INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

OPEN ACCESS

Received: 16-04-2024

Accepted: 01-07-2024

Published: 22-07-2024

Time Series Analysis for Air Pollution Prediction in High-Intensity Development Areas using Deep Learning

K V K Sasikanth[1]*, **B Sujatha**[2], **D Haritha**[3]

1 Research Scholar, JNTUK, Kakinada
2 Professor, Dept. of CSE, Godavari Institute of Engineering & Technology (A), Rajahmundry
3 Professor, Dept. of CSE, University college of Engineering, Jawaharlal Nehru Technological University Kakinada, Kakinada

Citation: Sasikanth KVK, Sujatha B, Haritha D (2024) Time Series Analysis for Air Pollution Prediction in High-Intensity Development Areas using Deep Learning. Indian Journal of Science and Technology 17(28): 2903-2913. https://doi.org/10.17485/IJST/v17i28.1117

* **Corresponding author**.

sasikanthkolanu@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objective: The purpose of this study is to design novel approach to predict air pollution indices accurately using hybrid approach in highly populated cities of the world. **Methods:** This study aimed to address the significant bottleneck of relying on a single model for time series analysis which we accomplished by developing a hybrid forecasting method. This blended method combines the advantages of CNN (Convolutional Neural Networks) and LSTM (Long Short Term Memory) hybrid model together, overcoming insufficiency of air pollution prediction. **Findings:** Due to the empirical operation that revealed in achieving proper predictions we have come to understand that relying on a single model for the estimation of time-series data results is inefficient and misleading, mostly due to the non-linear nature of raw data. On the other hand, by using this hybrid model forecasting, we can uncover more detailed patterns through longer period of time and predict them more accurate by using weather and air pollution related data.The hybrid CNN-LSTM Model achieved least values of 6.742 for MAE,12.921 for RMSE compared to other approaches, which indicatesmere perfection. **Novelty:** The novelty of our approach is to fuse LSTM and CNN models in order to perform exhaustive analysis on the dynamics of the air pollution throughout the entirety of the city. Through the combination of these two models, such a tool is made simpler to use, and there is an improved accuracy in predictions. This thus becomes a promising tool in the fight against air pollution in cities that have a very high population just like Delhi.

Keywords: Air Pollution; LSTM; CNN; Deep Learning; Time Series Analysis; Nonlinear Relationships

1 Introduction

Pollutants include ozone, nitrogen oxides, volatile organic compounds, Sulphur oxides, particulate matter and heavy metals. Studies have shown that when a person suffers from cardiovascular (relating to the heart and blood vessels), nerve or lung diseases his/her life is shortened by air pollution which may refer to both indoor and outdoor contamination. Air pollution is the greatest source of Disability-Adjusted Life Years (DALYs)[1]. Yet, climate change, ecosystems and ecologies can be affected in different ways by it. To measure and reduce it, it is necessary to monitor urban air pollution for public health and environmental protection purposes.

Due to fast economic growth, industrialization, urbanization, and rising energy consumption and car use, Indian towns like New Delhi, Varanasi, and Patna have been rated as world's most polluted cities by the World Economic Forum (WEF)[2]. Particularly in Delhi, where PM2.5 and PM10 concentrations are 15 times higher than WHO limits[3]. The number of people diagnosed with asthma in Delhi has risen dramatically, and the city urgently needs a strategy to lessen the health risks posed by air pollution[4]. Health effects from exposure to air pollution are shown in Table 1. Globally, governments and communities are worried about air pollution's effects on human health and sustainable development[5]. While developed countries have made substantial efforts to control their pollution levels, developing countries are facing a more significant challenge due to their exponential growth. The new global climate change pact known as ”The Paris Agreement” demonstrates that countries throughout the globe are working together to combat climate change.

The objective of this paper is to create a combined forecasting model that can analyze the meteorological and air pollution data, capture complex patterns over extended time periods, and make more accurate predictions of air pollution levels in highly populated cities like Delhi.

CNN's find the meaning of regions related to the degree of pollution in various locations while LSTM networks model the way the pollution level fluctuates with time. Accurate and reliable forecasting requires understanding when and where pollution will take place, achieved through a combination of these two techniques in making air quality predictions.

Table 1. Delhi's air pollution has serious health consequences

Study and year	Variable	Findings
Becerra-Rico J et.al., 2020[6]	Impacts of traffic-related air pollution on children	According to a study, there is a direct link between the levels of ambient PM10 and the occurrence of ADHD in children. The study found that the odds of having ADHD were 2.07 times higher when the levels of PM10 were increased. The 95% CI for the OR was between 1.08 and 3.99, indicating a statistically significant association between the two variables.
Ayturan YA et al., 2020[7]	Outdoor air	The cumulative probability of dying from all causes rises by 0.15 percentage points for every 10 μg/m3 increase in PM10.
Mani G et al., 2021[8]	Indoor air pollution	Suspended particle pollution, nitrogen dioxide, and sulphur dioxide are all linked to respiratory issues in children. Furthermore, homes with a smoking history tend to exhibit higher concentrations of these pollutants, thereby exacerbating the adverse effects on children's respiratory health.
Md Net al., 2020[9]	Indoor air	During the harsh winter season, the amounts of gaseous pollutants indoor increased to the extent they exacerbate respiratory problems in women and children. According to this corollary, pollution of indoor air can cause some negative effect on women in the reproductive period and include children, as well.
Amado TM, 2018[10]	Outdoor air	Findings confirmed a 10 g/m3 rise in pollutants was positively related to 1.004(similar to 4%) for NO2, 1.033(3%) for O3, and 1.006(0.06%) for RSPM relative-risk level website link, meaning small increments in pollution may affect adversely on respiratory
Shishegaran Aet al., 2020[11]	Outdoor air	During the winter, high levels of airborne pollution within the home have been linked to respiratory issues in women and children.
Danesh Yazdi M et al., 2020[12]	Indoor air pollution	There were considerably greater levels of Suspended Particulate Matter (SPM) inside, which has been linked to health problems in youngsters.
Londhe Met al., 2021[13]	Outdoor air	Winter months were found to be the risky times for the development of COPD suggesting that there were factors that influenced the likelihood of getting respiratory disease were season-related.
Conticini Eet al., 2020[14]	Outdoor air	Exposure to elevated air pollution can have negative health consequences and may lead to an increased risk of respiratory and cardiovascular problems.

