AIR QUALITY PREDICTION USING SUPERVISED MACHINE LEARNING APPROACH

[1] M.Prabu, [2] N.Prabhakaran
[1]Assistant Professor, [2]Assistant Professor,
[1] Department of ECE
[1] Misrimal Navajee Munoth Jain Engineering College, Chennai -97, India.

Abstract: Air pollution is a dangerous threat to both human health and the planet. It involves the release of harmful pollutants into the air that cause damage to animals, crops, and forests. To combat this issue, machine learning techniques are being used to predict air quality from pollutants in the transport sector. Consequently, air quality evaluation and prediction have become crucial research areas with the goal of achieving the most accurate results. To achieve this goal, a dataset analysis using supervised machine learning techniques (SMLT) is necessary. This analysis includes variable identification, uni-variate analysis, bi-variate and multi-variate analysis, missing value treatments, data validation, data cleaning and preparation, and data visualization. The analysis provides a comprehensive guide to sensitivity analysis of model parameters with regards to performance in predicting air quality pollution by accuracy calculation. In this study, we propose a machine learning-based method that accurately predicts the Air Quality Index value by comparing supervised classification machine learning algorithms and selecting the best accuracy prediction results. Furthermore, we compare and discuss the performance of various machine learning algorithms using the given transport traffic department dataset. Additionally, we evaluate a GUI-based user interface for air quality prediction based on attributes.

Index Terms - Pollutant, Machine Learning, SMLT, Visualization.

I. INTRODUCTION

Machine learning is a branch of artificial intelligence (AI) that enables computers to learn from past data without being explicitly programmed. Its primary goal is to predict future outcomes based on historical data. The process involves using specialized algorithms to train models with labelled or unlabelled data, and then making predictions on new, unseen data. Machine learning can be broadly categorized into three types: supervised learning, unsupervised learning, and reinforcement learning. Supervised learning involves training a model with labelled data, while unsupervised learning involves training with unlabelled data. Reinforcement learning is a dynamic process in which the model interacts with its environment and receives feedback to improve its performance. Data scientists use various machine learning algorithms in Python to uncover patterns and insights from data. These algorithms can be classified into two groups: supervised and unsupervised learning. In supervised learning, the model learns to predict output variables based on input variables, whereas in unsupervised learning, the model learns to uncover patterns or structure in data without any labelled data. Classification is a type of supervised learning that involves predicting the class or category of given data points. The goal of classification is to approximate a mapping function from input variables to discrete output variables. Examples of classification problems include speech recognition, handwriting recognition, biometric identification, and document classification.

Supervised learning is a widely used machine learning technique where input variables (X) and output variables (y) are available and an algorithm is used to learn the mapping function from input to output (i.e., $y = f(X)$). The objective is to learn the mapping function well enough so that the algorithm can accurately predict the output variables for new input data. Some of the techniques used in supervised machine learning algorithms include logistic regression, multi-class classification, decision trees, and support vector machines, among others. It is important to note that supervised learning requires labelled data for the algorithm to learn from. Supervised learning problems can be further classified into classification problems, where the goal is to construct a model that can predict the value of a dependent attribute from attribute variables. The dependent attribute can be numerical for regression or categorical for classification. In classification problems, the output variable is a category, such as "red" or "blue". Air pollution is a major threat to human health and the environment, caused primarily by the release of pollutants into the air through energy use and production. Air pollution has numerous negative effects, including damage to animals, crops, and forests, and contributes to the depletion of the ozone layer and global climate change. Climate change worsens air pollution by raising temperatures, which leads to the formation of smog and increases production of allergenic air pollutants such as mold and pollen. Air pollution can be classified into visible and invisible pollutants and is characterized by measurements of chemical, biological, or physical pollutants in the air.

