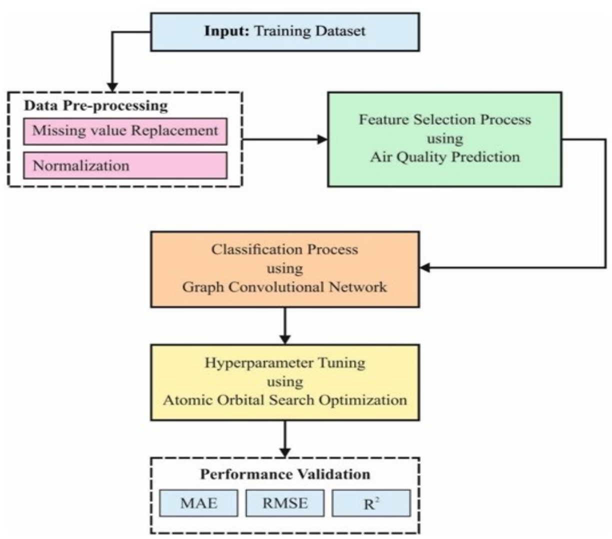
Global NEST Journal, Vol 25, No 5, pp 11-19

Deep learning and machine learning based air pollution prediction model for smart environment design planning

Karthikeyan B.[1], Mohanasundaram R.[2], Suresh P.[3] and Jagan Babu J.[4]
[1]Department of Information Technology, Panimalar Engineering College, Chennai, India
[2]School of Computer Science and Engineering, VIT, Vellore, India
[3]Department of Computer Science and Engineering, KPR Institute of Engineering and Technology, Coimbatore-641407, India
[4]Department of Electronics and Communication Engineering, R.M.D Engineering College, Kavaraipettai, Chennai, India
Received: 07/01/2023, Accepted: 31/01/2023, Available online: 04/02/2023
*to whom all correspondence should be addressed: e-mail: karthikeyanb953@gmail.com
https://doi.org/10.30955/gnj.004735

Graphical abstract



Abstract

For the past few decades, owing to human activities, urbanization, and industrialization, air pollution is becoming severe across several countries. Deep Learning (DL) and Machine Learning (ML) techniques had great contribution to the development of methods in various aspects of prediction, planning, and uncertainty analysis of smart cities and urban advancement in the current scenario. Many of the cities which are developed suffered from severe air quality (AQ) because of the rapid growth in industrialization and population. In this paper, we introduce a deep learning based air pollution prediction model for smart environment design planning (DLAPP-SEDP). The presented DLAPP-SEDP technique majorly intends to predict the level of air pollution in the smart environment. It follows a three-stage process namely data pre-processing, air pollution prediction, and hyperparameter tuning. At the initial stage, the presented DLAPP-SEDP technique performs various levels of data pre-processing such as missing value replacement, categorical value encoding, normalization, and feature selection. In the next stage, the DLAPP-SEDP technique employs graph convolutional network (GCN) model. Finally, the DLAPP-SEDP technique utilizes atomic orbital search optimization (AOSO) algorithm for optimal hyperparameter tuning process, showing the novelty of the work. To demonstrate the enhanced predictive efficiency of the DLAPP-SEDP method, a wide-ranging experimental analysis can be carried out. The experimental values assured the enhancements of the DLAPP-SEDP method over other recent techniques.

Keywords: Air pollution monitoring, smart environment, sustainability, deep learning, parameter optimization

1. Introduction

Urban areas and people who were living in those areas are often affected by environmental factors. It imposes novel issues for urban planners, like enhancing the air quality (AQ) and minimizing sound levels for building a friendly and clean environment for the population of a city. Additionally for avoiding adverse effects on the enterprises and residents, like dense snowstorms or bullying, severe meteorological conditions in a city should be controlled appropriately (Iskandaryan *et al.*, 2020). More stringent checks and tests, in the central areas of town car bans, Encouraging e-mobility, Greening of the city. Such things for enhancing AQ in cities and minimize the sound level. The vast and most unsatisfied capability of smart city technology for boosting the living standards (Ullo and Sinha, 2020). In the ecological sector-major variations have done and beyond the benefits of employment, safety, standard of housing, energy, interconnectivity, and wellness.

Air pollution contains an extensive range of impacts on humans which involve early death, breathing conditions, and pulmonary disease clinic (Masih, 2019). Ozone and nitrogen dioxide gas majorly affect individuals with conditions like liver cancer, asthma, and, respiratory problems that make the disease severe. The Air Quality Index (AQI) can be utilized to measure AQ. These metrics have a huge effect on the air pollution level and were computed by sensors in various SCs. The sensor systems connection in cities produces various data that are annotated in timely manner. Deep Learning (DL) forecasts large quantities of data (Sai *et al*., 2019). The important reason for success of DL was the enhanced chip processing capabilities, dramatically dropped cost of networking equipment, and current developments in data processing and artificial intelligence. As it is basically difficult for processing air pollution, its distribution pattern and temporal patterns were affected by several factors which include emissions of and accumulation of traffic flows, air pollutant, climatic conditions, and activities made by man, and many more (Ameer *et al*., 2019). The issue of utilizing conventional deep methods, mainly for providing quality representative of air pollution features, has enhanced.

The time series pollution data comprises long term dependence amongst all features. By the rapid advancement of machine learning (ML), artificial intelligence methods no longer stay as the existing methods (Al-Janabi *et al*., 2020). Numerous authors had carried out AQ modeling utilizing DL approaches and proved superior predictive methods compared to ML regarding temporal analysis of the air pollution data. DL methods displayed superior performance in medical image classification, sequential modeling, human detection, and other applications (Harishkumar *et al*., 2020). DL technique that is Gated Recurrent Unit (GRU), Recurrent Neural Network (RNN), LSTM, had served a significant part in predicting AQ. Some authors included a Convolutional Neural Networks (CNNs) layer with the shallow DL methods for capturing the spatial features in time-series dataset which is available that grants superior predictive performance through examining both the spatio-temporal features (Kabir *et al*., 2020).