1.1 Data- Delhi – Air Pollution

By 2030, Delhi might have as many as 36 million residents, which would increase pressure on the city's already poor air quality and health infrastructure[15]. Since 2000, there has been a significant growth in the number of automobiles on the road, which has contributed significantly to air pollution[16]. From January to September, Delhi's air quality index is typically moderate, but from October to December, it sharply declines to extremely bad or dangerous levels for a number of reasons. Air pollution levels are thought to be affected by factors including regional and synoptic meteorology, which are taken into account in this research. The monitoring stations in the Delhi-National Capital Region (NCR)[17] were established by a government organization, and the dataset includes the concentration levels of 12 distinct metrics of air pollution measured at those sites. High levels of PM2.5 and PM10 particles, exceeding permissible limits, have caused an air pollution crisis in Delhi. Exceedance is the concentration of a pollutant when it exceeds the predefined standards of air quality. The Exceedance factor (EF) can help categorize pollutants as low, moderate, high, or critical based on their violation of prescribed standards, giving a more holistic view of the issue[18].

According to the available statistics from the past 4 years of data available for 365 days from 2016-2019, just one percent of days in Delhi had acceptable air quality. In most cases, the wind during the rainy season is strong enough to carry away the pollutants, making those days ideal. Figure 1, represents the daily AQI in Delhi during the period 2016-2019.Real-time monitoring data plays animportant role in assessing the AQI(Air Quality Index), which considers multiple pollutants apart from PM2.5 for the analysis.

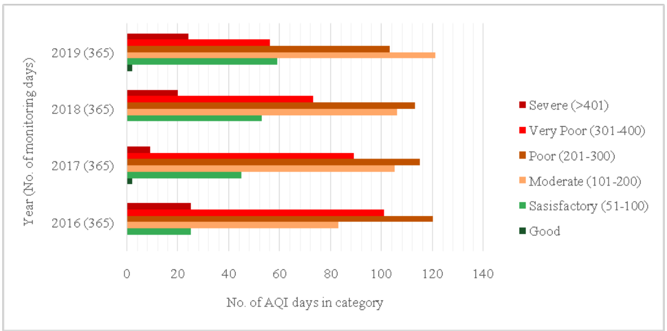

Fig 1. Delhi - Daily AQI (2016-2019)

*1.1.1 **AQI (Air Quality Index) in Delhi***

The AQI is a tool for tracking pollution levels in major cities throughout the United States. Since the AQI is calculated by looking at how various air contaminants affect people, it may not be suitable for environmentally delicate areas. The index considers a number of contaminants in order to provide an assessment of the air's qualityincludingPM10, PM2.5,CO, SO2, O3, NO2, and NH3. By keeping an eye on the AQI, we can learn about the dangers of breathing polluted air over the long term and take steps to improve the situation. In their article, "Air Pollution in the Capital: Concerns and the Approaches Taken to Ab-Atom," Sulian et al.[19] investigate the current state of air pollution in Delhi and its implications on the policies designed to tackle it along with people's health. Color-coded bands are used to make it easy for members of the public to comprehend and visualize the situation depicted in Table 2 through simple representation.

2 Methodology

2.1 Preprocessing

These steps cover grouping the data station wise, removing unneeded attributes, dealing with missing values, utilizing weighted classification if there is an absence of samples, scaling the features, creating training and test datasets and thus preparing air pollution data for the machine learning study. This guarantees the data to be truthful, full, and agreeable proceeding also to the conclusions and the decisions made on reliable information. Adaptive Synthetic Sampling (ADASYN) is a data-preprocessing method, and it helps to solve the imbalanced data case, too. Specifically, when it comes to monitoring stations, certain ones could be underrepresented in the dataset and when the topic is air pollution data analysis, ADASYN can be of big help. The pseudo-code for the ADASYN algorithm in recommendation system is described in Algorithm 1.

