## CHAPTER 3

## METHODOLOGY

### 3.1 INTRODUCTION

This chapter outlines the methodology employed in predicting air quality using machine learning models. The methodology is designed around a system architecture that guides the entire process, from data collection to model evaluation. The stages covered in this chapter include data collection, data preprocessing, data analysis, model development, and evaluation.

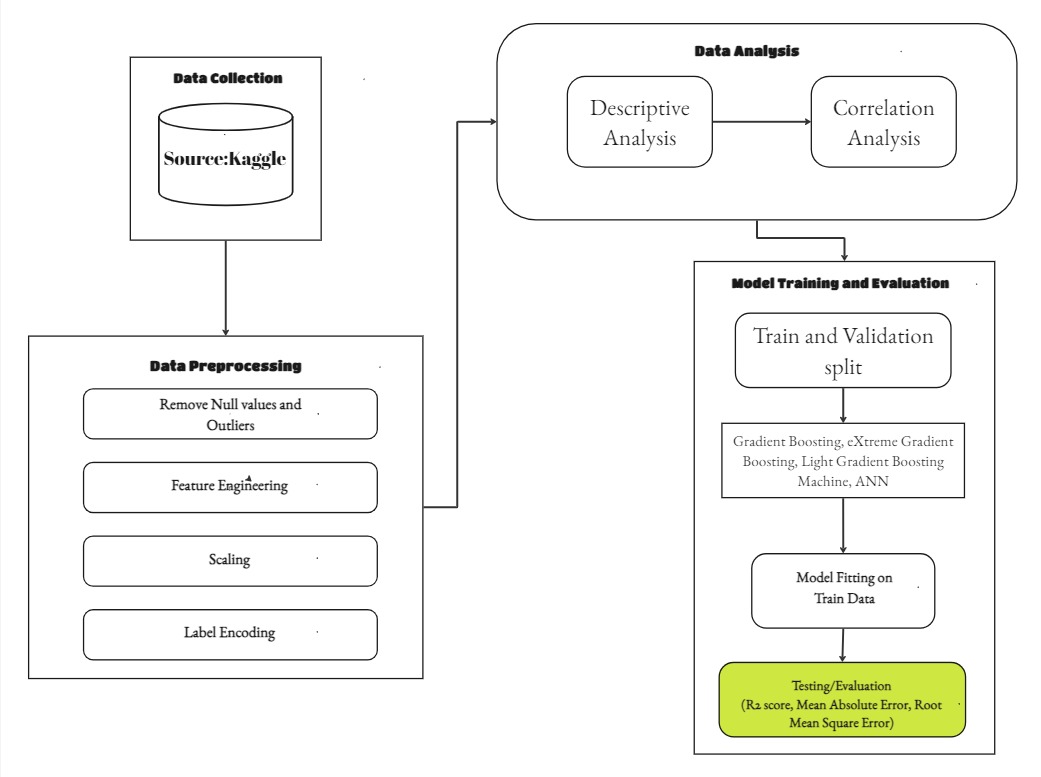


Figure 3. 1: Shows the System architecture of this research

### 3.2 SYSTEM ARCHITECTURE

The system architecture as shown in figure 3.1 serves as the framework within which the methodology is executed. It integrates various components, ensuring a seamless flow of data and processes. The architecture comprises the following key components:

* Data Collection: Acquiring the dataset from a reliable source.
* Data Preprocessing: Preparing the data for analysis.
* Data Analysis: Understanding and exploring the data.
* Model Development: Training and validating machine learning models.
* Model Evaluation: Assessing the performance of the models.

### 3.3 DATA COLLECTION

The dataset for this study is sourced from Kaggle, a well-known platform for data science competitions and datasets. The variables include Particulate Matter 2.5 micrometer (PM2.5), Particulate Matter 10 micrometer (PM10), Nitrogen dioxide(NO2), Carbon Mono-oxide (CO), Sulphur dioxide (SO2), and Ozone (O3), along with meteorological factors, temperature, humidity, and wind speed the dataset also contains 5811 entries for different times measuring the health impact of air quality at these specific times as shown in figure 3.2.

