International Journal of Future Generation Communication and Networking
Vol. 13, No. 4, (2020), pp. 925–945

# An Iot Based Air Quality Monitoring And Air Pollutant Level Prediction System Using Machine Learning Approach – Dlmnn

D.Saravanan[1], K.Santhosh Kumar[2], R.Sathya[3], U.Palani[4]

*Associate Professor[1], Department of CSE, IFET College of Engineering, Villupuram,*
*Assistant Professor [2], Department of IT, Annamalai University, Chidambaram,*
*Assistant Professor [3], Department of CSE, Annamalai University, Chidambaram,*
*Professor[4], Department of ECE, IFET College of Engineering, Villupuram.*
*saranmds@gmail.com[1], santhosh09539@gmail,com[2], sathya_vai@yahoo.com[3], palani.u@gmail.com[4]*

***Abstract:***

*Air pollution (AP) affects not only the environment but the body organs and respiratory system as well. Thus, a scientific air quality (AQ) monitoring in addition to the early system is needed for estimating the level of AP and also to precisely envisage the pollutant concentrations. The AQ prediction has considerably enhanced with the deployment of the sensor centered on the Internet of things (IoT). The entire existing techniques for AP predictions are costly and mostly have low accuracy. Thus, by employing Deep Learning Modified Neural Network (DLMNN) centered on IoT, a technique for monitoring and prediction is proposed here. The sensor values (SV) as of IoT devices are considered as input. After that, an attack detection mechanism is done on the received sensor data by utilizing the Gaussian kernel membership function (GMF) centered Adaptive Neuro-Fuzzy Inference System (ANFIS) termed as (GMF-ANFIS) to check whether the data is an attacked one or not. The system performs training to envisage the AP level. For training, the data as of the Air Quality Index (AQI) –UCI dataset is employed. In training, around '3' operations are performed. Initially, data is preprocessed. Secondly, utilizing the Modified Dragonfly Optimization Algorithm (MDOA), the pertinent features as of the preprocessed data are chosen. Thirdly, the chosen features are classified into '6' sorts of data by utilizing DLMNN. After training, testing is performed on the non-attacked SV. Finally, the results of testing are visualized. The experiment is implemented to analyze the proposed method's performance. The outcomes exhibited that the DLMNN performs better when weighed against other existing algorithms for AQ prediction.*

***Key words:*** *Air quality monitoring and prediction system, Internet of things (IoT), IoT based air quality prediction, Deep Learning Modified Neural Network (DLMNN), Adaptive Neuro-Fuzzy Inference System (ANFIS), Gravitation Search Optimization Algorithm (GSOA), Modified Dragonfly Optimization Algorithm (MDOA).*

## 1. INTRODUCTION

One such primal cause that is detrimental to a human's welfare is AP. The healths of 'seven-billion' humans are jeopardized because of AP as stated by the 'World Health Organization' (WHO). It is the preponderance and root cause for several health problems suchlike asthma, skin infections, heart problems, throat and eye disorders, bronchitis, lung cancer, and diseases concerned with the respiratory system [1, 2, and 3]. Typically, the aforementioned problem is caused by pollutants suchlike Sulfur dioxide (SO2), Nitrogen dioxide (NO2), Particulate material (PM10 and PM2.5), Lead (Pb), Ozone (O3) and Carbon monoxide (CO) [4, 5].

Besides the terrible consequence of humans health, it also influences the environment and engenders catastrophe suchlike acid rain, smog, ozone layer deterioration, along with global warming. Hence superintendence and control must be ensured over AP [6]. AP and the absence of control to monitor and examine AQ evince the constraints stumbled by the environment and the technology [7, 8]. To vanquish this effect, the industries emphasize on discovering a product that is dexterous and permits the enhancement



ISSN: 2233-7857 IJFGCN

of AQ further in network sites, supplying reference values where customary monitors flop to perform [9]. Customary air automatic monitors have comparatively intricate technology of the equipment, wobbly operation, and economically high. Wobbly performance and greater cost make it impracticable for an extensive installation and can merely be ensconced in key enterprise's primary monitoring areas that elicit the system data's absence to anticipate the overall pollution [10].

The (IoT) Internet of Things is employed to inspect and monitor live AQ over particular surroundings [11]. Myriad devices are possessed by IoT for mutual communication over-assisted interlinked nodal devices. Gadgets that are sensor embedded utilize this internet to gather data and transmit it through wired/wireless gadgets, where data analysis is done to implement requisite operations [12]. The progression of IoT is increasing every day and become renowned by virtue of its several applications over industries, and smart cities [13]. Intruders may hack the sensor data of IoT. Hence it is a prerequisite to safeguard the information and prevent such attacks. Devices employed for attack prevention must possess traits suchlike detection, privacy, internality, and undeniability.

The consolidation of such devices into existence procures enormous data which is annotated periodically. Prediction and treating such data resulted in the concept of Deep Learning (DL) [14]. Linear regression, support vector machine, artificial neural networks (ANN), [15] random forest are the machine learning techniques, which are widely employed to predict AQ. Yet, few techniques prioritize to concentrate on specific pollutant predictions suchlike Ozone or PM2.5. In AP control, Numerical models are crucial. With an eye towards predicting AQ, the government authorities and the Environmental Research Institute employs Mainstream AQ numerical models suchlike WRF-Chem [16], Community Multi-scale AQ Model (CMAQ), CAMx, and NAQPMS. The temporal data acquired from the monitored fumes is analyzed formerly to envisage the prospective values. Supposing the system possesses augmented precision and the attacks envisaged are null amid forecasting, then it is indicated as the best system.

The remaining part is designed as ensued: Section 2 exhibits the work that is related to paper. Section 3 elucidates the proposed methodology. The proposed method's results are disclosed in section 4, and eventually, section 5, renders the conclusion of the entire work.

## 2. LITERATURE REVIEW

Xiang Li *et al* [18] presented the SpatIo-Temporal deep learning (STDL) by to forecast the AQ that examined the interrelationships among spatial with temporal features intrinsically. A stacked auto-encoders model was employed for extracting intrinsic AQ features, in addition, trained on a greedy layer-wise mode. When contrasted with customary time-sequential paradigms, the implied paradigm effectively forecasted the AQ of every location concurrently and depicted the temporal permanence in entire seasons. Furthermore, when the paradigm was compared with other models, it evinced splendid execution.

Zhongshan and Jian [19] developed a system, which monitored the AQ and gave prior portent. In the monitoring of AQ, the primary contaminants were scientifically ascertained and examined by a fuzzy extensive analysis. A hybridization paradigm was merged with several techniques to enhance the monitoring of six primary components in AP. The results evinced that the principal pollutants in Xi'an and Jinan were PM10 as well as PM2.5 accordingly. The AQ of Xi'an was found enhanced to AQ of Jinan. The outcome evinced that the hybrid prototype was superior on grounds of greater stability and accuracy.

Dixian Zhu *et al.* [20] Predicated on the compiled metrological data acquired for previous days, developed a model that computed the hourly AP concentration. A beneficial standardization was done by enforcement of the prediction paradigms of sequential hours be nearer and various conventional regularizations suchlike standard Frobenius norm regularization, nuclear norm regularization, and $l_{2,1}$-norm regularization were contrasted with it. The formulations that diminish the parameters and sequential-hour-paradigms were superior in performance contrasted to the contemporary typical retrogression models and available methodologies.