Algorithm 1. Pseudocode for the ADASYN

Table 2. Values of the Air Quality Index and the Potential Health Effects of Exposure to High Ambient Concentrations

AQI Class	Range	Health Impact	Concentration Range: 24-hour (mg/m^3)						
			PM2.5	PM10	SO $_2$	NO $_2$	O $_3$	CO	NH $_3$
Good	(0, 50)	Minimal	0-50	0-30	0-40	0-40	0-50	0-1	0-200
Satisfactory	(51, 100)	Minor- Breathing	50-100	31-60	41-80	41-80	51-100	1.1-2	201-400
Moderately Polluted	(101, 200)	Individuals with respiratory problems experience difficulty breathing.	101-250	61-90	81-380	81-180	101-168	2.1-10	401-800
Poor	(201, 300)	Breathing discomfort to the people, on prolonged exposure	251-350	91-120	381-800	181-280	169-208	10-17	801-1200
Very Poor	(301, 400)	Individuals who are exposed to harmful pollutants for an extended period are at risk of developing respiratory illnesses.	351-430	121-250	801-1600	281-400	209-748	17-34	1200-1800
Severe	(401, 500)	Extended exposure to harmful pollutants causes respiratory ailments in individuals.	250+	430+	1600+	400+	748+*	34+	1800+

```
Input: The first column of matrix X consists of features as the target vector is y.
Output: X_resample: resampling of the feature matrix and y_resample: resampling of the target vector.
Parameters: N: number of synthetic examples to be generated, k: number of nearest neighbors to choose, beta: threshold for density distribution formula.,
Estimate r = 1 is the total number of all minority class samples / no, of samples in the majority class. That of minorities which are represented usually by not many samples.
Find the minority class samples: X_min = X[y is minority], y_min = y[y is minority].
Compute the number of synthetic examples to generate for each minority class sample:
G = (1 / r - 1) * N
Initialize a list to hold the synthetic examples: X_syn = [ ], y_syn = [ ] For each minority class sample ($x_i$, $y_i$) in (X_min, y_min):
5.1 Find the knn of $x_i$ in X:
indices = indices_of_knn (xi, X, k)
5.2 Compute the density distribution of the minority class at xi:
dist = distance_from_$x_i$_to_minority_class_samples (indices, X_min)
density = dist / sum(dist)
5.3 Compute the required number of synthetic examples to generate for xi:
$g_i$= G * density[i]
5.4 Generate $g_i$ synthetic examples for $x_i$:
syn_$x_i$ = generate_synthetic_examples (xi, indices, gi, beta)
5.5 Append the synthetic examples to X_syn and y_syn:
X_syn. append (syn_$x_i$)
y_syn. append ($y_i$)
Combine the original minority class samples with the synthetic examples:
X_resampled = concatenate (X_min, X_syn)
y_resampled = concatenate (y_min, y_syn)
Return the resampled feature matrix and target vector:
return X_resampled, y_resampled
```

2.2 Deep Learning Models

Human health and the health of the environment are both negatively impacted by air pollution. Predicting air pollution levels allows for preventative steps to be taken, lessening the toll that pollution has on people and the planet.

Moreover, the current accelerations concerning the use of DL (Deep Learning) models to predict air contamination has been observed, especially CNNs and LSTM networks. CNNs belonging to the category of deep neural networks are considered by the researchers as the best-known tools of image recognition. The link between time series analytics, such as forecasting pollutant, can be another prospect. The power of CNNs resides in their ability to discover essential features directly from the raw data (the spatial and time structures of air quality data), which is due to their inherent capacity to automatically extract features from unprocessed data. The sequences are seamlessly captured by RNN models like LSTM networks. This is one reason why LSTM networks are good actually because they can pick out long-term dependencies and sequence data of them. LSTM networks are capable of an in-depth learning of the underlying data patterns, e.g. seasonality trends and spike rates.

DL models as CNNs and LSTMs possess the unique feature of being able to precisely represent the atmospheric conditions which air quality is subjected to. This in turn boosts the overall performance of the air pollution prediction. Investment with better data and more precise reduction measures can be made as these models predict air pollution levels quickly and accurately.

Convolution Neural Networks – A CNN typically has to anyone three stages of operation and the outcome may be prediction or classification which is shown in the Figure 2. In a neural network, it all kicks off with the first layer, referred to as the input layer, and it is concerned with the data input and the subsequent stages. The hidden layers the second stage consists of the ones corresponding to the feature maps required in the process of learning and classification, and they are known as convolutional layers. It mirrors the way the visual cortex of an animal is laid out, which is why the work was done that shaped the deep learning algorithm. Convolving layers are not the only components of CNN architectures which include fully connected layers and pooling layers together with the convolutional layers. Convolutional layers are generally followed by pooling layers to reduce the filter map size to a compressed version only holding the most significant bits. To classify input data using characteristics learned in the preceding convolutional and pooling layers, fully connected layers are often placed at the network's output nodes.

The 1D-CNN architecture consists of a total of 11 layers shown in Table 3. Each of the three convolutional layers in this architecture is followed by a max-pooling layer to achieve dimensionality reduction, yielding feature maps. Next, a flattening layer makes the data into a one-dimensional array, and then two completely linked layers combine all the information from the preceding levels into a single vector. By using this route, the network may successfully identify relevant elements within the input data and utilize them to generate reliable predictions in the output phase.