Many prevailing predictive techniques will forecast air pollution levels for the following hours for a specific site. Forecasting air pollution levels for the entire research zone for a longer period could provide a benefit to receiving superior air pollution predictive outcomes (Castelli *et al*., 2020). Generally, air pollution forecasting performance for a longer time grants less accurateness compared to the shorter period. This may occur because the small number of samples will be performing long term AQ forecast (Janarthanan *et al.*, 2021). Thus, it becomes necessary to advance air pollution predictive methods that could efficiently achieve air pollution forecasting for the whole research area at a more important time resolution.

In this paper, we introduce a deep learning based air pollution prediction model for smart environment design planning (DLAPP-SEDP). The presented DLAPP-SEDP technique performs various levels of data pre-processing such as missing value replacement, categorical value encoding, normalization, and feature selection. In the next stage, the DLAPP-SEDP technique employs graph convolutional network (GCN) model. Finally, the DLAPP-SEDP technique utilizes atomic orbital search optimization (AOSO) algorithm for optimal hyperparameter tuning process. To demonstrate the enhanced predictive efficiency of the DLAPP-SEDP algorithm, a wide-ranging experimental analysis is carried out.

The rest of the paper is organized as follows. Section 2 provides the related works and section 3 offers the proposed model. Then, section 4 gives the result analysis and section 5 concludes the paper.

2. Literature review

Kalajdjieski *et al*. (2020) present a new technique assessing 4 different structures for estimating the air pollution in those areas by using camera images. Such images were enhanced by meteorological data for boosting the classifier accuracy. The presented technique will exploit generative adversarial networks (GAN) integrated with data augmenting methods for mitigating the class imbalance issue. Du *et al*. (2019) introduces a new DL algorithm for AQ (mainly PM2.5) prediction that learns interdependence of multivariate AQ related time series data by hybrid DL structure and the spatial-temporal relation features. Owing to the dynamic and nonlinear features of multivariate AQ time sequence data, the base modules of this method add Bi-directional Long Short-term Memory networks (Bi-LSTM) and 1D-CNNs. Previously extracted the spatial correlation features local and trend features, and the later was to study spatial and temporal dependencies.

In (Rao *et al*., 2019), this work presents a DL technique for prediction and quantification of ambient AQ. RNN-related structure having special structured memory cells called LSTM can be presented for capturing the dependences in several pollutants and performing AQ forecasting. Ma *et al*. (2019) devises a DL-related technique like transferred bi-directional LSTM (TL-BLSTM) method for predicting AQ. The techniques use the bi-directional LSTM method for learning the longer period dependencies of PM2.5 and implement TL for transferring the knowledge learnt from smaller to larger temporal resolutions. Chang *et al*. (2020) designed an Aggregated LSTM method (ALSTM) related to the LSTM-DL approach. In this novel technique, the author integrates the stations for external pollution sources, local AQ monitoring stations, and the station in nearby industrial areas. For enhancing prediction accuracy, the author aggregates 3 LSTM methods into a prediction technique for initial forecasting related to exterior sources of pollution and information from neighboring industrial AQ stations.

Le *et al*. (2020) offer the use of Convolutional LSTM (ConvLSTM) method, the grouping of CNN and LSTM that automatically uses both the spatial-temporal features. Particularly, the author presents the conversion way of the air pollution data into series of images that uses ConvLSTM technique for interpolating and predicting AQ for the whole city in the due course. In (Heydari *et al*., 2022), a novel hybrid intellectual method related to multi-verse optimization algorithm (MVO) and LSTM was advanced for predicting and analyzing air pollution gained from Combined Cycle Power Plants. In the presented method, LSTM method becomes forecaster engine for predicting the sum of produced SO2 and NO2 by the integrated Cycle Power Plant in which MVO technique can be employed for optimizing the LSTM variables for achieving a less prediction error.

3. The proposed model

In this paper, a novel DLAPP-SEDP algorithm was introduced to predict the level of air pollution in the smart environment. It follows a three-stage process namely data pre-processing, GCN based air pollution prediction, and AOSO based hyperparameter tuning. Figure 1 showcases the overall process of DLAPP-SEDP technique.

3.1. Data pre-processing

In the early phase, the proposed DLAPP-SEDP algorithm executes different levels of data pre-processing like normalization, missing value replacement, feature selection, and categorical value encoding (Abdellatif *et al.*, 2021). The values that were missing are sorted by linear spline imputation. The SL(x) equation could adapt to local anomaly without affecting the interpolate values at other points. The equation of the spline linear interpolation function as follows:

$$SL(x)=f(x_{i-1})\frac{x-x_i}{x_{i-1}-x_i}+f(x_i)\frac{x-x_{i-1}}{x_i-x_{i-1}}x\in[x_{i-1},x_i],i=1,2,3,\ldots,n \tag{1}$$

whereas x denotes the independent variable, $x_0,x_1,\ldots,x_n$ are well-known values of the spline and $SL(x)$ represents the linear spline which will interpolate f at these points.

To enhance the estimation accuracy, the author makes a normalization of the values utilizing the Min-Max normalization. If many features are entering the network for training purposes, discovering the relation among the target output values and those features minimizes the difficulty of training and enhances performance. The Pearson correlation becomes the well-known technique utilized for finding the relation between two variables.

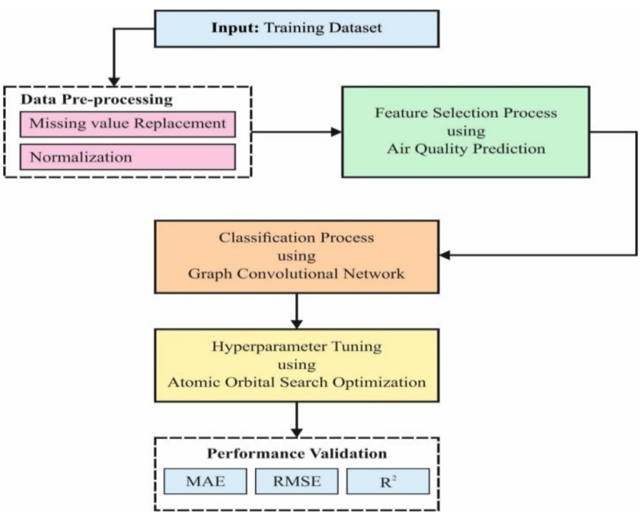


Figure 1. Overall block diagram of DLAPP-SEDP algorithm

3.2. Air pollution prediction using GCN model

For air pollution prediction process, the DLAPP-SEDP technique applied the GCN model. Based on the CNN, GCN refers to a multilayer neural network that functions straightway on graph and intends for extracting higher-level features via aggregating data from the neighborhood of graph node (Sofianos *et al.*, 2021). In GCN, an undirected graph can be generally described by $\mathrm{G}=(\mathrm{V},\mathrm{E})$ with V and E representing the set of nodes and edges, correspondingly. The notation A signifies adjacency matrix of G that represents the presence of edges among all pairs of the node, and its $(i, j)^{th}$ component is evaluated by Eq. (2)