II. RELATED WORKS

Explored three distinct machine learning (ML) algorithms such as support vector machines (SVM), M5P model trees, and artificial neural networks (ANN), to create precise forecasting models for both single-step and multi-step estimations of ground-level ozone (O_3), nitrogen dioxide (NO_2), and sulfur dioxide (SO_2) concentrations [1]. Machine learning techniques were applied to predict the quality of the air [2]. Summarized the overview of recent research on the health impacts associated with various forms of outdoor air pollution [3]. Reviewed various existing air quality forecasting techniques through soft computing [4]. A comprehensive evaluation framework was proposed to enhance prediction accuracy by incorporating pollution, weather, and chemical component forecasts from the WRF-Chem model as input features [5]. The concentration of PM2.5 serves as a significant measure of air pollution, as prolonged exposure to fine-grained PM2.5 particles can pose health risks. A novel model was introduced for estimating PM2.5 concentration by leveraging image quality captured by smartphones while taking into

account the impact of relative humidity [6]. A directed acyclic graph (DAG) is established to assess the air quality characteristics of urban areas using bias networks [7].

III. System Analysis

Exposure to outdoor air pollutants can have adverse health effects that are influenced by the composition and concentration of the pollutants. Common outdoor air pollutants in urban areas include ozone (O3), particle matter (PM), sulphur dioxide (SO2), carbon monoxide (CO), nitrogen oxides (NOx), volatile organic compounds (VOCs), pesticides, and metals, among others. Studies have found that increased concentrations of air pollutants, such as O3, PM, and SO2, are associated with higher mortality and morbidity rates. Despite reductions in emissions of O3 precursors, including VOCs, NOx, and CO, O3 levels continue to exceed the standards set by the Environmental Protection Agency (EPA) to protect public health. The size of the particles is important in determining where they deposit in the respiratory system; particles with a diameter of 2.5 micrometres or less (PM2.5) are of particular concern as they can be deposited in the gas-exchange region of the lungs. To develop a machine learning model for real-time air quality forecasting, several factors need to be considered, including accuracy of the training and testing datasets, specification, false positive rate, precision, and recall. By comparing different supervised machine learning algorithms using Python code, the goal is to create a model that predicts air quality with the highest possible accuracy, potentially replacing existing supervised classification models.

A method is proposed for estimating PM2.5 concentration using a photograph-based approach. The method relies on the observation that the saturation map of an image is sensitive to air quality, with high and low PM2.5 concentrations exhibiting different appearances. High PM2.5 concentrations cause a loss of structure and most pixel values tend towards 0. The structural information loss is quantified by computing the gradient similarity between the saturation and gray-scale maps. The saturation map is then fit to a Weibull distribution to derive a value for colour information. The PM2.5 concentration of an image is estimated by combining these two features and applying a nonlinear mapping procedure. The proposed method is effective, efficient, and low-cost compared to traditional instrument-based methods, which are expensive and labour-intensive. Experimental results on real data captured by a professional PM2.5 instrument demonstrate the method's effectiveness and efficiency. The proposed method is highly consistent with the real sensor's measures and requires a low implementation time. The automatic estimation of air quality can provide valuable guidance for both individuals and industry, given the worldwide concern for air pollution.

Implementing a photograph-based method requires critical evaluation of parameters, and the size of data taken is often high. To overcome this limitation, a machine learning approach can be implemented through a graphical user interface application. This involves combining multiple datasets from various sources to create a generalized dataset. Different machine learning algorithms are then applied to extract patterns and obtain results with maximum accuracy.

IV. Technology Used

Anaconda is a distribution of the Python and R programming languages for scientific computing, data science, machine learning applications, large-scale data processing, and predictive analytics. It is a free and open-source platform that simplifies package management and deployment. The package management system "Conda" manages package versions in the Anaconda distribution. Over 12 million users use the Anaconda distribution, which includes more than 1400 popular data-science packages suitable for Windows, Linux, and MacOS. Anaconda Navigator, a Conda package and virtual environment manager, comes bundled with the Anaconda distribution, which eliminates the need to learn how to install each library independently. Open-source packages can be installed individually from the Anaconda repository using the conda install command or the pip install command that is included with Anaconda. Pip packages provide many of the features of conda packages, and in most cases, they can work together.

