A Study on Air Quality Index Parameters for Covid-19 Period Using Machine Learning and Deep Learning Approaches

S. Ravishankar[1], Dr. P. Rajesh[2]

[1]Research Scholar, Department of Computer and Information Science, Annamalai University, Annamalainagar, Tamil Nadu, India
Email: thiru.ravishankar@gmail.com

[2]Assistant Professor, PG Department of Computer Science, Government Arts College, Chidambaram, (Deputed from Department of Computer and Information Science, Annamalai University, Annamalainagar) Tamil Nadu, India
Email: rajeshdatamining@gmail.com

Abstract:
The COVID-19 pandemic exerted a notable influence on pollution levels globally, yielding both favorable and adverse effects stemming from shifts in human conduct and economic activities. Machine learning and deep learning finds utility across diverse fields, and its pivotal role lies in automating processes and facilitating data-driven predictions and decision-making.This paper considers, Air Quality Index parameters (AQI) for Covid-19 period dataset like City, Date, PM2.5, PM10, NO, NO2, NOx, NH3, CO, SO2, O3, Benzene, Toluene, Xylene, AQI, AQI_Bucket. The machine learning and deep learning approaches is used to analysis and predict the dataset usingLogistic, Multilayer Perceptron, SMO, Decision Stump, Hoeffding Tree,J48,LMT, and Transformer deep learning models with Stochastic Gradient Descent approaches. Numerical illustrations are provided to prove the proposed results with test statistics or accuracy parameters.

Keywords: Machine learning, Deep learning, COVID-19 effect on AQI, decision tree, correlation coefficient, and test statistics.

DOINUMBER:10.48047/NQ.2022.20.19.NQ99449 NEUROQUANTOLOGY2022;20(19):4870-4883

1. INTRODUCTION AND LITERATURE REVIEW

Researchers and scientists have delved into multiple facets of this connection to comprehend the environmental repercussions of the global reaction to the virus.The proper technical documentation that explains any fundamental theory, topic survey, or proof of concept using a mathematical model or practical implementation. Data mining involves the examination and analysis of extensive datasets to discern patterns and extract valuable insights. Businesses employ data mining software to gain deeper insights into their clientele, enabling the formulation of more potent marketing strategies, revenue enhancement, and cost reduction. Deep learning, a branch of machine learning, encompasses using artificial neural networks capable of learning from data. The term "deep" denotes its employment of multi-layered neural network architectures, enabling the acquisition of complex data representations with multiple levels of abstraction.

The hybrid EAMA model is specifically tailored for making predictions

based on historical and current data. The study utilizes two datasets, one sourced from the Ministry of Health & Family Welfare of India and the other from World meters, to establish long-term predictions for both India and the global context. Remarkably, the predicted data closely aligns with real-time values. Furthermore, the research extends to state-wise predictions for India and country-wise predictions worldwide, which can be found in the Appendix [1].

The second route pertains to zoonotic infections, exemplified by the Middle East respiratory syndrome coronavirus (MERS-CoV). Lastly, there is a higher case fatality rate associated with severe acute respiratory syndrome coronavirus (SARS-CoV). In this context, machine learning techniques play a crucial role in classifying the three COVID-19 infection stages by employing feature extraction via data retrieval. The study employs TF/IDF for statistical evaluation of text data mining in COVID-19 patient records, enabling classification and prediction of coronavirus categories. This research demonstrates the viability of employing techniques to analyze blood tests and machine learning as an alternative to rRT-PCR for identifying COVID-19-positive patients [2].

The development of supervised machine learning models for COVID-19 infection is a key focus of this research, employing learning algorithms including logistic regression, decision trees, support vector machines, naive Bayes, and artificial neural networks. These models utilize labeled epidemiological datasets for positive and negative COVID-19 cases in Mexico. The research includes a correlation coefficient analysis to assess the strength of relationships between dependent and independent features in the dataset before model development. The study uses 80% of the dataset for model training and reserves the remaining 20% for testing. The results indicate that the decision tree model achieves the highest accuracy at 94.99%, the Support Vector Machine Model exhibits the highest sensitivity at 93.34%, and the Naïve Bayes Model displays the highest specificity at 94.30% [3].

COVID-19 patients can be harnessed and analyzed through advanced machine learning algorithms to better comprehend the viral spread patterns, enhance diagnostic speed and accuracy, formulate effective therapeutic strategies, and potentially identify individuals most susceptible to the virus based on personalized genetic and physiological traits. Notably, advanced machine learning techniques have been rapidly adopted for various applications since the outbreak of COVID-19, including taxonomic classification of COVID-19 genomes, CRISPR-based COVID-19 detection assays, survival prediction for severe COVID-19 cases, and the discovery of potential drug candidates against the virus [4].

The dataset comprises weekly confirmed cases and weekly cumulative confirmed cases spanning 35 weeks. The data distribution is scrutinized using the most up-to-date COVID-19 weekly case data, and statistical parameters are derived accordingly. The research also introduces a time series prediction model employing machine learning, featuring linear regression, multi-layer perceptrons, random forests, and support vector machines (SVM). The performance of these methods is compared using metrics like RMSE, APE, and MAPE, with SVM exhibiting the most promising trend. Projections suggest that the global pandemic will reach its peak at the end of January 2021, with approximately 80 million cumulative infections [5].

Data mining serves as a valuable tool for exploring extensive pre-existing databases to extract previously undiscovered, valuable insights. In one particular application, weather data is used, with attributes denoting conditions conducive to playing golf. Seven classification algorithms, including J48, Random Tree (RT), Decision Stump (DS), Logistic Model Tree (LMT), Hoeffding Tree (HT), Reduce Error Pruning (REP), and Random Forest (RF), are employed to assess accuracy. Among these algorithms, Random Tree emerges as the top performer with an accuracy of 85.714% [6].