$$A_{ij}=\begin{cases}e^{-\gamma\|x_i-x_j\|^2} & \textit{if } x_i\in N(x_j) \textit{ or } x_j\in N(x_i)\\ 0 & \textit{otherwise}\end{cases}, \tag{2}$$

In Eq. (2), the variable γ is empirically fixed as 0.2 in the experiment, x_i and x_j characterize two graph nodes (viz., image region), and $N(x_j)$ shows the set of neighbors of x_j.

Firstly, to conduct node embedding's for G, spectral filtering on the graph can be determined that is formulated by the multiplication of signal x with filter $g_\theta=diag(\theta)$ in the Fourier domain

$$g_\theta \,\varepsilon\, x=Ug_\theta U^T X, \tag{3}$$

In Eq. (3), U indicates the matrix of eigenvector of normalized graph Laplacian $L=I-D^{-\frac{1}{2}}AD^{-\frac{1}{2}}=U\Lambda U^T$, Λ means a diagonal matrix comprised of the eigenvalue of L, D denotes degree matrix having diagonal component $D_{ii}=\sum_j A_{ij}$, and I characterize identity matrix with appropriate size. Next, g_θ is understood as a function of eigenvalue of L, viz., $g_\theta(\Lambda)$. To decrease the computation cost of Eigen decomposition and estimated $g_\theta(\Lambda)$ by means of truncated expansion interms of Chebyshev polynomial $T_k(x)$ up to K^{th}-order,

$$g_{\theta'}(\Lambda)\approx\sum_{k=0}^{K}\theta'_k T_k(\tilde{\Lambda}), \tag{4}$$

In Eq. (4), θ' indicates a vector of Chebyshev coefficients, and $\tilde{\Lambda}=\frac{2}{\lambda_{\max}}\Lambda-I$ with $\lambda_{\max}$ being the maximum eigenvalues of L. Based on Chebyshev polynomial is represented by $T_k(x)=2xT_{k-1}(x)-T_{k-2}(x)$, where $T_0(x)=1$ and $T_1(x)=x$.

$$g_{\theta'} \,\varepsilon\, x\approx\sum_{k=0}^{K}\theta'_k T_k(\tilde{L})x, \tag{5}$$

In Eq. (5), $\tilde{L} = \frac{2}{\lambda_{\max}}$ L—I show the scaled Laplacian matrix and simply verified based on the fact that $(U\Lambda U^T)^k = U\Lambda^k U^T$. It is noted that, K^{th} order polynomial w.r.t Laplacian (*viz.*, K- localized). In another word, filtering relies merely on node viz., at K steps farther from the centralized node. In the CAD- GCN method, first-order neighborhood is taken into account, viz., $K = 1$, and therefore Eq. (5) becomes a linear function on graph Laplacian spectrum regarding L. Figure 2 depicts the infrastructure of GCN technique.

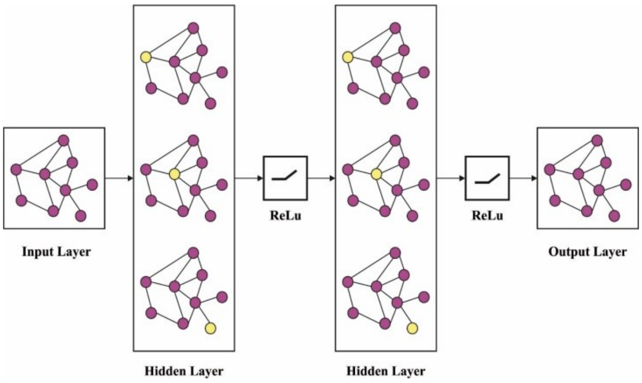


Figure 2. Framework of GCN

Later, a neural network depends on GCN is made through stacking more than one convolution layer, where all the layers are followed by element- wise nonlinear function (viz., softplus (.)). In that regard, different classes of convolution filter functions can be derived by stacking more than one layer of a similar configuration. By using linear formula, Kipf and Welling *Reject*17*Reject* estimated $\lambda_{\max} \approx 2$, consider the network parameter is adapted to this change in scale at the training model as follows

$$g_{\theta'} \ \varepsilon \ x \approx \theta'_0 x + \theta'_1 (L - I) x = \theta'_0 x - \theta'_1 D^{-\frac{1}{2}} A D^{-\frac{1}{2}} x, \tag{6}$$

In Eq. (6), θ'_0 and θ'_1 indicates two free parameters. Because decreasing the parameter count assists to prevent overfitting,

$$g_{\theta} \ \varepsilon \ x \approx \theta \left(I + D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \right) x, \tag{7}$$

By letting $\theta = \theta'_0 = -\theta'_1$. As I $+D^{-\frac{1}{2}} AD^{-\frac{1}{2}}$ has the eigenvalue range within $[0, 2]$, frequently employing these operators will lead to vanishing or exploding gradient and numerical instabilities. To resolve these shortage, the re-normalization trick $I + D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \rightarrow \tilde{D}^{-} - \frac{1}{2} \tilde{A} \tilde{D}^{-\frac{1}{2}}$ with $\tilde{A}$ $= A + I$ and $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$. Consequently, the convolutional function of GCN method is formulated by Eq. (8)

$$H^{(l)} = \sigma\left(\tilde{A} H^{(l-1)} W^{(l)} \right), \tag{8}$$

Now $H^{(l)}$ means the output of l^{th} layer, σ characterizes an activation function, namely softplus function applied in CAD-GCN, and $W^{(l)}$ shows trainable weight matrixes included in the l-th layer.