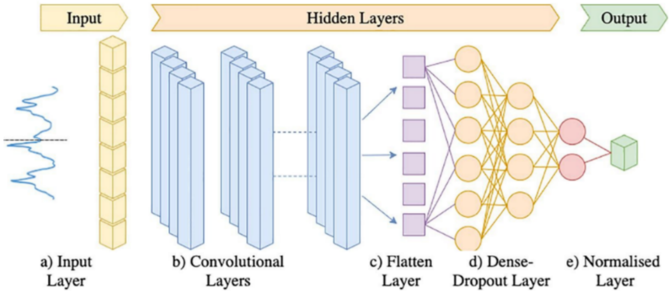

Fig 2. 1D-CNN Architecture

Long- and Short-Term Memory – In order to model sequential datalike LSTM, time series. LSTM is a popular RNN(Recurrent Neural Network) architecture. When compared to typical RNNs, LSTMs are superior because they employ a memory cell and three gates (input, forget, and output) to selectively retain and update information over time, thereby avoiding the vanishing gradient issue and making it easier to capture long-term relationships. Table 4 represents the parameters used for the LSTM structure.

2.3 Proposed Hybrid CNN-LSTM Model

LSTM networks and CNNs are popular and useful techniques in deep learning models, particularly for handling time series data. CNNs are fantastic at collecting significant characteristics from data and improving accuracy and effectiveness of the final prediction model, whereas LSTM networks excel at gathering sequential schema information. A typical CNN may face difficulties in processing such dependencies effectively. We suggested a hybrid model that uses a CNN and a LSTM in neural

Table 3. 1D-CNN Architecture layers description

Layer	Parameters	Output Size	Range	Value	Type
Input	1 × I1 × 10	–	–	49	–
Conv1-1D-ReLU	d1	1 × I1 × d1	[4, 64]	34	Int.
MaxPool-1	1×2	–	–	–	–
Conv2-1D-ReLU	d2	1 × I1 /2 × d2	[4, 64]	30	Int.
MaxPool-2	1×2	–	–	–	–
Conv3-1D-ReLU	d3	1 × I1 /4 × d2	[4, 64]	10	Int.
MaxPool-3	1×2	–	–	–	–
Flatten	–	1 ×I1 /8 × d2	–	–	–
FC-1	h1	h2	[0, 100]	110	Int.
Dropout-1	p_d	–	[0.3, 0.7]	0.63	Cont.
FC-2	h1	h2	[0, 100]	100	Int.
SoftMax	2	–	–	–	–
Value of Learning Rate	lr	–	–	–	10^{-6}
Best optimizer	Opt	–	–	–	Adam

Table 4. Notations of the variables in LSTM [20]

Notation	Formulae
Inputs: $x_{t\,,}$ $c_{t-1,}$ h_{t-1}	—
Outputs: c_t, h_t, y_t	$c_t = F_t \otimes C_{t-1} \oplus I_t \otimes (\tau(h_{t\,-1}\,W_c + x_tU_c + b_c))$, $h_t = O_t \otimes \tau(c_t)$
Nonlinearities: σ, τ	Sigmoid and Tanh activation function
Vectors: $\otimes$, $\oplus$	Element-wise multiplication and Element-wise addition
Weights: $U_f, U_i, U_c, U_o, W_f, W_i, W_c, W_o, b_f, b_i, b_c, b_o$	U_f, U_i, U_c, U_o—Recurrent weight vector W_f, W_i, W_c, W_o—Input weight vector b_f, b_i, b_c, b_o—Input bias vector
Gates: F_t, I_t,O_t	$F_t = \sigma\left(h_{t\,-1}\,W_f + x_tU_f + b_f\right)$, $I_t = \sigma\left(h_{t\,-1}\,W_f + x_tU_i + b_i\right)$, $\theta = \left(U_f, U_i, U_c, U_o, W_f, W_i, W_c, W_o, b_f, b_i, b_c, b_o\right]$
Loss Function: θ, y, $\hat{y}$	$argmin_\theta L(\theta) = \sum_{i=1}^{N} loss((y,\hat{y}),\ (MSE) isloss((y,\hat{y}) = \frac{1}{N}\sum_{I=1}^{N}(y_I - \hat{y})^2$

network to solve this problem. The model architecture is shown in Figure 2 ; it contains an input layer, a CNN for extracting features, a LSTM network for sequential information storage, and a fully connected layer. The structural configuration of parameters used for the hybrid CNN-LSTM structure are: convolutional layers - 2 with 64 and 128 filters, pooling layer - 1 with a pool size of 2, LSTM – 1 with 100 units, FC layer – 1 with 32 neurons, output neurons – 1, optimizer for model training – Adam, Activation function – ReLU, Training Set – monthly data set from 2001 to 2018, Testing set – monthly data for the year 2019, stopping condition – model trained for 1000 epochs with early stopping.

The input Layer as a supervised learning issue, reorganizing time-series data entails making predictions based on the value at the previous time step. This allows machine learning algorithms to be efficiently applied to time-series data, resulting in future value predictions that are more accurate.

CNN Layer - It is tasked with gleaning characteristics from the time series data provided as input. This is done by a series of convolutional and pooling attempts. Sliding the filters or kernel on top of the input data and then calculating the dot product at each point is the operative procedure of the convolution process. This function leads to generating a feature map which enables a person or a program to focus on particular features or patterns in the data. By pooling, the features dimensionality is reduced while the most important information is kept. This results in the saving of computing power, by avoiding the need of overfitting. The pooling layer's output is passed bottomwards to the LSTM layer below it where it is processed and stored sequentially.

