A NOVEL CONVOLUTIONAL NEURAL NETWORK FOR AIR POLLUTION FORECASTING

A DISSERTATION

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF

MASTER OF TECHNOLOGY
IN
INFORMATION SYSTEMS

Submitted by:
Bikash Kumar Sah
2K19/ISY/05

Under the supervision of
Dr. JASRAJ MEENA
(Assistant Professor Department of IT)

DEPARTMENT OF INFORMATION TECHNOLOGY
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi college of Engineering)
Bawana Road, Delhi-110042

(2019-2021)

CANDIDATE’S DECLARATION

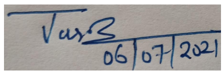
I, Bikash Kumar Sah, Roll No. 2K19/ISY/05 student of M.Tech, Information Systems, hereby declare that the project Dissertation titled “A Novel Convolution Neural Network Model for Air Pollution Forecasting ” which is submitted by me to the Department of Information Technology, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi **Bikash Kumar Sah**

Date: July 2021 **(2K19/ISY/05)**

CERTIFICATE

I hereby certify that the Project Dissertation" A Novel Convolution Neural Network Architecture Model for Air Pollution Forcasting" which is submitted by Bikash Kumar Sah, Roll No 2K19/ISY/05 Information Technology, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.



Place: Delhi **Dr. Jasraj Meena**

Date: July 2021 **SUPERVISOR**

ACKNOWLEDGEMENT

I express my gratitude to my major project guide **Dr. Jasraj Meena**, Assistant Professor Department of CSE, Delhi Technological University, for the valuable support and guidance he provided in making this major project. It is my pleasure to record my sincere thanks to my respected guide for his constructive criticism and insight without which the project would not have shaped as it has.

I humbly extend my words of gratitude to other faculty members of this department for providing their valuable help and time whenever it was required.

Bikash Kumar Sah

Roll No. 2K19/ISY/05

M.Tech (Information Systems)

Abstract

Air pollution was a global problem a few decades back. It is still a problem and will continue to be a problem if not solved appropriately.Various machine learning and deep learining approaches have been purposed for accurate prediction, estimation and analysis of the air polution. We have purposed a novel five layer one-dimensional convolution neural network architecture to forecast the PM2.5 concentration. It is a deep learning approach. We have used the five year air pollution dataset from 2010 to 2014 recorded by the US embassy in Beijing, China taken from the database from UCI machine learining repository [19]. The dataset we are considering is in the .csv format. The dataset consists of feature columns like “Number,” “year,” “month,” “day,” “PM2.5”, “PM10”, “S02”, “dew,” “temp,” “pressure,” “wind direction,” “wind direction,” “snow” and “rain.” The dataset consisted of a total of 43,324 rows and nine feature columns.The model yields the best results in predicting PM2.5 levels with an RMSE of 28.1309 and MAE of 14.9727. On statistical analysis we found that ur proposed prediction model outperformed the traditional forecasting models like DTR, SVR and ANN models for the air pollution forecasting.

CONTENTS

List of Figures

List of Tables

Chapter 1

INTRODUCTION

Air pollution is identified as a global problem since decades and its severity is increasing with the time. It causes various health issues in the human beings. According to Niharika et al [10] the various air pollutants are lead,flourides,chlorides, oxides of nitrogen, formaldehyde, oxides of nitrogen,CO2, carbon monooxide,sulphur dioxide,suspended particulate matter,Ammonia, ozone,etc. These air pollutants have always been identified as a major air pollutants for most of the researchers over the years in [1-5], [11] ,[16-17],[24-28]etc.

The causes that contributing to air pollution are well known to everyone and yet we ignore that and we keep on polluting the air by our activities. The root cause of air pollution starts from our houses and extends to most of the activities we have done and we have been doing in the name of modernization, technological advancement and so called urbanization and industraization. We are climbing the ladder of success at expenses of our lives.

Air pollution has a deleterious effect on human health. We all must make an attempt from all our levels to tackle this global issue. It has emerged as a problem few decades back , it is a matter of global concern today and if we don’t come up with efficient methods to hammer out this problem of air pollution it will engulf the entire possibilities of life on the Earth in no time.

1.1 Motivation

Thousands of researchers have published papers and their methods to tackle this air pollution considering various air pollutants. However, most of the researchers in the research community are have recognized mainly types of air pollution as indoor air pollution and outdoor air pollution as said in [1-3] and [10]. Most of the researchers are concerned mainly to solve air pollution by attempting to estimate , forecasting , and monitoring the pollutants levels as said in [1-12] There are some researchers who have also attempted to model the air polltuion transimssion behaviour and trying to provide you with the visual analytics of the air pollution propagation [13] and [12]. Meanwhile some researchers are interested in knowing the direct impact of air pollutants and estimate the air pollution health risks as said in [1] . Some are interestingly involved in providing you the best routes so that an individual gets very less exposed to air pollution risks as in [4]. Whether they are predicting the pollutants levels, monitoring the pollution , providing the analytics of the air polltuion transmission behaviour, trying to provide you the safest route so that people get less exposed to air pollution health risks , there must be some methods or techniques they must have adopted to cherish those research objectives. Some came up with solutions using the alreading available machine learning models like KNN, K-means clustering, Linear regression, MLP, SVM, decision tree, random forest , etc. Some came up with their own novel techniques to solve the air polltuion concerns as seen in [8],[9],[10],[11]and [12]. Some identified the air pollution as classification problem while some identified air polltuion as a regression problem and proposed their methods. While reseachers identified machine learning methods as major technique to solve the air polltuion they were doing a lot of hard work in extracting the appropriate features vectors that are going to have impact on the final target output as seen in [1-3]and[5]. Feature extraction is really an important step in handling any machine learning problem statement . With the advancement in science and technology, the computation power of the machines have drastically increased.

Today we have machines with high computation power using GPUS and TPUs which are thousand times faster than traditional CPUs. The availability of air pollution datasets is also not a problem as the entire world community has identified it as a global problem. The approaches to the handle huge datasets have become easy and cheap due to high storage capacity of computing machines and even if that is insufficient to you, cloud computing can used to handle those huge datasets. Sensors to sense various air polltuants have become cheaper with time and more available.Today even mobile devices come up with various sensors already installed in it. Due to high storage copacity, high availability of datasets, cheap prices of sensors, high computation power of machines , researchers have even come up with deep learning based solutions as in , cloud based solutions and IOT based solutions to solve the air pollution as backed by recent reasearch papers [9],[11] and [12]. Real life problems are more of unsupervised in nature and extracting relevant features is quite painful . We are just with datesets and huge datasets as input . In that case deep learning proves to be boon to solve problem statement of this type . Researchers in the recent years are analyzing air pollution as a deep learning problem statement and proposing the solution accordingly.

1.2 Objective

There are many researchers working in the area of air pollution forecasting, estimation, and analysis. Many research papers have have been published defending the effectiveness of their own purposed forcasting models. The old research papers were mainly using only the shallow machine learning techniques to resolve the challenging task of air pollution forcasting and analysis. With the elapsing time researchers working in the air pollution research domain have shown their interest in deep learning approaches to design the forecasting models.Most of the researchers have recognized PM2.5 as one of the major pollutant in the air.

The main objective of the thesis is to design a novel Convolution neural network model to forecast the PM2.5 concentration values. To ascertain the efficiency of the our forecasting model, we will be comparing its performance with the other contemporary computation models.

1.3 What is PM2.5 ?

$PM_{2.5}$ is minute particles, with diameters of 2.5 micrometers and smaller.This 2.5 micrometersis really really tiny. It is less than the size of one human hair. The average human hair is about 70 micrometers in diameter [29].The tiny size makes it harder to prevent PM2.5 from thriving into the body, and cause harm to our health. There is one more variant of PM particles called PM10 which is 10 microns in size and relatively larger.

In most of the recent research papers like [7],[25-28], the researchers have identified PM2.5 (particulate matter 2.5) as one of the significant air pollutant to be considered in one way or the other. They have all used datasets containing PM2.5 concentration values as one of the major component in air quality forecasting models or systems. A typical size comparision of PM2.5 is illutrated in the figure 1 :

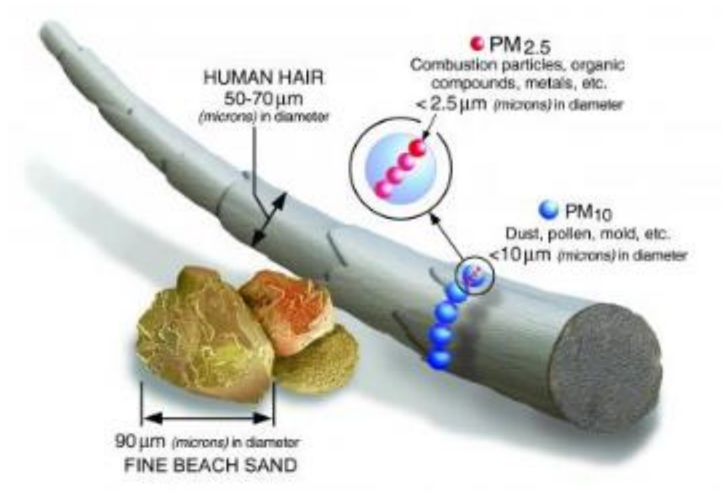


Figure 1: PM2.5 compared with human hair [29]

1.4 Training Models

1.4.1 Decision tree Regressor (DTR)

A decision tree regressor is a non-linear and non-continuous machine learning model.Suppose we have two variables X_1 and X_2 and you want to predict the third variable "y" based on the decision tree.Decision tree actually creates various splits for various cases of " X1" and "X2" and finnaly a tree like structure is obtained. All those splits are like intervals for various independent variable values.In the final decision tree many such intervals are formed. To predict the "y" variable on that interval the model just takes the average of the values in that interval. Decision tree regressor (DTR) can be easily applied for regression problems like prediction of PM2.5 concentration values [24].

