International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-11, Nov 2019 E-ISSN: 2347-2693

Study on Various Machine Learning Techniques for Pollution Forecasting

Ifshita Chaudhary[1*], Shruti Sharma[2], Preeti Sethi [3]

[1,2,3]Dept. of Computer Engineering, J.C. Bose University of Science and Technology, YMCA, Faridabad, Haryana, India

Corresponding Author: ifshita.chaudhary@ymail.com , Tel.: 8860935654

DOI: https://doi.org/10.26438/ijcse/v7i11.5663 | Available online at: www.ijcseonline.org

Accepted: 12/Nov/2019, Published: 30/Nov/2019

***Abstract*—** Because of a significant increment of pollution in the air, it is required to foresee the pollution of the following dates, months and years. Air pollution is quickly expanding because of different human factors and reasons, such as the generation of synthetic compounds, particulates, pollutants or in-organic materials and other substances which is even the reason for the loss of human lives and even additionally hurts the indigenous habitat like plants and animals, etc. Undoubtedly, air pollution is one of the significant natural problems in metropolitan and urban areas. In this way, Monitoring and safeguarding air quality is one of the most fundamental exercises in numerous modern and urban territories today. Consequently, air quality assessment, observing, and forecast has turned into significant research. The point of this paper is to explore different Machine Learning based strategies especially artificial neural network models for air quality determining in various conditions. This scheme for the future will elaborate on the distributed research results identifying with air quality index and forecast utilizing techniques predicting air quality of a particular area using Neural Networks. Therefore, as of now under this scheme, we will derive the comparative analyses of various neural network Algorithms from past researchers i.e. ANN, MLP, CNN, LSTM, CNN-LSTM, Encoder decoder, and Convolution LSTM. To find the efficiency and effectiveness in the area of air contamination and pollution.

***Keywords*—** Machine Learning, Artificial Neural Network, Multilayer Perceptron, Convolutional Neural Network, Long Short-Term Memory, Recurrent Neural Network, Encoding and Decoding.

I. INTRODUCTION

Air quality assessment is a significant method to screen and control air contamination or pollution. The qualities of air supply influence its appropriateness for a particular use. A couple of air contaminations or pollution called criteria air pollutants are common throughout the major urban cities in India and other countries. These toxins can harm wellbeing, hurt the earth and cause utter disaster harm. The present criteria toxins or pollutants are Carbon Monoxide (CO), Lead (Pb). Nitrogen Dioxide (NO2), Ozone (O3). Particulate matter (PM), Sulfur Dioxide (SO2), etc. The Air Quality Index (AQI) contains ambient air pollution data collected by the India Meteorological Department and other air pollution control agencies from thousands of monitors. However, descriptive information about each monitoring station (including its geographic location and its operator), and data quality assurance/quality control information. AQI data is used to assess air quality, assist in Attainment/Non-Attainment designations, evaluate state-wise Implementation Plans for Non-Attainment Areas, perform modeling for permit review analysis, and other air quality management functions. AQS information is also used to prepare reports for parliament as mandated by the AQI standards. The figure below elaborated on how the data concerning AQI standards are gathered for pollution forecasting.

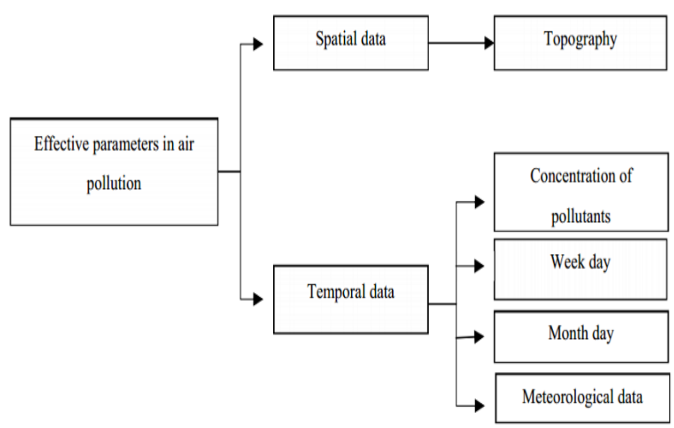

Figure 1: Representation Engaged with Data Collection for AIR Quality Index.

The AQI is a list for revealing day by day air quality. IMD and Central Pollution Control Board ascertain the AQI for five significant air contaminations controlled by the Clean Air Act: Ground-level Ozone, Molecule Contamination (otherwise called particulate issue), Carbon Monoxide, Sulfur Dioxide, and Nitrogen Dioxide. For every one of these contaminations, IMD and Central Pollution Control Board have set up National Air Quality measures to secure general wellbeing. The purpose of the AQI is to help you understand what local air quality means to your health. To make it easier to understand, the AQI is divided into six categories:

Table 1: IMD and Central Pollution Control Board have assigned a specific color to each AQI category to make it easier for people to understand quickly whether air pollution is reaching unhealthy levels in their communities. For example, the color orange means that conditions are "unhealthy for sensitive groups," while red means that conditions may be "unhealthy for everyone," and so on.