3.3. AOSO based Hyperparameter Prediction

Finally, the DLAPP-SEDP technique makes use of AOSO algorithm for optimal hyperparameter tuning process. The AOS is a recently designed optimization algorithm that is stimulated from the laws of quantum technicians whereby the standard arrangement of electrons around the nucleus (Azizi *et al.*, 2021). The AOS can be mathematically expressed in the following.

This study makes use of various solutions (X) as follows, and every solution (X) hold different decision parameters $(x_{i,j})$.

$$X = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_i \\ \vdots \\ X_N \end{bmatrix} = \begin{bmatrix} x_1^1 & x_1^2 & \cdots & x_1^j & \cdots & x_1^D \\ x_2^1 & x_2^2 & \cdots & x_2^j & \cdots & x_2^D \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_i^1 & x_i^2 & \cdots & x_i^j & \cdots & x_i^D \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_N^1 & x_N^2 & \cdots & x_N^j & \cdots & x_N^D \end{bmatrix}, \ i = 1,2,\ldots,N, j = 1,2,\ldots,D, \tag{9}$$

In Eq. (9), N signifies the used amount of solutions, and D specify the dimension length of the tested problem.

The initial solution is initialized arbitrarily in the following.

$$x_i^j = x_{i,\min}^j + rand \times \left(x_{i,\max}^j - x_{i,\min}^j \right), \tag{10}$$

In Eq. (10), x_i^j indicates the i-th location in the j-th solution, $x_{i,\min}^j$ and $x_{i,\max}^j$ specifies the lower and upper bounds of the i-th and j-th position.

A vector of energy value comprises the objective function of dissimilar solutions as given below.

$$E = \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_i \\ \vdots \\ E_m \end{bmatrix}, \tag{11}$$

In Eq. (11), E embodies a vector of objective value, and E_i denotes the energy level of solution i-th number.

The electron probability density chart describes solution position assessed by the Probability Density Function (PDF). Based on the particular definition of the individual by PDF, every imaginarily formulated layer comprises numerous solutions. With that regard, the mathematical models of the K_k position and E_k of the individual used n imaginary course is shown in the following:

$$XX = \begin{bmatrix} X_1^k \\ X_2^k \\ \vdots \\ X_i^k \\ \vdots \\ X_p^n \end{bmatrix} = \begin{bmatrix} x_1^1 & x_1^2 & \cdots & x_1^j & \cdots & x_1^d \\ x_2^1 & x_2^2 & \cdots & x_2^j & \cdots & x_2^d \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_i^1 & x_i^2 & \cdots & x_i^j & \cdots & x_i^d \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_p^1 & x_p^2 & \cdots & x_p^j & \cdots & x_p^d \end{bmatrix}, i = 1,2,3,\ldots, N, j = 1,2,3,\ldots,D, \quad (12)$$

$$E^k = \begin{bmatrix} E_1^k \\ E_2^k \\ \vdots \\ E_i^k \\ \vdots \\ E_p^n \end{bmatrix}, k = 1,2,\cdots,n, \quad (13)$$

Now X_i^k indicates the i - th number of solutions in the k - th IL number, and n signifies the amount of the generated IL. p show the solution number of k - th IL number. E_i^k is the objective value of i - th number of solutions in the k - th IL number.

With that regard, the requisite energy and state are determined for the solution in every IL by examining each solution's average position and objective value and it is mathematically expressed in the following:

$$BS^k = \frac{\sum_{i=1}^{p} X_i^k}{p} \quad (14)$$

$$BE^k = \frac{\sum_{i=1}^{p} E_i^k}{p} \quad (15)$$

Here, BS^k and BE^k denotes the requisite state and energy of the layer number k, respectively X_i^k and E_i^k stand for the position and fitness value of the solution number i in k - th layer.

Based on the presented item, the required energy and state of atom are described by approximating the mean position and objective value of the solution used:

$$BS = \frac{\sum_{i=1}^{m} X_i}{m} \quad (16)$$

$$BE = \frac{\sum_{i=1}^{m} E_i}{m} \quad (17)$$

Let BS and BE be the requisite state and energy of the atom.

The energy level (E_i^k) of tX_i^k in every IL is related to the requisite energy of layer (BE^k). Assume the energy ratio of existing solution in a specific layer is higher than the requisite energy (viz ., $E_i^k \geq BE^k$) hence, the photon emission is assessed. In these rules, the individuals are handling to transfer a photon with a cost of energy assessed by γ and β to simultaneously provide the requisite location of the atom (BS) and the location of electron with the lowest energy ratio (LE) in the atom and it is shown below:

$$X_{i+1}^k = X_i^k + \frac{\alpha_i(\beta_i \times LE - \gamma_i \times BS)}{k}, \ k = 1,2,\ldots, n, \ i = 1,2,\ldots,p \quad (18)$$

In the above expression, X_i^k and X_{i+1}^k denotes the present and estimated values for i - th individuals at k - th layers. α_i, β_i, and γ_i denotes random vector.

Supposing the energy ratio of solution in a specific layer is small when compared to the requisite energy ($E_i^k < BE^k$); then photon consumption is inspected and mathematically expressed in the following:

$$X_{i+1}^k = X_i^k + \alpha_i \times \left(\beta_j \times LE^k - \gamma_j \times BS^k\right) \quad (19)$$

While producing a random number ($\varnothing$) for all the individuals and it is valued lesser than PR (*i.e.*, $\varnothing < PR$) , the photon number on the solution isn't possible. As a result, the action of particles among different layers near the nucleus is assessed:

$$X_{i+1}^k = X_i^k + r_i \quad (20)$$

In Eq. (20), r_i indicates a vector of arbitrary numbers.

4. Experimental Validation

The proposed model is simulated using Python 3.6.5 tool on PC i5-8600k, GeForce 1050Ti 4GB, 16GB RAM, 250GB SSD, and 1TB HDD. The parameter settings are given as follows: learning rate: 0.01, dropout: 0.5, batch size: 5, epoch count: 50, and activation: ReLU. In this section, the air pollution prediction outcomes of the DLAPP-SEDP model are examined in detail. Table 1 provides an overall prediction performance of the DLAPP-SEDP model under day 1 and day 7 with varying batches.

