Optimized Neural Network Framework for Air Quality Index Prediction Using Adaptive Error Minimization and Regression Performance Enhancement

Dr Asha N[1], Dr M P Indra Gandhi[2],

[1] *Associate Professor, Department of Computer Science, Nrupathunga University (formerly Government Science College), Bangalore, India*

ORCID: 0000-0002-2615-5890

[2] *Associate Professor, Department of Computer Science, Mother Teresa Women's University, Kodaikanal, India*

email: ashan.gsc@gmail.com

Abstract: The climate, sustainable development, and human health are all greatly impacted by air pollution, which has become one of the most important environmental issues**.** Precise forecasting of the Air Quality Index (AQI) facilitates prompt actions, aids in environmental management, and helps legislators put into practice efficient pollution control measures. Traditional machine learning models often exhibit limitations in capturing the complex nonlinear relationships among atmospheric pollutants, leading to increased prediction errors and reduced regression performance. This research suggests a Artificial Neural Network-Based Error Optimization Algorithm (NNEOA) for precise AQI prediction with reduced error rate and improved regression fit in order to overcome the difficulties air pollution forcasting.

Keywords— Air Quality Index (AQI); Neural Network-Based Error Optimization Algorithm (NNEOA); Artificial Neural Network; Error Optimization; Regression Performance; Air Pollution Forecasting.

I. INTRODUCTION

One of the biggest environmental issues affecting ecosystems, the climate, and human health globally is air pollution. The Air Quality Index (AQI) is a standardised measure used to indicate the level of air pollution and its potential impact on human health. Accurate prediction of AQI enables governments, environmental agencies, and the public to take timely preventive measures, thereby reducing the adverse effects of poor air quality. The intricate and nonlinear interactions between different air contaminants and meteorological parameters are frequently difficult for traditional statistical approaches to grasp. Artificial Neural Networks (ANNs), inspired by the structure and functioning of the human brain, have emerged as powerful machine learning models capable of learning intricate patterns from large datasets. They are ideal for predicting air quality because they can handle nonlinear interactions. This project proposes an **Optimised Neural Network Framework for Air Quality Index Prediction Using Adaptive Error Minimisation and Regression Performance Enhancement**. The proposed framework focuses on improving prediction accuracy by minimising training errors through adaptive optimisation techniques and enhancing regression performance during the learning process. By continuously adjusting network parameters such as weights and biases, the model achieves faster convergence and better generalization, resulting in more reliable AQI forecasts.The optimized neural network is trained using historical air quality and meteorological data to identify hidden relationships among various pollutants. Performance is evaluated using regression metrics and error analysis to ensure high prediction accuracy. The proposed approach provides an efficient and intelligent solution for real-time AQI forecasting, supporting environmental monitoring, public health protection, and informed decision-making.

II. RELATED WORK

A computer and mathematical model created to mimic the operation of biological neurons is called an Artificial Neural Network (ANN), sometimes referred to as a Neural Network (NN) [1]. A biological neuron is a specialized cell in the human brain that receives, processes, and transmits information through electrical and chemical signals. A computational model that mimics the composition and operation of these biological neurons is called an Artificial Neural Network (ANN). It consists of a set of algorithms designed to simulate the brain's neural network, enabling machines to learn, recognize patterns, and make intelligent decisions [2]. The interconnected neurons in an Artificial Neural Network are represented using a set of algorithms that enable the system to learn from data and make decisions in a manner similar to the human brain. Although ANN is inspired by the principles of neuroscience, it does not precisely replicate the structure or functioning of the human brain [4,5]. Data mining, picture identification, employee access control systems, banking, language translation, and pattern recognition are just a few of the industries that have made extensive use of artificial neural networks (ANNs). ANNs are now among the most successful and efficient machine learning methods for resolving difficult real-world issues because of their capacity to handle massive amounts of complex data [6]. Artificial Neural Networks (ANNs) are available in various architectures, each designed to address specific types of problems. During the learning process, the network adjusts the weights of the connections between neurons to improve its performance. Based on the knowledge gained during training, the network can perform tasks such as pattern recognition and decision-making. ANN uses parametric algorithms during the training phase to optimize its performance and achieve accurate results[7,8]. Although it is unable to completely recreate the intricacy of actual brain cells, a computer neural network is intended to mimic the fundamental structure and operation of the human brain. It is built using fundamental processing units called neurons, which are interconnected to form a network. These neurons learn by identifying relationships and patterns within the dataset, enabling the network to analyze data effectively and work collaboratively to solve specific problems [9,10,11]. The structure of the network comprises three key parameters