Air Quality Index Levels of Health Concern	Numerical Value	Meaning
Good	0 to 50	Air quality is considered satisfactory, and air pollution poses little or no risk.
Moderate	51 to 100	Air quality is acceptable; however, for some pollutants there may be a moderate health concern for a very small number of people who are unusually sensitive to air pollution.
Unhealthy for Sensitive Groups	101 to 150	Members of sensitive groups may experience health effects. The general public is not likely to be affected.
Unhealthy	151 to 200	Everyone may begin to experience health effects; members of sensitive groups may experience more serious health effects.
Very Unhealthy	201 to 300	Health alert: everyone may experience more serious health effects.
Hazardous	301 to 500	Health warnings of emergency conditions. The entire population is more likely to be affected.

Figure 2

Appearance to particulate issues for quite a while can prompt respiratory and cardiovascular illnesses, for example, asthma, bronchitis, lung malignancy, and heart assaults. A year ago, the Global Burden of Disease contemplates stuck open-air contamination as the fifth biggest executioner in India after hypertension, indoor air contamination, tobacco smoking, and poor nourishment; around 720,000 early passing happened in India from air contamination related infections in 2016. The Central Pollution Control Board (CPCB) supported WHO says India positions among the world's most noticeably awful for its dirtied air. Delhi is among the most dirtied urban communities on the planet today. New Delhi's PM10 Levels over 10 years against Indian Standard and WHO Standard AIR QUALITY INDEX the examination that connections the poison, PM10 (particulate issue littler than 10 microns), to these diseases. The focal administrative expert as of late recommended stricter standards for various air poisons and contaminations yet overlooked modification of the standard for PM10.

II. RELATED WORK

Weibo Liua, Zidong Wanga, Xiaohui Liua, Nianyin Zengb, Yurong Liucd and Fuad E. Alsaadid [1] depicted that, the proposition of a quick learning calculation for deep belief networks organizes in 2006. The deep belief network measures have drawn frequently increasing assessment premiums as a result of their natural ability to defeat the drawbacks of Conventional Networks subject to hand-structured highlights. Deep belief networks learning methodologies have additionally been seen as reasonable for huge information examination with fruitful applications to PC vision, design acknowledgment, discourse acknowledgment, regular language preparing, and proposal frameworks. In this paper, details about some generally utilized profound learning models and their down to earth applications are mentioned. A modern review is given on four thoughtful learning structures, to be specific, autoencoder, Convolutional Neural Network, profound conviction arranges and confined Boltzmann Machine. Various kinds of profound neural systems are reviewed and ongoing indications of progress are condensed. Utilizations of profound learning strategies on some chose territories (discourse acknowledgment, design acknowledgment, and PC vision) are featured. A rundown of future research themes is at last given with clear avocations.

Mohammed Kamel Benkaddour, Abdennacer Bounoua [2] depicted that, the facial recognition has aroused the interest of the scientific community, this technique of biometric that is effective, non-intrusive and contactless has taken an increasingly important part in the field of research. This paper proposes a face recognition and classification method based on deep learning, in particular, Convolutional Neural Network (CNN), which are incredibly influential tools that have found enormous achievement in image classification and pattern recognition. In this work, the approach to this task is based on the Convolutional Neural Network (CNN) as a powerful feature extraction followed by Support Vector Machines (SVM) as a high classifier. To reduce the dimension of these features, a principal component analysis (PCA) technique is employed. An extensive evaluation of methods is conducted on the FERET dataset. The results obtained showed that the proposed method CNN combine with PCA and SVC solution provides a significant improvement in performance and enhance the recognition accuracy.

Chiou-Jye & Kuo, Ping-Huan [3] depicted that, current society air contamination is a significant theme as this pollution applies a fundamentally terrible impact on human wellbeing and nature. Among air poisons, Particulate Matter

(PM2.5) comprises of suspended particles with a distance across equivalent to or under 2.5 µm. Wellsprings of PM2.5 can be coal-terminated power age, smoke, or bits of residue. To screen and gauge the PM2.5 focus, Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) are joined and applied to the PM2.5 gauging framework. To think about the general execution of every calculation, four estimation files, Mean Absolute Error (MAE), Root Mean Square Error (RMSE) Pearson Relationship Coefficient and Index of Agreement (IA) are applied to the investigations in this paper. Contrasted and other AI techniques, the trial results demonstrated that the anticipating precision of the proposed CNN-LSTM model is checked to be the most noteworthy in this paper. For the CNN-LSTM model, its achievability and practicability to estimate the PM2.5 fixation are likewise confirmed in this paper. The fundamental commitment of this paper is to build up a profound neural system model that incorporates the CNN and LSTM structures, and through chronicled information, for example, cumulated long stretches of a downpour, cumulated wind speed, and PM2.5 focus. Later on, this investigation can likewise be applied to the counteractive action and control of PM2.5.