COVID-19 in Iran using data from the Google Trends website. Linear regression and long short-term memory (LSTM) models are employed to estimate the number of positive COVID-19 cases. The models undergo evaluation via 10-fold cross-validation, with the root mean square error (RMSE) serving as the performance metric [7].

The progress of COVID-19 vaccination worldwide using machine learning classification algorithms. The research determines which algorithm is most suitable for a given dataset. Real-world data is analyzed using Weka, with four classification algorithms, namely Decision Tree, K-nearest neighbors, Random Tree, and Naive Bayes, scrutinized for accuracy and performance period. The results reveal that the Decision Tree outperforms other algorithms in terms of both time and accuracy [8].

The research presented here seeks to predict the recovery rate of COVID-19 patients in South Asian countries based on healthy dietary patterns using data mining and various machine learning algorithms, including Random Forest, Support Vector Machine (SVM), and K-Nearest Neighbor (KNN) [9].

Data mining is a powerful tool for uncovering hidden information that aids in making predictions through stochastic sensing. This paper proposes an effective assessment of groundwater levels, rainfall, population, food grain data, and enterprise data using stochastic modeling and data mining approaches. The novel data assimilation analysis is introduced for groundwater level prediction, demonstrating the robustness of this approach [10] and [11].

The research uses a dataset related to chronic diseases, with attributes representing topics, questions, data values, low confidence limits, and high confidence limits for specific locations. The study employs five classification algorithms, and the M5P decision tree approach is identified as the most effective algorithm for building models in comparison to other decision tree methods [12].The tree's leaf nodes represent the final decision or the predicted outcome. The "CART" (Classification and Regression Trees) algorithm is the most used algorithm for building decision trees [13].

2. BACKGROUNDS AND METHODOLOGIES

2.1 Logistic Regression

Logistic Regression is a statistical method used for binary classification, which means it's used to predict the probability of an observation belonging to one of two classes (usually labeled as 0 and 1). It's a type of regression analysis that's particularly suited for categorical outcome variables. The formula for logistic regression involves the logistic function (also known as the sigmoid function) to transform the linear combination of input features into a value between 0 and 1, representing the predicted probability of the positive class. The formula is as follows:

$$P\left(Y=\frac{1}{X}\right) = \frac{1}{1+e^{-(\beta_0+\beta_1 X_1+\beta_2 X_2+\cdots+\beta_n X_n)}} \quad ...(1)$$

P(Y=1/X) is the probability that the dependent variable Y is the binary outcome equal to 1 given the input features $X_1 + X_2 + \cdots + X_n$. e is the base of the natural logarithm. $\beta_0 + \beta_1 + \cdots + \beta_n$are the coefficients that need to be estimated from the training data. $X_1 + X_2 + \cdots + X_n$. are the input features.

2.2 Multilayer Perception

A Multilayer Perceptron (MLP) is an artificial neural network consisting of multiple layers of interconnected nodes or neurons. It's a fundamental architecture in deep learning and is used for various tasks, including classification, regression, and more complex tasks like image recognition and natural language processing. The architecture of an MLP typically includes three types of layers:

i. Input Layer
ii. Hidden Layers
iii. Output Layer

2.3 SMO

SMO stands for "Sequential Minimal Optimization," an algorithm used for training support vector machines (SVMs), machine learning models commonly used for classification and regression tasks. The SMO algorithm is particularly well-suited for solving the quadratic programming optimization problem that arises during the training of SVMs.

Step 1. Initialization
Step 2. Selection of Two Lagrange Multipliers
Step 3. Optimize the Pair of Lagrange Multipliers
Step 4. Update the Model
Step 5. Convergence Checking
Step 6. Repeat

2.4 Decision Stump

A Decision Stump is a simple machine learning model that serves as a weak learner, often used in ensemble learning methods like boosting. It's a basic model that makes decisions based on a single feature (input) and a threshold value. Despite its simplicity, when combined with other decision stumps or more complex models, decision stumps can contribute to building stronger predictive models. Here's how a Decision Stump works:

Step 1. Input Feature
Step 2. Threshold
Step 3. Prediction
Step 4. Decision Rule

2.5Hoeffding Tree

A Hoeffding Tree, also known as VFDT (Very Fast Decision Tree) or Incremental Decision Tree, is a machine learning algorithm designed for online, incremental learning on streaming data. It's beneficial when you have large volumes of data that are continuously arriving, and you want to update your model in real-time without retraining the entire dataset. Here's a simplified overview of how the Hoeffding Tree algorithm works:

Step 1. Initialization
Step 2. Data Arrival
Step 3. Splitting Nodes
Step 4. Leaf Node Prediction
Step 5. Adaptation

2.6 J48

J48, also known as C4.5, is a popular decision tree algorithm used for classification tasks in machine learning and data mining. It was developed by Ross Quinlan and is an extension of the earlier ID3 (Iterative Dichotomiser 3) algorithm. J48 is widely used due to its effectiveness, ease of use, and ability to handle both categorical and numerical attributes. Here are the key features and steps of the J48 algorithm:

Step 1. Attribute Selection
Step 2. Splitting Nodes
Step 3. Recursion
Step 4. Pruning
Step 5. Handling Missing Values
Step 6. Post-Pruning
Step 7. Leaf Node Prediction

2.7 LMT

LMT (Logistic Model Trees) is a machine learning algorithm that combines decision trees with logistic regression to create a hybrid model for classification tasks. It aims to harness the strengths of both decision trees and logistic regression, mitigating their individual weaknesses. LMT was introduced as an alternative to traditional decision trees and has shown promise in improving predictive performance and interpretability. Here's how the LMT algorithm works:

Step 1. **Decision Tree Generation**
Step 2. **Leaf Node Transformation**
Step 3. **Predictions**

2.8 Transformer Deep Learning Models

Transformers have gained significant attention due to models like BERT, GPT (Generative Pre-trained Transformer), and their variations. These models excel in natural language understanding and generation tasks. The Transformer model has indeed revolutionized the field of natural language processing. Introduced by Vaswani et al. in the paper "Attention is All You Need," the Transformer architecture offers an alternative to traditional sequence modeling methods such as recurrent neural networks (RNNs) or convolutional neural networks (CNNs). Here's a breakdown of the components represented in the diagram:

Input Sequence (Tokenized): This is the original input sequence, which is typically tokenized into numerical representations before feeding it into the model.