1.4.2 Support Vector Regressor (SVR)

Support Vector Machine(SVM) is a popular machine larning algorithm. It can be suitable for regression as well as classification [11],[16]. In SVM what you do we analyze and plot data in any n number of dimensional space (where n symbolizes the total features you have) value of a particular coordinate in space would be the value of each feature. Now to achieve classification we need to search for a hyperplane which can equally differentiate two classes, either class 1 or class 2 and the distance between the hyperplane and successive classes should be uniform that is the best classification hyperplane solution we would prefer to achieve.

SVM tries to maximize the margin between the nearby datapoints and the nearby data points are called support vectors. In case of 2D space the boundary is a line. In case of 3D space the boundary is a plane . In case of nD (n-dimensional) plane/ n number of features its quite difficult to visualize and draw it. Mathematically still it is possible and that is called hyperplane.

Hyperplane is a plane in n-dimension that tries to separate out different classification groups.SVM creates a hyperplane in n-dimensional space such that the margin between classification groups gets maximized. A typical SVM hyperplane is shown in the Figure 2 which clearly shows the SVM hyperplane trying to maximize the margin between the nearby datapoints. The small circles and small rectangle in the Figure 2 are support vectors as seen. The decision boundary is also clearly seen in the Figure 2.

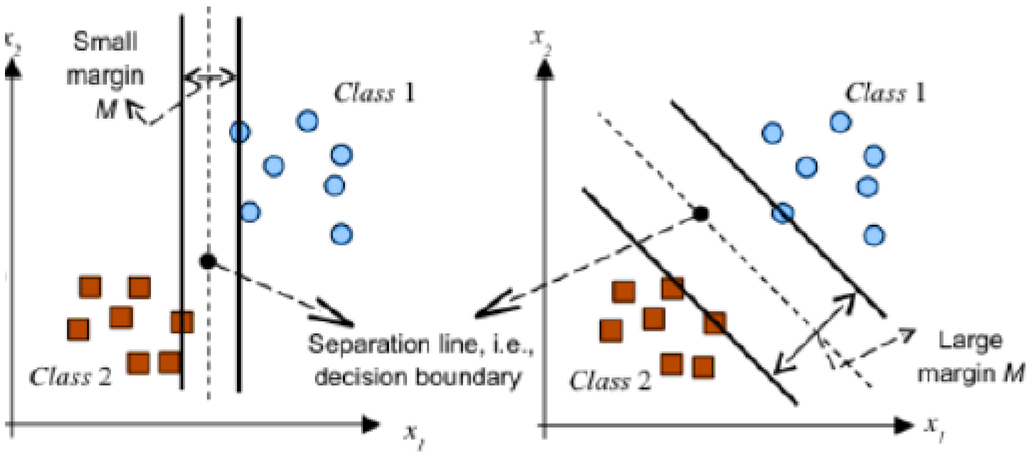


Figure 2: SVM Hyperplane

In recent papers like [16][11] we can see support vector machine have been succesfully applied to solve the regression problem of air pollution forecasting varying with time and weather and various other weather and meteorological condtions. We will also be implementing this Support vector regressor on the dataset used in our research to estimate and compare its effectiveness to make accurate PM2.5 predictions.

1.4.3 Artificial Neural Networks (ANNs)

A computer system consists of a series of simple and highly interconnected processing elements that processes information in response to the dynamic state of the external inputs.

Basic Structure of ANNs

The basic thought behind an ANN is to build an analogy with the working of the human brain. The human brain is composed of dendrites and neurons and in the same manner the ANN is built. So, the idea is to eliminate the working of human brain using wires and silicon as dendrites and silicon respectively.

There are billions of nerve cells in the human brain and they are connected to other thousands of cells by axons. Dendrites accept the stimuli from the external environment via sensory organs, further these stimuli are converted to the electric impulse which travels through the neural network and then the neurons communicate with the other neurons in order to handle the issue. According to [29] biological neural network can be illustrated as in the figure 3:

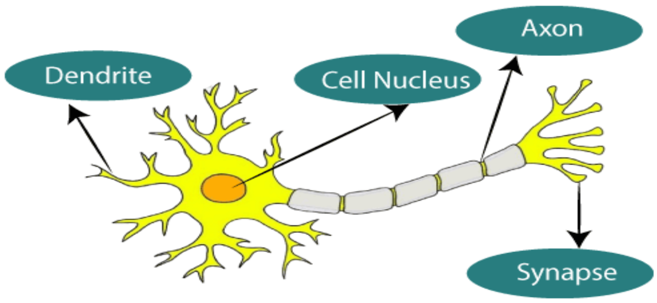


Figure 3: Biological neural network [29]

In the similar way, ANNs are made up of multiple nodes which work like neurons present in the human brain. The input data can be accepted by the nodes for the purpose of simple data operations and the results of these operations are passed to other nodes. The output of every node is known as its activation or node value. The nodes are connected with the links and these links are associated with some weights. Just like human brains, ANNs also posses the ability to learn and the learning process is carried out by adjusting the weights values of the links. The simple diagram of an ANN is in the figure 4:

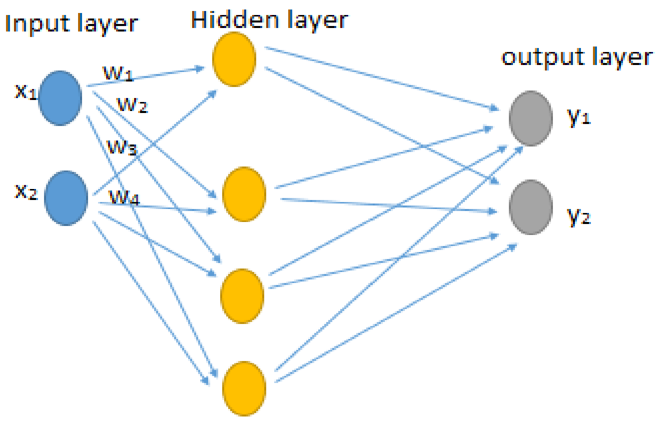


Figure 4: Artificial neural network

As shown in the figure 4 a typical ANN structure has three layers:

- Input layer
- Hidden layer and
- Output layer

Input layer:

Various input features say X_i (i=1,2,3……..n) ar fed to the input layer of the ANN. They are actually the independent varibles inour dataset.

Hidden layer:

The hidden layers in an ANN can be 1 or many. The various input features get multiplied to some weights as shown above in figure 5. The sum of producrs of weights wi and inputs Xi goes to hidden layer. Mathematically it is equivalent to saying that the yi (i=1,2,23,4…….n) goes to the i^{th} neuron or circle in figure 5.

$$yi = w1x1 + w2x2 + w3x3 + \ldots\ldots wnxn \qquad \text{Equation 1}$$

$$zi = Activation\_Function(yi) \qquad \text{Equation 2}$$

Activation function says which neuron will get triggered to pass the information to the next layer in the ANN. The famously used activaation functions are sigmoid funtion, threshold function, ReLu (rectilinear unit) , LeakyReLU, Hyperbolic tangent(tanh), etc.

Output layer:

Finally in the output layer we get the predicted values like y1 and y2 as shown in figure5. Mathematically, the ouputs of the ANNs can be indicated as y_i (i=1,2,23,4…….n).

We compare these predicted value with the actual value from the dataset and we just see the loss obtained. Then the ANN goes through backpropagation by readjusting all the weights w_i (i=1,2,23,4…….n) present on various links untill the loss is minimum between the actual and predicted value.

1.4.4 Convolution Neural Network (CNN)

A convolution neural network can capture the spatial and temporal dependencies successfully in an image by application of various relevant feature detectors oor kernels. The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and reusability of weights. In other words, the network can be trained to understand the complexities in the images better.

The main role of the CNN is reduction of times into something which is easier to process, without losing features which are for presiction purpose. Its major applications are in the areas of Computer vision, Image & Video recognition, Image Analysis & Classification, Recommendation Systems and Natural Language Processing. In a CNN model each input image passes through a series of convolution layers with kernels or feature detectors,Pooling ,flattening, fully connected layers (FC) and apply softmax function to classify an object with probablistic values between 0 and 1.

The steps followed in the convolution neural architecture for images generally follows the following steps :

- Convolution layer
- Polling
- Flattening
- Full Connection

Convolution layer: A convolution is basically the combined integration of the two functions ans it shows how one function modifies the shape of the other [31]. Quite popular in signal processing, image processing, etc. Mathematically the convolution operation is expressed as :

$$(f * g)(t) = \int_{-\infty}^{\infty} f(k)g(t-k)dk \qquad \text{Equation 3}$$

Convolution operation is applied between the input image “f” and kernel image “g” to get the feature map or say convolved feature. After convolution operation the image becomes smaller in dimension as only usefull features are picked and kept. The purpose of applying a kernel/feature detector is to pick the usefull features of an image like eyes, noes, tongue,etc .

ReLU layer: This layer is dedicated to increase non-linearity in our input images because our images are themselves higly non-linear. This alyer is an additional step on top of the convolution layer. On applying the applying the convolution operation between image and kernel we might risk into something linear. So to increase the non-linearity we this ReLU layer to breakup this linearity [32]. There are some other activation function used in CNN layer as said in [32].