Jiangshe and Weifu [21] calculated the intensity of pollutants in AP by utilizing a qualified intense learning machine that requisite the data from 2 monitoring locations from eight AQ parameters comprising Sham



ISSN: 2233-7857 IJFGCN

Shui Po along with Tap Mun. The results evinced that the paradigm's performance was found enhanced in quantity together with quality. The technology generated superior prediction outcomes with augmented R 2 and diminished error of root means square.

Xiaozheng Lai *et al.* [22] developed an instantaneous forecasting system premised onIoT together with edge computing that diminished IoT reliance on cloud computing. This algorithm utilizes the Kalman Filter (KF) that enhanced the sensors by twenty-seven percentage, was effectual in forecasting the denseness of AP. Centered upon the KF algorithm; the system attained an instant prediction of the AP concentration like SO2, NO2, and PM2.5 by joining the observations with errors. The optimal outcome was exhibited by the result with a reduced error of root means square further, depicted on the utilization of the algorithm in agriculture

Sandro*et al.* [23] devised a context-aware AP monitoring algorithm that preemptively cautioned the people through mobile when a highly polluted environment is approached. The system consistently discovered the AQ all through the given environment and transmitted the data to the people's mobile which were in turn accountable for frequently comparing a citizen's position in opposition to the regions of poor AQ.

Miles *et al.* [24] identified a decision support system (DSS) that scrutinized various abatement techniques to diminish AP. Initially, the DSS perceived critical AP and retaliated through the execution of full roads termination. It substantiated the feasibility of DSS to lessen AP as of attaining dangerous levels and enlisted the level at which abatement techniques should be implemented.

## 3. PROPOSED METHODOLOGY

Air pollution (AP) is a chief factor in global warming. Due to the increase in population, countless vehicles, industrialization, along with urbanization, the AP level has been augmenting every day. This affects all the human who all are exposed to pollution. Thus, this paper by employing DLMN proposed an IoT centered AP monitoring and prediction system. This technology can detect pollutants on the roads, measures numerous sorts of AP and reports the AP status in smart cities. Initially, the SV as of the IoT devices is given as the input. After that, the GMF-ANFIS classifier is used as the attack detection method to check whether the cloud server has attained the attacked or non-attacked sensor data. If attacked data is received, then that data is saved in a log-file and maintained with a cloud server for the purpose of future data analysis. And, the non-attacked data is additionally processed to find the AP level in the smart cities. This AP monitoring system undergoes training and testing. To analyze the AP level, the data values as of the AQI dataset are utilized in the training phase. Three different operations are performed during training: i) preprocessing, ii) feature selection, iii) classification.



ISSN: 2233-7857 IJFGCN

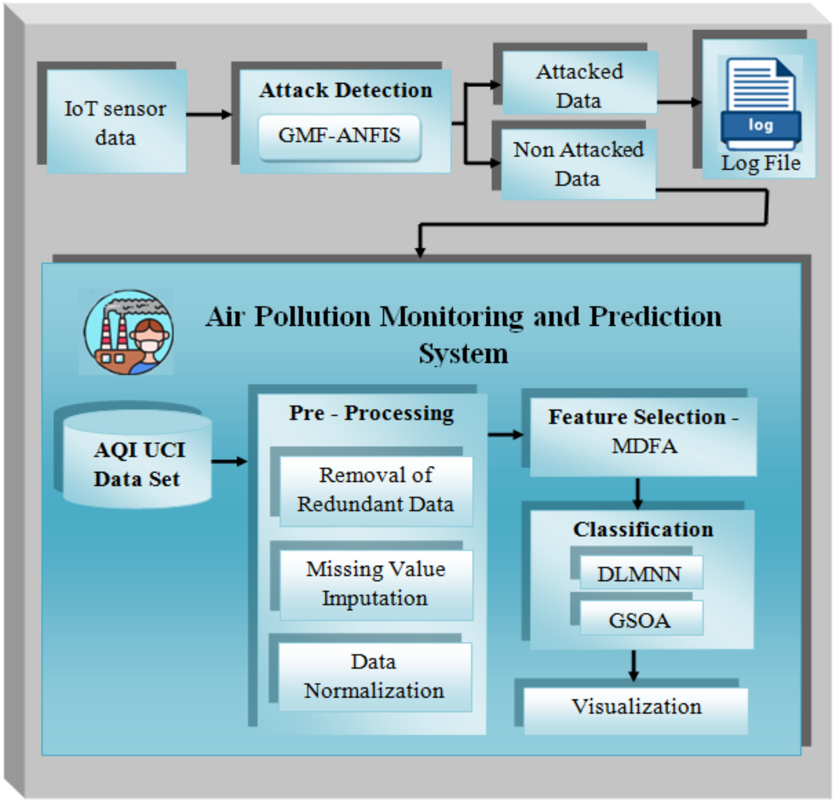



**Figure 1:** Architecture of the proposed method

The pre-processing has 3 stages: i) removal of redundant data, ii) imputation of missing values and iii) data normalization. Then, from the pre-processed data, the important features are chosen by utilizing MDOA. These chosen features are inputted to the DLMNN for the classification of data values into '6' classes of pollutant data as good, moderate, unhealthy for sensitive groups, very unhealthy, unhealthy, and Hazardous. The DLMNN predicts the level of pollutants. Subsequent to the process of training, the non-attacked data is tested by performing the same '3' operations as training. The method tests the non-attacked SV and gives the pollutant level by comparing the training results. The testing classification results are then visualized. The proposed method's architecture is exhibited in figure 1.

### 3.1 IoT Sensor Values

Initially, the SV of disparate IoT sensor devices is gathered as of numerous nodes, which are present in various locations. The sensor will gather several data as of the surrounding, which is indirectly or directly connected to IoT networks. These sensors gather the concentration of pollutants like Carbon monoxide, harmful gases, dust level, together with the meteorological parameters for instance temperature and GPS location. Then, for monitoring the level of AP, the collected values are sent out to the cloud server. But there stands a probability of attack taking place on these data and that data being sent out to the cloud. Thus, it is vital to know about whether the sensor data that is received from the IoT is attacked or not. This brings about the GMF-ANFIS, an attack detection technique that is explained below.

### 3.2 Attack Detection using GMF-ANFIS

Subsequent to attaining the SV, the Black Hole (BH) attack detection stage is performed utilizing GMF-ANFIS. BH is regarded as the most hazardous attack. The sensor nodes that were affected via a BH attack are termed malicious nodes. During a BH attack, the bad nodes promote the incorrect path to the network



ISSN: 2233-7857 IJFGCN

like advertising disparate along with a short route to the destination. These bad nodes' only intention is to create more traffic and drop the caught-up packets rather than forwarding them. BH attack is fundamentally a DOS attack, which deteriorates the network's performance. Thus, GMF-ANFIS is employed for the detection of the BH attacks. The GMF-ANFIS elucidation is provided as follows

ANFIS comes under ANN and it is formed on the basis of the Takagi–Sugeno fuzzy inference system. ANFIS incorporates neural networks with fuzzy logic principles, thus it encompasses the potential of both of them. Its inference system resembles a collection of fuzzy IF-THEN rules that encompass the learning capability for approximating nonlinear functions.

The ANFIS is made of ‘5’ layers. They are

- The fuzzification layer is the initial layer that ascertains the membership functions (MF) for the input values
- The rule-layer is the 2nd one that is accountable for creating the Firing Strengths (FS) for the entire rules.
- The 3rd layer normalizes the calculated FS. It is done by dividing every value for the total FS.
- The normalized values and the resulting parameter are inputted to the 4th layer, which returns the defuzzification values as the output.
- The outcome of the 4th layer is inputted to the last layer, which gives the last output.