Step 1. Input Embedding: The tokenized input sequence gets converted into high-dimensional vectors, capturing the semantic meaning of each token.

Step 2. Positional Encoding: Since the Transformer lacks inherent sequence information, positional encodings are added to the embeddings to indicate the position of tokens in the sequence.

Step 3. Transformer Encoder: The Transformer architecture involves a stack of encoder layers. Each layer contains self-attention mechanisms and feed-forward neural networks. The encoder processes the input sequence to create representations of the sequence.

Step 4. Transformer Decoder: In tasks such as machine translation, the Transformer uses both an encoder and a decoder. The decoder generates the output sequence based on the information encoded by the encoder. It also includes self-attention layers and feed-forward networks.

Step 5. Output Layer: This layer takes the output from the decoder and provides the final output sequence.

2.9 Stochastic Gradient Descent (SGD)

Stochastic Gradient Descent (SGD) is a fundamental optimization algorithm widely used in training machine learning models, including neural networks in deep learning. It's a variant of the standard Gradient Descent algorithm and is particularly beneficial in dealing with large datasets. Gradient Descent is an iterative optimization algorithm used to minimize the loss function of a model by updating its parameters. It operates by adjusting the model's weights in the direction that reduces the loss.

In SGD, rather than using the entire dataset to compute the gradient (as in standard Gradient Descent), it randomly selects a small subset (mini-batch) of data samples at each iteration to calculate the gradient of the loss function. It updates the model parameters more frequently but with noisier updates, as each mini-batch might not represent the entire dataset accurately. Steps of SGD:

Step 1. Initialization: Set initial values for the model parameters (weights and biases).

Step 2. Iteration: For each iteration (epoch):

Step 3. Randomly sample a mini-batch from the dataset.

Step 4. Calculate the loss on this mini-batch.

Step 5. Compute the gradient of the loss function with respect to the model parameters using the mini-batch data.

Step 6. Update the model parameters by taking a step in the direction opposite to the gradient to minimize the loss.

Step 7. Repeat the process until convergence or a specified number of epochs.

2.10Accuracy Metrics

The predictive model's error rate can be evaluated by applying several accuracy metrics in machine learning and statistics. The basic concept of accuracy evaluation in regression analysis is comparing the original target with the predicted one and using metrics like Kappa statistics, MAE, MSE, and RMSE to explain the errors and predictive ability of the model [14]. The Kappa, R-squared, MSE, MAE, and RMSE are metrics used to evaluate the prediction error rates and model performance in analysis and predictions [15] and [16].

The Kappa statistic, also called Cohen's Kappa or simply Kappa, is a statistical metric utilized to assess the level of agreement between two or more raters or classifiers when assigning categorical

ratings or labels to items. It goes beyond considering agreement by chance alone. The Kappa statistic is represented on a scale from -1 to 1. A Kappa value of -1 signifies perfect disagreement between the raters or classifiers. A Kappa value of 0 indicates agreement that is no better than chance. A Kappa value of 1 implies excellent agreement between the raters or classifiers. The calculation of Kappa employs the formula:

$$\text{Kappa} = \frac{P_o - P_e}{1 - P_e} \quad \text{...(2)}$$

$$P_o = \frac{\text{Number of items with agreement}}{\text{Total number of items}}$$

$$P_e = \sum \frac{\text{Total count in row} \times \text{Total count in column}}{\text{Total number of items}}$$

Where, $P_o$denotes the observed agreement, i.e., the proportion of items on which raters or classifiers agree. $P_e$represents the expected agreement, i.e., the agreement expected by chance.

MAE (Mean absolute error) represents the difference between the original and predicted values extracted by averaging the absolute difference over the data set.

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}| \quad \text{... (4)}$$

RMSE (Root Mean Squared Error) is the error rate by the square root of MSE.

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y})^2} \quad \text{... (5)}$$

Relative Absolute Error (RAE) is a metric used in statistics and data analysis to measure the accuracy of a forecasting or predictive model's predictions. It is particularly useful when dealing with numerical data, such as in regression analysis or time series forecasting.

$$\text{RAE} = \frac{\sum|y_i - \hat{y}_i|}{\sum|y_i - \bar{y}|} \quad \text{… (6)}$$

Root Relative Squared Error (RRSE) is another metric used in statistics and data analysis to evaluate the accuracy of predictive models, especially in the context of regression analysis or time series forecasting.

$$\text{RRSE} = \sqrt{\frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}} \quad \text{… (7)}$$

Where Y_i represents the individual observed (actual) values, $\hat{Y}_i$ represents the corresponding individual predicted values, $\bar{Y}$ represents the mean (average) of the observed values and Σ represents the summation symbol, indicating that you should sum the absolute differences for all data points.

The TP Rate, or True Positive Rate, is a frequently employed performance metric in binary classification scenarios, common in fields like machine learning and statistics. It's also recognized as Sensitivity, Recall, or Hit Rate. This metric gauges the proportion of actual positive instances that a model or classifier correctly identifies as positive. Its calculation involves the formula:

Similarly, the FP Rate, or False Positive Rate, is another vital performance measure in binary classification tasks within fields such as machine learning and statistics. It's also referred to as the False Alarm Rate. This metric assesses the proportion of actual negative instances that a model or classifier inaccurately categorizes as positive. The calculation employs the formula:

On the other hand, Precision is a performance metric employed in binary classification scenarios, providing an assessment of a model's ability to make accurate positive predictions. It's particularly valuable when the cost of false positives is substantial or when there's a need for high reliability in positive predictions.