1. Topology of the network
2. Learning rate
3. Activation function

The network Topology is classified into

- The Feed Forward Propagation of Artificial Neural Network
- The Backward Propagation of Artificial Neural Network

III. METHODOLOGY

The following goals were accomplished by predicting the Air Quality Index using a Multilayered Feed-forward and Backward Propagation neural network.

- The predicted output is concentrated to enhance the Prediction accuracy.
- And also, to minimize the error rate.
- The new framework is proposed based on the Neural Network model with an improvement on the momentum value to achieve greater prediction of the AQI index.

A. *Sample Collection of Pollution Data set*

The Pollution dataset from the KSPCB, Bangalore. The dataset included 746 samples collected from 17 different monitoring stations. Pollutant concentrations such as PM10, PM2.5, SO2, NO2, CO, and NH3 were used as inputs to the ANN for each station. Table I displays the neural network's input, hidden, and output layers used to predict AQI.

Table I: Layers of Neural Network

Layer	Structure	Description
Input Layer	Pollutant Concentration	PM10,PM2.5,SO2,NO2,CO,NH3 and its corresponding subindex
Hidden Layer	No of Hidden layers	40
Output Layer	Activation Function	Sigmoid
	Error Functions	Sum of Squares

B. *Neural Network Setup*

The Neural network was prepared initially before training the network with the Pollution data set.

Choosing Input layer:

The input layer denoted as I consist of 14 nodes. The following Attributes of the input layer are PM_{10}, PM_{10} sub-index, $PM_{2.5}$, $PM_{2.5}$ subindex, SO_2, SO_2 Subindex, NO, NO_2 subindex, CO, CO subindex, O_3, O_3 subindex, NH_3, NH_3 subindex.

Choosing the Hidden layer

The symbol for the hidden layer is H. Due to the enormous dimensions or features of the data, some hidden layers are first set at Hi=5 and increased in increments of five in order to obtain the best possible answer.

Choosing nodes in the Hidden layer

The size of the hidden neurons should fall between that of the output layer 1 and the input layer 14. Occasionally, there may be more hidden layers than the ideal scenario, and the number of nodes in hidden levels may also rise in later layers.

Choosing Output Layer:

The predicted AQI is the output layer, represented by the letter O. Recognizing the following training of our neural network is also crucial. Our data samples are divided by a decision boundary in order to do recognition.90% to establish a network

- 5% test data
- 5% is used for Validation

Initializing the parameter:

For error optimization, the weights and bias values are updated by the training process. It finds the best combination to create a network with good generalization by minimizing a combination of squared errors and weights.

The selection of activation is a crucial factor in neural network architecture design, depending on the goal and available data. The input neuron was given a smooth linear activation function, and the hidden neurons were given a non-linear log sigmoid function. Additionally, a constant bias was introduced to the output and hidden layers [13]. The following are the parameters initialized for Neural Network,

1. Input data contains 750 instances in excel format

2. The number of Neurons in Hidden Layers is calculated using the formula

Number of Neurons in each Layer= (no_of pollutants + Predicted AQI)$^{0.5}$ + (1 to hidden layer size)

……*Eq* (1)

Example 1: If the number of Pollutants concentration is 14 and one output layer, then substituting the values for *Eq* 1 we get, = $(14+1)^{0.5}+5$ =9 Neurons.