Anaconda Navigator: Anaconda Navigator is a graphical user interface (GUI) for desktops that comes bundled with the Anaconda distribution. It allows users to launch applications, manage conda packages, environments, and channels without having to use command-line commands. Navigator can search for packages on Anaconda Cloud or in a local Anaconda Repository, install them in an environment, run the packages, and update them. It is compatible with Windows, macOS, and Linux operating systems.

Conda: Conda is a package manager and environment management system that is open source, cross-platform, and language-agnostic. It is designed to install, run, and update packages and their dependencies. Originally created for Python programs, Conda is also capable of packaging and distributing software for other languages, such as R, and even multi-languages. The Conda package and environment manager is included in all versions of Anaconda, Miniconda, and Anaconda Repository.

Jupyter Notebook: The Jupyter Notebook is a freely available web-based tool that enables users to produce and distribute documents featuring interactive code, equations, visualizations, and explanatory text. Its diverse applications range from data cleaning and analysis to numerical simulation, statistical modelling, machine learning, data visualization, and beyond.

Kernel: In the context of Notebook documents, a kernel serves as a computational engine responsible for executing the code therein. The ipython kernel, which processes Python code, is the focus of this guide, but official kernels exist for a variety of other programming languages. Whenever a Notebook document is opened, the corresponding kernel is automatically launched. Subsequently, when the notebook is executed - either one cell at a time or all at once through the menu option Cell -> Run All - the kernel carries out the necessary computations and generates the associated output.

V. System Implementation

The report aims to prepare the dataset for analysis by loading and cleaning the data, ensuring that the document outlines the cleaning steps taken and justifies the cleaning decisions made. The collected dataset is divided into a Training set and Test set, usually in a 7:3 ratio, to predict the given data. Machine learning algorithms such as Random Forest, logistic regression, Decision tree, K-Nearest Neighbour (KNN), and Support Vector Classifier (SVC) are applied on the Training set, and based on their accuracy, Test set prediction is performed. The data may contain missing values, outliers, or variables that need to be converted, which can impact the algorithm's efficiency. Therefore, preprocessing is required to improve the accuracy of the model. In predicting air quality problems, the decision tree algorithm is an effective prediction model for classification problems. Raw data

cannot be used directly; hence it needs preprocessing before selecting an appropriate algorithm with a model. The model is trained and tested to ensure correct prediction with minimum errors, and the tuned model is continually improved to enhance accuracy.The system architecture is shown in below Figure.1.

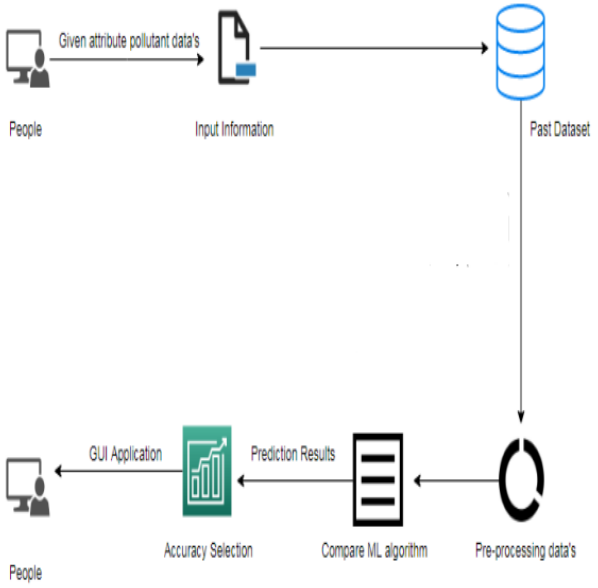

Figure.1 System Architecture

Data Validation Process: In machine learning, validation techniques are used to estimate the error rate of a ML model, which can be considered as close to the true error rate of the dataset. If the dataset is large enough and representative of the population, validation techniques may not be necessary. However, in real-world scenarios, working with samples of data that may not be representative of the population requires validation techniques to find missing values, duplicate values, and the data type (float or integer). The sample of data is used to provide an unbiased evaluation of a model fit on the training dataset while tuning hyperparameters. As the skill on the validation dataset is incorporated into the model configuration, the evaluation becomes more biased. The validation set is frequently used to evaluate a given model and machine learning engineers use this data to fine-tune the model hyperparameters. Data collection, analysis, and addressing content, quality, and structure can be time-consuming. During the data identification process, it is essential to understand the data and its properties to choose which algorithm to use to build the model. For instance, regression algorithms can be used to analyse time series data, while classification algorithms can be used to analyse discrete data.