Figure 3 reports an average MAE inspection of the DLAPP-SEDP model under day 1 and day 7. The figure implied that the DLAPP-SEDP model has attained reduced values of MAE under all aspects. For instance, on batch 8, the DLAPP-SEDP model has attained MAE of 4.604 and 4.270 under days 1 and 7 respectively. Similarly, on batch 16, the DLAPP-SEDP approach has gained MAE of 3.492 and 4.260 under days 1 and 7 correspondingly. Also, on batch 24, the DLAPP-SEDP technique has gained MAE of 5.203 and 3.257 under days 1 and 7 correspondingly. Finally, on batch 32, the DLAPP-SEDP approach has reached MAE of 5.362 and 4.692 under days 1 and 7 correspondingly.

Table 1. Result analysis of DLAPP-SEDP approach with distinct measures and runs

No. of Runs	Batch	Day-1				Day-7			
		MAE	MSE	RMSE	R2	MAE	MSE	RMSE	R2
Run-1	8	4.604	103.565	10.177	98.740	4.270	86.218	9.285	99.400
	16	3.492	97.289	9.864	98.760	4.260	92.555	9.621	98.790
	24	5.203	98.773	9.938	99.270	3.257	92.448	9.615	99.050
	32	5.362	105.203	10.257	99.100	4.692	86.502	9.301	99.000
	64	3.535	93.265	9.657	99.550	3.430	107.045	10.346	98.580
	Average	**4.439**	**99.619**	**9.979**	**99.084**	**3.982**	**92.954**	**9.634**	**98.964**
Run-2	8	5.745	106.900	10.339	99.370	5.364	95.150	9.754	99.370
	16	5.357	91.001	9.539	99.440	5.832	104.636	10.229	99.300
	24	3.902	109.500	10.464	98.590	3.434	97.553	9.877	98.690
	32	4.196	95.837	9.790	98.750	5.570	88.001	9.381	99.230
	64	5.055	82.402	9.078	98.860	5.366	87.120	9.334	98.560
	Average	**4.851**	**97.128**	**9.842**	**99.002**	**5.113**	**94.492**	**9.715**	**99.030**
Run-3	8	5.620	99.039	9.952	99.190	3.693	97.329	9.866	98.850
	16	3.043	104.940	10.244	99.510	4.501	105.386	10.266	99.270
	24	5.875	89.969	9.485	98.920	4.272	82.524	9.084	98.850
	32	4.051	81.976	9.054	99.320	4.800	93.449	9.667	99.489
	64	3.216	82.619	9.089	98.800	3.499	83.046	9.113	98.780
	Average	**4.361**	**91.709**	**9.565**	**99.148**	**4.153**	**92.347**	**9.599**	**99.048**
Run-4	8	4.335	106.156	10.303	99.010	3.533	85.338	9.238	99.200
	16	5.720	104.442	10.220	99.040	3.255	101.554	10.077	99.300
	24	5.107	84.592	9.197	98.860	5.748	88.572	9.411	99.400
	32	5.620	89.323	9.451	98.830	3.849	94.015	9.696	98.520
	64	3.013	101.569	10.078	99.340	4.332	102.148	10.107	98.800
	Average	**4.759**	**97.216**	**9.850**	**99.016**	**4.143**	**94.325**	**9.706**	**99.044**
Run-5	8	4.766	107.160	10.352	98.950	3.596	91.778	9.580	99.140
	16	5.918	107.856	10.385	99.000	3.088	96.884	9.843	98.930
	24	3.681	107.923	10.389	99.660	5.312	82.003	9.056	98.940
	32	4.294	91.552	9.568	98.630	5.413	109.974	10.487	99.160
	64	4.023	101.142	10.057	98.770	5.115	80.397	8.966	98.690
	Average	**4.536**	**103.127**	**10.150**	**99.002**	**4.505**	**92.207**	**9.586**	**98.972**

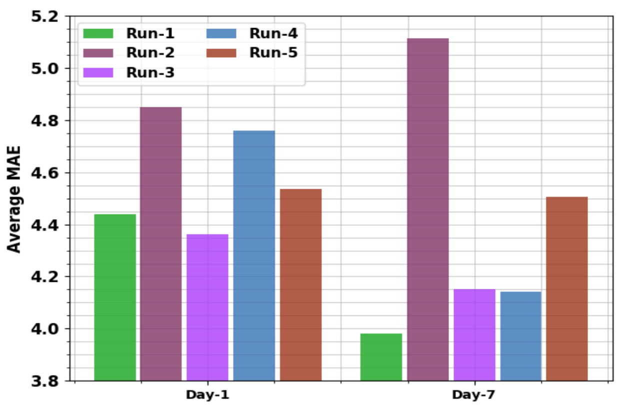


Figure 3. Average MAE analysis of DLAPP-SEDP approach with distinct runs

Figure 4 reports an average MSE analysis of the DLAPP-SEDP method under day 1 and day 7. The figure implied that the DLAPP-SEDP approach has reached reduced values of MSE under all aspects. For example, on batch 8, the DLAPP-SEDP methodology has gained MSE of 103.565 and 86.218 under days 1 and 7 correspondingly. Also, on batch 16, the DLAPP-SEDP algorithm has achieved MSE of 97.289 and 92.555 under days 1 and 7 correspondingly. Likewise, on batch 24, the DLAPP-SEDP technique has achieved MAE of 98.773 and 92.448 under days 1 and 7 correspondingly. At last, on batch 32, the DLAPP-SEDP method has gained MSE of 105.203 and 86.502 under days 1 and 7 correspondingly.