Pooling layer: When ReLU is applied to the convolved feature many feature maps are obtained . These feature maps are passed to the pooling layer. Various pooling operations that can be applied is MaxPooling, Mean Poolin and Sum Pooling [33]. In case of MaxPooling we pick the maximum of the values in the stride. In case of Mean Pooling we pick the mean of the values in the stride. In case of Sum Pooling we pick the sum of the values in the stride. The main pupose of this layer is to reduce the number of parameters and fasten the overall computation of the model with appropraite pooling operations.This also prevents the overfitting of the model. After pooling operations pooled feature maps are obtained.

Flattening: Here, the pooled feature maps are flattened as a long single column flattened vector. It implies that , from the pooled feature map ,row by row the values are picked and that is put into one.

Fully connected (FC): All those feature maps from flattening layer are fed to FC layer which performs the voting for the classes we are expecting.

Figure 5 shows the schematic diagram of the CNN architecture we are talking about with all the layers arranged one after the other in sequence.

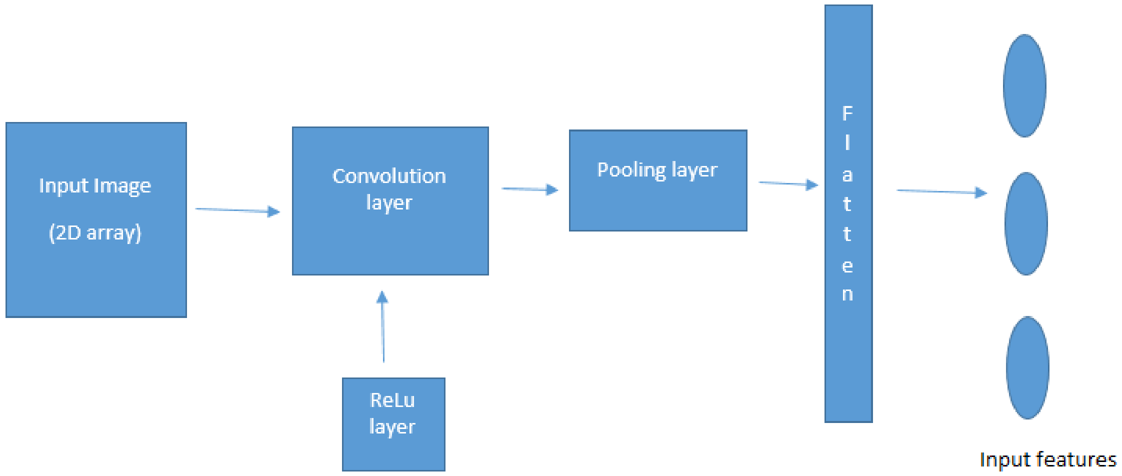


Figure 5: Schematic Diagram of CNN architecture

All of this training happens in forward as well as backward direction. In backward propagation of information, not only the weights are trained but also the feature detectors are trained and adjusted to come up with best feature maps.

Finally , when all these steps are over we get a well trained CNN model to perform the classification. CNN model can also used for both classification and regression problems[20-22],[30-33].

1.5 Organization of Dissertation

Chapter 2 has the discussion on the related work done in the research domain of air pollution forecasting.

Chapter 3 focuses on the problem identification part in the air pollution forecasting research domain.

Chapter 4 will have a detailed discussion on the proposed model/architecture for air pollution forecasting with schematic diagram of the model and lucid explanation.

Chapter 5 has a proper result analysis and model evalution for the proposed model.It incorporates various statistical approaches for the performance analysis of various forecasting model.

Chapter 6 talks about the dataset we have used, the simulation environment, the related softwares ,etc that assisted a lot in our experimentation purpose.

Chapter 7 delightfully presents the conclusions and the future work that can be extended to resolve this golbal problem air pollution problem by puposing some more efficient and reliable forecasting models.

Chapter 2

RELATED WORK

In this section we have reviewed the related work recently done in the field of air pollution forecasting. Air pollution has severe effects on every living organisms so the research community is showing its utmost interest in solving this problem since few decades.Various state-of-the-art methods have been purposed with their own unique computation models for forecasting, analysing and estimating the pollutants levels [1-12],[24-26].

If we trace the research papers approx 2 decades back, in a very old paper [27] they decided to focus on the outdoor pollution due to NO2 and successfully deployed Artificial Neural Network (ANN).Similarly, if we talk about the recent papers like [1] and [25-28], they have also recognized the ANN model as an important forcasting model for the air pollution forecasting. ANN model was a relevant forecasting model for air pollution 2 decades back and it is still relevant.

In most of the recent research papers like [7],[24-27], the researchers have identified PM2.5 (particulate matter 2.5) as one of the significant air pollutant to be considered in one way or the other. They have all used datasets containing PM2.5 concentration values as one of the major component in air quality forecasting models or systems.

As stated in [16], Support vector machine (SVM) regression model can be relaibly used to forecast the air pollution causing pollutants. They considered the outdoor polutants PM2.5,SO2,CO,NO2,O3,etc as the major air pollutants . The air pollutant data in three major cities of China i.e., Beijing, Taianjin and Shijianhuay was taken into design the air pollution forecasting model.Their model used six pollutants, five meterlogical condition and air quality index for pollution forecasting.

Decision tree regressor (DTR) can be easily applied for regression problems like prediction of PM2.5 concentration values [23].A decision tree regressor is a non-linear and non-continuous machine learning model.Suppose we have two variables X_1 and X_2 and you want to predict the third variable "y" based on the decision tree.Decision tree actually creates various splits for various cases of " X1" and "X2" and finnaly a tree like structure is obtained. All those splits are like intervals for various independent variable values.In the final decision tree many such intervals are formed. To predict the “y” variable on that interval the model just takes the average of the values in that interval. We have also re-implemented this DTR forecasting model to evaluate the efficacy of our own model.

As stated in [9] they have proposed a novel STDL i.e. spatiotemporal deep learning based air quality prediction method.A stacked autoencoder (SAE) model is used to extract implicit air quality features . The output of this stacked autoencoder layer is fed as input to the Logistic regression to make real value predictions .Their prediction model outperformed tradtional prediction model like the spatiotemporal artificial neural network (STANN), auto regression moving average (ARMA), and support vector regression (SVR) models.

As discussed in [17] PM2.5 is recognized as the major component of the smog and the improvement of the air quality can be achieved by controlling the PM2.5 concentration. In order to predict the PM2.5 concentration [16] proposed convolution based bidirectional gated recurrent network (CBGRU). It combines one dimensional convnets with bidirectional GRU. Similarly, [18] proposes a recurrent neural network with LSTM(long short term model) model to forecast the PM2.5 concentration values for the next four hours at 66 stations around the Taiwan. The preprocessed dataset is fed to the Autoencoder to extract the useful features that actually affect the PM2.5 values. The output from Autoencoder is given as input features for further analysis into the Bi-LSTM layer and finally we get PM2.5 prediction values.

With the elapsing time, researchers have used different deep learning approaches to forecast the air pollution so that is also our motivation to opt for deep learning approach in our research work.

Chapter 3

PROBLEM IDENTIFICATION

This chapter is thesis primarily focuses in the research problem identification part. As discussed in the previous two chapters various forecasting models have been purposed to resolve the problem of air pollution. Various methods like Decision tree regressor (DTR), Support vector regression (SVM) , Artificial neural networks(ANNs) , LSTM RNN, CBGRU,etc have proved their efficiency and reliability to achieve accurate air pollution forecasting [11],[16],[17],[24],[25-28].

In most of the old papers only shallow machine learning models were being used to tackle the air pollution research domain. With the elapsing time, deep learning approaches also came into the picture to resolve the air pollution forecasting problem with even more accuracy and precision.

Many of the state-of-the-art research methods have recognized PM2.5 as one of the significant pollutant causing the air pollution and severe health impacts on living organisms. Meanwhile both machine learning methods and deep learning approaches are found effective in air pollution forecasting. After the deep inspection of all these scenario we decided to adopt the deep learning approach to design our air pollution forecasting . We are aiming to predict the PM2.5 concentration values using our own deep learning approach.

Chapter 4

PROPOSED MODEL/ARCHITECTURE

We have proposed a novel CNN (convolution neural network) architecture to predict the PM2.5 concentration values.To ascertain the effectiveness of our proposed model we have compared it with other forecasting models like SVR,DTR and ANN. Genaerally CNN architecture is used for the classification tasks like image classification, object detection,facial expression recoginition,etc. The CNN architecture famously revolves around images as stated in past research papers like [20-23].However, we have attempted to use CNN architecture for PM2.5 prediction.Working with images on CNN is just like working with 2d array or 3d array as input feature to the CNN model.

The novel CNN architecture we have proposed is illustrated in the figure(6). It is a one dimensional convolution neural network architecture. We have kept this model one dimensional as we considering only PM2.5 concentraion values varying over time for our research work. Our novel CNN model has a five layer deep convolution layer each. Each convolution layer has 60 2*1 kernals applied in them and a ReLU (rectilinear unit) activation is applied to each convolution. Each layer within our one dimensional convolution layer is stacked one over the other as shown in figure (6).

Input layers takes all the input features X_i (i=1,2,3,4……….n) which is actually the PM2.5 values fed into the model in the timesteps of 10,20,30,45,60,75,90 and 120 as X_train i.e. training dataset. These inpuit features is first fed into the first one dimensional convolution layer spread over 60 neurons.Then a 2*1 kernel is applied to the input feature and the convolution operation is performed between input features and the filter. Once the convolution operation is done , ReLU activation is applied to break up the linearities as our problem is more of a non-linear PM2.5 values varying over time. The output of the first one dimensional convolution layer is input to second convolution layer .The convolution operation is performed again in the second layer followed by ReLu activation. This is repeated in 3rd, 4th and 5th convolution layer.