3. Weights:

The strength of the relationship between units is represented by weight. Neuron 1 has more impact over neuron 2 if the weight from node 1 to node 2 is larger. The I-H-O topology is used to determine the number of weights in our training model.

I – is the input Layer

H- is the Hidden Layer

O- is the output layer

As,
num of Weight Elements=(I+1) *H+(H+1) *O

……*Eq*(2)

Example 2: We know for the first 5 Layers

I=14, H=5, 0=1

The Number of Weight elements required is = (14+1) *5+(5+1) *1=81 weights will be created, and weights are Initialized between -1 to 1 using random functions.

4. Bias:

Processing of each neuron is done by the following formula,

Output= sum(weights*inputs) +bias ……*Eq* (3)

Bias is like the intercept added in a linear equation

Y=m x + c ……*Eq* (4)

Where,

Y is the Predicted AQI,

m is the weight from every pollutant neuron to every neuron of hidden layers,

c is the bias.

Bias is added to the sum of weights coming from every neuron to get nearer to the predicted value.

5. Epoch

The neural network is first trained with all of the training pollutant data for one cycle using epoch=1. We use every piece of data exactly once during each period. An epoch is defined as the sum of a forward pass and a backward pass.

Forward Pass in Neural Network to Predict AQI

Forward pass in Neural Network to predict AQI is carried out in four steps.

1. Input Layer Computation

Synapses link the input neurons to the hidden neurons.

• Let wij represent the weight of the arc connecting the jth hidden layer to the ith input neuron.
• The hidden neuron receives all contaminants as input, which is the weighted total of the input neurons' outputs.

$I_{HP}= X_{01}*W_{1p} + X_{02}*W_{2p} + X_{03}*W_{3p} +\dots\dots\dots X_{Ii}*W_{ip}$ ……*Eq* (5)

Example 3:

$I_{H15}= X_{151}*W_{151} + X_{152}*W_{152} + X_{153}*W_{153} +\dots\dots\dots X_{Ii}*W_{Ip}$

……*Eq*(6)

Where,

I_{H15} = Input to 15th neuron Hidden Layer

X_{151} = Output of neuron 1 to 15th neuron of hidden layer

W_{151} =Weights of the arc between the input of the 1st neuron to the 15th neuron.

2.Hidden Layer Computation

The output of every neuron is applied with the transfer function [75] as

$O_{HP}= \frac{1}{1+e\ -OHP}$ ......*Eq* (7)

Example 4:

$O_{H15}\ = \frac{1}{1+e\ -OH15}$

Where,

$O_{HP}= \frac{1}{1+e\ -OH15}$

O_{H15}= Output of the P^{th} Hidden Neuron

There are 15 input neurons and 24 output neurons. P is the input neuron to the output neuron, which is the weighted sum of the hidden neurons' outputs determined by equation (7).

3.Output Layer Computation

Once more, the output from the qth neuron is represented using a transfer function as the input to the output neuron.

$$O_{Oq}= \frac{1}{1+e\ -OOQ} \quad ......\text{Eq (8)}$$

Example 5:

$O_{O\ 24}= \frac{1}{1+e\ -OO24}$

The input neuron is Q, the Output neuron is T is the weighted sum of the output of the hidden neuron calculated using Equation (8). Here the input neuron is 15 and the output neuron is 24. The output of the r^{th} neuron is applied with an activation function using Equation (8).

Calculation of error

Consider the rth neuron, for the target value T,

Calculate the error

E_r=1/2 (Target-0utput)^2 ......Eq (9)

Example 6:

E_{21}=1/2 (Observed AQI-Predicted AQI) ......Eq (10)

Where,

r is the 21st neuron,

T is the Target value,

O is the Output value.

This error function is for one training pattern if we consider the same technique for all training patterns.

Backward Pass in Neural Network to Predict AQI

With backpropagation, we want to minimize the error for each output neuron and the network overall by updating each weight in the network to make the actual output closer to the intended output..