The proposed work in the initial layer utilizes GMF in the place of the bell-MF for enhancing the rule generation process’ performance. So, the ANFIS is called GMF based ANFIS (GMF-ANFIS). The two basic rules of GMF-ANFIS are specified in the equations below.

**Rule 1:** If $d_1$ is $p_i$ and $d_2$ is $q_i$ then,

$$R_i = u_i d_i + v_i d_{i+1} + w_i \tag{1}$$

**Rule 2:** If $d_1$ is $p_{i+1}$ and $d_2$ is $q_{i+1}$ then,

$$R_{i+1} = u_{i+1} d_i + v_{i+1} d_{i+1} + w_{i+1} \tag{2}$$

Where $p_i$, $q_i$, $p_{i+1}$ and $q_{i+1}$ specifies the fuzzy sets. $d_i$ and $d_{i+1}$ represents the different SV obtained from IoT sensor devices. $u_i$, $v_i$, $w_i$, $u_{i+1}$, $v_{i+1}$ & $w_{i+1}$ represents the parameter set. The function of ‘5’ layers in GMF-ANFIS are explained as follows

**Layer 1**: Each node encompassed here stands as an adaptive node having a node function

$$K_{1,i} = \psi_i(d_i) \tag{3}$$

Here, $d_i$ implies the input to the node $i$. Every node acclimatizes to a function parameter. The output from each node stands as a degree of membership value that is provided by the input of the MF. The MF that is utilized in the proposed work is Gaussian kernel MF. This is specified in the succeeding equation.

$$\psi_i = exp\left( -\frac{\left\| u_i - v_i \right\|^2}{2w_i^{\,2}} \right) \tag{4}$$

Where, $u_i$, $v_i$ and also $w_i$ are the MFs parameters which can change the MF shape. The parameters are alluded to as the premise parameters.

**Layer 2:** Every node stands as a fixed node termed GMF-ANFIS. And the product of the entire incoming signals gives the output $K_{2,i}$ that implies the rule’s FS.

$$K_{2,i} = G_i = \varphi_i(d_i) \times \psi_i(d_{i+1}) \tag{5}$$

**Layer 3:** Here also every node stands as a fixed node but labeled N. The $i$ th node estimates the ratio of the FS of the ‘ $i$ th’ rule to the addition of the entire rules' FS.



ISSN: 2233-7857 IJFGCN

$$K_{3,i} = \overline{G}_i = \frac{G_i}{G_i}, \quad i = 1,2....6 \tag{6}$$

This layer's outputs are termed normalized FS.

**Layer 4:** Similar to layer-1, each node here stands as an adaptive node encompassing a node function.

$$K_{4,i} = \overline{G}_i . R_i \tag{7}$$

Here, $\overline{G}_i$ implies the normalized FS as of the previous layer and $R_i$ signifies the system's rule. The parameters that are employed are termed as succeeding parameters.

**Layer 5**: The sole node stands as a fixed node termed GMF-ANFIS that calculates the general outcome as the sum of the entire incoming signals. Here, the circle node is labeled as $\sum$

$$K_{5,i} = \sum_i \overline{G}_i R_i = \frac{\sum_i G_i R_i}{\sum_i G_i} \tag{8}$$

**3.3 Air Pollution Monitoring and Prediction System**

After the attacked and non-attacked data are found, the attacked data is saved in a log file which is maintained in a cloud server for future utilization and the non-attacked data goes through additional processes via the AP monitoring system. For finding the AP level, the system uses the training and testing phase. In training, the data values from the AQI dataset are used. The three processes like a) preprocessing b) feature selection c) classification are executed under training. The brief explanation of these '3' processes are provided in the section below

**3.3.1 Pre-processing**

The AQI database is inputted to the pre-processing step. The pre-processing encompasses '3' steps i) removal of redundant data, ii) imputation of missing values and iii) normalization of data. In the initial step, the redundant attribute (irrelevant ones) are eradicated to diminish the data size. In the second step, the missing value of a specific parameter is swapped by other parameters (CO (GT), PT08.S1 (CO), NMHC (GT), etc) if most of its value from any date and time matched the other parameter. Finally, in the 3rd step, the data is scaled for it to come under a particular gamut and Min-Max Normalization is utilized. This transforms a value $d$ to $N_r$ that fits in the gamut [0.1]. It is provided by the formula,

$$N_r = \left( \left( \frac{(d - d_{mn})}{(d_{mx} - d_{mn})} \right) * (1 - 0) + 0 \right) \tag{9}$$

Where, $N_r$ denotes the normalization, zero and one denotes the range, $d_{mx}$ and $d_{mn}$ specifies the maximum and minimum values of data. Normalizing the data attempts to give all attributes an equal weight.

**3.3.2 Feature Selection using MDOA**

There are numerous parameters present within the AQI dataset and when inputting all these parameters into the classifier, it will take loads of time to process each parameter. Thus, the feature selection phase is included here to decide on the pertinent features for the forecast of the AQ level. The modified dragonfly optimization algorithm is utilized here to pick the features of preprocessed data. The main objective that the dragonfly swarms encompass is their survival, which they work-out by only going after food sources and deflecting from all the enemies. Such behaviors are mathematically designed. Here, '3' chief rules are followed by means of each other (nature of swarms)

- Separation ( $S_c$ ) - The dragonflies avoid each other so as to circumvent collision in a still position as of the neighborhood.
- Alignment ( $A_l$ ) – the velocity of every dragonfly coordinates with another one in neighborhood



ISSN: 2233-7857 IJFGCN

- Cohesion ($C_e$) - The dragonflies soar in the directions of the midpoint of the cluster of the neighborhood.

The proposed system employs a customized version of the conventional dragonfly algorithm by integrating crossover and mutation techniques to get effective optimization results. The steps embraced in the MDF algorithm are delineated below.

**Step 1**: Initialize the population of every individual. Then devise the objective function which minimizes the weights.

$$O_b = min(W_{jk_i}) \tag{10}$$

**Step 2**: Computes the $S_c, A_l, C_e, E$ and $F$ values. The separation is computed as,

$$S_c = -\sum_{j=1}^{N} L - L_j \tag{11}$$

Where, $L$ implies the position of the current individual, $L_j$ exhibits the position $j$-th neighboring individual, and $N$ implies the number of neighboring individuals. The alignment is computed as,

$$A_l = \frac{\sum_{j=1}^{N} v_j}{N} \tag{12}$$

Where, $v_j$ signifies the velocity of the $j$-th neighboring individual. The cohesion is computed as,

$$C_e = \frac{\sum_{j=1}^{N} L_j}{N} - L \tag{13}$$

The attraction towards a food source is evaluated utilizing the mathematical relation that is specified

$$F_s = L^{+} - L \tag{14}$$

Where, $L$ signifies the current individual's position and $L^{+}$ denotes the food source position. The Distraction towards the enemy is formulated as,

$$E = L^{-} - L \tag{15}$$

Where, $L$ signifies the current individual's position in addition $L^{-}$ is the enemy's position.

**Step 3**: For the updation of the dragonflies' position in a search space as well as to reproduce their movements, the position and step vector are considered. The step vector ($\Delta Y$) specifies the dragonflies' direction and is written as,

$$\Delta Y_{t+1} = (sS_c + aA_l + cC_e + fF_s + eE) + w\Delta L_t \tag{16}$$

Where, $t$ implies the current iteration, $s$ is the separation weight, $S_c$ signifies the separation of the $i$-th individual, $a$ indicates the alignment weight, $A_l$ indicates the alignment of the $i$-th individual, $c$ cohesion weight, $C_e$ specifies the cohesion of the $i$-th individual, $f$ is the food factor, $F_s$ specifies the food source of the $i$-th individual, $e$ is the enemy factor, $E$ signifies the enemy's position of the $i$-th individual, indicates the weight of inertia as well as $t$ specifies the iteration counter.

