 0.49354568 0.6367874
0.5082964 0.75032943 0.6109985 0.48847294 0.63735425 0.67693025
0.7329857 0.5170391 0.325847 0.6734162 0.66368157 0.6492993
1. 0.16443966 0.6014765 0.3722812 0.7346861 0.7338784
0.6649852 0.82619417 0.74619186 0.7693027 0.7168464 0.51062024
0.9610758 0.62064815 0.80129796 0.9252405 0.98855084 0.6581412
0.23891573 0.45710117 0.13916087 0.04201324 0.40239185 0.40229267
0.88334066 0.6192453 0.41474786 0.8162045 0.8943647 0.43970072
0.24089949 0.237102 0.6137333 0.6368158 0.6268828 0.72123903
0.5430264 0.6726085 0.70086294 0.9803466 0.71584034 0.40511245
0.7024783 0.66739404 0.54094344 0.9672963]

Fig 7.4.5 Predictions of LSTM model on the lung cancer dataset

The predictions generated are based on a percentage probability metric.

i.e. 0.472913 value generated by the model would mean that there is a 47% chance of the person contracting lung cancer in the current conditions they are based on their genetic history, AQI, etc.

```
loss, mae = model.evaluate(x_test, y_test)

7/7 [=====] - 1s 15ms/step - loss: 0.1241 - mae: 0.3265
```

Fig 7.4.6 Error analysis

The mean absolute error is 0.32 which is quite low and proves the model is it suggests that the model's predictions are very close to the actual observed values in the dataset.

7.5 IoT STATION IMPLEMENTATION:

IoT station:

The IoT station consists of two sensors - MQ-7 gas sensor for measuring the concentration of CO gas in the atmosphere and GP2Y1010AUF sensor for measuring the concentration of PM 2.5 dust particles in the atmosphere. The ESP8266 WiFi module will take reading from the sensors and upload it to the ThingSpeak cloud platform.

- **MQ-7 gas sensor:** The MQ 7 gas sensor plays a critical role in detecting CO. We provide power to MQ 7 from digital pin 1. We take readings from it at analog 0 pin.
- **GP2Y1010AUF particulate matter sensor:** The GP2Y1010AU0F dust sensor is instrumental in particulate matter detection. We provide power to it from digital pin 2. We take its reading at analog pin 0.

This hardware configuration enables the system to gather data from diverse sources, covering gasses and particulate matter. The flexibility in the ESP board selection and dust sensor options allows for adaptability based on factors like cost and availability, making the project versatile and well-suited for various operational scenarios. The elaborate and delicate wiring ensures the efficient flow of data between the hardware components and the ESP board, thereby facilitating accurate and reliable air quality monitoring.

IoT Circuit Diagram:

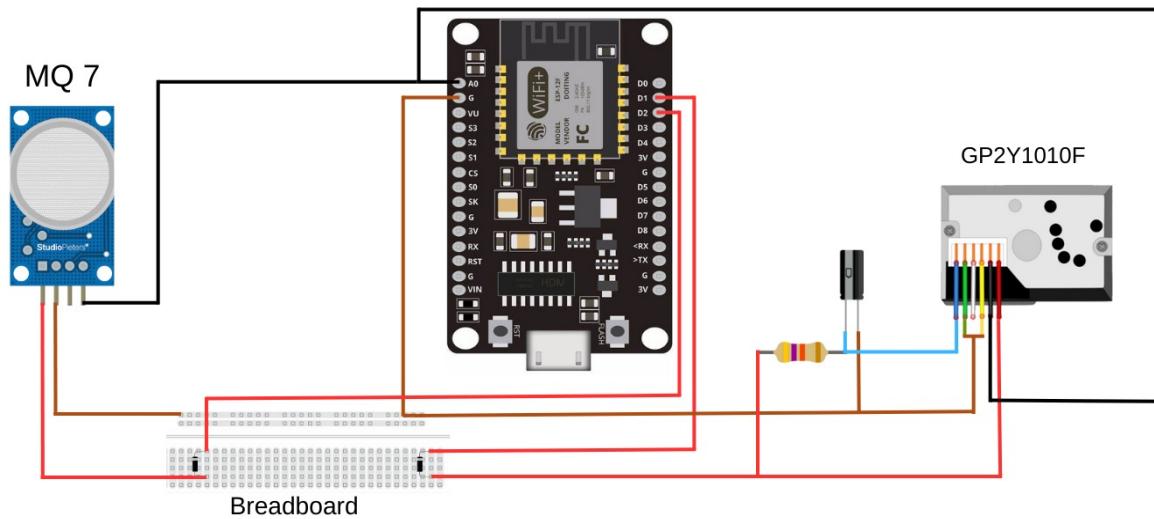


Fig 7.5.1 Circuit diagram

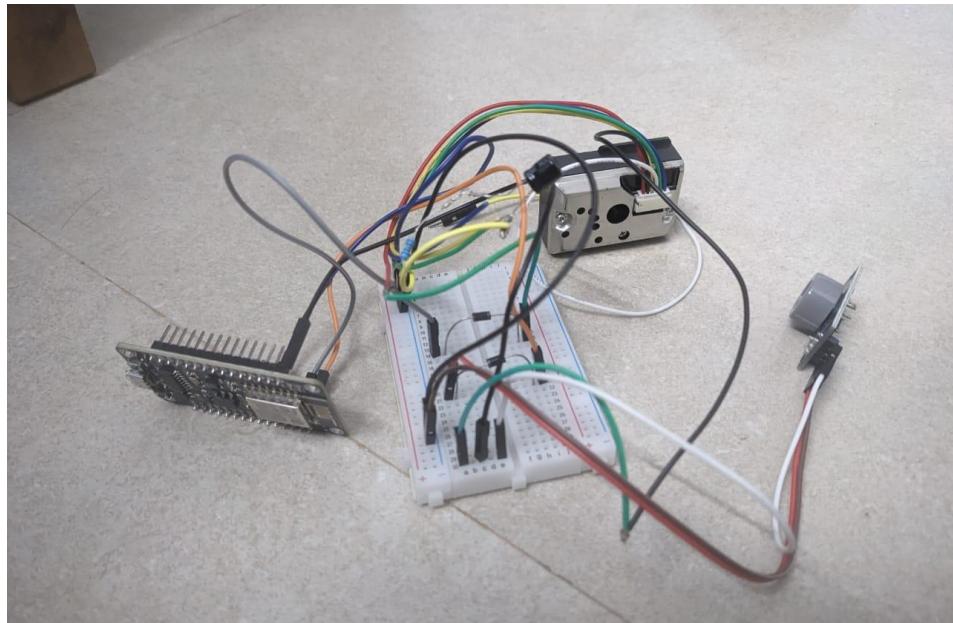


Fig 7.5.2 Real life diagram

Collection of data on ThingSpeak cloud:

The data pollutant readings collected by the sensors are analyzed and converted into meaningful values, and uploaded on the ThingSpeak cloud platform to be fed to the ML model..

```
final.ino
1 #include <ESP8266WiFi.h>
2 #include "ThingSpeak.h"
3 #include "MQ7.h"
4
5 MQ7 mq7(A0,5.0);
6
7 char ssid[] = "Phone";
8 char pass[] = "hereyougo2021";
9 WiFiClient client;
10
11 const char * myWriteAPIKey = "ZL1K7BIINL8Z2EY0";
12 float dust = 0;
13 float co = 0;
14 float co_read = 0;
15 float calcVoltage = 0;
16 float dustDensity = 0;
17 int Pin_D1 = 4;
18 int Pin_D2 = 5;
19
20 void setup() {
21   Serial.begin(9600);
22   pinMode(Pin_D1, OUTPUT);
23   pinMode(Pin_D2, OUTPUT);
24   WiFi.mode(WIFI_STA);
25   ThingSpeak.begin(client);
26   if(WiFi.status() != WL_CONNECTED){
27     while(WiFi.status() != WL_CONNECTED){
28       WiFi.begin(ssid, pass);
29       Serial.print(".");
30       delay(5000);
31     }
32     Serial.print("Connected.");
33   }
34 }
```

```
35
36 void dustSensor() {
37     digitalWrite(Pin_D2, LOW);
38     delay(1000);
39     digitalWrite(Pin_D1, HIGH);
40     delay(10000);
41 }
42
43 void coSensor() {
44     digitalWrite(Pin_D1, LOW);
45     delay(1000);
46     digitalWrite(Pin_D2, HIGH);
47     delay(10000);
48 }
49

50 void loop() {
51
52     dustSensor();
53     dust = analogRead(0);
54     calcVoltage = dust * (3 / 1024.0);
55     dustDensity = 0.17 * calcVoltage - 0.1;
56     Serial.print("Dust Density = ");
57     Serial.println(dustDensity);
58     Serial.println("");
59     int x = ThingSpeak.writeField(2253626, 3, dust, myWriteAPIKey);
60     if(x == 200){
61         Serial.println("Channel update successful.");
62     }
63     else{
64         Serial.println("Problem updating channel. HTTP error code " + String(x));
65     }
66

67     coSensor();
68     co_read = analogRead(0);
69     co = mq7.getPPM();
70     Serial.print("CO = ");
71     Serial.println(co);
72     Serial.println("");
73     int y = ThingSpeak.writeField(2303832, 1, co, myWriteAPIKey);
74     if(y == 200){
75         Serial.println("Channel update successful.");
76     }
77     else{
78         Serial.println("Problem updating channel. HTTP error code " + String(y));
79     }
80 }
81
82 }
```

Fig 7.5.3 Arduino IDE code for collecting readings from the sensors and connection to ThingSpeak cloud platform

7.6 CLOUD PLATFORM

In our project, we are excited to leverage the Streamlit platform as a pivotal component of our cloud-based system. Streamlit offers an intuitive, user-friendly interface that seamlessly pulls from a GitHub repository to build our application. By doing so, we ensure that our system is always up-to-date with the latest code, making it easy to maintain and extend as our project evolves. The cloud hosting provided by Streamlit allows us to host our application on a specially chosen link, ensuring it's readily accessible to our users. This cloud-based approach simplifies deployment and eliminates the need for users to install any software locally, enhancing the accessibility of our system.

Moreover, Streamlit not only offers hosting but also provides an interactive interface for users to input data and interact with our IoT module and machine learning model. This seamless interconnectivity between Streamlit, the IoT module, and the ML model streamlines the user experience and enhances the effectiveness of our project. As we move forward, the combination of Streamlit's capabilities with our IoT and ML components ensures a comprehensive and user-friendly solution for continuous air quality monitoring and health hazard prediction.

Following are various screenshots of our working application hosted on Streamlit.

Welcome to your dashboard

Please enter your age:

What's your gender?

 Male
 Female

On a Scale of 1 to 8, how allergic are you to dust particles?

1  8

On a scale of 1 to 8, how would you classify your occupational hazards?

1  8

On a scale of 1 to 8, how would you classify your genetic risk of lung cancer?

1  8

Do you currently have any chronic lung disease? If yes how drastic?

1  7

On a scale of 1-8 how often do you smoke?

1  8

On a scale of 1 to 8, what would be your exposure to cigarette smoke?

1  8

Have you noticed any clubbing of finger nails? If yes how extreme is it?

1  9

On a scale of 1 to 7, how frequently do you contract a cold?

1  7

Submit

Click on me to check your risk.

Data from your station retrieved! Running ARIMA...

AQI forecast ready! Running LSTM...

Model has finished running.

Your risk of lung cancer is: 20.4%

Welcome to your dashboard

Please enter your age:

What's your gender?