After the convolution layer is over, the convolved feature map is passed through subsequent MaxPooling1D layer, Flattten layer and finnaly to the Dense layer. In the MaxPooling1D layer we pick the maximum value from each stride and finally get our pooled feature map. The output of pooled feature map is stacked one over the other in one single long column in the flattening layer. Finally, we have a single neuron layer Dense layer The output from the dense layer produces the predicted value of PM2.5 concentration values .

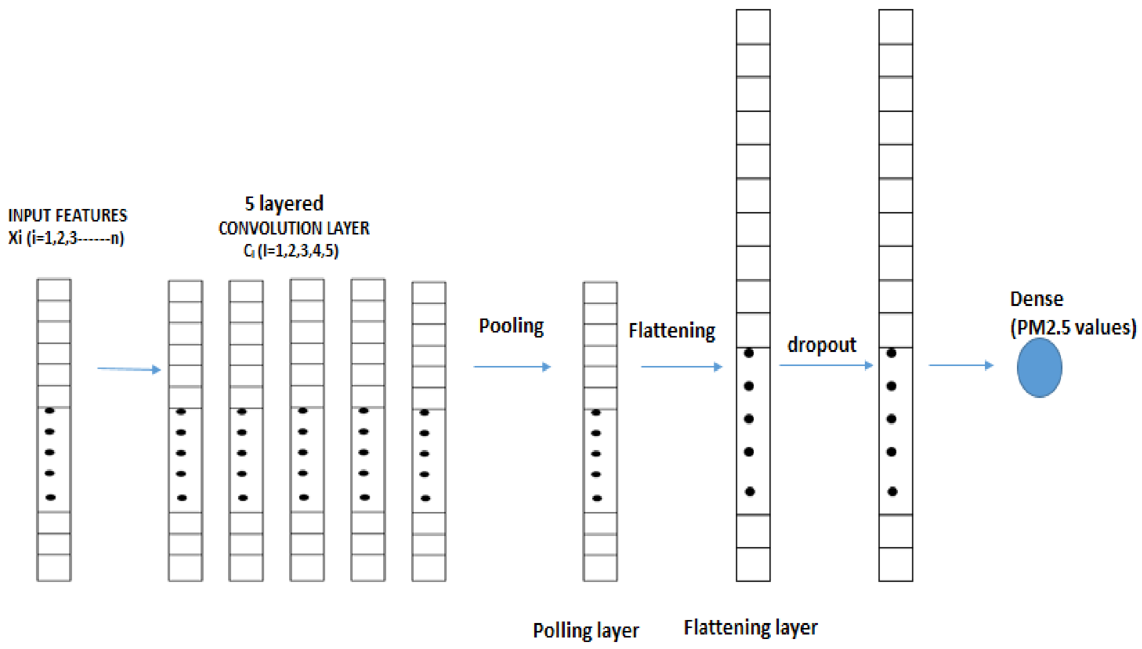


Figure 6: A novel CNN Architecture for PM2.5 prediction

The model summary of our novel one dimensional convolution network for PM2.5 concentration can be seen in the Figure 7 .The model summary clearly shows the enitre working process of our novel CNN architecture. It has a total of 36,901 parameters and it is evidently seen that the all the parameters are trainable. The model summary of our novel CNN architecture also shows that the proposed model has 5 layer deep one dimensional convolution layer. After the convolution layer the convolved feature map passes through the ReLU function in each convolution layer. This final convolved feature map is fed to the pooling layer. In the pooling layer we can see the size of feature map has further reduced and that is shown also in Figure 7 depicting the model summary. The max pooled feature maps are then given to flattening layer. The long flattened vector of single column has the size equal to the product of dimension of pooled feature map and this is also justfiable from figure 7 showing the model summary . Finally , we do a dropout and enter to the dense layer of just single neuron and no activation funtion applied to it.In dense layer itself PM2.5 concentration values will be predicted and compared with actual PM2.5 values.

```
Model: "sequential_3"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d_13 (Conv1D)           (None, 19, 60)            180
_________________________________________________________________
conv1d_14 (Conv1D)           (None, 18, 60)            7260
_________________________________________________________________
conv1d_15 (Conv1D)           (None, 17, 60)            7260
_________________________________________________________________
conv1d_16 (Conv1D)           (None, 16, 60)            7260
_________________________________________________________________
conv1d_17 (Conv1D)           (None, 15, 60)            7260
_________________________________________________________________
conv1d_18 (Conv1D)           (None, 14, 60)            7260
_________________________________________________________________
max_pooling1d_3 (MaxPooling1 (None, 7, 60)             0
_________________________________________________________________
flatten_3 (Flatten)          (None, 420)               0
_________________________________________________________________
dropout_3 (Dropout)          (None, 420)               0
_________________________________________________________________
dense_3 (Dense)              (None, 1)                 421
=================================================================
Total params: 36,901
Trainable params: 36,901
Non-trainable params: 0
```

Figure 7 : Model summary of our novel CNN architecture.

Chapter 5

THE EXPERIMENTAL APPROACH

In this chapter we talk about the experimental approach we have used to design our novel CNN model/architecture. The following system configuration has been required while conducting the experiments:

- Processor: Intel Core i5
- Main Memory: 8 GB
- Hard Disk Capacity: 1 TB (for faster processing)
- Software Used: Spyder 3.6 and Anaconda.
- Graphics: AMD Radeon

Some of the experiments were also conducted on Google collaboratory.

5.1 DATASET

We have used the five year air pollution dataset from 2010 to 2014 recorded by the US embassy in Beijing, China taken from the database from UCI machine learining repository [19]. The dataset we are considering is in the .csv format. The dataset consists of feature columns like "Number," "year," "month," "day," "PM2.5", "PM10", "S02", "dew," "temp," "pressure," "wind direction," "wind direction," "snow" and "rain." The dataset consisted of a total of 43,384 rows and nine feature columns. On the same dataset, we are applying our proposed model and the other forecasting models like SVR (support vector regressor), DTR(decision tree regressor) and ANN(artificial neural network). A sample of the dataset used for our research work is shown in figure 8.

Number	year	month	day	hour	pm2.5	DEWP	TEMP	PRES	wind dire	wind spee
1	2010	1	1	0	NA	-21	-11	1021	NW	1.79
2	2010	1	1	1	NA	-21	-12	1020	NW	4.92
3	2010	1	1	2	NA	-21	-11	1019	NW	6.71
4	2010	1	1	3	NA	-21	-14	1019	NW	9.84
5	2010	1	1	4	NA	-20	-12	1018	NW	12.97
6	2010	1	1	5	NA	-19	-10	1017	NW	16.1
7	2010	1	1	6	NA	-19	-9	1017	NW	19.23
8	2010	1	1	7	NA	-19	-9	1017	NW	21.02
9	2010	1	1	8	NA	-19	-9	1017	NW	24.15
10	2010	1	1	9	NA	-20	-8	1017	NW	27.28
11	2010	1	1	10	NA	-19	-7	1017	NW	31.3
12	2010	1	1	11	NA	-18	-5	1017	NW	34.43
13	2010	1	1	12	NA	-19	-5	1015	NW	37.56
14	2010	1	1	13	NA	-18	-3	1015	NW	40.69
15	2010	1	1	14	NA	-18	-2	1014	NW	43.82
16	2010	1	1	15	NA	-18	-1	1014	cv	0.89
17	2010	1	1	16	NA	-19	-2	1015	NW	1.79
18	2010	1	1	17	NA	-18	-3	1015	NW	2.68
19	2010	1	1	18	NA	-18	-5	1016	NE	1.79
20	2010	1	1	19	NA	-17	-4	1017	NW	1.79
21	2010	1	1	20	NA	-17	-5	1017	cv	0.89
22	2010	1	1	21	NA	-17	-5	1018	NW	1.79

Figure 8 : Raw Dataset

5.2 DATA PREPROCESSING

The dataset needs to be preprocessed before moving ahead with designning the model. We parsed the date-time into a pandas data frame in the appropriate format. We have given proper names for all our columns present in the dataset. The column named "Number" was of no use. We got rid of that column. The columns like "year", "month", "day" and "hour" were merged into one column called "date" .After this change the dataset appears as shown in figure 9.

	pollution	dew	temp	pressure	w_dir	w_speed	snow	rain
year_month_day_hour								
2010-01-02 00:00:00	129.0	-16	-4.0	1020.0	SE	1.79	0	0
2010-01-02 01:00:00	148.0	-15	-4.0	1020.0	SE	2.68	0	0
2010-01-02 02:00:00	159.0	-11	-5.0	1021.0	SE	3.57	0	0
2010-01-02 03:00:00	181.0	-7	-5.0	1022.0	SE	5.36	1	0
2010-01-02 04:00:00	138.0	-7	-5.0	1022.0	SE	6.25	2	0

Figure 9: After merging year,month,day and hour into single column

On running the info() method on our dataset, we learned about the datatype of all our feature columns which is illustrated in figure 10. Meanwhile, it also helped us analyze the various null values present in the dataset.

```
dataset.info()

<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 43824 entries, 2010-01-01 00:00:00 to 2014-12-31 23:00:00
Data columns (total 9 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   No      43824 non-null  int64
 1   pm2.5   41757 non-null  float64
 2   DEWP    43824 non-null  int64
 3   TEMP    43824 non-null  float64
 4   PRES    43824 non-null  float64
 5   cbwd    43824 non-null  object
 6   Iws     43824 non-null  float64
 7   Is      43824 non-null  int64
 8   Ir      43824 non-null  int64
dtypes: float64(4), int64(4), object(1)
memory usage: 3.3+ MB
```