**Step 4**: The position vector ($Y$) can well be calculated from the $\Delta Y$. It is written as follows.



ISSN: 2233-7857 IJFGCN

$$Y_{t+1} = Y_t + \Delta Y_{t+1} \tag{17}$$

Where, $t$ implies the current iteration.

**Step 5**: This step is carried out once the stopping criterion is attained. Once the stopping criteria are fulfilled, there is an additional way of ascertaining the position vector. The dragonfly's position is updated utilizing the below equation.

$$Y_{t+1} = Y_t + L_v(di).Y_t \tag{18}$$

Where, $di$ implies the position vector's dimension and $L_v$ represents the Levy flight.

**Step 6**: After that, Crossover and mutation are implemented if the dragonfly does not encompass even '1' neighboring dragonfly, which makes the optimization more effectual. The crossover type that is utilized here is the 2-point crossover. This is performed by utilizing the crossover points.

$$c_1 = \frac{|Y_{t+1}|}{3} \tag{19}$$

$$c_2 = c_1 + \frac{|Y_{t+1}|}{2} \tag{20}$$

Where $c_1$ and $c_2$ are the '2' points that are chosen as crossover points.

**Step 6**: Mutation is performed by swapping the genes as of every chromosome with new ones. The swapped genes are arbitrarily created ones that don't encompass any repetition inside the chromosome. In the proposed work, the chromosomes are the assortment of parameters that state the solution. The MDOA pseudo-code is evinced in figure 2

**Input:** IoT non attacked sensor values
**Output:** Selected Features

**Begin**
  Initialize the population of dragonflies as $W_{jk_i} = W_{jk_1}, W_{jk_2}, \ldots W_{jk_n}$
  Initialize the step vectors
  **While** the stopping criteria is not met
    **Calculate** the objective function for all $W_{jk_i}$
    **Calculate** $S_e, A_l, C_e, E$ and $F$
    **Update** the neighborhood radius
    **If** the current dragonfly has at least one neighbouring dragonfly
      **Update** the step vector using equ (16)
      **Update** the position vector using equ (17)
    **Else**
      **Update** the position vector using equ (18)
      **Select** two crossover points
      **Perform** mutation
    **End** if
    **Check** and correct the new positions on new boundaries of variables
  **End** while
**End**

**Figure 2:** Pseudo code for MDOA

### 3.3.3 DLMNN Classifier

Here, the selected features are provided as input to the DLMNN. As there is just '1' hidden layer (HL) in the existing ANN, it takes more time to train the data. To resolve this problem, this proposed method uses more than three HLs and generates optimized weight value between the HLs and input layer, and HLs and output layers utilizing Gravitational Search Optimization Algorithm (GSOA) which is termed as DLMNN. The DLMNN structure could be comprehended using Fig 3,



ISSN: 2233-7857 IJFGCN

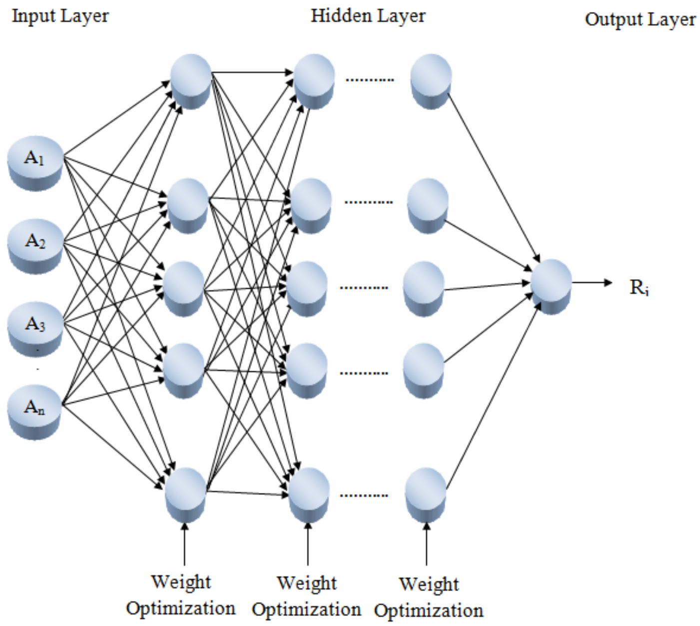



**Figure 3:** Structure of DLMNN

In Fig 3, $\{A_1, A_2, A_3, .......A_n\}$, are the input feature values.

**Step 1:** Input the selected feature values with their equivalent weights as follows

$$A_i = \{ A_1, A_2, A_3 .....A_n \} \tag{21}$$

$$U_i = \{ U_1, U_2, U_3 .....U_n \} \tag{22}$$

Where,

$A_i$ - Input value that indicates $n$ number of selected features, i.e. $A_1, A_2, A_3 .....A_n$

$U_i$ - Weight value of $A_i$ that includes $n$ number of weights, i.e. $U_1, U_2, U_3 .....U_n$

**Step 2:** Multiply the arbitrarily selected $A_i$ with $U_i$ and then total the resultant values.

$$S_u = \sum_{i=1}^{n} A_i U_i \tag{23}$$

Where

$S_u$ - Summed value.

**Step 3:** Determine the activation function which is mathematically depicted as $Af_i$ in eqn. (24),

$$Af_i = C_i (\sum_{i=1}^{n} A_i U_i) \tag{24}$$

And

$$C_i = e^{-A_i^2} \tag{25}$$

Where

$C_i$ - Exponential of $A_i$ .

**Step 4**: Here, the output of the next HL's is stated as,

$$y_i = z_i + \sum C_i U_i \tag{26}$$

Where,



ISSN: 2233-7857 IJFGCN

$z_i$ - Bias value,

$U_i$ - Weight between the input and the HLs.

**Step 5**: Here, the steps as of 2 to 4 are done for every-layer in the DLMNN. Lastly, assess the output unit by totaling the entire input signals' weights for attaining the output layer neurons' value as in,

$$R_i = z_i + \sum V_i U_i \tag{27}$$

Where,

$V_i$ - Value of the layer that precedes the output one

$R_i$ - Output unit.

**Step 6:** This step contrasts the network output against the target value. The difference of these '2' values is termed an error signal $E_s$ which is mathematically evaluated as,

$$E_s = T_i - R_i \tag{28}$$

Where,

$E_s$ - Error signal,

$T_i$ - Target output.

**Step 7:** This step contrasts the output unit and the targeted value for ascertaining the related error. Grounded on this error, a value $\eta_i$ is evaluated and also it is employed to allocate the error at the output back to all other units on the network.

$$\eta_i = E_s [ f(R_i)] \tag{29}$$

**Step 8:** Implement the back-propagation model for assessing weight correction. This relation is proffered as,

$$U_{ci} = \alpha \delta_i (A_i) \tag{30}$$

Where

$U_{ci}$ - Weight correction,

$\alpha$ - Momentum term,

$\eta_i$ - Error that is distributed in the network.