 Male
 Female

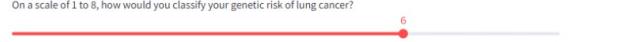
On a Scale of 1 to 8, how allergic are you to dust particles?

1  8

On a scale of 1 to 8, how would you classify your occupational hazards?

1  8

On a scale of 1 to 8, how would you classify your genetic risk of lung cancer?

1  8

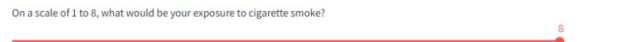
Do you currently have any chronic lung disease? If yes how drastic?

1  7

On a scale of 1-8 how often do you smoke?

1  8

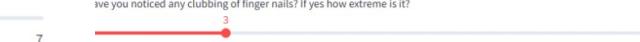
On a scale of 1 to 8, what would be your exposure to cigarette smoke?

1  8

Have you noticed any clubbing of finger nails? If yes how extreme is it?

1  9

On a scale of 1 to 7, how frequently do you contract a cold?

1  7

Submit

Click on me to check your risk.

Data from your station retrieved! Running ARIMA...

AQI forecast ready! Running LSTM...

Model has finished running.

Your risk of lung cancer is: 89.7%

Fig. 7.6.1 Working application on Streamlit

CHAPTER VIII

RESULTS AND DISCUSSION

In this study, we harnessed the strengths of ARIMA and LSTM models to develop a hybrid prediction model for lung cancer incidence and have deployed this model on the cloud, alongside an IoT station to collect real time pollutant data. This unique approach capitalizes on ARIMA's ability to capture temporal patterns while leveraging LSTM's powerful sequential learning capabilities. It also combines the flexibility and scalability of cloud with the real life deployed IoT stations. To assess our hybrid model's performance, we compared it with two alternative models: the GRU model, a variation of LSTM focused on sequential learning, and the SARIMA model, renowned for its time series forecasting efficiency.

Following a comprehensive evaluation using various performance metrics such as Mean Squared Error (MSE), Mean Absolute Percentage Error (MAPE), and Root Mean Squared Error (RMSE), our hybrid ARIMA-LSTM model consistently outperformed both SARIMA and GRU models across all metrics. Our model yielded impressive results, with an MSE value of 686.16, RMSE of 79.144, MAPE of 0.47, MAE of 69.24, and a minimal loss of 0.124. In contrast, SARIMA obtained an MSE of 782.62, RMSE of 27.97, and MAPE of 311.37, while GRU achieved an accuracy of 1.13.

The IoT station deployed consists of the GP2Y1010AUF dust sensor to measure the concentration of PM 2.5 and the MQ-7 gas sensor to measure the concentration of CO gas. The ESP8266 has a shortcoming with only one analog pin. We are accepting analog inputs from two sensors. To

circumvent this, we are alternating power to the sensors. We achieve this by giving them power using digital pin 1 and 2 and changing their state in the code.

The data collected is uploaded to the ThingSpeak cloud platform and fed directly to the ARIMA model to generate forecasts. These forecasts are subsequently fed to the LSTM model to make predictions on the lung cancer incidence rate of the user.

The models are created in notebooks and they are then converted into regular python. This is then rewritten into a standard Streamlit application. Then we create a cloud website link using Streamlit community cloud. Then we link the github repository to the Streamlit cloud. Then Streamlit fetches the code and builds and hosts the application.

This research introduces a novel ARIMA-LSTM hybrid model for lung cancer risk prediction, demonstrating its superiority over traditional SARIMA and GRU models. Our comparative study underscores the enhanced prediction accuracy and potential to enhance early risk assessment and intervention strategies in the realm of lung cancer prevention. This innovative fusion of LSTM and ARIMA frameworks not only advances risk prediction techniques but also holds promise for applications in public health and personalized healthcare. The inclusion of an IoT station to collect live pollutant data broadens the horizon and allows us to extend the applications of our study into smart home systems as well. Governments can also use our system to identify pollution hotspots and create policies and regulations to reduce the harmful emissions and issue public health warnings in high risk areas.