The data is carefully selected to ensure it covers a broad temporal and spatial range, providing a comprehensive basis for model training and evaluation. The dataset is imported into the environment using pandas (A library in python for importing and transforming dataset), ready for preprocessing.

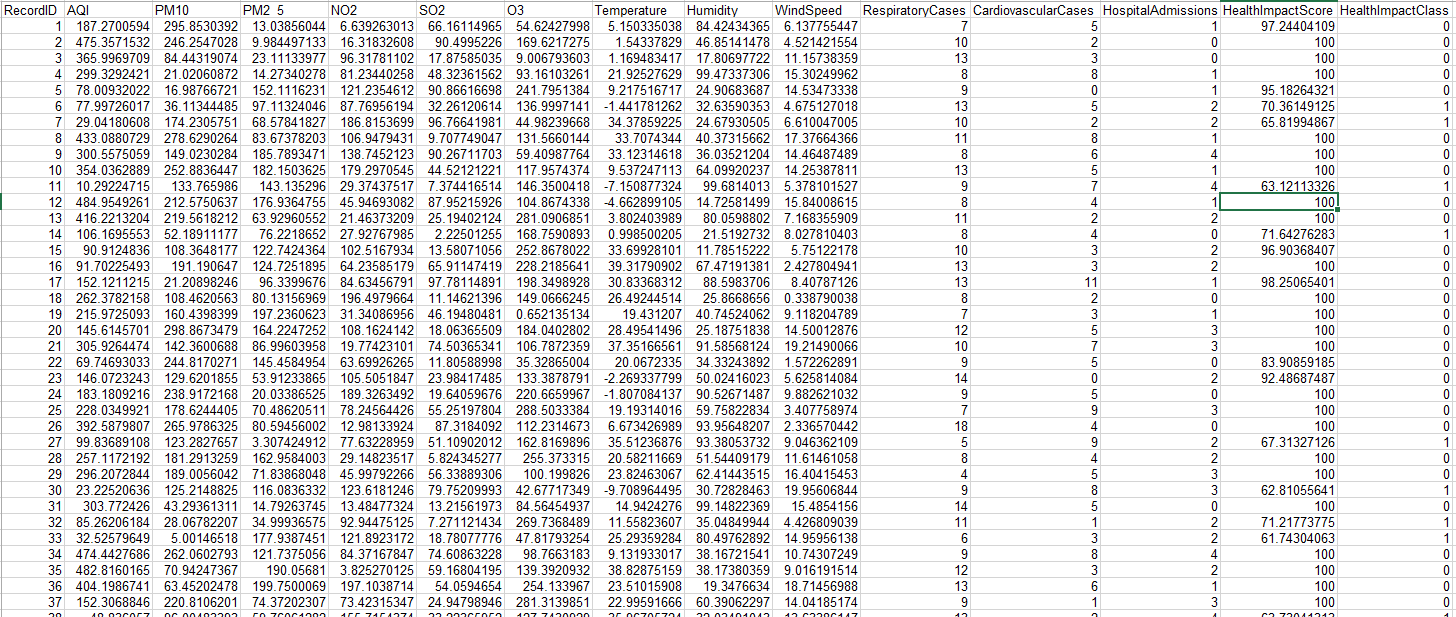


Figure 3. 2: Dataset overview

### 3.4 DATA PREPROCESSING

The raw dataset often contains inconsistencies, missing values, and irrelevant features that need to be addressed before analysis. The data preprocessing phase includes the following steps:

#### 3.4.1 HANDLING MISSING VALUES AND OUTLIERS

Outliers: Detected using IQR (Interquartile Range) and either removed or transformed to minimize their impact using pandas and matplotlib.

#### 3.4.3 SCALING AND NORMALIZATION

To ensure that all features contribute equally to the model and improve convergence speed, the data is scaled or normalized and for this project we would be using Standard Scaling which is a type of scaling that is Applied to normalize the data to a mean of zero and a standard deviation of one using StandardScaler() in Scikit learn library.