The weight values are now optimized utilizing the GSOA which is delineated in the succeeding section as,

#### 3.3.3.1 GSOA

GSOA stimulated by Newton's law of gravity and motion is employed for optimizing the weight values. The solutions in the GSOA populace are termed as agents and these agents inter-communicate via the gravitational force (GF). The agents are regarded as objects and their performance is gauged by their respective masses. All objects travel in the direction of other objects with heavy masses on account of the GF. This step signifies the global movement of the object whilst the agent with a heavy mass that travels slowly assures the exploitation step of the algorithm. Here, a solution having heavy mass is concerned as the best solution.



ISSN: 2233-7857 IJFGCN

The eqn. 31 gives a gravitational constant $H$ at the iteration $t_r$,

$$H(t_r) = H_0 e^{-\beta t_r / T_n} \tag{31}$$

Where

$T_n$ - Total iterations.

$H_0$ and $\beta$ are initialized at the commencement of the search, and their values would be lessened at the time of search. The masses obey the law of gravity as evinced in Equation (32) and the law of motion in Equation (33).

$$M_G = H\left(Y_1 Y_2 / d^2\right) \tag{32}$$

$$f = M_G / Y \tag{33}$$

Where

$M_G$ - Magnitude of the GF,

$Y_1$ - Mass of the 1st object,

$Y_2$ - Mass of the 2nd object,

$d$ - Distance in-between the two objects

As per the Newton law, the GF between 2-objects in addition to the product of their respective masses is directly proportional to one another. Moreover, the GF between 2-objects and the square of their distance are inversely proportional and it could be comprehensible from equation (32). Whilst for Equation (33), Newton's 2nd law evinces that when a force, $M_G$ is employed to an object, its acceleration, $f$ is contingent on the force and its mass $Y$.

In GSOA, the agent comprises '4' parameters, say inertial mass, position, passive gravitational mass, along with active gravitational mass. The mass' position specifies the solution for the issue, wherein the inertial along with gravitational masses are ascertained utilizing a fitness function. This algorithm is executed via adjusting those masses, and every mass proffers a solution. However, the heaviest mass attracts those masses and hence, it could proffer an optimal solution in its search space. The GSOA steps are:

- ✓ Initially assign the position of '$N$', i.e., number of agents arbitrarily utilizing the below equation



ISSN: 2233-7857 IJFGCN

International Journal of Future Generation Communication and Networking
Vol. 13, No. 4, (2020), pp. 925–945

$$P_i = \left(p_i^1, p_i^2, \ldots p_i^d, \ldots. p_i^n\right) \text{ for } i = 1,2,\ldots N \tag{34}$$

Where,

$p_i^d$ - Positions of the $i$ th agent in the $d$ th dimension,

$n$ - Space dimension.

- ✓ Then, determine the fitness value (FV) and the best FV. For the maximization-minimization problems, the FV evaluation is done by assessing the worst and best FV for the entire agents at all iteration. The worst and the best FV for the minimization issue could be evaluated as,

$$b(t_r) = min\ ft_{j(t_r)} \text{ Where } j \in 1,2,\ldots N \tag{35}$$

$$w(t_r) = max\ ft_{j(t_r)} \tag{36}$$

Where

$ft_{j(t_r)}$ - The $j$ th agent's FV value at iteration $t_r$

$b(t_r)$ and $w(t_r)$ - Best and worst FV, respectively at iteration $t_r$

The best-worst FV for minimization problems could be evaluated utilizing the same formulae of eqn. (35) and (36).

- ✓ Subsequently, Inertia and Gravitational masses for every agent are evaluated utilizing the below equations

$$Y_{ai} = Y_{pi} = Y_{ii} = Y_i \text{ Where } i = 1,2,\ldots N \tag{37}$$

$$y_i(t_r) = \frac{ft_i(t_r) - w(t_r)}{b(t_r) - w(t_r)} \tag{38}$$

$$Y_i(t_r) = \frac{y_i(t_r)}{\sum_{i=1}^{N} y_j(t_r)} \tag{39}$$

Where

$Y_{ai}$, and $Y_{pi}$ - Active and passive gravitational masses



ISSN: 2233-7857 IJFGCN

$Y_{ii}$ - The $i$ th agent's Inertia mass.

- ✓ The $i$ th agent's acceleration at iteration $t_r$ is evaluated as,

$$f_i^d(t_r) = \frac{W_i^d(t_r)}{Y_{ii}(t_r)} \tag{40}$$

Where,

$W_i^d(t_r)$-total force acting on $i$ th agent

$$W_i^d(t_r) = \sum_{j \in Bbest, j \neq i} Rd_j W_{ij}^d(t_r) \tag{41}$$

Where $Bbest$ signifies the collection of first $B$ agents having the biggest mass and best FV. $Bbest$ would diminish linearly with time. In the end, there would be merely one agent to apply force to the other agents. $W_{ij}^d(t_r)$ is evaluated utilizing the succeeding equation:

$$W_{ij}^d(t_r) = H(t_r).\left(Y_{pi}(t_r) \times {Y_{aj}(t_r)}\Big/{E_d(t_r)} + \in\right) \cdot \left(p_j^d(t_r) - p_i^d(t_r)\right) \tag{42}$$

$W_{ij}^d(t_r)$ - Force influencing on agent $i$ as of the agent $j$ at $d$ th dimension at $t_r$ th iteration

$E_d(t_r)$ - Euclidian distance of agents $i$ and $j$

$\in$ - Small constant.

- ✓ Position and Velocity of the agents at subsequent iteration (t+1) are evaluated centered on the succeeding equations:

$$r_i^d(t_r + 1) = rd_i\, p\, r_i^d(t_r) + f_i^d(t_r) \tag{43}$$

$$P_i^d(t_r + 1) = p_i^d(t_r) + r_i^d(t_r + 1) \tag{44}$$

- ✓ Re-execute steps 2 to 6 till the iterations reach their maximal limit. The best FV at the final iteration is evaluated as the global FV whilst the position of the respective agent at specified dimensions is evaluated as the global solution of that specific problem.

The classification outcomes of DLMNN comprise six sorts of data that signify the AP level in the cities. The six classes are i) Good, ii) Moderate, iii) Unhealthy for the sensitive groups, iv) Unhealthy, v) Very



ISSN: 2233-7857 IJFGCN

Unhealthy, and vi) Hazardous. Each class of data indicates an AP level that could be recognized by their AQI value and that would be visualized once testing the IoT values.

### 3. 4 Testing and Visualization

After aforesaid training, the non-attacked sensor data are tested utilizing the same processes of training. Subsequently, the testing classification outcomes are visualized which is expounded in the section below. The classified data is visualized in several formats in the network, for instance, the base of severity in the equivalent locations are offered in the Google map utilizing disparate colors as enumerated below.

i) Good (AQI≤50): The visualization "Outdoor air is safe to breathe" is marked with green color if the classification outcome is good.

ii) Moderate (AQI: 51–100): The visualization "unusually sensitive people may avert prolonged or heavy outdoor exertion" is marked with yellow color if the outcome is moderate.

iii) Unhealthy for the sensitive groups (AQI: 101–150): Here, the visualization "Sensitive people (e.g., children, old people, outdoor workers, and patients having lung disease, say asthma) need to lessen long or heavy outdoor exertion" is marked with orange.

iv) Unhealthy (AQI: 151–200): Here, the visualization **"**Sensitive people should avert prolonged or heavy outdoor exertion. Everyone else has to lessen long or heavy outdoor exertion" is marked with red.

v) Very unhealthy (AQI: 201–300): Here, the visualization is "Sensitive people should avert each outdoor exertion. Everyone else has to reduce outdoor exertion" in purple.

vi) Hazardous (AQI≥301): Here, the visualization is "Everyone should avoid all outdoor exertion" in maroon color.