LSTM Layer - In LSTM layer, the main problem associated with the classic RNNs is their inability to identify time of dependencies that are present in consecutive data. This problem is due to the vanishing gradients which emerge during the backpropagation process. There are three types of gates: layers of the LSTM algorithm include input, output, and forget gates to address this shortcoming. The long-term memory of the LSTM layer is known as ”cell states,” and these gates modulate how data passes from and to such states. The forget gate is tasked with deciding what details from the memory should be forgotten, the input gate job is to decide what data should be included, and the output gate determines the data that should be read out. LSTMs capture input data dependencies through a longer time length by controlling the information flow via these gates.

The structure of layers in a LSTMis created in a way to simplify the processing of sequential input data (x1,x2,...,xk).Each j-th time step in the hidden sequence has two values assigned, the hidden state (hj) and the cell state (cj). These values are maintained for the lifetime of the decoder. The LSTM implementation involves the hidden state to be output at every time step in contrast to the cell state that is responsible for keeping the long-term layer memories. The precedence relationship of the LSTM network is demonstrated in Figure 3 below to reflect a resulting process where intricate patterns and correlations within the input data are identified by manipulating the information at one step to the next.

The recurrent regression is the main technique of an LSTM model that allows the networks several steps ahead depending on what it learned (hj-1,cj-1)) and the current input. This enhances the model's ability to make more accurate time hops in changing patterns in its input data. By recursively transferring information from past time steps throughout the entire LSTM loop, the model can effectively learn and adapt to changing trends in air pollution data.

During training, an optimization technique like ADAM[(21)] is used to fine-tune the LSTM and fully-connected layer parameters. To do this, we need to compute gradients of the loss function with relating to the parameters of the network and then use these gradients to guide our parameter updates towards a loss minimization. To enhance the ability to make predictions of the model, the parameters of the LSTM and fully connected layers are optimized together.

Adam Optimization - Adam algorithm is a deep learning optimization method that is mostly used to refine a network while using it to train the network. Adaptive learning rates are applied to each weight, making this method different from that used in the usual Stochastic Gradient Descent (SGD).

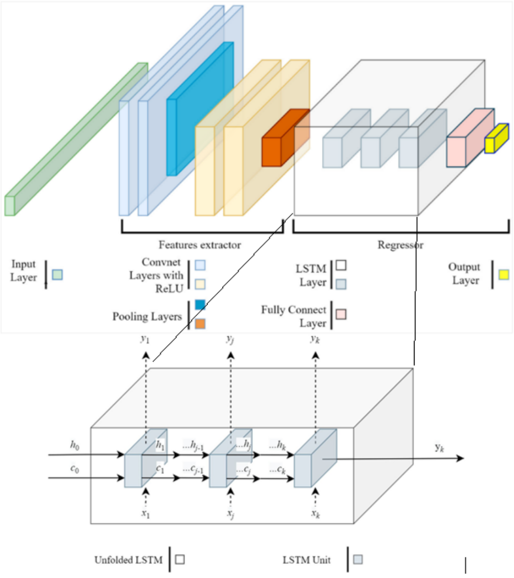

Fig 3. Proposed hybrid model of CNN–LSTM architecture & detailed structure of LSTM

The procedure calculates a moving average of the gradients that is exponentially weighted at two different times: the first instant with its the value (m_t) and second with its magnitude (v^x_t). Gradient and mean and variance of the gradient as at now are used together with moving averages to calculate adaptive learning rate for each of the parameters respectively. The Adam algorithm also consists of a bias correction step that acts to correct the false zero values at the commencing stages. The quality of the inputs being learned during training is improved by this correction, thus, the reliability of the weight updates done during this process is enhanced.

Let's pretend f is the goal function and are the θ tuning parameters. The gradient, denoted by gt in Equation (6), may be written out in greater detail, and further formulae for this section are provided in[22,23].

$$g_t = \nabla_\theta f_t\,(\theta_{t-1})$$

where$f_{i=1\cdots t}\,(\theta)$represent the function values from time step t to t. m_t, v_t stand for first and second moment estimates of the gradient.

$$m_t = \beta_1 \cdot m_{t-1} + (1-\beta_1)\cdot g_t$$
$$v_t = \beta_2 \cdot v_{t-1} + (1-\beta_2)\cdot {g_t}^2$$

where$[\beta_1, \beta_2]$ ranging from [0, 1]. The final, revised parameter formula is as follows:

$$\theta_t = \theta_{t-1} - \alpha \bullet \frac{m_t}{\mathrm{p}v+\epsilon}$$

Where α stands forlearning rate, by default it is set to 0.001. Here is the pseudo code for the Adam optimization algorithm:

```
Algorithm 2: Pseudocode for the Adam optimization
1. Initialize parameters: learning_rate = 0.001, beta_1 = 0.9, beta_2 = 0.999, epsilon = 1e-8
m = 0 (initial mean)
v = 0 (initial variance)
t = 0 (initial time step)
2. For each iteration until convergence:
t = t + 1
Compute the gradient of the loss function with respect to the parameters m = beta_1 * m + (1 - beta_1) * gradient
v = beta_2 * v + (1 - beta_2) * (gradient ** 2)
m_hat = m / (1 - beta_1 ** t) (correct bias in mean)
v_hat = v / (1 - beta_2 ** t) (correct bias in variance)
3. Update the parameters: parameter = parameter - learning_rate * m_hat / (sqrt(v_hat) + epsilon)
```

The gradient of the loss function is used by this approach to inform modifications to weights of neural network. The adaptive learning rate for each parameter is calculated using the gradients mean and variance. The bias correction procedures to enhance the precision of the updates.