Vidushi Chaudhary, Anand Deshbhratar, Vijayanand Kumar, Dibyendu Paul [4] depicted that, air pollution has been a foremost concern in India for the last few years. Speedy industrialization has led to a tremendous increase in air pollution. To monitor air pollution, particularly in major pollution prone cities like Delhi, the government of India has installed pollutants measuring sensors. The sensors regularly monitor the level of air pollutants like PM2.5, PM10, CO, NO2, SO2, O3. However, capturing current pollutants concentration does not help decrease/avoiding pollution exposure. This paper proposes a deep learning-based LSTM model to predict future air pollutant's concentration and formulate the problem of predicting pollutant's concentration as a time series-based problem where current pollutants level is dependent on the previous pollutant, meteorological, traffic data, festivals and national holidays information. A sample dataset of historic pollutant levels, meteorological and traffic data, and identify discriminatory features to predict pollutants concentration for the next hours. Empirical analysis reveals that certain meteorological features (temperature, pressure, humidity, wind speed, wind direction, UV Index, cloud cover, rain), traffic information, festival information can be used to forecast pollutants. A series of experiments are conducted to validate the proposed solution approach and present evidence to demonstrate the effectiveness of the proposed framework with average RMSE of pollutants is less than 15 for the next 12-hour prediction, less than 8 for the next 6-hour prediction and less than 5 for next hour prediction. The model accuracy is compared with real-time data fetched from CPCB (Central Pollution Control Board).

Y. Tsai, Y. Zeng and Y. Chang [5] With the advance of technology, it is increasingly exhausted emissions have caused air pollution. In particular, PM2.5 (Particulate Matter) has been proven that it has a great correlation with human health. Therefore, the detection and prediction of PM 2.5 air pollution is an important issue. There are countries around the world that have built a variety of sensing devices for monitoring PM2.5 concentrations. There were also many studies that have been constructed to predict and forecast various air pollution. Therefore, how to accurately forecast PM2.5 becomes an important issue in recent years. In this paper, the proposal is to forecast PM2.5 concentration using RNN (Recurrent Neural Network) with LSTM (Long Short-Term Memory). Keras is exploited which is a high-level neural networks API written in Python and capable of running on top of TensorFlow, to build a neural network and run RNN with LSTM through TensorFlow. The training data used in the network is retrieved from the EPA (Environmental Protection Administration) of Taiwan from the year 2012 to 2016 and is combined into 20-dimensions data, and the forecasting test data is the year 2017. Conducted experiments to evaluate the forecasting value of PM2.5 concentration for the next four hours at 66 stations around Taiwan. The result shows that the proposed approach can effectively forecast the value of PM2.5.

S Geetha & L Prasika [6] depicted that, smog triggered due to air pollutants and fog. Deep Learning techniques are applied to predict the smog severity. This paper presents a deep learning-based predictive model for various air pollutants (NO2, NOx, CO, SO2, O3, PM2.5, PM10) for metropolitan area air pollution dataset. Central Pollution Control Board (CPCB) is monitoring air, water, waste, etc through nationwide programs. Through the National Air Quality Monitoring program, the primary and secondary air pollutants are captured and available online. In this paper, two traditional predictive models along with deep learning techniques Long-Short Term Memory (LSTM) are used for predicting air pollutants. Before training the model, the missing values and noise in the dataset were imputed using the mean value. Then, the models are built with LR, ARIMA, and LSTM. Finally, the model's performance is measured using Mean Absolute Error and Root Mean Square Error (RMSE). LSTM performed better than LR and ARIMA.

Chiou-Jye Huang and Ping-Huan Kuo [7] depicts that, in modern society, air pollution is an important topic as this pollution exerts a critically bad influence on human health and the environment. Among air pollutants, Particulate Matter (PM2.5) consists of suspended particles with a diameter equal to or less than 2.5 µm. Sources of PM2.5 can be coal-fired power generation, smoke, or dust. To monitor and estimate the PM2.5 concentration, Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) are combined and applied to the PM2.5 forecasting system. To compare the overall performance of each algorithm, four measurement indexes, Mean Absolute Error (MAE), Root Mean Square Error (RMSE) Pearson correlation coefficient and Index of Agreement (IA) are applied to the experiments in this paper. Compared with other Machine Learning methods, the experimental results showed that the forecasting accuracy of the proposed CNN-LSTM model is verified to be the highest in this paper. For the CNN-LSTM model, its feasibility and practicability to forecast the PM2.5 concentration are also verified in this paper. The main contribution of this paper is to develop a deep neural network model that integrates the CNN and LSTM architectures, and through historical data such as cumulated hours of rain, cumulated wind speed and PM2.5 concentration. In the future, this study can also be applied to the prevention and control of PM2.5.