## 4. RESULT AND DISCUSSION

In this phase, the performance is examined for the IoT based AQ monitoring and prediction system utilizing DLMNN by implementing in the working platform of JAVA. The proposed methods utilized in attack detection, and classification are contrasted to the existing methodologies and their comparisons are expounded in the below sections.

### 4.1 Dataset Description

AQI- UCI dataset is utilized in the proposed work for examining its performance. A set of five chemical metal oxide sensors is fixed in an AQ Chemical Multi-sensor Device. From which, the AQI- UCI dataset comprises 9358-events of hourly averaged responses. The afore-said device was fixed in Italy, specifically in the field of a highly polluted area at road level. Co-located reference certified analyzer proffered Ground Truth hourly averaged concentrations of Non-Metanic Hydro-carbons, CO, Benzene, Total Nitrogen Oxides (NOx) along with Nitrogen Di-oxide (NO2).

### 4.2 Performance Analysis of GMF-ANFIS



ISSN: 2233-7857 IJFGCN

Here, the performance shown by GMF-ANFIS utilized in attack detection is contrasted to the existing ANFIS concerning the performance measures, say packet loss ($P_l$) along with packet delivery ratio ($P_{dr}$) and this could be elucidated using table 1. A packet drop occurs in the loT network because of the higher temperature, dense foliage, noises, and rain. Once the IoT sensor values are attained, the sensor nodes' $P_l$ and $P_{dr}$ could be checked to find whether the received sensor node is attacked or not utilizing the GMF-ANFIS. If the system attains the data packets with the lowest $P_l$ and highest $P_{dr}$, then the system is highly reliable and there is a possibility to have a very low number of attacks in the nodes. The $P_l$ and $P_{dr}$ are computed as,

$$P_l = \frac{N_{pnr}}{N_{ps}} \tag{45}$$

Where

$N_{pnr}$ - Number of lost packets

$N_{ps}$ - Total number of packets sent

$$P_{dr} = \frac{R_{dp}}{G_{dp}} \tag{46}$$

Where

$R_{dp}$ - Received data packets by the destination

$G_{dp}$ - Generated packets by the source

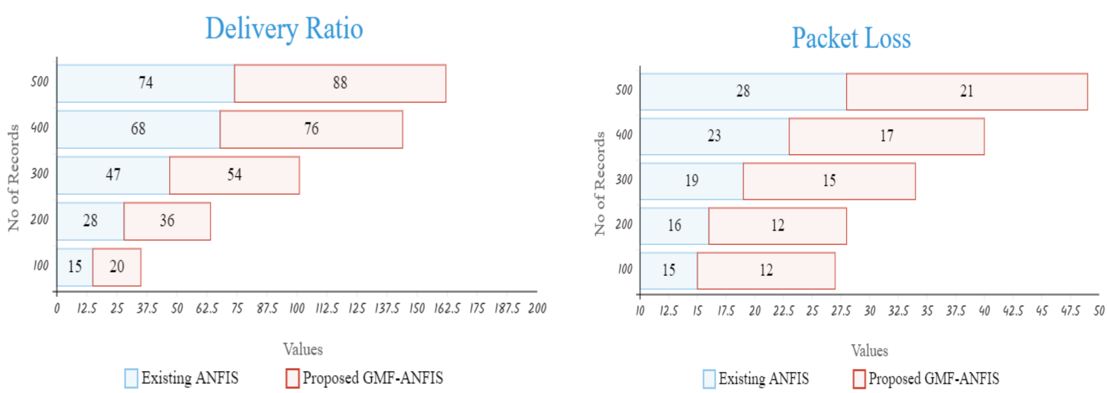
**Table 1:** Performance of the GMF-ANFIS with existing ANFIS

| **No of Records** | $P_{dr}$ | | $P_l$ | |
|---|---|---|---|---|
| | **Proposed GMF-ANFIS** | **Existing ANFIS** | **Proposed GMF-ANFIS** | **Existing ANFIS** |
| 100 | 20 | 15 | 12 | 15 |
| 200 | 36 | 28 | 12 | 16 |
| 300 | 54 | 47 | 15 | 19 |
| 400 | 76 | 68 | 17 | 23 |
| 500 | 88 | 74 | 21 | 28 |

Table 1 contrasted the proposed GMF-ANFIS and existing ANFIS in respect of $P_{dr}$ and $P_l$. $P_{dr}$ and $P_l$ are perceived for the records ranging from 100 to 500. For 100 records, the GMF-ANFIS proffers $P_{dr}$


ISSN: 2233-7857 IJFGCN

of 20 and $P_l$ of 12, whereas, the existing ANFIS renders 15 $P_{dr}$ and 15 $P_l$. Here $P_{dr}$ of the GMF-ANFIS is higher when contrasted to the ANFIS and $P_l$ of GMF-ANFIS is lower on considering the ANFIS. Likewise for the remaining number of records, the proposed GMF- ANFIS gives the lowest $P_l$ and highest $P_{dr}$ when contrasted to ANFIS. If the system has the lowest packet loss value then the system is known as the best system. So it is stated that the proposed GMFANFIS achieves better results in attack detection. The graphical demonstration of table 1 is plotted in figure 4.




**(a)** **(b)**

**Figure 4:** Performance graph for GMF-ANFIS and ANFIS

The above figure 4 compares the performance graph for the proposed GMF-ANFIS and existing ANFIS in respect of $P_{dr}$ and $P_l$. For 200 records, the GMF-ANFIS proffers $P_{dr}$ of 36 and $P_l$ of 12, whereas, the existing ANFIS renders 28 $P_{dr}$ and 16 $P_l$. Here $P_{dr}$ of the GMF-ANFIS is higher when contrasted to the ANFIS and $P_l$ of GMF-ANFIS is lower on considering the ANFIS. Likewise for the remaining number of records (100, 300, 400, and 500), the proposed GMF- ANFIS gives the lowest $P_l$ and highest $P_{dr}$ when contrasted to ANFIS. So from the comparison it is clearly known that the proposed GMFANFIS achieves better results in attack detection.

### 4.3 Performance Analysis of DLMNN

Here, the performance of the DLMNN for AP monitoring and prediction system is analyzed by contrasting to the existing LaSVM in respect of precision ( $p_r$ ), recall ( $r_c$ ), f-measure ( $f_m$ ), accuracy ( $a_c$ ), regression coefficient ( $rg_c$ ), and root mean squared errors (RMSE or $R_m$ ). The computation of $p_r$, $r_c$, $f_m$, and $a_c$ is done centered on True Positives, False Negatives, True Negatives, and False Positives. The computation of $a_c$, $R_m$ and $rg_c$ are provided as,

$$a_c = \frac{C_{nr}}{T_{nt}} \tag{47}$$



ISSN: 2233-7857 IJFGCN

Where,

$C_{nr}$ - Number of correctly predicted values

$T_{nt}$ - Total test data.

$$R_m = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left|R_i^* - R_i\right|^2} \tag{48}$$

$$rg_c = 1 - \sum_{i=1}^{n}\left(R_i^* - R_i\right)^2 \Big/ \sum_{i=1}^{n} R_i - \overline{R} \tag{49}$$

Where

$R_i^*$ and $R_i$ - The predicted and observed values

$\overline{R}$ - Mean of observed data in eqn (49)

The comparison of DLMNN with LaSVM is evinced in table 2.