Figure 10: Datatype of all feature columns with non-null values in them

. We replaced all the null values present in the dataset with mean value of that data column. After replacing null values in the dataset with appropriate values, the statistical analysis of the dataset can be obtained using the dataset.describe() on the Google colab. A sample of few important statistical values on the given dataset is shown in figure 11.

	pollution	dew	temp	pressure	w_speed	snow	rain
count	43800.000000	43800.000000	43800.000000	43800.000000	43800.000000	43800.000000	43800.000000
mean	98.613215	1.828516	12.459041	1016.447306	23.894307	0.052763	0.195023
std	89.877910	14.429326	12.193384	10.271411	50.022729	0.760582	1.416247
min	0.000000	-40.000000	-19.000000	991.000000	0.450000	0.000000	0.000000
25%	31.000000	-10.000000	2.000000	1008.000000	1.790000	0.000000	0.000000
50%	77.000000	2.000000	14.000000	1016.000000	5.370000	0.000000	0.000000
75%	132.250000	15.000000	23.000000	1025.000000	21.910000	0.000000	0.000000
max	994.000000	28.000000	42.000000	1046.000000	585.600000	27.000000	36.000000

Figure 11: Few important statistical values in dataset

Here, we are interested in predicting the PM2.5 concentration values. So, we get rid of all columns from the dataset except the columns “PM2.5” and “date.” We even renamed our PM2.5 column as pollution level as we are very much interested in forecasting the PM2.5 concentration in this research work.Final sample of dataset that will be fed into the proposed model is shown in figure 12:

	pollution
year_month_day_hour	
2014-12-31 19:00:00	8.0
2014-12-31 20:00:00	10.0
2014-12-31 21:00:00	10.0
2014-12-31 22:00:00	8.0
2014-12-31 23:00:00	12.0

Figure 12: Preprocessed Dataset

5.3 IMPLEMENTATION

Before implementing our actual forecasting model , we needed some more data preparation for the dataset we have taken from UCI machine leairning repository [19]. We normalized our dataset within a small range of 1 and 0 using MinMaxScaler from sklearn python library. The normalized dataset can affect the ultimate performance of any computational model, so we incorporate this critical step before actually executing the model. We transformed our problem into a supervised learning problem.

We have split our dataset into training dataset and the test dataset.We have corrresponding X_train and Y_train sliced out of the training dataset. Based on the timestamps of 10, 20,30,45,60,75,90 and 120 timesteps , we have attempted to predict the PM2.5 concentration values of the 11th, 21st, 31st, 46th, 61st, 76th,90th and 121st timestep. As for instance , if we are taking 10 timesteps as X-train part , then Y_train part would be 11th timestep PM2.5 concentration value. Similarly, it goes the same way for the rest of the timestamps considered for the experimentation.

5.4 MODEL FITTING

We have a colossal dataset extended over 48324 rows. We have split the data into train(80 %) and test (20%) datasets. We have trained our one dimensional convolution network model on the training data with its X_train and Y_train values. We also have a new dataset section with Y_test and X_test values used to test the model.

Here, we have purposed a novel CNN architecture to forecast PM2.5 values. We are feeding our input feature vector into our first 1D convolution layer and the subsequent 4 layers 1D convolution layer receive input from the previous convolution layer. Then those input vector passes through the hidden layer with 60 neurons. Each neuron in the hidden layer process the input features by applying convolution operation between input feature and the kernel. In the optimal scenario we have kept kernel sizes like 2x1in each 1D convolution layer. Then after convolution ReLu activation is applied onto that reduced feature map. Many feature maps are obtained with relatively lesser dimensions.

The activation function used in the hidden layer is Relu function i.e. rectified linear unit activation function. Relu is frequently used activation function. Relu function is half rectified function i.e. when the input is negative the output will be 0. Relu activation function can be seen in the Figure 13 . A typical equation of the Relu activation function is given as

$$y = \max(0,x) \qquad \text{Equation (4)}$$

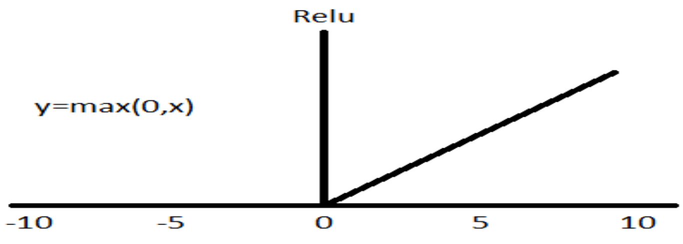


Figure 13 : ReLU activation function

After the convolution layer, the max pooling is applied which further picks more valuable features for the forecasting model. By this time, pooled feature maps are seen to have even more reduced size. After obtaining those pooled feature maps , we take them as a single large vector in the flattening layer. Since the flattening layer produces an long one column vector , we apply the dropout rate of 0.4 to avoid overfitting of our model. Finally, we feed these values to a dense layer with only one neuron to predict the PM2.5 concentration values. PM2.5 values and the predicted PM2.5 values varies during the initial run of the model. In order to minimize the errors between the actual and predicted PM2.5 values we need an optimizer.

To optimize the model , to minimize the loss and to get a more accurate model, we apply the “adam” optimizer in the model. We did not go for “sgd” optimizer aas it would have been hectic to adjust weights now and then to minimize the overall loss of the prediction model.With the use of “adam” optimizer it becomes quite easy to avoid the problems of gradient descent problem and the explodding gradient problem in the prediction model we have aimed for.

Chapter 6

RESULTS ANALYSIS

We evaluated our novel five layer one dimensional convolution model for various timestamps of 10, 20, 30 , 45 ,60 ,75 ,90 and 120 timesteps and we predicted the corresponding 11th, 21st, 31st, 46th,61st,76th,91st and 121st timesteps PM2.5 prediction values. We trained our model on the training dataset and we tested our model on the test dataset. After the succesfull implementation of the model, we measured the RMSE (i.e. root mean squared error) values for the various timestamps considered.As we go on incresing the timestamp sizes , the flattened vector size in the model goes on increasing .For the experimentation conducted on the various timesteps, the proposed model shows the results as shown in th**e Table 1 and Table 2 .**

Table 1 illustrates the effect of varying the timesteps on the normalized dataset and running that on our proposed novel CNN model. As discussed in section 5.3 our dataset was normalized and brought in the range of 0 and 1 before actual model training. This was done to increase the overall model performance and fasten its computation . Here we started with the timesteps of 10 as input feature to the novel CNN model and yeild the RMSE of 0.0281 , MSE of 0.009 and MAE of 0.0143. Then till timesteps of 30 we got high RMSE, MSE and MAE value . At timestep 45 , we got the RMSE, MSE and MAE values reduced (RMSE = 0.0281, MSE = 0.009 and MAE = 0.0143). We further extended our experimentation for the timesteps of 60,75, 90 and 120 timsteps and expected the prediction for 61st, 75th, 90th, and 121st timesteps. Finally we found that the proposed novel CNN model performs optimal at 45 timesteps and it performs worst for 120 timesteps. The results in Table 1 is obtained from the experimentation done on our novel CNN architecture on the Google collaboratory platform. The results in Table 1 are based the following parameters settings :

- 60 2*1 filters applied on each convolution layer
- ReLU applied on top of each convolution layer
- Dropout 0.4
- 50 epochs
- batch size of 64
- adam optimizer

Table 1: Effect of varying timesteps on our proposed model (normalized dataset)

Timesteps Sizes	Rmse (root mean sqaured error)	MSE(mean squared error)	MAE(mean absolute error)	Flattened vector size
10	0.0395	0.0015	0.0229	240
20	0.0399	0.0016	0.0230	540
30	0.0403	0.0016	0.0231	840
45	**0.0281**	**0.0009**	**0.0143**	**1320**
60	0.0394	0.0394	0.0228	1740
75	0.0326	0.0010	0.0183	2220
90	0.0397	0.0015	0.0231	2640
120	0.0405	0.0016	0.0249	3540

Table 2 depicts the effect of varying the timesteps on our novel CNN architecture. After the model training part is over, we rescale the scaled dataset back to original dataset applying the inverse transform and some other minor changes. First wee started with applying the timesteps of simply 10 timesteps to the proposed model and we got the RMSE = 39.3465 and MAE of 23.7071 .Again we applied 20 and 30 timesteps and the model was still performing bad with higher values of RMSE and MAE . Then we further varied the timesteps as 45, 60 , 75 , 90 and 120 timesteps. We observed that the model yields least error for the 45 timesteps taken (RMSE =28.1309 and MAE=14.9727). The model performs the highest RMSE and MAE value for 120 timesteps . The results in Table 1 is obtained from the experimentation done on our novel CNN architecture on the Google collaboratory platform. The results in Table 2 are based the following parameters settings :

- 60 2*1 filters applied on each convolution layer
- ReLU applied on top of each convolution layer
- Dropout 0.4
- 50 epochs
- batch size of 64
- adam optimizer

Table 2: Effect of varying timesteps on our proposed model (unscaled original dataset)

Timesteps Sizes	Rmse (root mean squared error)	MAE (mean absolute error)	Flattened vector size
10	39.3465	23.7071	240
20	39.5523	23.8311	540
30	40.1409	24.1857	840
45	**28.1309**	**14.9727**	1320
60	39.1912	23.6135	1740

75	32.1434	19.3671	2220
90	39.5516	23.8297	2640
120	40.3345	24.2923	3540

By varying the timesteps we are calculating RMSE, MSE and MAE values for our proposed model in both the unscaled and scaled versions of the dataset, we conducted our experimentation for our novel CNN model.