Recall, also known as Sensitivity or True Positive Rate, is a performance metric designed to quantify a model's capability to accurately identify all positive instances within a dataset.

Lastly, the F-Measure, often referred to as the F1 Score, is a performance metric tailored for binary classification tasks, aiming to strike a balance between precision and recall. It comes in handy when dealing with imbalanced class distributions and when you want to evaluate a model's performance while considering both false positives and false negatives.

TP Rate = (True Positives) / (True Positives + False Negatives) ... (8)

FP Rate = (False Positives) / (False Positives + True Negatives) ... (9)

Precision = (True Positives) / (True Positives + False Positives) ... (10)

Recall = (True Positives) / (True Positives + False Negatives) ... (11)

Where True Positives (TP) are the instances that are actually positive and are correctly predicted as positive by the model. False Negatives (FN) are the instances that are actually positive but are incorrectly predicted as negative by the model. False Positives (FP) are the instances that are actually negative but are incorrectly predicted as positive by the model. True Negatives (TN) are the instances that are actually negative and are correctly predicted as negative by the model.

F1 Score = 2 * (Precision * Recall) / (Precision + Recall) ... (12)

Where Precision is the ratio of true positives to all instances predicted as positive. Recall is the ratio of true positives to all actual positive instances.

MCC, short for Matthews Correlation Coefficient, serves as a widely adopted performance metric, especially valuable for assessing the effectiveness of binary classification models, especially in scenarios with imbalanced datasets. It offers a holistic assessment by considering true positives, true negatives, false positives, and false negatives. The computation of the Matthews Correlation Coefficient follows this formula:

MCC = (TP * TN - FP * FN) / √((TP + FP) * (TP + FN) * (TN + FP) * (TN + FN)) ... (13)

The MCC value ranges from -1 to +1. A higher MCC indicates a better overall performance of the model. An MCC of +1 represents a perfect prediction, an MCC of 0 suggests that the model's predictions are no better than random, and an MCC of -1 indicates a completely inverse relationship between the prediction and the actual values.

The ROC (Receiver Operating Characteristic) is a visual representation that assesses a binary classification model's performance across different threshold settings. Its primary purpose is to analyze and contrast the trade-offs between a model's True Positive Rate (sensitivity) and its False Positive Rate (1 - specificity). The ROC curve itself materializes by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR) at varying threshold levels, revealing how adeptly a model distinguishes between positive and negative classes across diverse decision thresholds.

The True Positive Rate, also referred to as Sensitivity or Recall, quantifies the ratio of correctly identified positive cases concerning all actual positives:

TPR = TP / (TP + FN). ... (14)

False Positive Rate (FPR), also known as Fall-out, is the ratio of false positives to all actual negatives:

FPR = FP / (FP + TN) ... (15)

An exemplary ROC curve is typically depicted as TPR (y-axis) versus FPR (x-axis) and generally originates from the origin (0, 0), advancing upward. The diagonal line represents a random classifier with no predictive power, while an effective classifier's ROC curve endeavors to maximize its distance from this diagonal, preferably reaching the upper-left corner of the chart.

The Area Under the ROC Curve (AUC-ROC) serves as a prominent summary statistic for the ROC curve. Models with AUC-ROC values close to 1 exhibit superior discrimination ability, while those near 0.5 are no more effective than random chance. This metric facilitates model comparisons, where a higher AUC-ROC signifies an enhanced overall performance.

The PRC (Precision-Recall Curve) is a graphical portrayal of a binary classification model's effectiveness, with a specific emphasis on the balance between precision and recall under varying threshold conditions. This visualization proves especially valuable in scenarios with imbalanced datasets, where one class significantly outweighs the other. To construct a PRC, consider the following steps:

1. Vary the classification threshold of your model to obtain different sets of predictions.
2. For each threshold setting, calculate True Positives (TP), False Positives (FP), False Negatives (FN), and True Negatives (TN).
3. Calculate precision and recall for each threshold setting:
   Precision = TP / (TP + FP)
   Recall (Sensitivity) = TP / (TP + FN)
4. Plot precision on the y-axis and recall on the x-axis for each threshold setting. Connect the data points to form the PRC.

An ideal classifier's Precision-Recall Curve (PRC) would commence at the lower-left point (0, 0) and ascend to the upper-right point (1, 1). In real-world applications, the PRC is frequently

employed alongside the AUC-PRC (Area Under the Precision-Recall Curve) to derive a unified performance metric encapsulating the model's overall effectiveness across various thresholds. A higher AUC-PRC signifies superior performance.Equation 3 to 15 are used to find the model accuracy, which is used to find the model performance and error.

Numerical Illustrations

The corresponding dataset was collected from the open souse Kaggle data repository. The COVID-19 effect on pollutiondatasetincludes16 parameters whichhavedifferent categories of data likeCity, Date, PM2.5, PM10, NO, NO2, NOx, NH3, CO, SO2,O3, Benzene,Toluene, Xylene,AQI, AQI_Bucket[17]. Adetailed description of the parameters is mentioned in the following Table 1.