III. METHODOLOGY

Machine Learning Models: Machine learning (ML) is the branch of computer science that makes computers capable of performing a task without being explicitly programmed. Many research papers focus on the classification of air quality evaluation using Machine Learning algorithms. Most of these articles use different scientific methods, approaches and ML models to predict air quality. S. Y. Muhammed et al. [8] point out that machine learning algorithms are best suited for air quality prediction. Some of them are discussed below:

Artificial Neural Network (ANN) Model: The artificial neural Network model attempts to mimic the structures and systems inside the human cerebrum. The engineering of neural systems comprises nodes that produce a sign or stay quiet according to a sigmoid enactment work as a rule. A. Sarkar et al. [9] bring up that the ANNs are prepared with a preparation set of sources of info and referred to yield information as appeared in Figure 2. For preparing, the edge loads are controlled to diminish the preparation mistake. One basic preparing system is the backpropagation connect with two hidden layers or units. The predetermined checking station is a divert arranged in the State of New Delhi in India. Their strategy incorporates two stages recorded underneath.

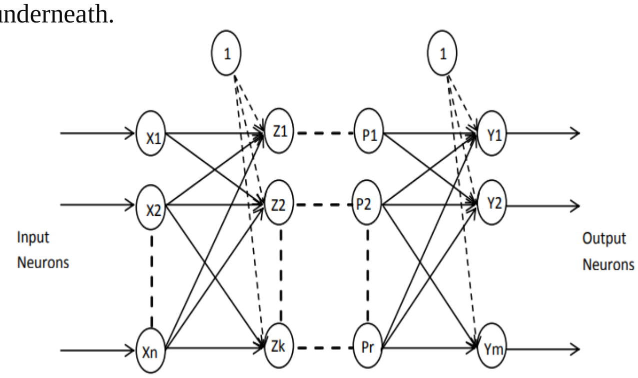

Figure 3: ANN Model for Neurons

Multilayer Perceptron (MLP) : It is one of the essential neural system structures from which a few others were determined. The fundamental component of the MLP is a neuron. A few neurons are composed into layers – input, covered up (at least one) and yield layer. Every neuron has a basic structure that imitates the usefulness of the neuron found in creatures and the entire structure of layers copies the cerebrum structure. This closeness offers to ascend to the name. Every neuron initially condenses the weighted information esteems and after that goes the whole through the exchange work. If that the exchange capacity is nonlinear, for example, a fundamental sigmoid capacity or hyperbolic digression, at that point the entire structure obtains its extraordinary capacity as an all-inclusive approximator. The neurons in the info layer take the qualities from the model information factors and pass the qualities to the neurons in the concealed layer, the shrouded layer neurons pass the qualities to the higher concealed layers lastly to the yield layer that gives the model yield esteem. The yield of every neuron is passed to the contribution of all neurons in the following higher layer. Every one of the associations between neurons is weighted. These interconnection loads are the fundamental parameters of the model that are balanced during the learning procedure. Model information sources take their qualities from the info highlights – estimated parameters that decide the yield of the model. Model output(s) speaks to the phenomenon that is being reconstructed (approximated). Outputs are called output features. The estimations of one specific acknowledgment of all information sources are known as the info vector, and the model reveals structure the output vector. The two vectors together structure an example. An example is, accordingly, similar to one domain into the multivariable domain or space laying on the outside of the capacity the model is approximating. The entire thought of building a model to inexact a multivariable capacity is the accompanying: Firstly, enough examples ought to be accessible (for example from the estimations) with known information based on input and output results which highlights the context. These examples ought to be consistently spread over the entire explored area. At that point, the model topology is planned by the quantity of input and output results. The model learning stage comprises of a few changes of model interconnection loads– to limit the normal mistake between the genuine estimated output value and the output values that are delivered by the neural system. One of the calculations that can be utilized for this reason for existing is the back-propagation calculation. During the time spent learning the MLP takes the data (about the speculate under examination) that is accessible in the learning designs and when learning is finished (the model built) it can give the outcomes for beforehand obscure examples – where just info esteems are displayed to the system. This is conceivable if there were comparative examples (to the obscure example) in the learning set. This is the supposed summing up the ability of the MLP. The comparability is numerically the separation between two examples. The essential standard of MLP model development is subsequently to give data-rich learning designs. There are some fundamental advances and strategies that ought to be utilized in the model development procedure to get compelling models. However, Learning occurs in the perceptron by changing connection weights after each piece of data is processed, based on the amount of error in the output compared to the expected result. This is an example of supervised learning and is carried out through backpropagation, a generalization of the least mean square's algorithm in the linear perceptron. We can represent the degree of error in an output node j in the n^{th} data point (training example) by $e_i(n)=d_i(n)-y_i(n)$ where d is the target value and y is the value produced by the perceptron. The node weights can then be adjusted based on corrections that minimize the error in the entire output. Therefore, the learning rule for the multilayer perception is known as "the generalized delta rule" or the "backpropagation rule. The generalized delta rule repetitively calculates an *error function* for each input and back propagates the error from one layer to the previous one. The weights for a particular node are adjusted in direct proportion to the error in the units to which it is connected.