MSE(mean square error) is calculated as the average of squares of the difference of the actual values and the predicted values by our model . Here the actual value is the actual PM2.5 values and the predicted value will be the predicted PM2.5 values by our proposed forecasting model. Mathematically the MSE(mean squared error) is given by :

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y - \hat{y})^2 \qquad \text{Equation (5)}$$

RMSE(root mean square error) is calculated as the square root of MSE values simple. Mathematically, RMSE is given by the formula :

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y - \hat{y})^2} \qquad \text{Equation (6)}$$

Mean Absolute Error(MAE) is the average of absolute difference of the actual values and the predicted values.Mathematically MAE can be represented as :

$$\text{MAE} = \frac{1}{m} \sum_{1=1}^{m} |\, y - \hat{y} \,| \qquad \text{Equation (7)}$$

In all the equations for RMSE, MSE and MAE "y" indiactes the actual values and "$\hat{y}$" indicates the predicted values by the model. The actual values are actual PM2.5 concentration values and the predicted values are the predicted PM2.5 values resulted from the forecasting model.

It is seen that the proposed model has the least error values for the 45 timesteps experimented both on the scaled and the unscaled dataset . The proposed model best predicts the PM2.5 concentration values for the 45 days timestamps taken for unscaled original dataset and yields the least RMSE value of just 28.1309. as compared to the rest of the timestamps considered in the proposed model. Similarly , if we consider the experimental results found by applying the proposed model on the scaled dataset we get the least RMSE value of only 0.0281 for 45 days timestamps.

We have also analyzed the training and validation accuracy of our novel CNN mode over various epochs for 45 timesteps which was giving optimal performance for the model. We can see Figure 14 that illustrates the training and validation loss when the proposed model is run till 300 epochs with 64 batch size.The graph in Figure 14 shows that the training and validation loss line dipicted by red line and blue starts moving apart after 50 epochs. Generally, we should aim for epochs such that the lines for training loss and validation loss are as close as possible.The values for the training and validation loss should be as minimum as possible. From the results obtained in Table 1, Table 2 and Figure 14 below, we found that our proposed model gives the best results for 50 epochs with the least difference between the training loss and validation loss.

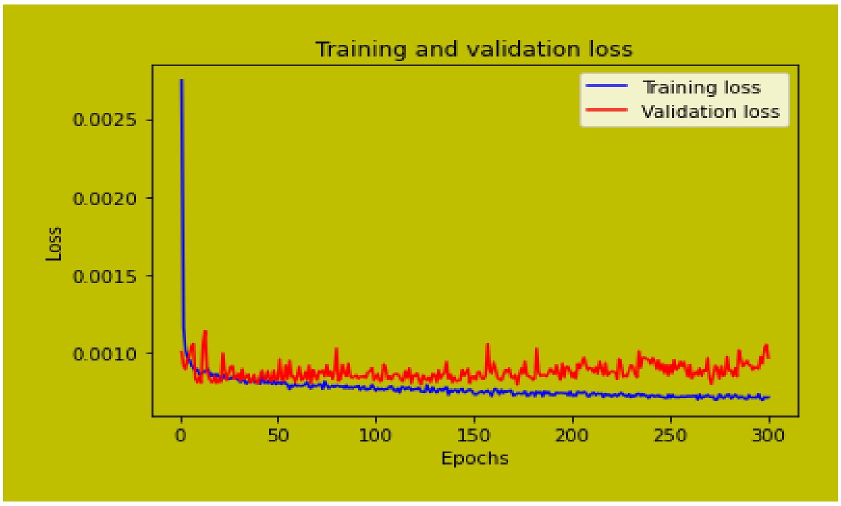


Figure 14: Training and validation loss for 45 timesteps,64 batch size and 300 epochs

The experimentation on observing the effect of varying the timestamps on the actual versus predicted pollution level (i.e. PM2.5 values) values graph is also deeply analysed .
Figure 15 shows the graphical representaion of the actual pollution level and the predicted pollution level for the timesteps of 10. In this case we obtained the RMSE as 39.3465 and MAE as 23.7071. We the actual and predicted value prediction lines are trying to follow the trend but they produced a bad RMSE value and MAE value.

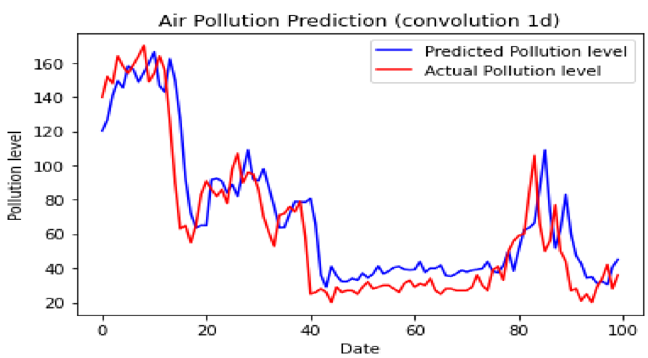


Fig 15 : Pm2.5 prediction based on 10 timesteps

Similarly, figure 16 and figure 17 shows the graphical representaion of the actual pollution level and the predicted pollution level for the timesteps of 20 and timesteps of 30.We also the predicted value prediction lines tries its best to follow the actual value line but still fails with a bad RMSE and MAE value.

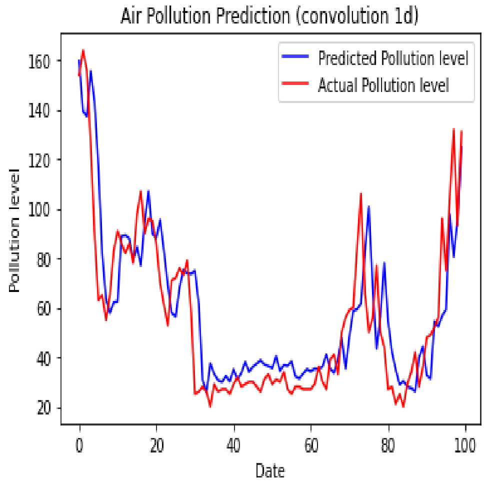


Fig 16 : Pm2.5 prediction based on 20 timesteps

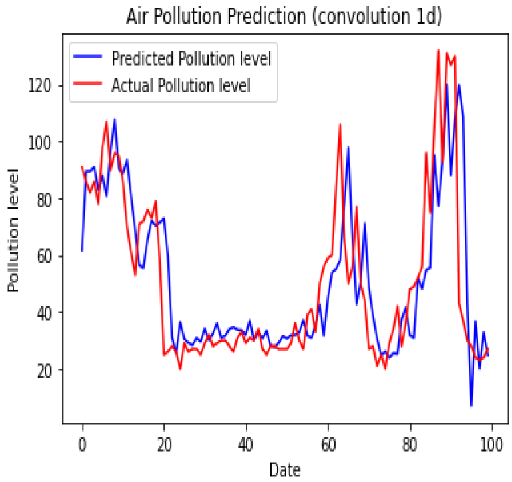


Fig 17: Pm2.5 prediction based on 30 timesteps

Figure 18 shows the graphical representaion of the actual pollution level and the predicted pollution level for the timesteps of 45.In this case the actual pollution level and predicted pollution level overlaps very closely over one another. Resultantly, we get the least value of RMSE and MAE values (RMSE= 28.1309 and MAE = 14.9727).

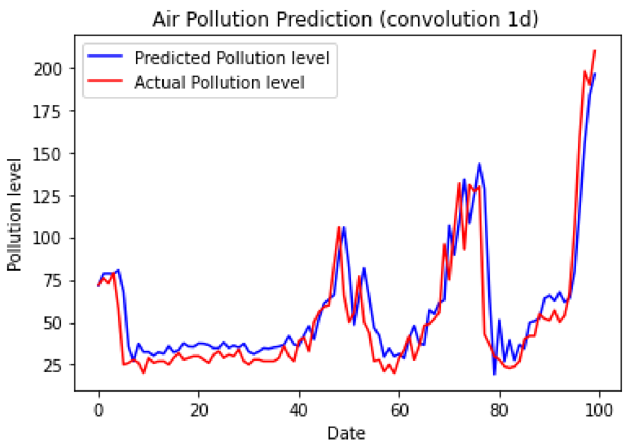


Fig 18 : Pm2.5 prediction based on 45 timesteps [best]

Figure 19 , Figure 20 and Figure 21shows the graphical representaion of the actual pollution level and the predicted pollution level for the timesteps of 60,75,90 timesteps . The RMSE and MAE values for these consideration of timesteps shows better resuts than results shown in figure 22.