Table 1. COVID-19 effect on pollutionsampledataset

Date	PM2.5	PM10	NO	NO2	NOx	NH3	CO	SO2	O3	Benzene	Toluene	Xylene	AQI	AQI_Bucket
11/5/2018 8:00	10	29.5	4.82	8.92	8.7	4.5	0.49	6.15	31.85	0.05	0.08	0.1	53	Satisfactory
11/5/2018 11:00	22	50	4.18	12.45	10.03	5.9	0.61	7.15	52.93	0.05	0	0.2	53	Satisfactory
17-05-2018 10:00	11.25	34	4.15	5.27	6.15	5.25	2.08	5.12	33.32	0.2	0	0.1	101	Moderate
27-08-2018 19:00	14.75	48.25	3.03	13.92	9.85	16	0.75	8.38	30.87	0.1	0.18	0.1	76	Satisfactory
28-08-2018 20:00	27.75	91.25	20.7	57.9	47.63	12.1	1.83	9.05	15.68	0.1	0.18	0.1	92	Satisfactory
19-11-2017 00:00	154.96	208.34	16.28	43.11	48.27	40.89	1.1	9.09	44.94	2.16	5.32	0.08	307	Very Poor
19-11-2017 01:00	130.84	176.36	15.1	39.62	44.18	37.48	1.05	7.94	41.99	1.92	4.16	0.08	306	Very Poor
19-11-2017 02:00	123.4	174.3	13.53	38.41	42.14	36.39	1.13	10.66	40.12	1.79	4.3	0.08	304	Very Poor
19-11-2017 03:00	113.89	150.43	13.11	37.49	38.68	36.75	1.14	10.12	42.39	1.72	4.16	0.08	303	Very Poor
19-11-2017 04:00	104.87	138.8	13.21	34.41	37.04	35.07	1.24	8.84	40.57	1.41	3.3	0.08	302	Very Poor

Table 2: Machine Learning Models with Correctly and Incorrectly Classified Instances

Function and Trees	Correctly Classified Instances	Incorrectly Classified Instances
Logistic	81546.0000	432.0000
Multilayer Perceptron	80537.0000	1441.0000
SMO	79087.0000	2891.0000

Decision Stump	53783.0000	28195.0000
Hoeffding Tree	81976.0000	2.0000
J48	81978.0000	0.0000
LMT	81961.0000	17.0000
TMSGD	81978.0000	0.0000

Table 3: Machine Learning Models with Correctly Classified Instances (%) and Incorrectly Classified Instances (%)

Function and Trees	Correctly Classified Instances (%)	Incorrectly Classified Instances (%)
Logistic	99.4730	0.5270
Multilayer Perceptron	98.2422	1.7578
SMO	96.4734	3.5266
Decision Stump	65.6066	34.3934
Hoeffding Tree	99.9976	0.0024
J48	100.0000	0.0000
LMT	99.9793	0.0207
TMSGD	100.0000	0.0000

Table 4: Machine Learning Models with Kappa statistic

Function and Trees	Kappa statistic
Logistic	0.9929
Multilayer Perceptron	0.9764
SMO	0.9525
Decision Stump	0.4758
Hoeffding Tree	1.0000
J48	1.0000
LMT	0.9997
TMSGD	1.000

Table 5: Machine Learning Models with Mean Absolute Error and Root Mean Squared Error

Function and Trees	MAE	RMSE
Logistic	0.0036	0.0402
Multilayer Perceptron	0.0092	0.0624
SMO	0.2230	0.3116
Decision Stump	0.1665	0.2886
Hoeffding Tree	0.0004	0.0033
J48	0.0000	0.0000
LMT	0.0032	0.0281
TMSGD	0.0000	0.0000

Table 6: Machine Learning Models with Relative Absolute Error (%) and Root Relative Squared Error (%)

Function and Trees	RAE (%)	RRSE (%)
Logistic	1.4401	11.4144
Multilayer Perceptron	3.7094	17.7126

SMO	89.8215	88.4329
Decision Stump	67.0746	81.8995
Hoeffding Tree	0.1469	0.9245
J48	0.0000	0.0000
LMT	1.2715	7.9672
TMSGD	0.0000	0.0000

Table7: Machine Learning Models with Time Taken to Build Model (Seconds)

Function and Trees	Timetaken(seconds)
Logistic	43.0200
Multilayer Perceptron	343.3700
SMO	8.0300
Decision Stump	2.5000
Hoeffding Tree	7.9400
J48	1.9100
LMT	528.8400
TMSGD	100.5452

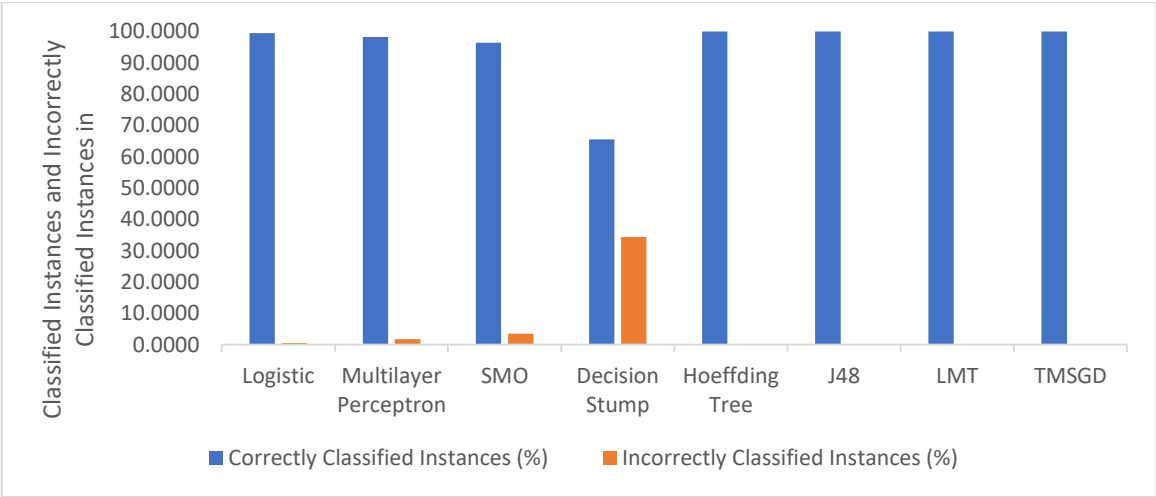

Fig. 1. Machine Learning Models with Correctly and Incorrectly Classified Instances (%)