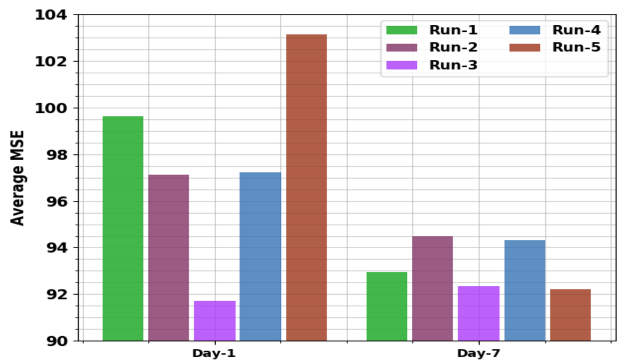


Figure 4. Average MSE analysis of DLAPP-SEDP approach with distinct runs

Figure 5 reports an average RMSE inspection of the DLAPP-SEDP model under day 1 and day 7. The figure implied that the DLAPP-SEDP model has attained reduced values of RMSE under all aspects. For instance, on batch 8, the DLAPP-SEDP model has attained RMSE of 10.177 and 9.285 under days 1 and 7 respectively. Likewise, on batch

16, the DLAPP-SEDP model has attained RMSE of 9.864 and 9.621 under days 1 and 7 correspondingly. Similarly, on batch 24, the DLAPP-SEDP approach has gained RMSE of 99.270 and 9.615 under days 1 and 7 correspondingly. At last, on batch 32, the DLAPP-SEDP algorithm has attained RMSE of 99.100 and 9.301 under days 1 and 7 correspondingly.

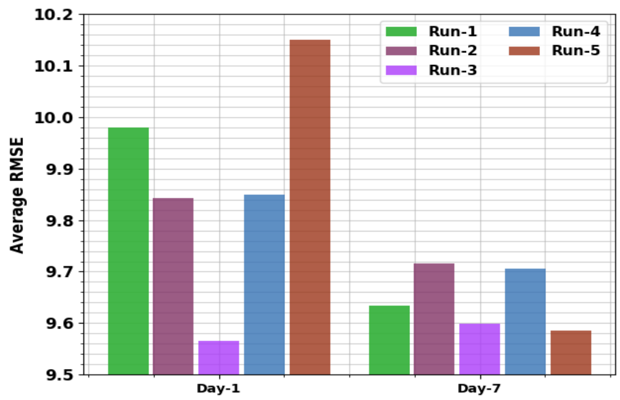


Figure 5. Average RMSE analysis of DLAPP-SEDP approach with distinct runs

An average R2 examination of the DLAPP-SEDP model under varying batches is given in Figure 6. The results inferred that the DLAPP-SEDP model has gained maximum prediction outcomes. For instance, on batch 8, the DLAPP-SEDP model depicted R2 of 98.740 and 99.400 under days 1 and 7 respectively. Besides, on batch 16, the DLAPP-SEDP algorithm has depicted R2 of 98.760 and 99.790 under days 1 and 7 correspondingly. Concurrently, on batch 24, the DLAPP-SEDP method has depicted R2 of 99.270 and 99.050 under days 1 and 7 correspondingly. Simultaneously, on batch 32, the DLAPP-SEDP approach has depicted R2 of 99.100 and 99.000 under days 1 and 7 correspondingly.

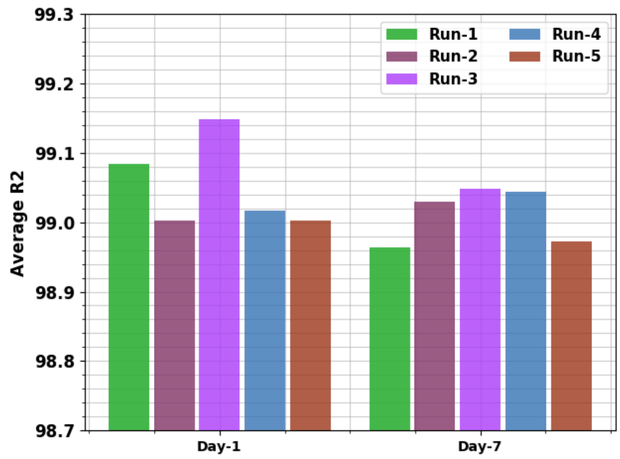


Figure 6. Average R2 analysis of DLAPP-SEDP approach with distinct runs

Table 2 offers a brief comparison predictive outcome of the DLAPP-SEDP model with existing models [18]. Figure 7 exhibits a comparative MAE and RMSE inspection of the DLAPP-SEDP model with recent models on day 1. The figure demonstrated that the DLAPP-SEDP model has shown enhanced performance with minimal MAE and RMSE values. With respect to MAE, the DLAPP-SEDP model has reached minimal MAE of 4.361 whereas the GRU, LSTM, Bi-LSTM, Bi-GRU, CNN, CNN-LSTM, and CNN-GRU models have resulted to maximum MAE of 9.841, 9.396, 9.216, 9.505, 9.622, 8.187, and 9.551 respectively. Likewise, With respect to RMSE, the DLAPP-SEDP approach has attained minimal RMSE of 9.565 whereas the GRU, LSTM, Bi-LSTM, Bi-GRU, CNN, CNN-LSTM, and CNN-GRU algorithms have resulted in maximum RMSE of 16.826, 16.314, 16.036, 16.453, 17.087, 15.385, and 17.344 correspondingly.

Table 2. Comparative analysis of DLAPP-SEDP approach with existing algorithms

Methods	Day-1			Day-7		
	MAE	RMSE	R2	MAE	RMSE	R2
DLAPP-SEDP	4.361	9.565	99.148	4.153	9.599	99.048
GRU	9.841	16.826	97.800	12.020	19.751	96.600
LSTM	9.396	16.314	98.000	11.728	19.198	97.100
Bi-LSTM	9.216	16.036	98.100	11.816	18.994	97.000
Bi-GRU	9.505	16.453	98.000	11.664	19.138	97.000
CNN	9.622	17.087	97.800	10.785	18.409	97.400
CNN-LSTM	8.187	15.385	98.300	9.327	17.062	97.800
CNN-GRU	9.551	17.344	97.800	9.743	18.251	97.100

A detailed R2 assessment of the DLAPP-SEDP model with other models is made in Figure 8. These results indicated the betterment of the DLAPP-SEDP model with higher R2 value of 99.148. At the same time, the other existing models such as GRU, LSTM, Bi-LSTM, Bi-GRU, CNN, CNN-LSTM, and CNN-GRU models have provided decreased R2 values of 97.800, 98.000, 98.100, 98.000, 97.800, 98.300, and 97.800 respectively.