Output Layer - This layer aims to transform the features learned by the preceding layers into a final prediction of air pollution levels for a given time step. To make a comparison between the projected and observed levels of air pollution, a loss function like mean squared error is often used during training for this layer. By feeding the input data through the full CNN-LSTM structure and retrieving the final prediction from the output layer, the trained model may be used to forecast fresh data. This process can be repeated for multiple time steps to generate a sequence of air pollution predictions. The output layer of a CNN-LSTM model is significant because it converts the learnt representations of the input data into a forecast of the air pollution levels. By combining the temporal and spatial features learned by the preceding layers, the output layer can produce accurate predictions that can help to inform public health and environmental policies related to air pollution.

The data from these monitoring sites will be used to create a model that can accurately anticipate Delhi's air pollution levels. From March 1, 2016 through February 28, 2019, a total of 35,064 recordings were collected, including 12 various criteria such as PM10, PM2.5, O_3, SO_2, CO, NO_2, dew point, air pressure, wind speed, snow hours, wind direction, temperature and rain. Due to machine failure or other uncontrolled circumstances, missing values and outliers may occur and have an impact on data mining outcomes, making data quality review essential. The information comes from the CPCB India (Central Pollution Control Board of India). Preprocessing techniques such as handling missing values by using ”Fill” function, removing unused features using the ”Drop” function, separating the data by monitoring station using the "groupby" function from the Pandas library, addressing imbalanced data by "ADASYN" algorithm and creating training and test datasets are necessary. After preprocessing, ML (Machine Learning) algorithms are applied to implement an accurate air pollution prediction model, which is evaluated for accuracy.

In hybrid CNN-LSTM model, we have incorporated two convolutional layers each having 64 and 128 filters respectively, a pooling layer with a pool size of 2, after which comes an LSTM layer having 100 units and lastly it has a fully connected layer with 32 neurons and an output neuron which employs the Adam optimizer and makes use of ReLU activation function. The training dataset consists of monthly data spanning from January 2001 to December 2018 and it is tested on data from January 2019. The train and test data ratio is maintained as 80%:20%. The input variables include monthly levels of O_3, NO_2, CO, PM, VOCs, SO_2, heavy metals, and relevant environmental data, while the output variable is air pollution index and particular polluting levels. The training goes on for as long as 1000 epochs among which stopping is done in the beginning to avoid over-fitting.

2.4 Performance Evaluation

The effectiveness of the models is measured using several evaluation measures. There are two types of evaluation metrics: range-dependent metrics and percentage metrics shown in Table 5. Range-dependent metrics are used to compare different models applied to the same dataset, whereas percentage measurements are neutral with regards to datasets when comparing models.

Table 5. Performance evaluation metrics

Percentage metrics	Range-dependent metrics
$Mean\ absolute\ percentage\ error\ (MAPE) = \frac{100}{n}\sum_{i=1}^{n}\frac{\left(y_i - \widehat{Y}_i\right\vert}{y_i}$	$Mean\,square\,error\ (MSE) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \widehat{Y}_i\right)^2$
$Accuracy\ (Acc) = 1 - \frac{\sum_{i=1}^{n}\left(y_i - \widehat{Y}_i\right\vert}{\sum_{i=1}^{n} y_i}$	$Root\,mean\,square\,error\ (RMSE) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \widehat{Y}_i\right)^2}$
$R\,squared\ \left(R^2\right) = \sqrt{\frac{\sum_{i=1}^{n}\left(\widehat{y}_i - y_i\right)^2}{\sum_{i=1}^{n}\left(y_i - \widehat{Y}_i\right)^2_i}}$	$Mean\,absolute\,error\,(MAE) = \frac{1}{n}\sum_{i=1}^{n}\left\vert y_i - \widehat{Y}_i\right\vert$
$Recall/\ Sensitivity\,(R) = \frac{TP}{TP+FN}$	$Mean\,relative\,error\,(MRE) = \frac{1}{n}\sum_{i=1}^{n}\frac{\left\vert y_i - \widehat{Y}\right\vert_i}{y_i}$
$Positive\,predictive\,value\ (P) = \frac{TP}{TP+FP}$	$y_i$current value and $\widehat{Y}$
$F\ measure\ (F1) = \frac{P-R}{P+R}$	TP = number of true positives,
	FP = number of false positives, and
	FN = number of false negatives; and;
	n = number of observations.

3 Results and Discussion

The correlations between each input variable (pollutant) and the goal variable (air quality index) may be calculated using a correlation-based feature selection technique. Next, we pick out the factors that have the highest association with our dependent variable. The performance of ML algorithms may also be significantly impacted by outliers in the dataset; hence, a correlation-based statistical outlier identification approach was used to isolate these cases. The primary characteristics between the AQI(Air Quality Index) and other contaminants were analyzed. The investigation found that PM10, PM2.5, CO, NO2, SO2, NOX, and NO are the primary contributors to poor air quality[24]. The AQI is connected with these contaminants when the correlation values are greater than 0.4. The precise values of each contaminant in the dataset's connection with AQI are shown in Table 6. The suggested method attempts to predict PM2.5 concentrations at specified times and dates by using meteorological characteristics such as temperature, relative humidity, and wind speed. The PM2.5 concentration was predicted using simulated versions of CNN, LSTM, Bi-LSTM, and CNN-LSTM deep learning models. Month, hour, wind speed, relative humidity and temperature were shown to be the top five parameters for PM2.5 prediction using a confusion matrix.