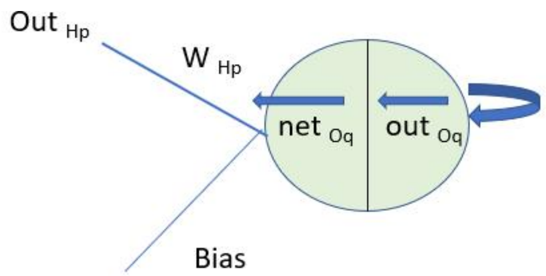

Figure I. Backward flow of signals to update weights

a. Computations in Output Layer

From the Equation 9

1. To know how much the change of weight of WHp Affects the total error to Out0q. The chain rule is applied

$\frac{\partial \epsilon_{total}}{\partial \epsilon_{HP}} = \frac{\partial \epsilon_{total}}{\partial Out_{oq}} * \frac{\partial Out_{oq}}{\partial net_{oq}} * \frac{\partial net_{oq}}{\partial W_{HP}}$ ......*Eq* (11)

a. $E_{total} = \frac{1}{2}$ (Target$_{oq}$-Out$_{oq}$)^2 ......*Eq* (12)

The partial derivative of above Equation 12 w.r.t Out_{oq} is

$\frac{\partial \epsilon_{total}}{\partial out_{oq}} =-\ (\text{Target} - \text{Outoq})$

b. $Out_{oq} = \frac{1}{1+e^{netoq}}$ ......*Eq* (13)

The partial derivative of the above Equation 13 w.r.t Netoq

$$\frac{\partial outoq}{\partial netoq} = Outoq(1 - Outoq)$$

c. Net_{oq}= W_{HP} + O_{HP} + Bias ......*Eq* (14)

The partial derivative of the above Equation 14 w.r.t to W_{HP} is

$\frac{\partial netoq}{\partial wHP} = O_{HP}$

d. Substituting Eq 12, 13, and 14 in Eq 11 we get

$$\frac{\delta\epsilon_{total}}{\delta w_{HP}} = -(Targetoq - Outoq) * Outoq(1 - Outoq) * OutHP \quad ......Eq\ (15)$$

The new weights calculated are subtracted from the previous weight to decrease the error.

e. The learning rate is applied to the new weight

$$W_{HP} = W_{HP} - \eta * \frac{\delta\epsilon_{total}}{\delta w_{HP}} \quad ......Eq\ (16)$$

Where,

η - is defined as the learning rate

The above Eq (16) is used to calculate the new weights and the same process is repeated.

NN based-Error Optimization Algorithm to predict AQI with Minimum error rate and best Regression fit

ALGORITHM 1: NN based-Forward and Backward pass to predict AQI

Step 1: Start

Step 2: Choose an ***Optimization Algorithm***

Step 3: **Forward Propagate** an input

Begin

1. To Calculate the output of the hidden layer node using the sum of squares plus the bias for every input node connected to every other node in the hidden layer.

2. Calculate the **Activation function** using **Sigmoid.**

3. Utilising the sum of squares plus the bias for each hidden layer node linked to the output node in the output layer, determine the output of the output layer node.

4. Repeat these steps from steps 1 to 3 for nodes in the hidden layer to the output node in the output layer.

End

Step 4: Calculate the error and mean square error of Predicted AQI and Observed AQI

Step 5: If the predicted output is not acceptable

Step 6: **Backward Propagate**

Begin

1. Calculating the partial Derivative of Error

2. Apply the Chain Rule to the output node's input with respect to weight in the output layer.

3. Calculate the new weight using the learning rate

4. Repeat steps 1 to 3 for all nodes in the backward pass to change the new value of weights.

End

Step 7: Stop

NN based-Error Optimization Algorithm to predict AQI

ALGORITHM 2: NN based-Error Optimization Algorithm to predict AQI

Step 1:_ Initialize weights and Bias

Step 2: Initialize epoch=1 and Hiddenlayersize=5

Step 3: For Layer 1 to Hiddenlayersize

Begin

Step 3.1: Call Forward Propagate (carry out Forward Pass as Step 3 of Algorithm 4.3)