Exploration Data Analysis of Visualization: Data visualization is a crucial skill in both applied statistics and machine learning. While statistics focuses on quantitative descriptions and estimations of data, data visualization offers a set of tools to gain a qualitative understanding. This can be beneficial when exploring and familiarizing oneself with a dataset, and can aid in identifying patterns, corrupt data, outliers, and more. With some domain knowledge, data visualizations can effectively express and demonstrate key relationships in plots and charts that are more impactful and understandable for stakeholders than measures of association or significance. Data visualization and exploratory data analysis are entire fields in themselves, and it is recommended to delve deeper into some of the books mentioned at the end.

Training the Dataset: The first line of code imports the iris dataset from the sklearn module, which contains information about various varieties in the form of a table. Additionally, the algorithm and train_test_split class are imported from the sklearn and numpy modules for use in the program. The load_data() method is encapsulated in the data_dataset variable, and the dataset is further divided into training and test data using the train_test_split method. The variable names prefixed with X denote the feature values, while those prefixed with y denote the target values. The train_test_split method divides the dataset into training and test data randomly, typically in a ratio of 67:33 or 70:30. After dividing the dataset, any desired algorithm is encapsulated. The next line fits the training data into this algorithm to train the computer using the provided data. Once this training process is complete, the program moves to the next step.

Testing the Dataset: The dimensions of new features are stored in a NumPy array called 'n'. The objective is to predict the species of these features using the predict method, which takes the 'n' array as input and produces the predicted target value as output. After prediction, the resulting target value is found to be 0. To determine the accuracy of the predictions, the program calculates the test score, which is the ratio of the number of correct predictions to the total number of predictions made. Additionally, the program calculates the accuracy score using a method that compares the actual values of the test set with the predicted values.

VI. Conclusion and Future Scope

The analytical process began with data cleaning and processing, addressing missing values, exploratory analysis, and finally model building and evaluation. The achieved accuracy on the public test set is high, indicating a good accuracy score for predicting air quality based on the given attributes. This application could assist the India Meteorological Department in predicting future air quality conditions, allowing them to take necessary actions. Air quality assessment process can be automated in real-time by displaying the prediction results in a web or desktop application. To optimize this process, Artificial Intelligence could be implemented.

REFERENCES

[1] Bashir Shaban, K., Kadri, A and Rezk, E. 2016. Urban Air Pollution Monitoring System with Forecasting Models. IEEE Sensors Journal.16 (8): 2598-2606.

[2] Kalapanidas, E and Avouris, N. 2017. Applying machine learning techniques in air quality prediction in Proc. ACAI, 99.

[3] Luke Curtis, William Rea, Patricia Smith-Willis. 2006. Adverse health effects of outdoor air pollutants.

[4] Niharika, V.M and Rao, P.S. 2014. A survey on air quality forecasting techniques. International Journal of Computer Science and Information Technologies. 5(1):103-107.

[5] Xi, X., *et al*. 2015. A comprehensive evaluation of air pollution prediction improvement by a machine learning method. IEEE International Conference on Service Operations and Logistics, And Informatics (SOLI), Yasmine Hammamet, Tunisia, 176-181.

[6] Yang, B and Chen, Q. 2017. PM2.5 Concentration Estimation Based on Image Quality Assessment. 2017 4th IAPR Asian Conference on Pattern Recognition (ACPR), Nanjing, China. 676-681.

[7] Yang, R., Yan, F and Zhao, N. 2017. Urban air quality based on Bayesian network. IEEE 9th International Conference on Communication Software and Networks (ICCSN), Guangzhou, China 2017. 1003-1006.