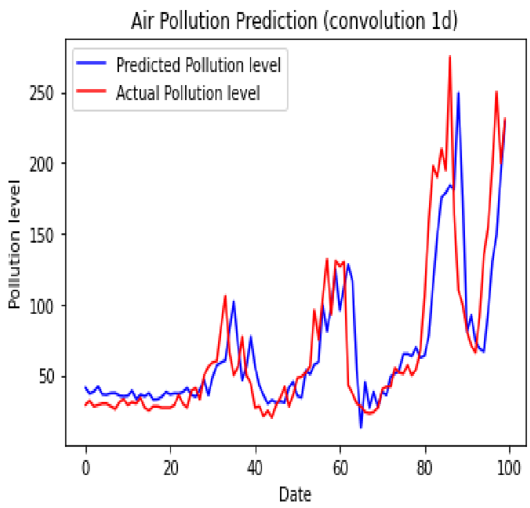


Fig 19 : Pm2.5 prediction based on 60 timesteps

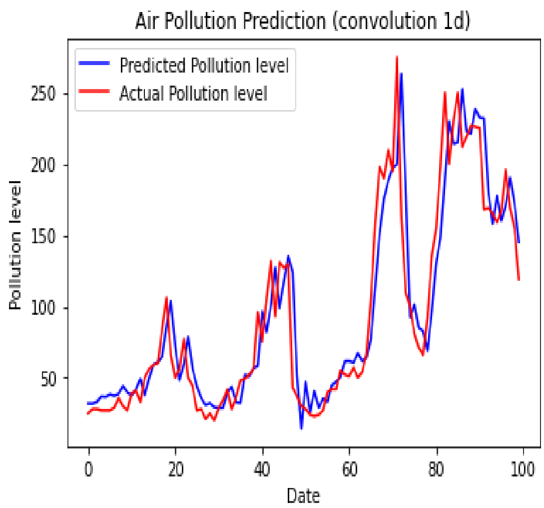


Fig 20 : Pm2.5 prediction based on 75 timesteps

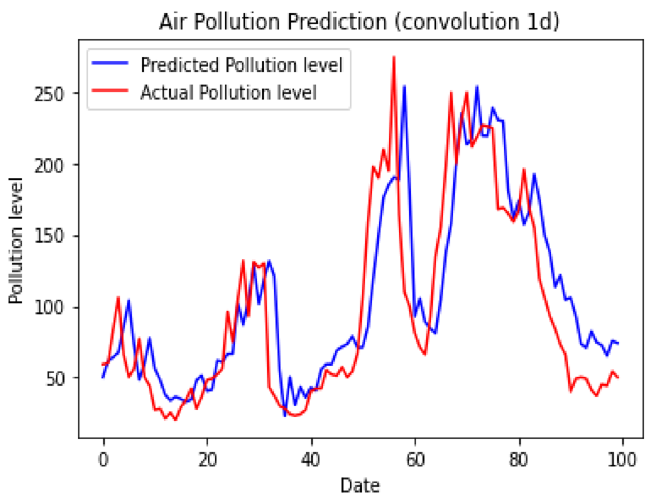


Fig21 : Pm2.5 prediction based on 90 timesteps

Figure 22 shows the worst results for 120 timesteps with the highest value of RMSE and MAE. Its predicted pollution level line did not follow the trend of actual pollution level over the time.At main points the actual pollution level red line is seen above the predicted pollution level line as seen in Figure 22. And at many points the value of predicted pollution level has surpassed the value of actual pollution level as seen in figure 22.

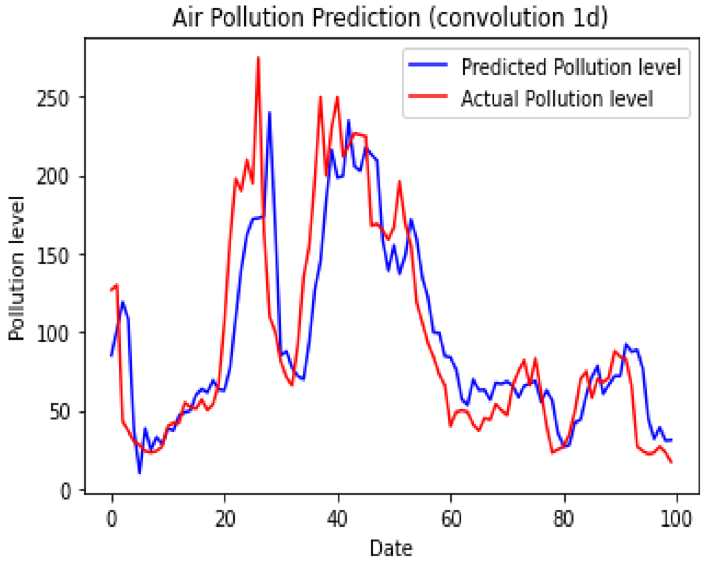


Fig22 : Pm2.5 prediction based on 120 timesteps

The results illustrated in graphs shown in Fig 15 to Fig 22 and the tables Table 1 and Table 2 are based on 50 epochs, each convolution layer with 60 2x1 filters, batch size of 64 and the dropout of 0.4.

In the nutshell, the results obtained from the above 8 predictions graphs and the RMSE values from above two tables Table 1 and Table 2 , we see that the proposed model gives the best prediction results for 50 epochs and 45 timesteps on the given dataset. For the 45 timesteps, the proposed model yields the least RMSE values of 28.1309 only. Even from the graphical analysis of actual versus predicted pollutant levels (i.e. PM2.5 values) it is very much evident that the 45 timesteps shows the actual value line and the prediction line in the graph is very much overlapping as compared to the other seven graphs. It implies that our proposed model gives best results for 45 timesteps (RMSE = 28.1309) and worst result for the 120 timesteps (RMSE= 40.1567).

To ascertain the effectiveness of our propopsed novel CNN model, we have compared its performance with the other forecasting models like SVR, DTR and ANN as shown in Table 3. As stated in past papers [11],[16],[24], [25-28] researchers have used support vector regression(SVR), decision tree regression(DTR) and artificial neural network(ANN) for air pollution forecasting. They have got really good results and they have defended the better performance of thier approach. We have re-implemented these traditional computational model DTR, ANN and SVR used in old research papers. We have trained our proposed model till the model converges and became optimal. We have trained our model on the training set and we tested our model on the test set. Each training set data points were fed as 10, 20,30,45,60,75, 90 and 120 timesteps for training the model. When the model converged to optimality, we compared our model performance with the traditional forecasting models already in this research domain.

We have tabulated our results in the **Table 3** using the statistical approaches root mean squared error, mean squared error and mean absolute error for evaluating the various forecasting models and conducted our comparitive analysis.

Table 3 shows that the traditional prediction models like SVR, DTR and ANN have relatively higher RMSE and MAE value as compared to our proposed model. The ANN forecasting model for PM2.5 prediction produces the RMSE and MAE value relatively higher than all other model. The performance of DTR forecasting model is found better than SVR model in PM2.5 prediction due its relatively lower RMSE and MAE values. It is also evident from the Table 3 that DTR has lower RMSE value than both SVR and ANN forecasting model. The SVR, DTR and ANN model have relatively higher RMSE and MAE value than our proposed model.

Our novel CNN architecture predicts the PM2.5 concentration values with the RMSE of 28.1309 and MAE of 14.9727 which is the least obtained errror value. By model performance comparision, ou purposed model outperforms the forecasting models like DTR,SVR and ANN. It clearly shows the effectivenes of our novel CNN model for PM2.5 forecasting.

Table 3: Comparision of Model Performance of various forecasting models

Method	**Rmse (root mean sqaured error)**	**MAE(mean absolute error)**
Support Vector Regression [11],[16]	54.6629	30.0943
Decision tree Regression [24]	44.0914	24.4900
Artificial neural network [25][26][27][28]	91.5041	49.7031
Novel CNN	**28.1309**	**14.9727**

To be more specific , the results shown by SVR model uses dataset with columns as shown in Figure 23 . The SVR forecasting model uses all the feature columns as independent variables except “pollution” column and “w_dir” column. The “pollution” feature column is considered as dependent variable.The SVR forecasting model produces the best PM2.5 prediction when following parameter settings are kept:

- kernel = rbf (radial bias function)
- C=32
- gamma value=0.055156
- Training dataset = 80%
- Test dataset = 20%