Step 3.2: Calculate the error and mean square error of the Predicted AQI and Observed AQI

Step 3.4: If MAE>=0 and MAE<=1.1632

or Hiddenlayersize<=40

go to step 4

else

Step 3.5: Call Backward Propagate(carry out Backward Pass as Step 5 of Algorithm 4.3)

Step 3.6: Epoch=Epoch+1

Step 3.7: Hiddenlayersize=Hiddenlayersize+5

End for

Step 4: Stop

IV. *RESULTS AND DISCUSSION*

The following Figure 1.2 shows that there are 14 input nodes in the Input layer with the Pollutant Concentration PM_{10}, $PM_{2.5}$, SO_2, NO_2, CO, Ozone, NH_3, and its corresponding subindex. The output layer has only one node with a target variable as Observed AQI. The Hidden layers are increased from 1 in steps of 5 to arrive at the best predictions of AQI.

The input to each hidden layer is the weighted sum of the inputs from each node of the input layer to every node in the hidden layer plus the bias value (Ex: PM_{10} *-0.7064 + b) to adjust the Predicted AQI with Observed AQI. The output layer is the output of the activation function of each node multiplied by its weight values plus the bias (Ex: O_{HP1} *-1.7040 +b). Here O_{HP1} output of the pth hidden neuron. The architecture is a generalized structure to build by considering the weights and bias values generated during the execution of the 40th layer using an optimization algorithm. One such instance is shown in the architecture that for the Observed AQI of 162, the Predicted AQI is 161.7157.

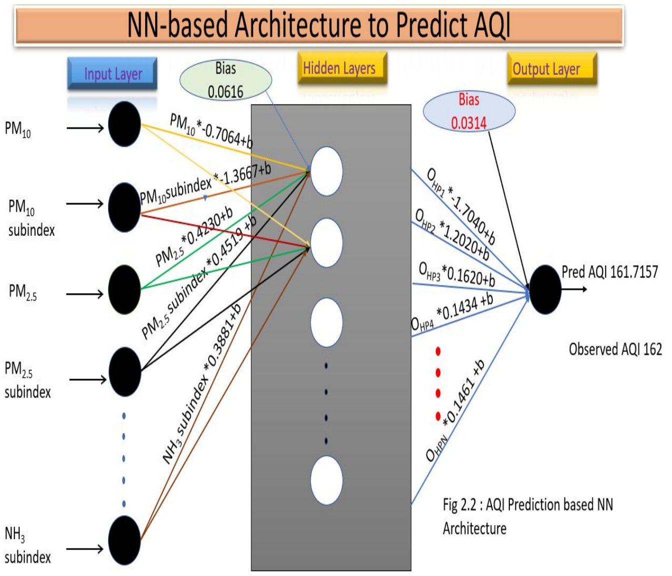

Figure 1.2: Proposed Neural Network-based Architecture to Predict AQI

Also, Table 1.2 shows the results of 50 layers in terms of iterations, time, epoch, and, regression fit. The table depicts that as the number of layers increases in steps of 5, the number of iterations increases with an increase in time in seconds. The neural network is first trained using all of the training pollutant data for one cycle at epoch=12. Every piece of data is used exactly once in each epoch. An epoch is defined as the sum of a forward pass and a backward pass. A Regression fit determines the correlation between the pollutant concentration observed in a data set and quantifies whether the correlation is significant statistically. So as the number of layers increases the regression fit is also getting better.