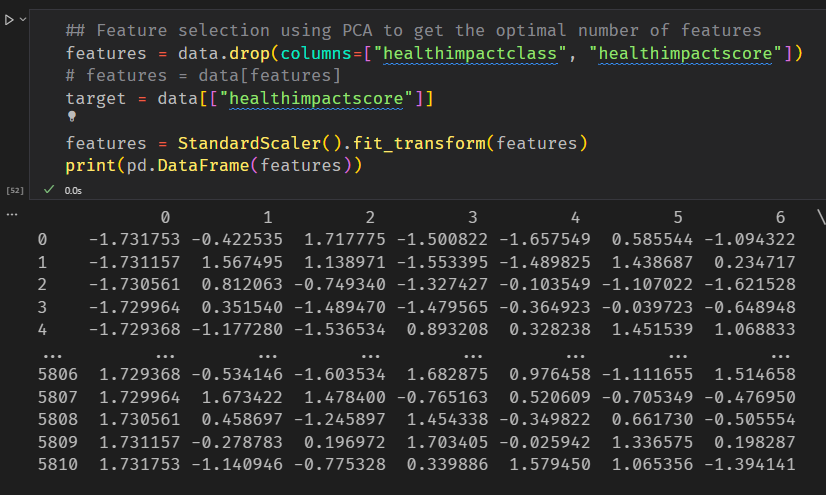


Figure 3. 3 shows the scaled dataset

#### 3.4.4 LABEL ENCODING

If the dataset includes categorical variables, they are converted into numerical values using label encoding. This process ensures that machine learning algorithms can interpret the categorical data.

### 3.5 DATA ANALYSIS

Before building models, it's crucial to understand the dataset through exploratory data analysis (EDA). The following analyses are conducted

#### 3.5.1 DESCRIPTIVE ANALYSIS

Descriptive statistics, mean, median, mode, standard deviation, and distribution plots are used to summarize and understand the characteristics of the data as shown in figure 3.4 below.

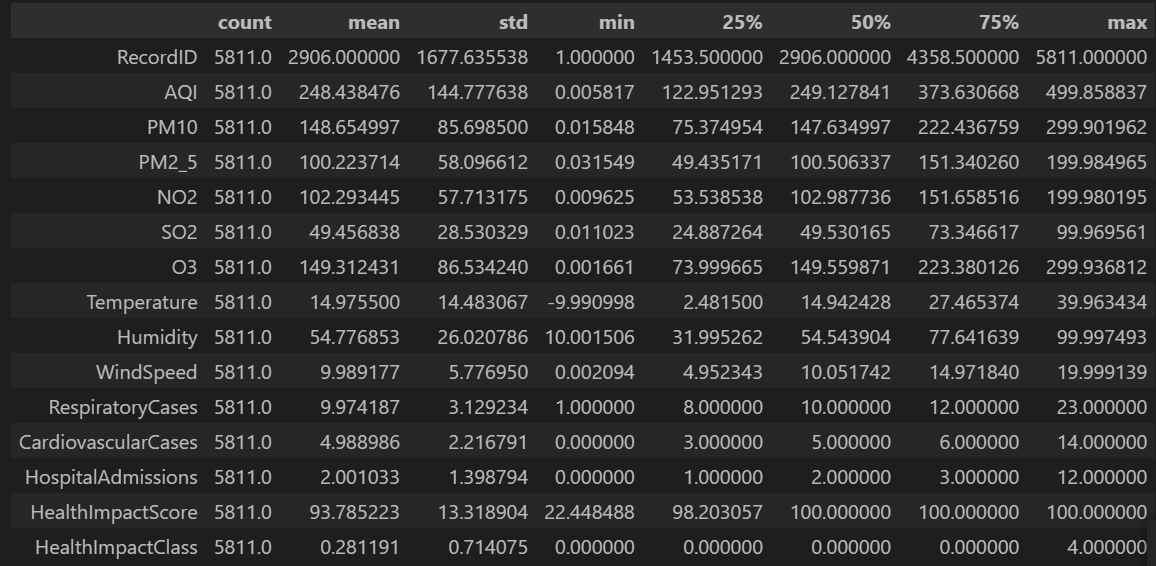


Figure 3. 4: Shows a descriptive table of the data

#### 3.5.2 CORRELATION ANALYSIS

Correlation analysis is performed to identify relationships between variables. Pearson’s correlation coefficient is calculated to understand the linear relationships between features, helping in feature selection and engineering as shown in figure 3.5.