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 43800 entries, 0 to 43799
Data columns (total 12 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   year       43800 non-null  int64
 1   month      43800 non-null  int64
 2   day        43800 non-null  int64
 3   hour       43800 non-null  int64
 4   pollution  43800 non-null  float64
 5   dew        43800 non-null  int64
 6   temp       43800 non-null  float64
 7   pressure   43800 non-null  float64
 8   w_dir      43800 non-null  object
 9   w_speed    43800 non-null  float64
 10  snow       43800 non-null  int64
 11  rain       43800 non-null  int64
dtypes: float64(4), int64(7), object(1)
memory usage: 4.0+ MB
```

Figure 23: Information of all feature columns for DTR

Similarly, DTR also uses the same feature columns as shown the independent variables used in the SVR model and predicts the PM2.5 values keeping it as dependent variable.The prediction graph for the best case of DTR forecasting model to predict PM2.5 prediction is shown in Figure 24 .The original value for PM2.5 very closely follows the predicted value regression line as shown in the prediction graph.

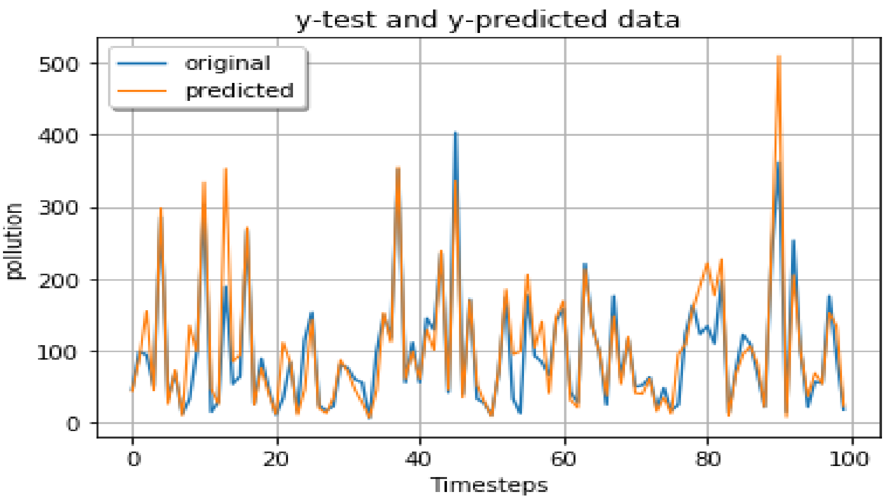


Figure 24: Decision Tree Regressor model prediction graph for best case

The performance of the DTR model was analysized for different depth the tree and it is tabulated in the Table 4. It produces the best forecasting results when following parameter settings are kept:

- Maximum depth of decision tree = 23
- Training dataset = 80%
- Test dataset = 20%

The RMSE value of DTR model in the best case is 44.0914 as shown in Table 4.

Table 4 : Variation in RMSE value of DTR model with change in depth of tree

Maximum Depth of tree	**RMSE value**
2	83.4744
8	66.3071
6	72.6471
10	59.1902
12	52.6526
16	45.4898
18	44.8126
20	44.7042
23	**44.0914**
25	44.0920
28	44.2621
30	44.9404

To predict the effectiveness of our proposed model we also re-implemented the ANN forecasting model. The layers are fed the features as shown in Figure 23 . It also uses the same input variables as DTR and SVR forecasting model and aims to predict the PM2.5 concentration values. The input layer consists of 10 input features supplied effectively.It consists of 2 hiddeen layers each of 5 neurons wide. The first hidden layer uses ReLU as the activation and the second hidden layer uses Tanh as the activation function.The output layer has only one neuron predicting the Pm2.5 values.

In most of the cases the ANN model is predciting the PM2.5 values with RMSE valus slightly greater than 91. In the best case the ANN model shows the RMSE value of 91.5041 as shown in Table 5.

Table 5 : Variation in RMSE value of ANN model with change epochs and batch-size

Batch size	Epochs	RMSE
20	5	93.4152
20	10	91.7676
20	50	91.7272
20	100	**91.5041**
15	5	91.7239
15	10	91.7240
15	50	91.7229
15	100	91.7249

10	5	94.1768
10	10	91.7237
10	50	91.7241
10	100	91.7229
5	5	91.7871
5	10	91.7226
5	50	91.7236
5	100	91.7216

Chapter 7

CONCLUSIONS AND FUTURE WORK

A novel one dimensional CNN architecture model has been proposed for predicting the PM2.5 concentration level.The proposed model has five layer deep one dimensional convolution layer as a significant novel work.This novelity in the proposed forecasting model yields the RMSE of 28.1309 and MAE of 14.9727 whis is a relatively better performance scenario as compared to traditional models like the SVR, DTR and the ANN forecasting model. Th best results for PM2.5 prediction are seen for 45 timesteps for our data driven model. The worst results are seen for 120 days timestamps (RMSE =40.3345 and MAE=24.2923). Our model can be used as an effective tool for making more precise PM2.5 prediction with lesser error as compared to its contemporary forecasting models SVR,DTR and ANN model.

In the future work , I will try to implement and analyse the results if Graph Neural Networks can be used for air pollution efficient forecasting. We can even think of implementing some other deep leaning approaches forecasting the air pollution.

Refrences:

[1] Araujo, Lilian N., et al. "Ensemble method based on Artificial Neural Networks to estimate air pollution health risks." *Environmental Modelling & Software* 123 (2020): 104567.

[2] Athanasiadis, I.N.; Kaburlasos, V.G.; Mitkas, P.A.; Petridis, V. "Applying machine learning techniques on air quality data for real-time decision support." In Proceedings of the First international NAISO Symposium on Information Technologies in Environmental Engineering (ITEE2003), Gdansk, Poland, 2427 June 2003.

[3] Jiang, D.; Zhang, Y.; Hu, X.; Zeng, Y.; Tan, J.; Shao, D. Progress in developing an ANN model for air pollution index forecast. Atmos. Environ. 2004, 38, 70557064.

[4] Zou, Bin, et al. "Healthier routes planning: A new method and online implementation for minimizing air pollution exposure risk." *Computers, Environment and Urban Systems* 80 (2020): 101456. [5] Corani, G. Air quality prediction in Milan: Feed-forward neural networks, pruned neural networks and lazy learning. Ecol. Model. 2005, 185, 513529.

[6] Fu, M.; Wang, W.; Le, Z.; Khorram, M.S. Prediction of particular matter concentrations by developed feed-forward neural network with rolling mechanism and gray model. Neural Comput. Appl. 2015, 26,17891797.

[7] Ni, X.Y.; Huang, H.; Du, W.P. Relevance analysis and short-term prediction of PM 2.5 concentrations in Beijing based on multi-source data. Atmos. Environ. 2017, 150, 146161.

[8] 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

[9] Li, Xiang, et al. "Deep learning architecture for air quality predictions." *Environmental Science and Pollution Research* 23.22 (2016): 22408-22417.

[10] Niska, Harri, et al. "Evolving the neural network model for forecasting air pollution time series." *Engineering Applications of Artificial Intelligence* 17.2 (2004): 159-167.

[11] Liu, Bing-Chun, et al. "Urban air quality forecasting based on multi-dimensional collaborative support vector regression (svr): A case study of beijing-tianjin-shijiazhuang." *PloS one* 12.7 (2017): e0179763.

[12] Li, Xiang, et al. "Long short-term memory neural network for air pollutant concentration predictions: Method development and evaluation." *Environmental pollution* 231 (2017): 997-1004.

[13] Saleh, P. R., Banik, A., Banerjee, D., Gupta, R. K., & Srivathshan, K. S. (2020). The AirSense: Smart air quality monitoring and reporting tool using IoT devices and cloud service. *Probyto Journal of AI Research*, *1*(01).

[14] Deng, Z., Weng, D., Chen, J., Liu, R., Wang, Z., Bao, J., ... & Wu, Y. (2019). AirVis: Visual Analytics of Air Pollution Propagation. *IEEE transactions on visualization and computer graphics*, *26*(1), 800-810.

[15] Kalapanidas, E.; Avouris, N. Short-term air quality prediction using a case-based classi_er. Environ. Model.Softw. 2001, 16, 263272.

[16] Liu, Bing-Chun, et al. "Urban air quality forecasting based on multi-dimensional collaborative support vector regression (svr): A case study of beijing-tianjin-shijiazhuang." PloS one 12.7 (2017): e0179763.

[17] Tao, Qing, et al. "Air pollution forecasting using a deep learning model based on 1D convnets and bidirectional GRU." IEEE access 7 (2019): 76690-76698.

[18] Tsai, Yi-Ting, Yu-Ren Zeng, and Yue-Shan Chang. "Air pollution forecasting using RNN with LSTM." 2018 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 and Technology Congress (DASC/PiCom/DataCom/CyberSciTech). IEEE, 2018.

[19] https://archive.ics.uci.edu/ml/datasets/Beijing+PM2.5+Data

[20] Al-Saffar, Ahmed Ali Mohammed, Hai Tao, and Mohammed Ahmed Talab. "Review of deep convolution neural network in image classification." *2017 International Conference on Radar, Antenna, Microwave, Electronics, and Telecommunications (ICRAMET)*. IEEE, 2017.

[21] Traore, Boukaye Boubacar, Bernard Kamsu-Foguem, and Fana Tangara. "Deep convolution neural network for image recognition." *Ecological Informatics* 48 (2018): 257-268.

[22] Gross, Jan, et al. "High-Performance Detection of Corneal Ulceration Using Image Classification with Convolutional Neural Networks." *Proceedings of the 54th Hawaii International Conference on System Sciences*. 2021.

[23] Wang, Y., Li, Y., Song, Y., & Rong, X. (2020). The influence of the activation function in a convolution neural network model of facial expression recognition. *Applied Sciences*, *10*(5), 1897.

[24] Birant, D. (2011). Comparison of Decision Tree Algorithms for Predicting Potential Air Pollutant Emissions with Data Mining Models. *Journal of Environmental Informatics*, *17*(1).

[25] Lagesse, Brent, et al. "Predicting PM2. 5 in Well-Mixed Indoor Air for a Large Office Building Using Regression and Artificial Neural Network Models." *Environmental Science & Technology* 54.23 (2020): 15320-15328.

[26] Liu, H., & Dong, S. (2020). A novel hybrid ensemble model for hourly PM2. 5 forecasting using multiple neural networks: a case study in China. *Air Quality, Atmosphere & Health*, *13*(12), 1411-1420.

[27] Maleki, Heidar, et al. "Air pollution prediction by using an artificial neural network model." *Clean Technologies and Environmental Policy* 21.6 (2019): 1341-1352.

[28] Kolehmainen, M., Martikainen, H., & Ruuskanen, J. (2001). Neural networks and periodic components used in air quality forecasting. *Atmospheric Environment*, *35*(5), 815-825.

[29] https://www.epa.gov/pm-pollution/particulate-matter-pm-basics

[30] https://www.javatpoint.com/artificial-neural-network

[31] Wu, Jianxin. "Introduction to convolutional neural networks." *National Key Lab for Novel Software Technology. Nanjing University. China* 5 (2017): 23.

[32] Kuo, C-C. Jay. "Understanding convolutional neural networks with a mathematical model." *Journal of Visual Communication and Image Representation* 41 (2016): 406-413.

[33] Scherer, Dominik, Andreas Müller, and Sven Behnke. "Evaluation of pooling operations in convolutional architectures for object recognition." *International conference on artificial neural networks*. Springer, Berlin, Heidelberg, 2010.