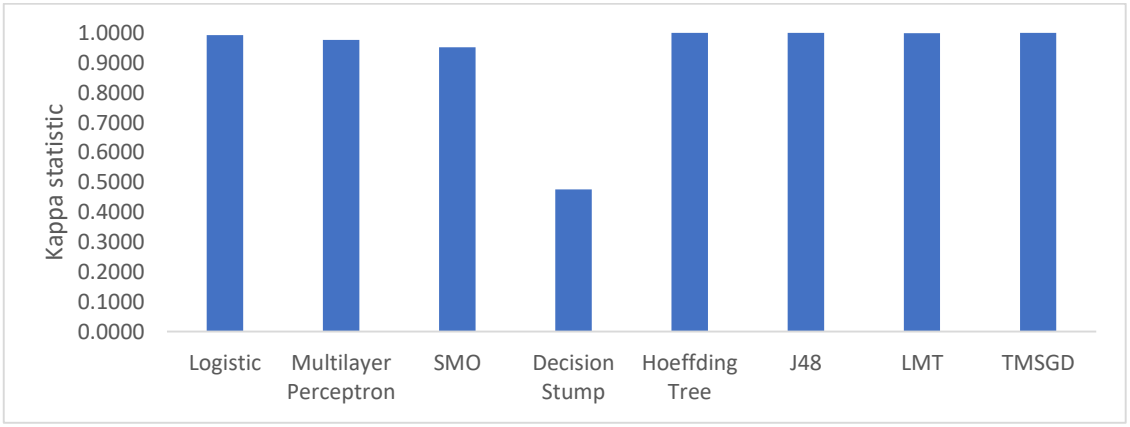

Fig. 2. Machine Learning Models with Kappa statistic

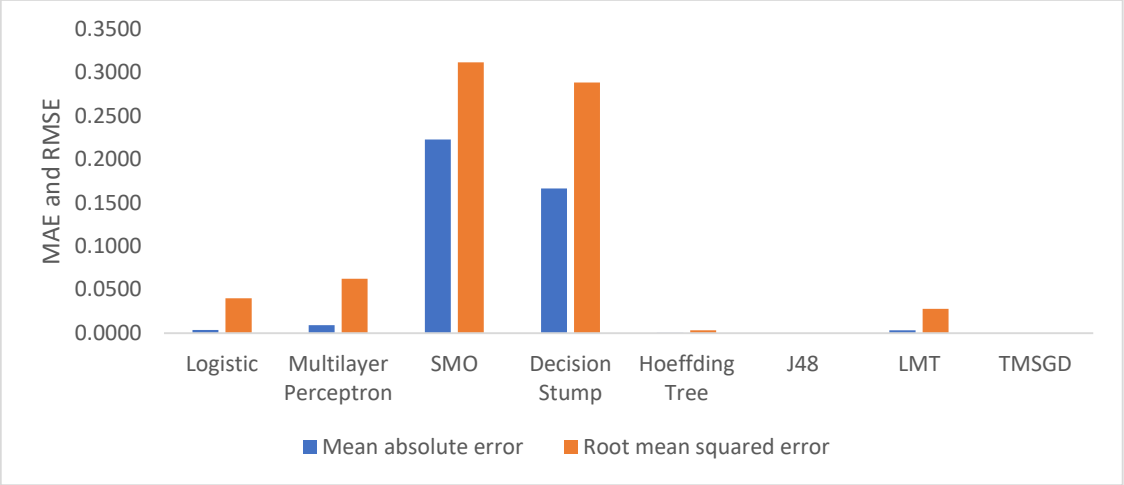

Fig.3.Machine Learning Models with MAE and RMSE

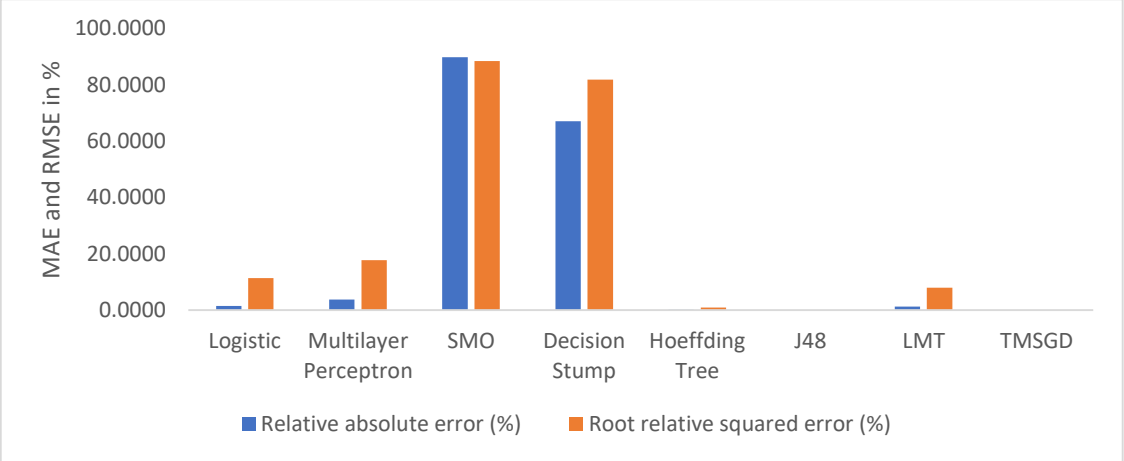

Fig. 4. Machine Learning Models with RAE (%) and RRSE (%)