Table 1.2: Table representing Layers, Iterations, Time, Epochs and Regression Fit

Layers	Iterations	Time in secs	Epoch	Regression Fit
1	28	0	12	0.98521
5	513	1	512	0.98867
10	538	3	295	0.99583
15	793	6	792	0.99482
20	639	4	638	0.98627
25	523	9	522	0.98909
30	1000	15	1000	0.98914
35	1000	20	719	0.99155
40	1000	24	1000	0.99569
45	1000	28	625	0.99581
50	1000	34	820	0.99594

Table 1.3 shows the behavior of the Observed AQI of 4 samples out of 746 instances at hidden layers from 1-40, Predicted AQI, and the Error. The Error is the error difference between the Observed AQI and the Predicted AQI.

Table 1.3: Observed AQI and Predicted AQI at each layer

Observed AQI	Hidden Layer Size	Predicted AQI	Error
162	1	157.645	-4.355
	5	158.429	3.5708
	10	161.010	0.9899
	15	160.799	1.2001
	20	160.461	1.5388
	25	161.887	0.1123
	30	161.546	0.4536
	35	162.639	-0.6395
	40	161.715	0.2843
133	1	128.296	-4.704
	5	129.618	3.3815
	10	134.539	-1.539
	15	133.841	-0.8414
	20	133.492	-0.4921

	25	133.013	0.0131
	30	132.960	0.0396
	35	133.639	-0.6390
	40	133.016	-0.0164
	1	109.582	-4.5827
	5	103.028	1.9717
	10	101.403	3.5960
	15	102.691	2.3099
105	20	105.691	-0.6913
	25	97.558	7.4411
	30	106.232	-1.2000
	35	102.917	2.0824
	40	102.734	2.2652
	1	127.997	11.0029
	5	139.965	-0.9655
	10	141.840	-2.8409
	15	138.771	0.2286
139	20	138.99	0.0310
	25	139.487	-0.4878
	30	135.649	-3.351
	35	138.490	0.5100
	40	138.772	0.2280

The graphical depiction of how the number of hidden layers reduces the mean absolute error of observed AQI and predicted AQI is shown in Figure 1.3. For the air pollution data set, the neural network is also trained for 45 and 50 hidden layers; nevertheless, the mean absolute error stays constant and stabilizes after training for 40 hidden layers.

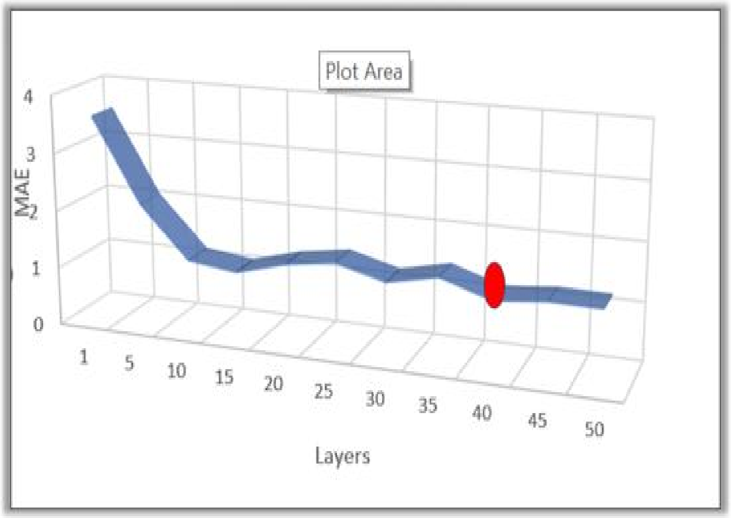

Figure 1.3: Function Fitting Neural Network at layer 15

Conclusion:

The experimental results show that training the model with several hidden-layer configurations for AQI prediction can successfully reduce the neural network's error rate. The implementation of a multilayer neural network, employing deep learning techniques during the training and testing phases on the air pollution dataset, significantly improved prediction performance. With a **Mean Absolute Error** (MAE) of 1.1632 and an **accuracy** (R^2) of 0.99569, the suggested model demonstrated outstanding agreement between the observed and predicted AQI values. Furthermore, the evaluation of the optimized multilayer neural network led to the development of a robust **Neural Network Architecture for AQI Prediction**, providing an effective framework for accurate air quality forecasting.