To evaluate the correspondence between PM2.5 data obtained through historical observation and those generated by ANNs, several metrics were employed. These indicators make possible this juxtaposition of modeled and observed concentrations PM2.5 values. The result of the ANNs calculations is drawn with weather observations from previous years. The capability of the ANN models to PM2.5 levels from input data may be better understood by comparing these two sets of data. Apart from that, MAE (Mean Absolute Error), R^2 (R-squared), and RMSE (Root Mean Squared Error) are other commonly used measures as well. Some of the training procedures included a phase whereby the batches ranged from 24, 32, 64, and 128**.**MSE (Mean Squared Error) was employed as the cost function, and the number of iterations through neural networks was referred to as an "epoch." Table 6 shows that, among the models tested under the same circumstances and with varying batch sizes, the CNN-LSTM with a batch size of 32 achieved the greatest one-hour prediction accuracy. Additionally, the 32-batch-size CNN-LSTM performed better across the board, especially at the 1-day latency. The models' performance was assessed by calculating indicators such as RMSE, R^2 (correlation coefficient) and MAE.

The results shown in the Table 6 for Hybrid CNN LSTM model clearly states a level ahead prediction compared to the other experimented models. The values of RMSE (Root Mean Squared Error), MAE (Mean Absolute Error) have decreased, R^2 have been increased for Day 1 as well as Day 7 compared to the previous works which represents a significant efficiency and outperformance of our hybrid CNN-LSTM model.

Table 6. Comparisonof Model Outcomes at 1 and 7 Day Delays (The top responses are highlighted.)

Model	Batch	1Day			7Day		
		MAE	RMSE	R 2	MAE	RMSE	R 2
LSTM	24	9.102	15.999	0.980	11.916	19.255	0.969
	32	9.503	16.217	0.980	11.623	19.154	0.972
	64	9.252	16.044	0.980	11.551	19.155	0.970
	128	9.725	16.997	0.978	11.823	19.227	0.971
Bi-LSTM	24	8.947	15.710	0.982	12.204	20.050	0.966
	32	8.868	15.597	0.982	11.253	18.488	0.974
	64	9.561	16.380	0.980	12.055	19.323	0.969
	128	9.488	16.456	0.979	11.753	18.113	0.971
CNN	24	9.663	17.062	0.978	10.693	17.962	0.973
	32	9.591	16.981	0.979	11.150	18.606	0.975
	64	9.261	16.667	0.979	10.621	18.431	0.975
	128	9.974	17.636	0.977	10.674	18.636	0.974
CNN-LSTM	24	9.198	16.523	0.981	9.353	16.724	0.978
	32	**6.742**	**12.921**	**0.989**	**9.034**	**16.625**	**0.979**
	64	7.869	15.757	0.982	9.885	18.373	0.976
	128	8.940	16.337	0.980	9.037	16.524	0.979

When comparing the performance of CNN-LSTM models trained on data with 1-day and 7-day lags, it was observed that the RMSE and MAE increased, while the R^2 (R-squared) decreased. Specifically, the MAE increased from 6.742 to 9.034, and the RMSE increased from 12.921 to 16.625.The given data indicate that a 7-day lag-based training for the CNN-LSTM models yields a remarkable distinction between the forecast and actual output. This decrease in accuracy could be attributed to the fact that a longer lag time may lead to a loss of important information and patterns in the data.

4 Conclusion

In order to investigate PM2.5 concentration in the metropolitan of Delhi region, a hybrid model based on both LSTM and CNN approaches were constructed. Selecting variables that showed a stronger correlation with PM2.5, like meteorological data from many sites, was the aim of the analysis based on historical data from various sites. This analysis also covered the cross-station correlations. The hybrid model has the ability of gathering the time-related information for the more satisfying and regular prediction which helps to identify the good qualities being the external appearance and inner properties of distinct things. Delhi PM2.5 air pollution forecasting model proved to be very accurate and stable for forecasting. We chose to use 24-hour inputs since pollution data was the same daily (constant). When comparing it to the other models, this model, CNN-LSTM, demonstrates a superior performance by achieving the highest accuracy and prediction stability of all models.Hybrid CNN-LSTM not only maintains long-term dependencies but also gives accurate forecasts over long-term time series.