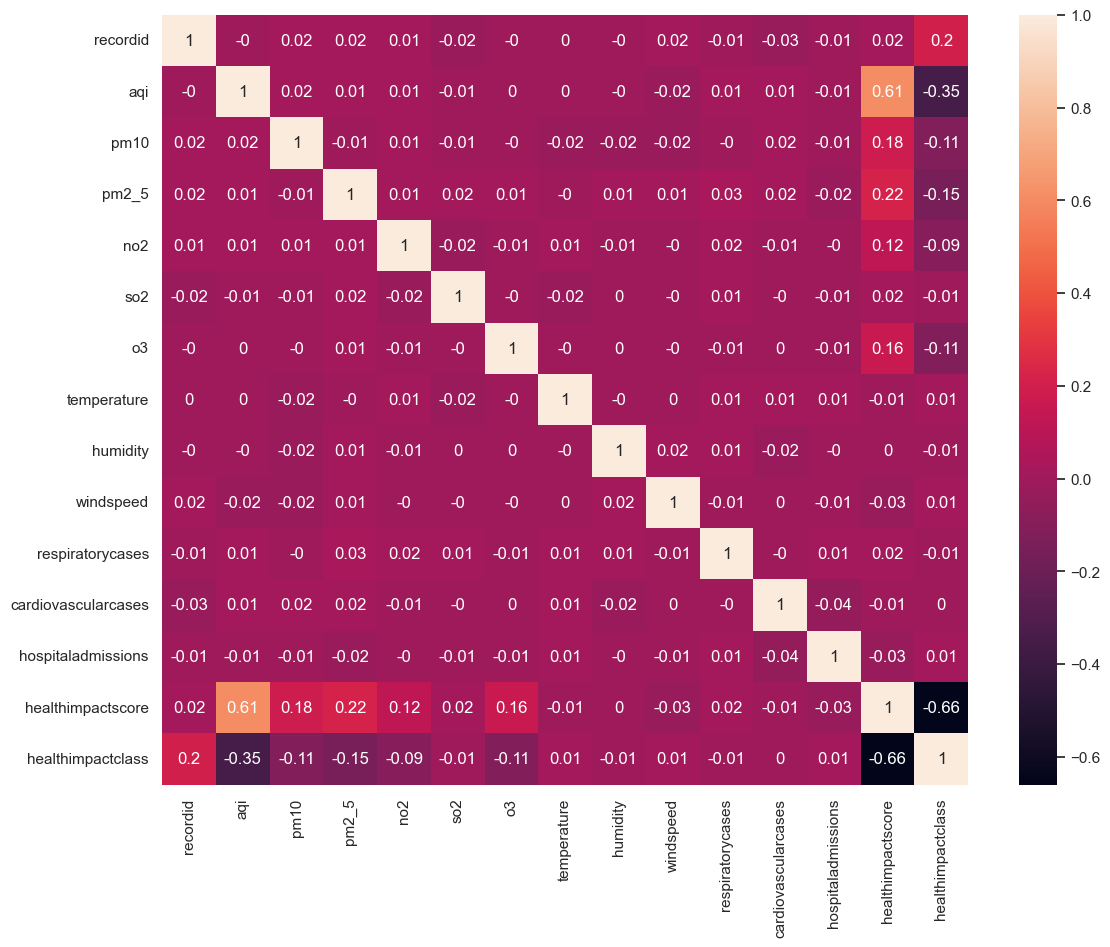


Figure 3. 5 Shows the correlation plot showing relationship between variables

### 3.6 MODEL DEVELOPMENT

The model development phase as displayed in figure 3.6 below involves splitting the data into training and validation sets, selecting appropriate machine learning models, and training these models using Extreme Gradient Boosting, Light Gradient Machine, and scikit learn libraries for machine learning models and Tensorflow for neural network model.

#### 3.6.1 TRAIN-VALIDATION SPLIT

The dataset is split into training and validation sets using a typical split ratio of 70:30. This ensures that the model is trained on a substantial portion of the data and evaluated on unseen data to test its generalization ability.