Figure 9 displays a detailed MAE and RMSE analysis of the DLAPP-SEDP approach with recent algorithms on day 7. The figure established that the DLAPP-SEDP technique has shown enhanced performance with minimal MAE and RMSE values. With respect to MAE, the DLAPP-SEDP approach has gained minimal MAE of 4.153 whereas the GRU, LSTM, Bi-LSTM, Bi-GRU, CNN, CNN-LSTM, and CNN-GRU techniques have resulted in maximum MAE of

12.020, 11.728, 11.816, 11.664, 10.785, 9.327, and 9.743 correspondingly. Also, with respect to RMSE, the DLAPP-SEDP algorithm has attained minimal RMSE of 9.599

whereas the GRU, LSTM, Bi-LSTM, Bi-GRU, CNN, CNN-LSTM, and CNN-GRU approaches have resulted in maximum RMSE of 19.751, 19.198, 18.994, 19.138, 18.409, 17.062, and 18.251 correspondingly.

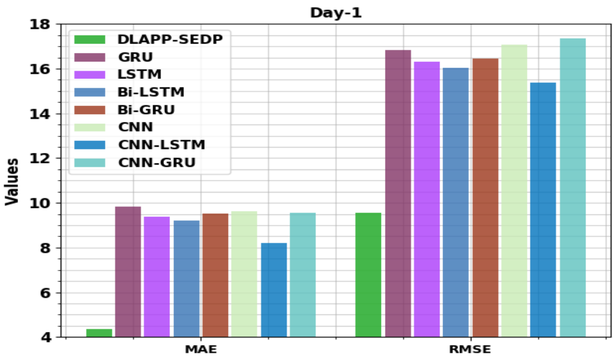


Figure 7. MSE and RMSE analysis of DLAPP-SEDP approach with existing algorithms under day 1

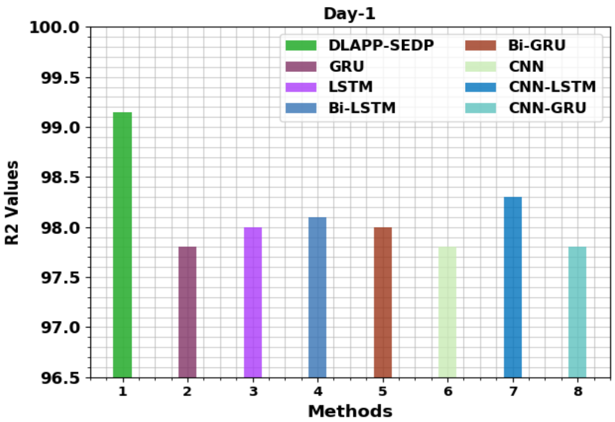


Figure 8. R2 analysis of DLAPP-SEDP approach with existing algorithms under day 1

A comprehensive R2 assessment of the DLAPP-SEDP approach with other models is made in Figure 10. These results indicated the betterment of the DLAPP-SEDP method with higher R2 value of 99.048. In the meantime, the other existing methodologies such as GRU, LSTM, Bi-LSTM, Bi-GRU, CNN, CNN-LSTM, and CNN-GRU approaches have offered decreased R2 values of 96.600, 97.100, 97.000, 97.000, 97.400, 97.800, and 97.100 correspondingly.

Table 3 and Figure 11 provides actual vs prediction outcomes of the DLAPP-SEDP model under several time step. The experimental values indicated that the DLAPP-SEDP model has effectually predicted the PM2.5 values. For instance, on 20 time step and actual value of 90.49, the DLAPP-SEDP model has attained predicted value of 88.73.

Similarly, on 40-time step and actual value of 71.94, the DLAPP-SEDP method has achieved predicted value of 64.94. In addition, on 60-time step and actual value of 14.08, the DLAPP-SEDP approach has achieved predicted value of 6.89. Also, on 80-time step and actual value of 53.39, the DLAPP-SEDP technique has gained predicted value of 57.65. At last, on 100-time step and actual value of 146.13, the DLAPP-SEDP approach has achieved predicted value of 140.95. These values assured the enhanced air quality prediction performance of the DLAPP-SEDP model.

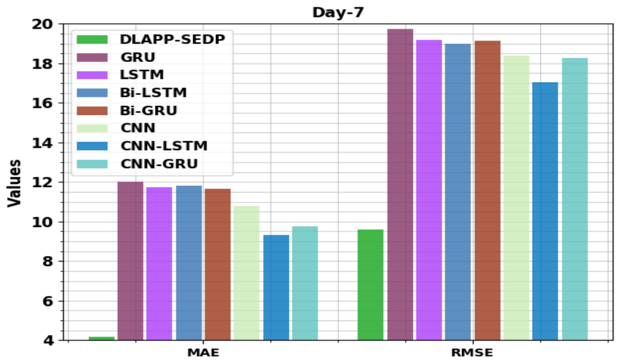


Figure 9. MSE and RMSE analysis of DLAPP-SEDP approach with existing algorithms under day 7

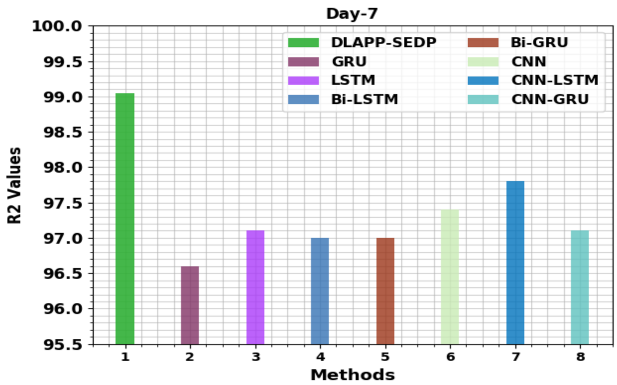


Figure 10. R2 analysis of DLAPP-SEDP approach with existing algorithms under day 7