VII. ACKNOWLEDGMENTS

*"This article is derived from the author's **Dr Asha N and Dr M P Indra Gandhi** PhD thesis titled 'Predictive Analysis, Forecasting, and Control of Air Pollution Using Data Analytic Techniques' submitted to Mother Teresa Women's University and available on Shodhganga.",**but not published elsewhere.***

VIII. REFERENCES

[1] Y. Singh and A. S. Chauhan, "Neural Networks in Data Mining," Journal of Theoretical and Applied Information Technology, Jul. 2010, doi:10.1007/978-0-387-09823-4_21.

[2] J. M. Nazzal, I. M. El-Emary, and S. A. Najim, "Multilayer Perceptron Neural Network (MLPs) for Analyzing the

Properties of Jordan Oil Shale," World Applied Sciences Journal, vol. 5, no. 5, pp. 546–552, 2008.

[3] A. Sharma, A. Mitra, S. Sharma, and S. Roy, "Estimation of Air Quality Index from Seasonal Trends Using Deep Neural Network," in Artificial Neural Networks and Machine Learning (ICANN 2018), Lecture Notes in Computer Science, vol. 11141, Springer, 2018, pp. 511–521.

[4] M. Chalfen and J. Kamińska, "Identification of Parameters and Verification of an Urban Traffic Flow Model: A Case Study in Wrocław," ITM Web of Conferences, vol. 23, 2018.

[5] H. Maleki, A. Sorooshian, and G. Goudarzi, "Air Pollution Prediction by Using an Artificial Neural Network Model," Clean Technologies and Environmental Policy, vol. 21, pp. 1341–1352, 2019, doi:10.1007/s10098-019-01709-w.

[6] M. Z. Rehman and N. M. Nawi, "Improving the Accuracy of Gradient Descent Back Propagation Algorithm (GDAM) on Classification Problems," International Journal on New Computer Architectures and Their Applications, vol. 1, no. 4, pp. 838–847, 2011.

[7] H. Yan, Y. Jiang, J. Zheng, C. Peng, and Q. Li, "A Multilayer Perceptron-Based Medical Decision Support System for Heart Disease Diagnosis," Expert Systems with Applications, vol. 30, pp. 272–281, 2006.

[8] S. M. Cabaneros, J. K. Calautit, and B. R. Hughes, "A Review of Artificial Neural Network Models for Ambient Air Pollution Prediction," Environmental Modelling & Software, vol. 119, pp. 285–304, 2019, doi:10.1016/j.envsoft.2019.06.014.

[9] A. Heydari, M. Majidi Nezhad, D. Astiaso Garcia, F. Keynia, and L. De Santoli, "Air Pollution Forecasting Application Based on Deep Learning Model and Optimization Algorithm," Clean Technologies and Environmental Policy, 2021.

[10] A. Perera, H. M. D. Azamathulla, and U. Rathnayake, "Comparison of Different Artificial Neural Network Training Algorithms to Predict the Atmospheric Temperature in Tabuk, Saudi Arabia," Mausam, vol. 71, no. 2, pp. 233–244, Apr. 2020.

[11] S. Christy and V. Khanaa, "Data Mining in the Prediction of Impacts of Ambient Air Quality Data Analysis in Urban and Industrial Area," International Journal on Recent and Innovation Trends in Computing and Communication, vol. 4, no. 2, pp. 153–157, 2016.

[12] S. Rong and Z. Bao-wen, "The Research of Regression Model in Machine Learning Field," MATEC Web of Conferences, vol. 176, Art. no. 01033, 2018, doi:10.1051/matecconf/201817601033.

[14] L. Chen and T. Y. Pai, "Comparisons of GM (1,1) and BPNN for Predicting Hourly Particulate Matter in Dali Area of Taichung City, Taiwan," Atmospheric Pollution Research, vol. 6, pp. 572–580, 2015.