Let E_p = error function for pattern p

t_{pj} = target output for pattern p on node j

o_{pj} = actual output for pattern p on node j

w_{ij} = weight from node i to node j

The error function E_p is defined to be proportional to the square of the difference $t_{pj} - o_{pj}$

$$E_p = 1/2 \sum_j (t_{pj} - o_{pj})^2 \quad (1)$$

The activation of each unit j, for pattern p, can be written as

$$net_{pj} = \sum_i w_{ij} o_{pi} \quad (2)$$

For this investigation, data were divided into training data (60%), testing data (20%) and validation data (20%). For better examination and results, information was scaled to fall between the scopes of [0,1].

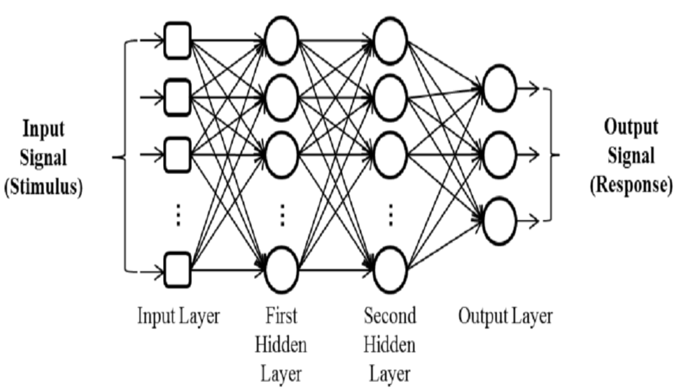

Figure 4: Architecture of Multilayer Perceptron with Two Hidden Layers

Convolutional Neural Network (CNN): CNN consists of several convolutional layers and some fully connected layers. Between two adjacent convolutional layers, rule sampling operations take place. Inputs of CNN are of the format (w,h,c) where w is width, h is the height of an object in pixels and c is the number of colour channels of each pixel. The Output of CNN is a vector of q probability units depending on the number of categories. The convolution operation between kernel and input object is performed. The typical convolution layer contains k filters with size (i,j,c) where i is the width, j is the height. Filters are less than width w and height h of the input object. Colour channel of the input object is always c. Each of the filters is independently combined with input objects followed by non-linear transformation and generates k feature maps which are the input for the next layer. The Dot product is calculated between the entry of the filter and the local region it is connected to in the input object. Learnable filters are set as the parameter. Certain regions produce larger output than others when the filter goes through all the inputs. Regardless of where this feature is present in the input, it can be extracted and preserved in the feature maps and passed to the next layer. The smaller sample is taken from an existing sample generated from the convolution layer over a common region ($s.t$) where s is the width and t is the height. As the depth increases, the resolution of the feature object becomes harsher. In a fully connected layer, all hidden units are connected to a previous layer. In the last layer, the fully connected layer features vectors from the previous layer on high-level reasoning to produce a final class score for the object. The deep CNN has a total of 4 learnable layers, including 2 convolutional layers and 2 fully connected layers. CNN architecture that is capable of discriminating between different variables based on the event type and capable of handling events at various spatial scale in pollution forecasting. Convolutional neural network layers can be of three types which are discussed below:

Convolutional: Convolutional layers comprise a rectangular network of neurons. It necessitates that the past layer likewise is a rectangular framework of neurons. Every neuron takes contributions from a rectangular segment of the past layer; the loads for this rectangular segment are the equivalent for every neuron in the convolutional layer. In this way, the convolutional layer is only a picture convolution of the past layer, where the loads indicate the convolution channel. Also, there might be a few frameworks in each convolutional layer; every network takes contributions from every one of the matrices in the past layer, utilizing conceivably various channels.

Max-Pooling: After each convolutional layer, there may be a pooling layer. The pooling layer takes small rectangular blocks from the convolutional layer and subsamples it to produce a single output from that block. There are several ways to do this pooling, such as taking the average or the maximum, or a learned linear combination of the neurons in the block. Our pooling layers will always be max-pooling layers; that is, they take the maximum of the block they are pooling.

Fully-Connected: Finally, after several convolutional and max-pooling layers, the high-level reasoning in the neural network is done via fully connected layers. A fully connected layer takes all neurons in the previous layer (be it fully connected, pooling, or convolutional) and connects it to every single neuron it has. Fully connected layers are not spatially located anymore (you can visualize them as one-dimensional), so there can be no convolutional layers after a fully connected layer.

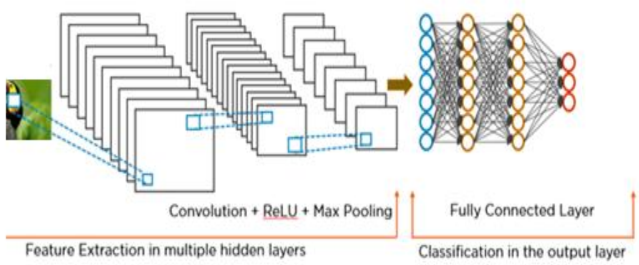

Figure 5: CNN Example with Convolutional layers, Max-Pooling, and Fully-Connected Layers