Table 3. PM2.5 analysis of DLAPP-SEDP model under several time steps

	PM 2.5	
Time Step	Actual	Predicted
0	5.17	14.03
20	90.49	88.73
40	71.94	64.94
60	14.08	6.89
80	53.39	57.65
100	146.13	140.95
120	2.95	2.54
140	83.07	79.46
160	20.75	25.32
180	27.43	32.61
200	91.97	94.74
220	66.75	61.25
240	47.46	45.32
250	38.56	47.33

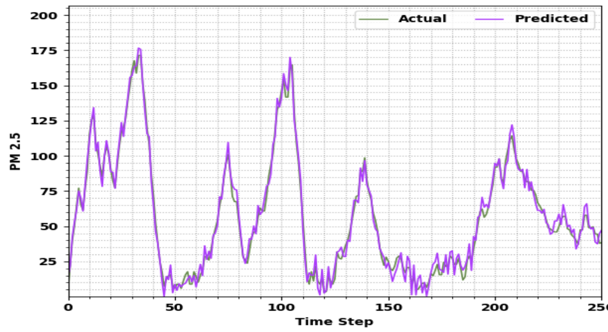


Figure 11. PM2.5 analysis of DLAPP-SEDP model under several time steps

5. Conclusion

In this paper, a novel DLAPP-SEDP technique has been introduced to predict the level of air pollution in the smart environment. It follows a three-stage process namely data pre-processing, air pollution prediction, and hyperparameter tuning. At the initial stage, the presented DLAPP-SEDP technique performs various levels of data pre-processing such as missing value replacement, categorical value encoding, normalization, and feature selection. In the next stage, the DLAPP-SEDP technique applied the GCN model. In the last stage, the DLAPP-SEDP technique makes use of AOSO algorithm for optimal hyperparameter tuning process. To demonstrate the enhanced predictive efficiency of the DLAPP-SEDP technique, a wide-ranging experimental analysis is carried out. The experimental values assured the enhancements of the DLAPP-SEDP technique over other recent approaches. In future, the proposed model can be extended to the IoT enabled air pollution monitoring system in real time.

References

Abdellatif B., Badr H., Samira D. and Khadija D. (2021). Air-pollution prediction in smart city, deep learning approach. *Journal of Big Data*, ***8***(1).

Al-Janabi S., Mohammad M. and Al-Sultan A. (2020). A new method for prediction of air pollution based on intelligent computation. *Soft Computing*, ***24***(1), 661–680.

Ameer S., Shah M.A., Khan A., Song H., Maple C., Islam S.U. and Asghar M.N. (2019). Comparative analysis of machine learning techniques for predicting air quality in smart cities. *IEEE Access*, ***7***, 128325–128338.

Azizi M., Talatahari S. and Giaralis A. (2021). Optimization of engineering design problems using atomic orbital search algorithm. *IEEE Access*, ***9***, 102497–102519.

Castelli M., Clemente F.M., Popovič A., Silva S. and Vanneschi L. (2020). A machine learning approach to predict air quality in California. *Complexity*, *2020*.

Chang Y.S., Chiao H.T., Abimannan S., Huang Y.P., Tsai Y.T. and Lin K.M. (2020). An LSTM-based aggregated model for air pollution forecasting. *Atmospheric Pollution Research*, ***11***(8), 1451–1463.

Du S., Li T., Yang Y. and Horng S.J. (2019). Deep air quality forecasting using hybrid deep learning framework. *IEEE Transactions on Knowledge and Data Engineering*, ***33***(6), 2412–2424.

Harishkumar K.S., Yogesh K.M. and Gad I. (2020). Forecasting air pollution particulate matter (PM2. 5) using machine learning regression models. *Procedia Computer Science*, ***171***, 2057–2066.

Heydari A., Majidi Nezhad M., Astiaso Garcia D., Keynia F. and De Santoli L. (2022). Air pollution forecasting application based on deep learning model and optimization algorithm. *Clean Technologies and Environmental Policy*, ***24***(2), 607–621.

Iskandaryan D., Ramos F. and Trilles S. (2020). Air quality prediction in smart cities using machine learning technologies based on sensor data: a review. *Applied Sciences*, ***10***(7), 2401.

Janarthanan R., Partheeban P., Somasundaram K. and Elamparithi P.N. (2021). A deep learning approach for prediction of air quality index in a metropolitan city. *Sustainable Cities and Society*, ***67***, 102720.

Kabir S., Islam R.U., Hossain M.S. and Andersson K. (2020). An integrated approach of belief rule base and deep learning to predict air pollution. *Sensors*, ***20***(7), 1956.

Kalajdjieski J., Zdravevski E., Corizzo R., Lameski P., Kalajdziski S., Pires I.M., Garcia N.M. and Trajkovik V. (2020). Air pollution prediction with multi-modal data and deep neural networks. *Remote Sensing*, ***12***(24), 4142.

Le V.D., Bui T.C. and Cha S.K. (2020). February. Spatiotemporal deep learning model for citywide air pollution interpolation and prediction. In *2020 IEEE international conference on big data and smart computing (BigComp)* (55–62). IEEE.

Ma J., Cheng J.C., Lin C., Tan Y. and Zhang J. (2019). Improving air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques. *Atmospheric Environment*, ***214***, 116885.

Masih A. (2019). Machine learning algorithms in air quality modeling. *Global Journal of Environmental Science and Management*, ***5***(4), 515–534.

Rao K.S., Devi G.L. and Ramesh N. (2019). Air quality prediction in Visakhapatnam with LSTM based recurrent neural networks. *International Journal of Intelligent Systems Applications*, ***11***(2), 18–24.

Sai K.B.K., Subbareddy S.R. and Luhach A.K. (2019). IOT based air quality monitoring system using MQ135 and MQ7 with machine learning analysis. *Scalable Computing: Practice and Experience*, ***20***(4), 599–606.

Sofianos T., Sampieri A., Franco L. and Galasso F. (2021). Space-time-separable graph convolutional network for pose forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (11209–11218).

Ullo S.L. and Sinha G.R. (2020). Advances in smart environment monitoring systems using IoT and sensors. *Sensors*, ***20***(11), 3113.