CHAPTER IX

CONCLUSION AND FUTURE WORK

Accurate air quality monitoring and forecasting carry substantial theoretical and practical significance for the general populace. By alerting individuals to potential health hazards in their immediate surroundings, this research endeavors to offer valuable insights that could assist both government entities and the public. The proactive dissemination of information regarding air pollution may lead to a reduction in health-related issues and enhance emergency response capabilities, particularly during days of heightened pollution levels. The study introduces a hybrid model, harmonizing ARIMA and Adaptive LSTM, harnessing air pollutant concentration data to predict air quality thresholds that may pose health risks. Empirical findings demonstrate the efficacy of both models on time series data, with Adaptive LSTM exhibiting superior performance in the real-time and large data domains. In essence, the research establishes that these time-series prediction models demonstrate proficiency, particularly when used in tandem as a hybrid system. Furthermore, the incorporation of a live Internet of Things (IoT) station for real-time air pollutant monitoring and a cloud hosted application for predictive capabilities augments the project's significance.

To enhance the project's efficacy, more comprehensive and granular data collection efforts could be undertaken. The integration of geographical data would unveil regional variations in air pollution effects. This versatile model can be further extended to investigate various health hazards linked to air pollution. Moreover, the IoT station could benefit from improvements, including the substitution of the ESP8266 module with a more potent counterpart featuring additional analog input pins. A deeper exploration of sensor technology and the incorporation of sensors capable of detecting a

broader spectrum of gases would broaden the station's capabilities. The inclusion of a more dependable dust sensor would bolster data accuracy. While the project exhibits economic viability, exploration into cutting-edge technologies could further optimize its scope.

REFERENCES/BIBLIOGRAPHY

- [1] An Application of IoT and Machine Learning to Air Pollution Monitoring in Smart Cities
By: Muhammad Taha Jilani, Husna Gul A.Wahab
[\[https://ieeexplore.ieee.org/document/8981707\]](https://ieeexplore.ieee.org/document/8981707)
- [2] How Is the Lung Cancer Incidence Rate Associated with Environmental Risks? Machine-Learning-Based Modeling and Benchmarking
By: Kung-Min Wang, Kun-Huang Chen, Shieh-Hsen Tseng
[\[https://www.mdpi.com/1660-4601/19/14/8445\]](https://www.mdpi.com/1660-4601/19/14/8445)
- [3] Assessment of indoor air quality in academic buildings usng IOT and deep learnings
By: Mohammad Marzouk and Mohammad Atef
[\[https://www.mdpi.com/1667822\]](https://www.mdpi.com/1667822)
- [4] Household Ventilation May Reduce Effects of Indoor Air Pollutants for Prevention of Lung Cancer: A Case-Control Study in a Chinese Population.
By: Jin Z-Y, Wu M, Han R-Q, Zhang X-F, Wang X-S, et al.
[\[https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0102685\]](https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0102685)
- [5] Determination of Air Quality Life Index (AQLI) in Medinipur City of West Bengal(India) During 2019 To 2020 : A contextual Study
By: Samiran Rana.
[\[https://www.researchgate.net/publication/360622768_Determination_of_Air_Quality_Life_Index_Aqli_in_Medinipur_City_of_West_BengalIndia_During_2019_To_2020_A_contextual_Study\]](https://www.researchgate.net/publication/360622768_Determination_of_Air_Quality_Life_Index_Aqli_in_Medinipur_City_of_West_BengalIndia_During_2019_To_2020_A_contextual_Study)
- [6] Air pollution and skin diseases: Adverse effects of airborne particulate matter on various skin diseases,
By: Kim Kyung Eun, Cho Daeho, Park Hyun Jeong
[\[https://pubmed.ncbi.nlm.nih.gov/27018067/\]](https://pubmed.ncbi.nlm.nih.gov/27018067/)

[7] The spatial association between environmental pollution and long-term cancer mortality in Italy.