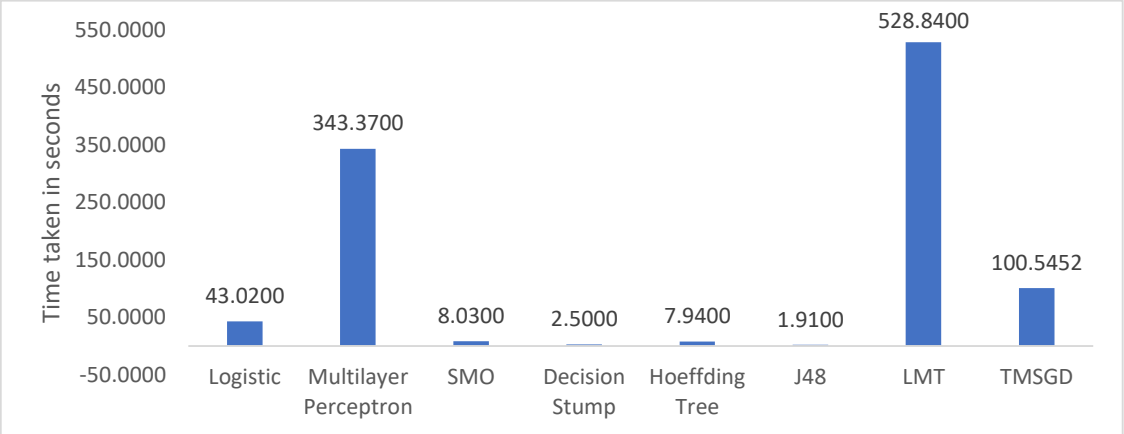

Fig. 5. Machine Learning Models and its Time Taken to Build the Model (Seconds)

Table 8:ML Approaches with Performance

ML Approaches	Logistic	Multilayer Perceptron	SMO	Decision Stump	Hoeffding Tree	J48	LMT	TMSGD
TP Rate	0.9907	0.9777	0.9497	0.3333	1.0000	1.0000	1.0000	1.0000
FP Rate	0.0010	0.0040	0.0080	0.0897	0.0000	0.0000	0.0000	0.0000
Precision	0.9898	0.9840	0.0080	0.2295	1.0000	1.0000	0.9998	1.0000
Recall	0.9907	0.9777	0.9497	0.3333	1.0000	1.0000	1.0000	1.0000
F-Measure	0.9900	0.9807	0.9572	0.2702	1.0000	1.0000	0.9998	1.0000
MCC	0.9892	0.9767	0.9495	0.2363	1.0000	1.0000	0.9998	1.0000
ROC Area	1.0000	0.9998	0.9913	0.7760	1.0000	1.0000	1.0000	1.0000
PRC Area	0.9993	0.9988	0.9392	0.3333	1.0000	1.0000	1.0000	1.0000

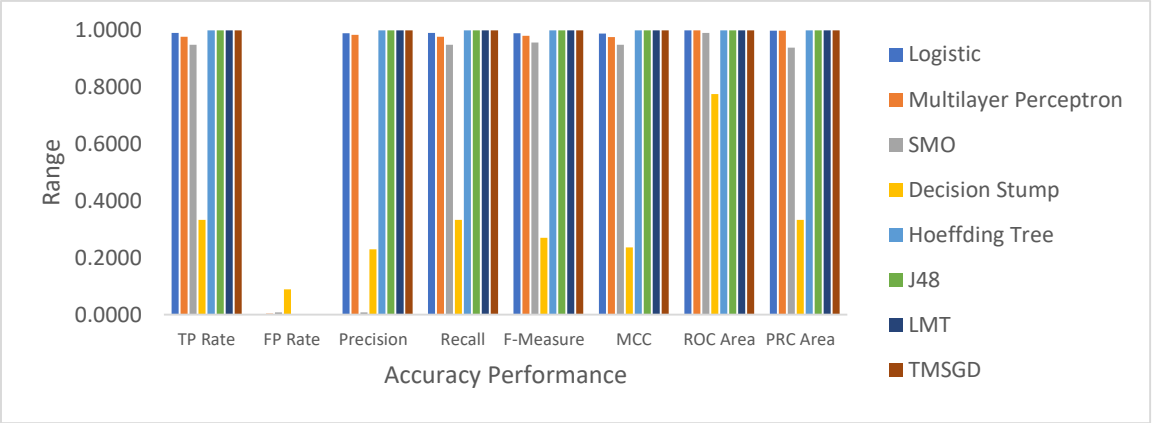

Fig. 6. ML Approaches with Performance

3. RESULT AND DISCUSSION

Table 1 explains 16 parameters, including different data categories like Soil micronutrients, macronutrients, and various agriculture product Information. Based on the dataset, it is evident that seven additional machine learning approaches, namely Logistic, Multilayer Perceptron, SMO, Decision Stump, Hoeffding Tree, J48, and LMT, are used to find the hidden patterns, and which is the best or influencing parameter to decide future predictions. Related results and numerical illustrations are shown between Table 1 to Table 8 and Figure 1 to Figure 6.

Table 2 explains the quality of given data, which means correctly and incorrectly classified instances. Similarly, table 3 indicates correctly and incorrectly classified instances in percentage. In this case, a maximum of the ML and DL approaches return better correctly classified instances. The related results are shown in Figure 1.

They are based on Equation 2, table 4, and Figure 2, which is used to find the kappa statistics to find a measure of inter-rater agreement or reliability for categorical data, often used to assess the consistency of ratings or classifications between two or more observers. In this case, the decision stumpreturns a minimum kappa value, and remaining ML Approaches produce nearly 0.95, which means better kappa values. The deep learning approach TMSGD returns better results of 1.0000. The related visualization is shown in Figure 2.

The MAE is used to find model errors using Equations 3. Seven machine-learning algorithms will be used in this case. All the seven ML approaches return the best error performance, nearly 0. The RMSE (root mean square error) measures the difference between predicted and actual values using Equation 4. In this case, all the ML approaches return the best error performance, nearly 0. In these

cases, some of ML approaches and Deep learning approaches return 0% error. The related numerical illustration is shown in Table 5 and Figure 3.