References

1) Abdullah S, Ismail M, Ahmed AN, Abdullah AM. Forecasting particulate matter concentration using linear and non-linear approaches for air quality decision support. *Atmosphere*. 2019;10:667–667. Available from: https://doi.org/10.3390/atmos10110667.
2) Yang G, Huang J, Li X. Mining sequential patterns of PM2.5 pollutions in three zones in China. *J Clean Prod*. 2018;170:388–398. Available from: https://doi.org/10.1016/j.jclepro.2017.09.162.
3) and LD. India State-Level Disease Burden Initiative Air Pollution Collaborators. Health and economic impact of air pollution in the states of India: The Global Burden of Disease Study 2019. *Lancet Planet Health*. 2019;5(1):30298–30307. Available from: ;https://doi.org/10.1016/S2542-5196(20)30298-9.
4) Amarpuri L, Yadav N, Kumar G, Agrawal S. Prediction of CO2 emissions using deep learning hybrid approach: a case study in Indian context. *2019 Twelfth International Conference on Contemporary Computing (IC3) IEEE; 2019*;p. 1–6. Available from: https://doi.org/10.1007/s11356-023-30428-5.
5) Lv CX, Qiao ASY, Wu BJ, W. Time series analysis of hemorrhagic fever with renal syndrome in mainland China by using an XGBoost forecasting model. *BMC Infect Dis*. 2021;21:1–13. Available from: https://doi.org/10.1186/s12879-021-06503-y.
6) Becerra-Rico J, Aceves-Fernández, Esquivel-Escalante MA, Pedraza-Ortega K, C J. Airborne particle pollution predictive model using gated recurrent unit (GRU) deep neural networks. *Earth Sci Inf*. 2020;13:821–834. Available from: https://doi.org/10.3390/app12010256.
7) Ayturan YA, Ayturan ZC, Ho A, Kongoli C, Tuncez FD, Dursun S, et al. Short-term prediction of PM2.5 pollution with deep learning methods. *Global NEST J*. 2020;22(1):126–131. Available from: https://doi.org/10.30955/gnj.003208.

8) Mani G, Viswanadhapalli JK, Stonie AA. Prediction and forecasting of air quality index in Chennai using regression and ARIMA time series models. *J Eng Res*. 2021. Available from: https://doi.org/10.36909/jer.10253.

9) Md N, Wai Y, Ibrahim N, Rashid Z, Mustafa N, Hamid H, et al. Particulate matter (pm2.5) as a potential sars-cov-2 carrier. 2020. Available from: https://doi.org/10.1038/s41598-021-81935-9.

10) Amado TM, and DC. Development of Machine Learning-based Predictive Models for Air Quality Monitoring and Characterization. *IEEE*. 2018. Available from: https://doi.org/10.1109/TENCON.2018.8650518.

11) Shishegaran A, Saeedi M, Kumar A, Ghiasinejad H. Prediction of air quality in Tehran by developing the nonlinear ensemble model. *J Clean Prod*. 2020;259. Available from: https://doi.org/10.1016/j.jclepro.2020.120825.

12) Yazdi D, Kuang M, Dimakopoulou Z, Barratt K, Suel B, Amini E, et al. Predicting fine particulate matter (pm2.5) in the greater London area: an ensemble approach using machine learning methods. 2020. Available from: https://doi.org/10.3390/rs12060914.

13) Londhe M. Data mining and machine learning approach for air quality index prediction. *Int J Eng Appl Phys*. 2021;1(2):136–153. Available from: https://ijeap.org/ijeap/article/view/28.

14) Conticini E, Frediani B, Caro D. Can Atmospheric Pollution Be Considered a Co-factor in Extremely High Level of SARS-CoV-2 Lethality in Northern Italy? *Environ Pollut*. 2020;261. Available from: https://doi.org/10.1016/j.envpol.2020.114465.

15) Wang B, Kong W, Zhao P. An air quality forecasting model based on improved convnet and RNN. *Soft Comput*. 2021;25. Available from: https://doi.org/10.1038/s41598-022-12355-6.

16) Freeman. Forecasting air quality time series using deep learning. *J Air Waste Manage Assoc*. 2018;68(8):866–886. Available from: https://doi.org/10.1080/10962247.2018.1459956.

17) The worlds cities in 2016: data booklet. June 2016. United Nations; 2016. 2016. Available from: https://www.un.org/en/development/desa/population/publications/index.asp.

18) Singh V, Singh S, Biswal A. Exceedances and trends of particulate matter (PM2.5) in five Indian megacities. *Science of The Total Environment*. 2021;750. Available from: https://doi.org/10.1016/j.scitotenv.2020.141461.

19) Usual suspects: Vehicles, industrial emissions behind foul play. August 2018. Times of India. 2018. Available from: https://timesofindia.indiatimes.com/city/delhi/usual-suspects-vehicles-industrial-emissions-behind-foul-play-all-year/articleshow/66228517.cms.

20) Spyrou ED, Tsoulos I, Stylios C. Applying and Comparing LSTM and ARIMA to Predict CO Levels for a Time-Series Measurements in a Port Area. *Signals*. 2022;3:235–248.

21) Madhuri VM, Samyama GG, Kamalapurkar S. Air pollution prediction using machine learning supervised learning approach. *Int J Sci Technol Res*. 2020;9(4):118–141. Available from: https://www.ijstr.org/final-print/apr2020/Air-Pollution-Prediction-Using-Machine-Learning-Supervised-Learning-Approach.pdf.

22) Mishra A, Gupta Y. Comparative analysis of Air Quality Index prediction using deep learning algorithms. *Spat Inf Res*. 2024;32:63–72. Available from: https://doi.org/10.1007/s41324-023-00541-1.

23) Maciąg PS, Bembenik R, Piekarzewicz A, Ser JD, Lobo JL, Kasabov NK. Effective air pollution prediction by combining time series decomposition with stacking and bagging ensembles of evolving spiking neural networks. *Environmental Modelling Software*. 2023;170:105851–105851. Available from: https://dx.doi.org/10.1016/j.envsoft.2023.105851.

24) Predicting Air Pollution Levels in New Delhi. . Available from: https://iq.opengenus.org/predicting-air-pollution-levels-part2/.