Long Short-Term Memory (LSTM): Another important technology of ANN is Recurrent Neural Network (RNN), which differs from CNN and MLP in its consideration of the time sequence. LSTM is one of the RNN models. The schematic of LSTM is shown in Figure 4, where σ is a sigmoid function, as shown in Equation (1). LSTM contains an input gate, an output gate, and a forget gate. The intelligent activity among these three gates makes LSTM have the adequate capacity to tackle the issue of long-haul conditions which general RNNs can't learn. Also, a typical issue in profound neural systems is called inclination disappearing, i.e., The learning velocity of the recently concealed layers is slower than the more profound shrouded layers. This wonder may even prompt a diminishing in precision rate as shrouded layers increment. Be that as it may, the keen structure of the memory cell in LSTM can viably take care of the issue of slope disappearing in backpropagation and can get familiar with the info grouping with longer time steps. Consequently, LSTM is usually utilized for tackling applications identified with time-sequential issues. The particular recipe inference of LSTM is shown in Equations beneath:

1) $sigmoid(x) = \frac{1}{1+e^{-x}}$
2) $\bar{z}^t = W_z x^t + R_z y^{t-1} + b_z$
3) $z^t = \tanh(\bar{z}^t)$
4) $\bar{\imath}^t = W_i x^t + R_i y^{t-1} + p_i \odot c^{t-1} + b_i$
5) $i^t = sigmoid(\bar{\imath}^t)$
6) $\bar{f}^t = W_f x^t + R_f y^{t-1} + p_f \odot c^{t-1} + b_f$
7) $f^t = sigmoid(\bar{f}^t)$
8) $c^t = z^t \odot i^t + c^{t-1} \odot f^t$
9) $\bar{o}^t = W_o x^t + R_o y^{t-1} + p_o \odot c^{t-1} + b_o$
10) $o^t = sigmoid(\bar{o}^t)$
11) $y^t = \tanh(c^t) \odot o^t$

where W_z, W_i, W_f, and Wo are input weights; R_z, R_i, R_f, and R_o are recurrent weights, p_i, p_f, and p_o are peephole weights; b_z, b_i, b_f, and b_o are bias weights; z^t is the block input gate; f^t is the forget gate; c^t is the cell; o^t is the output gate; y^t is the block output; and ⊙ represents point-wise multiplication. To reach the goal of parameter optimization, either CNN or LSTM can use backpropagation to adjust the parameters of the model during the process of training.

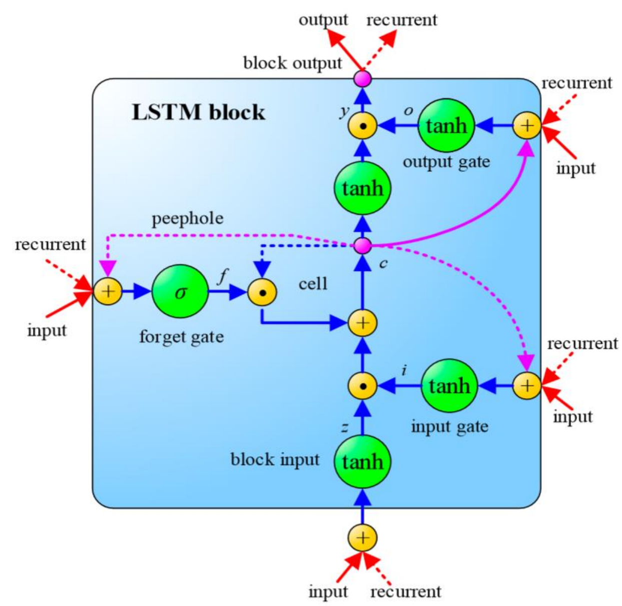

Figure 6: LSTM Example with ANN is Recurrent Neural Network (RNN), which differs from CNN and MLP in its consideration of the time sequence

CNN-LSTM Model: CNN-LSTM method for predicting air pollution consists of a series connection of CNN and LSTM. CNN-LSTM can extract complex features among multiple sensor variables collected vide demand forecasting based on datasets and can store complex irregular trends. First, the upper layer of CNN-LSTM consists of CNN. The CNN layer can receive various variables that affect intensity and subcategorization. Also, modelled as meta information in the CNN layer can produce the features. CNN consists of an input layer that accepts sensor variables as inputs, an output layer that extracts features to LSTMs, and several hidden layers. The hidden layer typically consists of a convolution layer, a ReLU layer, an activation function, and a pooling layer. The convolution layer applies the convolution operation to the incoming multivariate series sequence and passes the results to the next layer. The convolution operation emulates the response of individual neurons to visual stimulation. Each convolution neuron processes data only for the receptive field. The convolutional operation can reduce the number of parameters and make the CNN-LSTM network deeper. If x^0_i ¼ = {x_1; x_2; x_n} is power consumption input vector and n is the number of normalized 60 min unit per window. Equation is the result of the vector y^1_{ij} output from the first convolutional layer, y^1_{ij} is calculated by output vector x^1_{ij} of the previous layer, b^1_i represents the bias for the ***j th*** feature map, ***w*** is the weight of the kernel, m is the index value of the filter, and ***s*** is the activation function like ReLU. The equation below is the result of the vector y_{lj} output from the l^{th} convolution layer as depicted:

$$y^1{}_{ij} = \sigma(b^1{}_j + \sum_{m=1}^{M} w^1_m,\ {}_jx^0_i + m - 1, j)$$

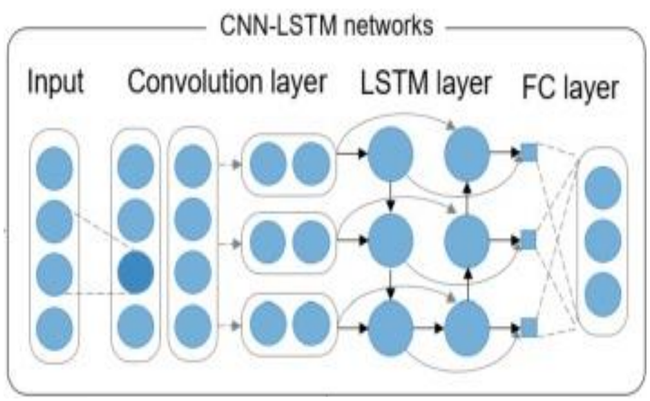

Figure 7: The CNN-LSTM Network Diagram

Convolutional LSTM: To encode the pollution based spatial information in the data sequence, in which all the inputs $X_1, \ldots, X_t$, cell outputs $C_1, \ldots, C_t$, hidden states $H_1, \ldots, H_t$, and gates it, ft, ot of the network are 3D tensors. Given the outputs of CNN features from the layer ***conv5***, denoted as $X = [X_1, X_2, \ldots, X_t]$ where X_i is of size $a \times b \times M$, $a \times b$ is the size of filtered objects of ***conv5***, and M is the number of convolutional filters (in our case, $a = 14$, $b = 6$ and $M = 512$). Specifically, CNN-based feature maps at frame-level have a pollution based spatial grid of size $a \times b$ and M measurements which vary over time. A typical LSTM model i.e., fully-connected LSTM deals with spatiotemporal data by full connections in input-to-state and state-to-state transitions in which spatial information is not encoded. To explicitly encode spatial priors, a better way is to determine the future state of a certain cell in the spatial grid by the inputs and past states from its local neighbourhood. This can be achieved by implementing a convolutional operator with a local receptive field for the input-to-state and state-to-state transitions. The structure of convolutional LSTM is illustrated in the equation below. Accordingly, the hidden states of convolutional LSTM can be computed by the following equations:

$$i_t = \sigma(W_{xi} * X_t + W_{hi} * H_{t-1} + W_{ci} \circ C_{t-1} + b_i)$$
$$f_t = \sigma(W_{xf} * X_t + W_{hf} * H_{t-1} + W_{cf} \circ C_{t-1} + b_f)$$
$$C_t = f_t \circ C_{t-1} + i_t \circ \tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c)$$
$$o_t = \sigma(W_{xo} * X_t + W_{ho} * H_{t-1} + W_{co} * C_t + b_o)$$
$$H_t = o_t \circ \tanh(C_t)$$

where $*$ denotes the convolutional filter and $\circ$ denotes the requisite object or products. In the equation above the state, H_t with the same spatial dimensions as the inputs can be seen as the hidden representation of objects. In this sense, a convolutional LSTM model with a larger convolution kernel e.g., 5×5 can capture faster motions while that with a smaller kernel can capture slower motions.

Encoding and Decoding: An encoder network and a decoder network. Both networks have a stack of two convolutional LSTM layers. In the initial states and cell outputs of the decoder, the network is copied from the last state of the encoding network. Intuitively, the encoder LSTM compresses the input sequence into a fixed-length representation (encoded by hidden state tensor), and the decoder LSTM unfolds this representation to reconstruct the input sequence (i.e., CNN features at frame-level), which can be formulated as follows:

$$\hat{X}_1, \ldots, \hat{X}_t \approx g_{decoding}(f_{encoding}(X_1, \ldots, X_t))$$

IV. CONCLUSION

While adhering to various ANN models example MLP, CNN, CNN-LSTM, Encoder-Decoder and Convolution LSTM on the architectural scenarios we achieved that, CNN formed the convolutional layer passing the objects based max pooling with the fully connected node for the formation of results based on various features extracted from hidden layers. Whereas, LSTM vide recurrent neural network which passes the parameters and features not by objects but by values from one hidden layer to another hidden layer which carrying the existing values as a feed-forward model for better and effective results. However, the Encoder and Decoder compressed the values formed by features and passed from one layer to another layer respectively resulting features do not lose its relevance or sequence due to complexity or mammoth values. On the contrary, Convolution-LSTM passes the featured data from input to state or state to state among the hidden layers without encoding and decoding over the filtered objects. Subsequently, the CNN-LSTM with root means a square error which created model APNET delivers potentially effective are resourceful architectures amongst the other as adhered to in the scheme.