Relative Absolute Error (RAE) measures accuracy using equation 6 to compare the difference between predicted and actual values in percentage. This research considers seven ML classification algorithms. Based on the RAE error performance, the SMO and Decision Stump returns the maximum error rate for solving this problem. The remaining five ML approaches yield the best performance and minimum error. Similar error approaches are reflected in RRSE for using ML and DL approaches. Similar numerical illustrations are shown in Table 6 and Figure 4.

Time taken is one of the significant tasks in machine-learning approaches. Based on Table 7 and Figure 5, for Logistic, multi-layer perceptions, and LMT taking maximum time to solve this type of problem. Multilayer Perceptron, Hoeffding Tree and J48 take minimum time to build these models. Subsequently, the logistic regression approach also takes minimum time to make the model for the next level of algorithms. Similar approaches are reflected in the mentioned visualization.

Based on table 8 and figure 6, decision stump return lowest TP rate. Remaining ML and ML approaches return very strong TP Rate nearly 0. The FP rate means false positive, in this case, except decision stump remaining returns better results. In precision return best accuracy performance except SMO and decision stump. Recall test statistics return best performance for using all ML approaches except decision stump. Similarly remaining accuracy parameters returns best performance for using ML and DL approaches except decision stump. The related results are shown in table 8 and figure 6.

4. CONCLUSION AND FURTHER RESEARCH

In this research clearly state that, climate change analysis and prediction indicate all the parameters is very useful for solve climate change problems using various ML approaches and its accuracy parameters. Additionally, propose possible enhancements or future steps, such as exploring additional data sources, investigating better algorithms or hyperparameters, and fine-tuning the model to enhance its performance. This research benefits government and climate change research. Further research to learn how to reduce climate change related issues effectively using deep learning.

5. REFERENCE

[1]. Mohan, S., Abugabah, A., Kumar Singh, S., Kashif Bashir, A. and Sanzogni, L., 2022. An approach to forecast impact of Covid-19 using supervised machine learning model. Software: Practice and Experience, 52(4), pp.824-840.

[2]. Ramanathan, S. and Ramasundaram, M., 2021. Accurate computation: COVID-19 rRT-PCR positive test dataset using stages classification through textual big data mining with machine learning. The Journal of supercomputing, 77(7), pp.7074-7088.

[3]. Muhammad, L.J., Algehyne, E.A., Usman, S.S., Ahmad, A., Chakraborty, C. and Mohammed, I.A., 2021. Supervised machine learning models for prediction of COVID-19 infection using epidemiology dataset. SN computer science, 2(1), pp.1-13.

[4]. Alimadadi, A., Aryal, S., Manandhar, I., Munroe, P.B., Joe, B. and Cheng, X., 2020. Artificial intelligence and machine learning to fight COVID-19. Physiological genomics, 52(4), pp.200-202.

[5]. Ballı, S., 2021. Data analysis of Covid-19 pandemic and short-term cumulative case forecasting using machine learning time series methods. Chaos, Solitons & Fractals, 142, p.110512.

[6]. Rajesh, P. and Karthikeyan, M., 2017. A comparative study of data mining algorithms for decision tree approaches using the Weka tool. Advances in Natural and Applied Sciences, 11(9), pp.230-243.

[7]. AyyoubZadeh, S.M., Ayyoubzadeh, S.M., Zahedi, H., Ahmadi, M. and Kalhori, S.R.N., 2020. Predicting COVID-19 incidence through analysis

of google trends data in Iran: data mining and deep learning pilot study. JMIR public health and surveillance, 6(2), p.e18828.

[8]. Abdul Kareem, N.M., Abdulazeez, A.M., Zeebaree, D.Q. and Hasan, D.A., 2021. COVID-19 world vaccination progress using machine learning classification algorithms. Qubahan Academic Journal, 1(2), pp.100-105.

[9]. Hossen, M.S. and Karmoker, D., 2020, December. Predicting the Probability of Covid-19 Recovered in South Asian Countries Based on Healthy Diet Pattern Using a Machine Learning Approach. In 2020 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI) (pp. 1-6). IEEE.

[10]. Rajesh, P., Karthikeyan, M. and Arulpavai, R., 2019, December. Data mining approaches to predict the factors that affect the groundwater level using a stochastic model. In AIP Conference Proceedings (Vol. 2177, No. 1). AIP Publishing.

[11]. Rajesh, P. and Karthikeyan, M., 2019. Data mining approaches to predict the factors that affect agriculture growth using stochastic models. International Journal of Computer Sciences and Engineering, 7(4), pp.18-23.

[12]. Rajesh, P., Karthikeyan, M., Santhosh Kumar, B. and Mohamed Parvees, M.Y., 2019. Comparative study of decision tree approaches in data mining using chronic disease indicators (CDI) data. Journal of Computational and Theoretical Nanoscience, 16(4), pp.1472-1477.

[13]. Kohavi, R., &Sahami, M. (1996). Error-based pruning of decision trees. In International Conference on Machine Learning (pp. 278-286).

[14]. Akusok, A. (2020). What is Mean Absolute Error (MAE)? Retrieved from https://machinelearningmastery.com/mean-absolute-error-mae-for-machine-learning/

[15]. S. M. Hosseini, S. M. Hosseini, and M. R. Mehrabian, “Root mean square error (RMSE): A comprehensive review,” International Journal of Applied Mathematics and Statistics, vol. 59, no. 1, pp. 42–49, 2019.

[16]. Chi, W. (2020). Relative Absolute Error (RAE) – Definition and Examples. Medium. https://medium.com/@wchi/relative-absolute-error-rae-definition-and-examples-e37a24c1b566

[17]. https://www.kaggle.com/code/parulpandey/breathe-india-covid-19-effect-on-pollution