**Table 1:** Performance comparison of DLMNN with LaSVM

(a)

| No of records | Proposed DLMNN | | Existing LaSVM | |
|---|---|---|---|---|
| | $p_r$ | $r_c$ | $p_r$ | $r_c$ |
| 100 | 92.5925 | 92.5925 | 90.9090 | 91.5567 |
| 200 | 98.0392 | 93.6574 | 96.1538 | 92.3432 |
| 300 | 97.6562 | 94.4656 | 96.1538 | 93.6565 |
| 400 | 98.0392 | 96.4633 | 97.2222 | 95.3432 |
| 500 | 98.6842 | 97.3454 | 97.8260 | 96.8343 |

(b)

| No of records | Proposed DLMNN | | Existing LaSVM | |
|---|---|---|---|---|
| | $a_c$ | $f_m$ | $a_c$ | $f_m$ |
| 100 | 92 | 92.5925 | 91 | 91.7431 |
| 200 | 93.5 | 95.8466 | 92 | 94.9367 |
| 300 | 94.6666 | 96.8992 | 93.3333 | 96.1538 |
| 400 | 95.75 | 97.6290 | 94.25 | 96.8188 |
| 500 | 96.8 | 98.2532 | 95.2 | 97.4026 |

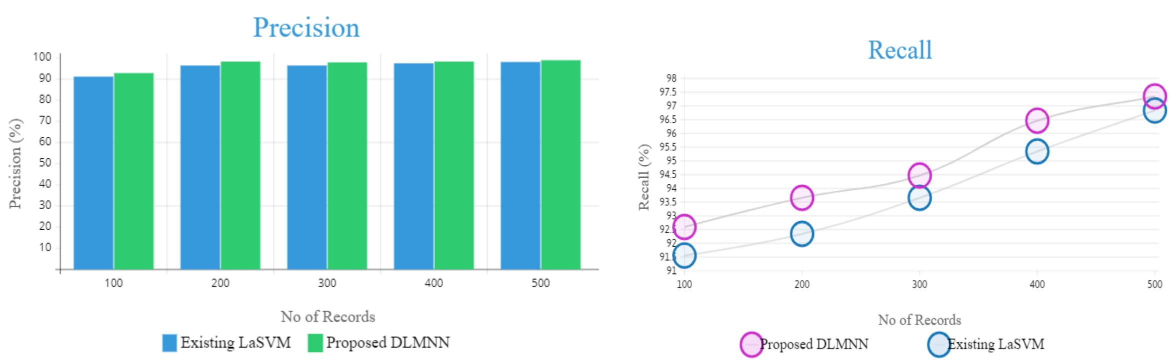


ISSN: 2233-7857 IJFGCN

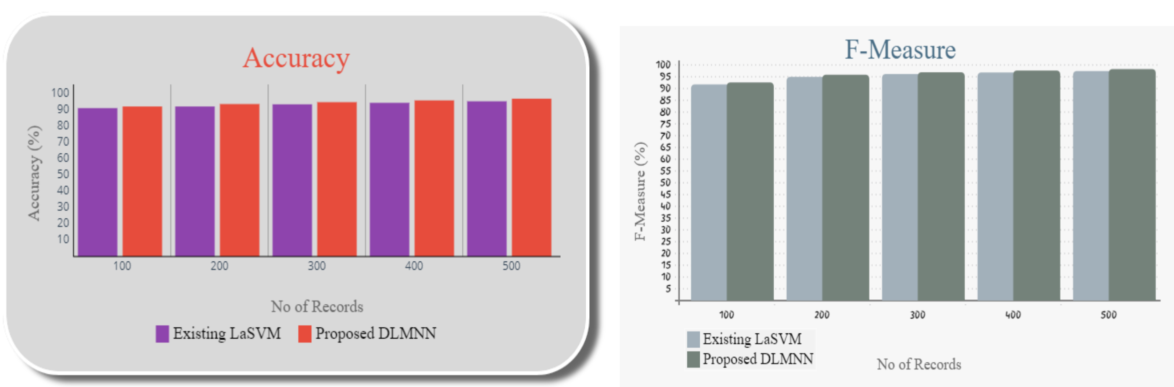
(c)

| No of records | Proposed DLMNN | | Existing LaSVM | |
|---|---|---|---|---|
| | $R_m$ | $rg_c$ | $R_m$ | $rg_c$ |
| 100 | 92.5925 | 92.5925 | 90.9090 | 91.5567 |
| 200 | 98.0392 | 93.6574 | 96.1538 | 92.3432 |
| 300 | 97.6562 | 94.4656 | 96.1538 | 93.6565 |
| 400 | 98.0392 | 96.4633 | 97.2222 | 95.3432 |
| 500 | 98.6842 | 97.3454 | 97.8260 | 96.8343 |

Table 2 contrasted the performance shown by the proposed DLMNN and existing LaSVM concerning the performance measures, say (a) $p_r$ and $r_c$, (b) $a_c$ and $f_m$, and (c) $R_m$ and $rg_c$. These measures are evaluated for 500 records. On considering100 records, the proposed DLMNN proffers the values of 92.5925, 92.5925, 92, 92.5925, 92.5925, and 92.5925, whereas, the existing LaSVM technique gives 90.9090, 91.5567, 91, 91.7431, 90.9090, and 91.5567 for $p_r$, $r_c$, $a_c$, $f_m$, $R_m$, and $rg_c$ , respectively. Here, the existing techniques have lower values on considering the DLMNN. The proposed DLMNN attains the topmost accuracy level for all records. And, for the remaining 200 to 500 records, the DLMNN proffers the highest values for all measures when contrasted to LaSVM. So it is inferred that the DLMNN works well for the AP prediction system and it accurately recognizes the air pollutant level. Table 2 is plotted graphically in figure 5.



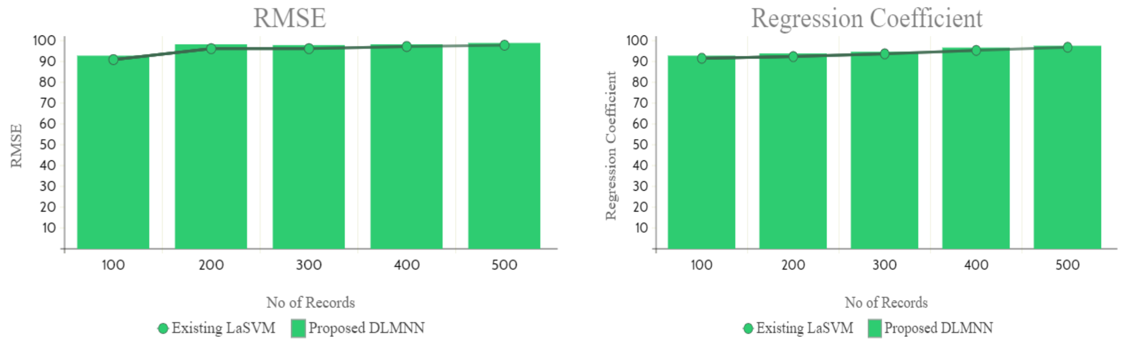

(a)




(b)



ISSN: 2233-7857 IJFGCN




(c)

**Figure 5:** Performance analysis graph for DLMNN and LaSVM regarding (a) (a) $p_r$ and $r_c$, (b) $a_c$ and $f_m$, and (c) $R_m$ and $rg_c$.