REFERENCES

[1] Weibo Liua, Zidong Wanga , Xiaohui Liua, Nianyin Zengb, Yurong Liuc,D And Fuad E. Alsaadid ,"*A Survey Of Deep Neural Network Architectures And Their Applications"*, Brunel University Research Archive(BURA), May 14 – 2018.

[2] Mohammed Kamel Benkaddour, Abdennacer Bounoua, "*Feature Extraction And Classification Using Deep Convolutional Neural Networks, Feature Extraction And Classification Using Deep Convolutional Neural Networks"*, Laboratory Communication Network & Architecture Multimedia RCAM & DJILLALILIABBES University Sidi Bel Abbes 22000, Algeria. University Kasdi Marbah, FNTIC Faculty, Ouargla 30000, Algeria June, 2019.

[3] Huang, Chiou-Jye & Kuo, Ping-Huan, "*Deep CNN-LSTM Model For Particulate Matter (PM2.5) Forecasting In Smart Cities",* Sensors. 18. 2220. 10.3390/S18072220, 2018

[4] Vidushi Chaudhary, Anand Deshbhratar, Vijayanand Kumar, Dibyendu Paul, "*Time Series Based LSTM Model To Predict Air Pollutant's Concentration For Prominent Cities In India",* Harbin Institute Of Technology (Shenzhen) February, 2017

[5] Y. Tsai, Y. Zeng And Y. Chang, "*Air Pollution Forecasting Using RNN With LSTM* ", IEEE 16th Intl Conf On Dependable, Autonomic And Secure Computing, 16th Intl Conf On Pervasive Intelligence And Computing, 4th Intl Conf On Big Data Intelligence And Computing And Cyber Science AndTechnologyCongress(DASC/Picom/Datacom/Cyberscitech), Athens, 2018, Pp. 1074-1079

[6] S Geetha & L Prasika, "*SMOG PREDICTION MODEL USING TIME SERIES WITH LONG-SHORT TERM MEMORY",* International Journal Of Mechanical Engineering And Technology (IJMET) Volume 10, Issue 01, January 2019

[7] Huang, Chiou-Jye & Kuo, Ping-Huan, "*A Deep CNN-LSTM Model for Particulate Matter (PM2.5) Forecasting in Smart Cities*" Sensors. 18. 2220. 10.3390/S18072220, 2018

[8] Muhammad, Salisu & Makhtar, Mokhairi & Rozaimee, Azilawati & Aziz, Azwa & Jamal, Azrul Amri, "*Classification Model for Water Quality Using Machine Learning Techniques. International Journal of Software Engineering and Its Applications*" 9. 45-52. 10.14257/Ijseia.2015.9.6.05, 2015

[9] Sarkar, Archana & Pandey, Prashant, "*River Water Quality Modelling Using Artificial Neural Network Technique*" Aquatic Procedia.4.1070-1077. 10.1016/J.Aqpro.2015.02.135, 2015

[10] V. M. Niharika And P. S. Rao, "*A Survey on Air Quality Forecasting Techniques*," International Journal of Computer Science and Information Technologies, Vol. 5, No. 1, Pp.103-107, 2014

[11] Shubham Billus, Shivam Billus, Rishab Behl, "*Weather Prediction through Sliding Window Algorithm and Deep Learning*", Isroset-Journal (IJSRCSE), Vol.6 , Issue.5 , pp.20-24, Oct-2018

[12] N.S. Lele, "*Image Classification Using Convolutional Neural Network*", Vol.6 , Issue.3 , pp.22-26, Jun-2018

Authors Profile

Ifshita Chaudhary pursed B.Tech from KR Mangalam University, Haryana, India in 2017 and is currently pursuing M.Tech in Computer Engineering Department of J. C.Bose University of Science and Technology,YMCA, Faridabad (Formerly YMCAUniversity of Science & Technology, Faridabad), Haryana, India.
Email: ifshita.chaudhary@ymail.com, Phone No. 8860935654

Shruti Sharma has a work experience of 13 years and currently working as an Assistant Professor in Computer Engineering Department of J. C. Bose University of Science and Technology, YMCA, Faridabad (Formerly YMCA University of Science & Technology, Faridabad). She received her B.Tech. in Computer Science and Engineering from LIMAT, Faridabad affiliated to MDU, Rohtak and M.Tech. in Computer Engineering from YMCA University of Science & Technology, Faridabad in 2004 and 2006, respectively. She is currently pursuing Ph.D. in CE from YMCA University of Science & Technology, Faridabad. She has published 7 research papers in reputed journals and conferences.

Preeti Sethi has a work experience of 17 years and currently working as an Assistant Professor in Computer Engineering Department of J. C. Bose University of Science and Technology, YMCA, Faridabad (Formerly YMCA University of Science & Technology, Faridabad). She received her B.E. in CE from M.D. University Rohtak and M.E. in CE from M.D. University Rohtak in 2001 and 2007, respectively. She has completed her Ph.D. from YMCA University of Science & Technology, Faridabad. She has published 15 research papers in reputed journals and conferences.