#### 3.6.2 MODEL SELECTION

Several machine learning models are selected based on their suitability for regression tasks and ability to handle complex datasets. The models chosen were Gradient Boosting, Extreme Gradient Boosting, Light Gradient Machine, and Artificial Neural Network.

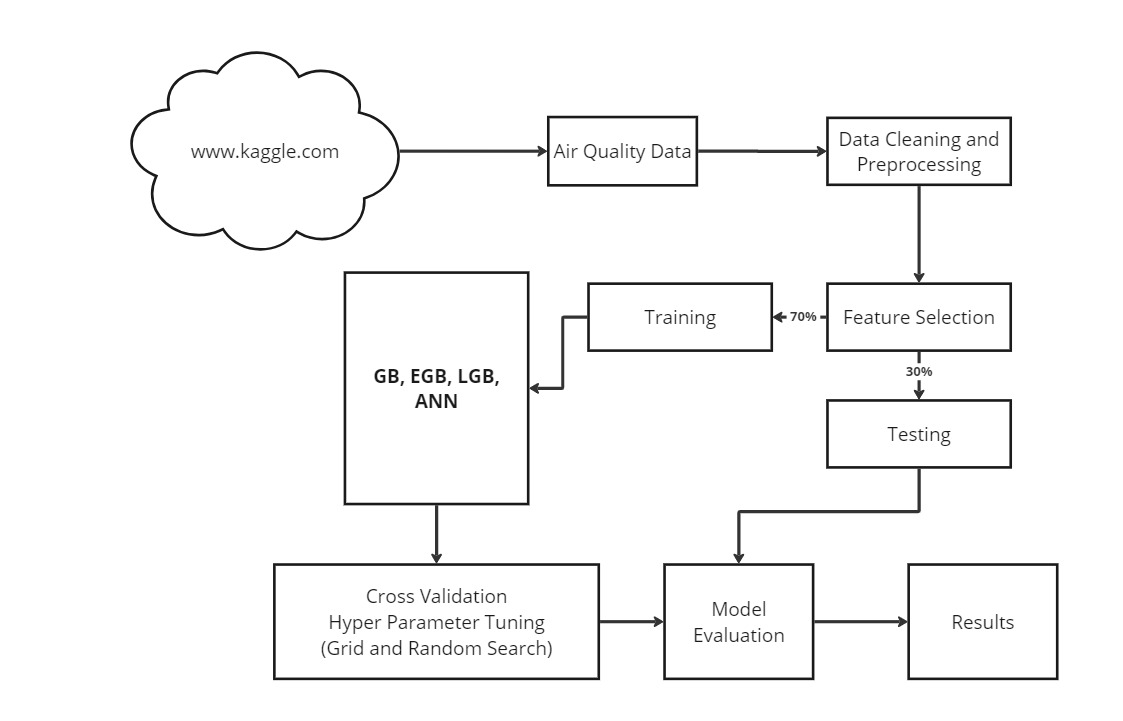


Figure 3. 6: Model architecture for this research

#### 3.6.3 MODEL TRAINING

Each model is trained on the training dataset using appropriate hyper-parameters. cross-validation was used as shown in figure 3.6 may be employed to fine-tune the models and prevent overfitting.

### 3.7 MODEL EVALUATION

Once the models are trained, they are evaluated using the validation set. The following evaluation metrics are used to assess the performance of each model:

* Mean Absolute Error (MAE): The average of the absolute differences between the predicted values and the actual values. It provides a straightforward measure of prediction accuracy by calculating the average magnitude of errors without considering their direction.
* Root Mean Squared Error (RMSE): The square root of the Mean Squared Error. RMSE provides an error measure in the same units as the target variable, offering an intuitive understanding of the model's prediction errors.
* R-squared (Coefficient of Determination): A statistical measure that indicates the proportion of variance in the dependent variable explained by the independent variables. An R-squared value closer to 1 signifies that the model explains a large portion of the variance.

The model with the best performance metrics will be selected for further analysis and potential deployment.

### 3.8 SUMMARY

This chapter provided a detailed overview of the methodology used to predict air quality using machine learning models. The methodology is anchored in a robust system architecture that integrates data collection, preprocessing, analysis, model development, and evaluation. The next chapter will present the results obtained from the models and discuss their implications.