By: Roberto Cazzolla Gatti, Arianna Di Paola, Alfonso Monaco, Alena Velichevskaya, Nicola Amoroso, Roberto
[\[https://www.sciencedirect.com/science/article/pii/S0048969722055383#:~:text=We%20studied%20the%20links%20between%20cancer%20mortality%20and%20environmental%20pollution%20in%20Italy.&text=Tumor%20mortality%20exceeds%20the%20national%20average%20when%20environmental%20pollution%20is%20higher.&text=Air%20quality%20ranks%20first%20for,to%20the%20average%20cancer%20mortality.\]](https://www.sciencedirect.com/science/article/pii/S0048969722055383#:~:text=We%20studied%20the%20links%20between%20cancer%20mortality%20and%20environmental%20pollution%20in%20Italy.&text=Tumor%20mortality%20exceeds%20the%20national%20average%20when%20environmental%20pollution%20is%20higher.&text=Air%20quality%20ranks%20first%20for,to%20the%20average%20cancer%20mortality.)

[8] The nexus between COVID-19 deaths, air pollution and economic growth in New York state: Evidence from Deep Machine Learning

By: Cosimo Magazzino , Marco Mele , Samuel Asumadu Sarkodie
[\[https://www.sciencedirect.com/science/article/pii/S0301479721003030\]](https://www.sciencedirect.com/science/article/pii/S0301479721003030)

APPENDIX A

DEFINITIONS, ACRONYMS AND ABBREVIATIONS

- ARIMA - AutoRegressive Integrated Moving Average.
- LSTM - Long Short-Term Memory
- IoT - Internet of Things
- ML - Machine Learning
- WHO - World Health Organization
- AQI - Air Quality Index
- PM - Particulate Matter
- PMS3003- Plantover Particulate Matter Sensor
- INAAQS - Indian National Ambient Air Quality Standards
- ANN- Artificial Neural Network
- LCD - Liquid Crystal Display
- CNN - Convolutional Neural Network
- LSTM - Long ShortTerm Memory
- IAQ - Indoor Air Quality
- SVM - Support Vector Machine
- IDE - Integrated Development Environment
- OS - Operating System
- AWS - Amazon Web Services
- ROI - Return On Investment