The above figure 5 shows the performance comparison of the proposed DLMNN with existing LaSVM concerning the performance measures, say (a) $p_r$ and $r_c$, (b) $a_c$ and $f_m$, and (c) $R_m$ and $rg_c$. These measures are computed for 500 records. On considering 200 records, the proposed DLMNN proffers the values of 98.0392, 93.6574, 93.5, 95.8466, 92.5925, and 92.5925, whereas, the existing LaSVM technique gives 96.1538, 92.3432, 92, 94.9367, 96.1538, and 92.3432 for $p_r$, $r_c$, $a_c$, $f_m$, $R_m$, and $rg_c$, respectively. Comparing all, the proposed DLMNN attains the topmost accuracy level for all records. And, for the remaining records, the DLMNN proffers the highest values for all measures when contrasted to LaSVM.

## 5. CONCLUSION

This work proposes the IoT based AQ monitoring and prediction system utilizing DLMNN. The attack detection phase is executed on the received sensor values. Subsequently, the non-attacked sensor values are tested and the pollutant level of the air is predicted. Totally six sorts of pollutant levels are found and visualized. The outcomes of the proposed method are assessed by contrasting their performance to the existing techniques. The proposed GMF-ANFIS attains the lowest $P_l$ and highest $P_{dr}$ for 100 to 500 records when contrasted to the existing ANFIS. In addition, the performance shown by the proposed DLMNN for AQ prediction is contrasted to the existing LaSVM. The DLMNN attains the highest values for $p_r$, $r_c$, $a_c$, $f_m$, $R_m$, and $rg_c$ when weighed against the existing LaSVM. So, from the performance comparison, it is stated that DLMNN is suitable for the AQ monitoring and prediction system.

## REFERENCES

1. choi, W., hwang, D., kim, J., & lee, J.,"Fine dust monitoring system based on Internet of Things", International Conference on Information and Communication Technology Robotics, 2018.
2. Yohan Han, Byungjun Park, and Jongpil Jeong, "A novel architecture of air pollution measurement platform using 5g and blockchain for industrial iot applications", Procedia Computer Science, vol. 155, pp. 728-733, 2019.



ISSN: 2233-7857 IJFGCN

3. Saba Ameer, Munam Ali Shah, Abid Khan, Houbing Song, Carsten Maple, Saif Ul Islam, and Muhammad Nabeel Asghar, "Comparative analysis of machine learning techniques for predicting air quality in smart cities", IEEE Access, vol. 7, pp. 128325-128338, 2019
4. Hernández-Vega, J.I Varela, E.R., Romero, N.H., Hernández-Santos, C., Cuevas, J.L.S. and Gorham, D.G.P, "Internet of Things (IoT) for monitoring air pollutants with an unmanned aerial vehicle (UAV) in a smart city", In Smart Technology, Springer, Cham, pp. 108-120, 2018.
5. Temesegan Walelign Ayele, and Rutvik Mehta, "Air pollution monitoring and prediction using IoT", In Second International Conference on Inventive Communication and Computational Technologies (ICICCT), pp. 1741-1745. IEEE, 2018.
6. Grace Kingsy, R., Manimegalai R., Devasena MS Geetha, Rajathi S., Usha K., and Baseria N. Raabiathul, "Air pollution analysis using enhanced K-Means clustering algorithm for real time sensor data", In IEEE Region 10 Conference (TENCON) IEEE, pp. 1945-1949, 2016.
7. Huang, Le Hui, and Bin Gui, "Discussion on Air Pollution and Its Control Measures", Advanced Materials Research, pp. 1010- 1012, pp. 839, 2014.
8. Kumar S. and Katoria D, "Air Pollution and its Control Measures", International Journal of Environmental Engineering and Management, vol. 4, no. 5, pp. 445-450, 2013.
9. Nitin Sadashiv Desai, and John Sahaya Rani Alex, "IoT based air pollution monitoring and predictor system on Beagle bone black", In International Conference on Nextgen Electronic Technologies: Silicon to Software (ICNETS2), pp. 367-370. IEEE, 2017.
10. Gagan Parmar, Sagar Lakhani, and Manju K. Chattopadhyay, "An IoT based low cost air pollution monitoring system", In International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE), IEEE, pp. 524-528, 2017.
11. Kiruthika, R., and A. Umamakeswari, "Low cost pollution control and air quality monitoring system using Raspberry Pi for Internet of Things", In International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), IEEE, pp. 2319-2326, 2017.
12. Jhanvi Arora, Utkarsh Pandya, Saloni Shah, and Nishant Doshi, "Survey-pollution monitoring using IoT", Procedia Computer Science, vol. 155, pp. 710-715, 2019.
13. Debadri Dutta, Akshit Pradhan, Acharya O. P., and Mohapatra S. K., "IoT based pollution monitoring and health correlation: a case study on smart city", International Journal of System Assurance Engineering and Management, vol. 10, no. 4, pp. 731-738, 2019.
14. İbrahim Kök, Mehmet Ulvi Şimşek, and Suat Özdemir, "A deep learning model for air quality prediction in smart cities", In IEEE International Conference on Big Data (Big Data), IEEE, pp. 1983-1990, 2017.
15. Ling Wang, Xi-yuan Xiao, and Jian-yao Meng, "Prediction of air pollution based on FCM-HMM Multi-model", In 35th Chinese Control Conference (CCC), IEEE, pp. 2057-2062, 2016.
16. Spiridonov, V., Jakimovski, B., Spiridonova, I., & Pereira, G, "Development of air quality forecasting system in Macedonia, based on WRF-Chem model. Air Quality, Atmosphere & Health, pp. 1-12, 2019.
17. Xia Xi, Zhao Wei, Rui Xiaoguang, Wang Yijie, Bai Xinxin, Yin Wenjun, and Don Jin, "A comprehensive evaluation of air pollution prediction improvement by a machine learning method", In IEEE International Conference on Service Operations And Logistics, And Informatics (SOLI), IEEE, pp. 176-181, 2015.
18. Xiang Li, Ling Peng, Yuan Hu, Jing Shao, and Tianhe Chi, "Deep learning architecture for air quality predictions", Environmental Science and Pollution Research, vol. 23, no. 22, pp. 22408-22417, 2016.
19. Zhongshan Yang, and Jian Wang, "A new air quality monitoring and early warning system: Air quality assessment and air pollutant concentration prediction", Environmental research, vol. 158, pp. 105-117, 2017.
20. Dixian Changjie Cai Zhu, Tianbao Yang, and Xun Zhou, "A machine learning approach for air quality prediction: Model regularization and optimization", Big data and cognitive computing, vol. 2, no. 1, pp. 5, 2018.



ISSN: 2233-7857 IJFGCN

21. Jiangshe Zhang, and Weifu Ding, "Prediction of air pollutants concentration based on an extreme learning machine: the case of Hong Kong", International journal of environmental research and public health, vol. 14, no. 2, pp. 114, 2017.
22. Xiaozheng Lai, Ting Yang, Zetao Wang, and Peng Chen, "IoT implementation of kalman filter to improve accuracy of air quality monitoring and prediction", Applied Sciences, vol. 9, no. 9, pp. 1831, 2019.
23. Sandro Rodriguez Garzon, Sebastian Walther, Shaoning Pang, Bersant Deva, and Axel Küpper, "Urban air pollution alert service for smart cities", In Proceedings of the 8th International Conference on the Internet of Things, pp. 9, 2018.
24. Alexander Miles, A. Zaslavsky, and Chris Browne, "IoT-based decision support system for monitoring and mitigating atmospheric pollution in smart cities", Journal of Decision Systems, vol. 27, no. 1, pp. 56-67, 2018.



ISSN: 2233-7857 IJFGCN