APPENDIX B SUPPORTING DATA

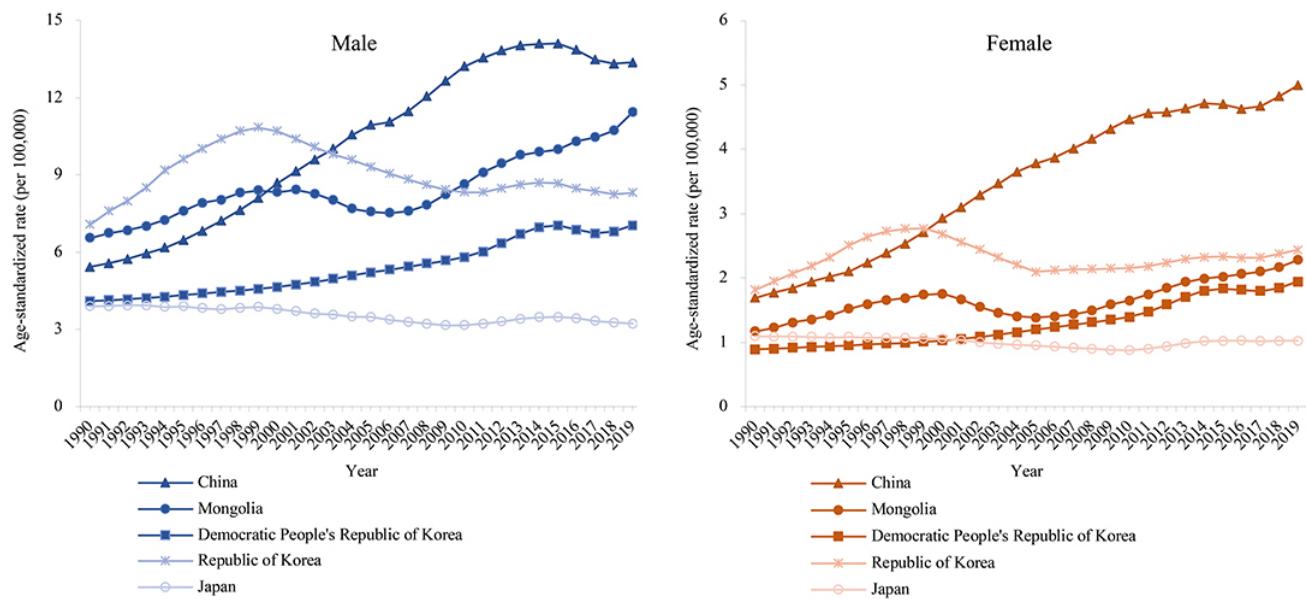


Fig 10.1.1: Lung Cancer death attributable to Long-Term Ambient Particulate Matter (PM2.5)

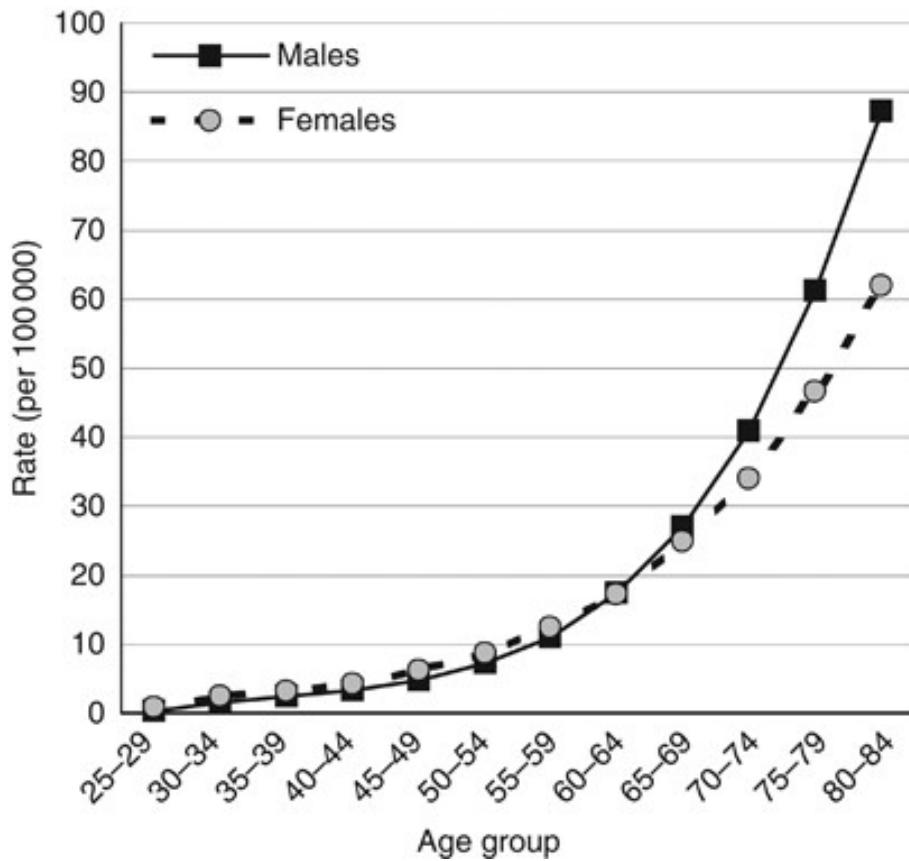


Fig 10.1.2 : Tobacco Attributable Cancer trends in Males and Females