Weather AQI Prediction using Artificial Neural Networks and Finding Hardware Requirements

by

Aditya Kumar Singh, Kartik Singh, Shivam Ruhil

2020BCS-004, 2020BCS-046, 2020BCS-068

A report submitted for Course Project in CAD for VLSI

Bachelor of Technology

in

CSE

विश्वजीवनामृतं ज्ञानम्

ATAL BIHARI VAJPAYEE-

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY AND MANAGEMENT

GWALIOR - 474015, MADHYA PRADESH, INDIA

Report Certificate

I hereby certify that the work, which is being presented in the report, entitled Weather AQI Prediction using Artificial Neural Networks and Finding Hardware Requirements, **Bachelor of Technology**
in **CSE** and submitted to the institution is an authentic record of my/our own work carried out during the period *August-2022* to *November-2022* under the supervision of Dr Gaurav Kaushal. I also cited the reference about the text(s)/figure(s)/table(s) from where they have been taken.

Dr Gaurav Kaushal

Date ________________

The final copy of this report has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.

Abstract

Air quality has a significant impact on human health. Degradation in air quality leads to a wide range of health issues, especially in children. The ability to predict air quality enables the government and other concerned organizations to take necessary steps to shield the most vulnerable, from being exposed to the air with hazardous quality. Traditional approaches to this task have very limited success because of a lack of access of such methods to sufficient longitudinal data. In this paper, we use a Artificial Neural Network (ANN) model to forecast the PM2.5 concentration and hence the air quality index, using the data of collected using web-scraping and Central Pollution Control Board in Kolkata. The model predicts the PM2.5 concentration with RMSE value of 26.15 $\mu g/m^3$.

Acknowledgments

We am extremely grateful to Dr. Gaurav Kaushal for giving us the freedom to develop and experiment with new ideas. We would like to take this opportunity to express our sincere gratitude to him for our academic and personal mentoring, his interest in our idea, and the ongoing support, motivation, and confidence-building sessions that were very successful and helped us gain confidence and trust in the growth and development of the current work primarily because of his insightful advice, suggestions, good judgment, and constructive criticism as well as his desire for excellence. Our mentor never let us feel like a newbie by always listening to our opinions, respecting and enhancing them, and providing us complete freedom in our project. He always responded to all of our questions with a smile and an abundance of patience. The current work has only progressed to this far because of his intense interest and supportive demeanour.

In closing, I want to convey my gratitude to our organisation whose unwavering support enabled me to focus my attention and energy on this project while maintaining my motivation.

Contents

Tables

Table

Figures

Figure

Chapter 1

Introduction

This chapter presents an overview about the context as a part of Weather AQI Prediction project in section 1.1. In section 1.2 the problems and motivations are presented. Next, in section 1.3, project objectives are presented. Finally, in section 1.4, research workflow is presented.

1.1 Context

This project was done as a part of our B.Tech. course CAD for VLSI. The course *CAD for VLSI* is a part of our Computer Science and Engineering degree program. This project focuses on developing a neural networks model for predicting the AQI using the features such as average temperature, maximum temperature, wind speed, wind direction, atmospheric pressure, etc. for the city of Kolkata . We have implemented various algorithms and found that Artificial Neural Network model predicts the AQI with least errors.

1.2 Problem/Motivation

Air pollution is caused by the introduction of harmful or excessive quantities of certain substances into the atmosphere. Such substances include solid particles, liquid droplets and gases. Air pollutants are classifieds into primary and secondary pollutants. Primary pollutants are the pollutants that are directly released from their source directly into the atmosphere. Sources of primary air pollutants may be natural, like volcanic eruptions, sand storms, etc. or man-made, like burning of fossil fuels, leaking gases from appliances, etc. Primary pollutants include sulfur dioxide

(SO2), oxides of nitrogen (NOx), particulate matter (PM) and carbon monoxide (CO). Secondary pollutants are formed in the atmosphere due to chemical or physical interactions between primary pollutants. Secondary pollutants include photochemical oxidants and secondary particulate matter.

The most common pollutants are called criteria pollutants and correspond to the most prevalent health threats. These include SO2, ground level ozone (O3), NO2, lead and PM. It has been demonstrated that there is a correlation between exposure to such pollutants for a short while and health issue like inflamed respiratory track in healthy people, increased respiratory symptoms in people with asthma, difficulty meeting high oxygen requirements while exercising and critical respiratory situations, especially in children and the elderly [3].

National agencies like CPCB and many others have set standards for acceptable levels of air quality. Air quality index (AQI) is used to indicate the levels of the criteria pollutants in the air. However this project only focuses on predicting the levels for particulate matter under 2.5 microns size i.e. PM2.5 which is a significant indicator for poor air quality. Serious health symptoms may be experienced shortly after exposure to polluted air or in the long run. Such symptoms may also vary based on the age and health conditions of the particular person being exposed.

1.3 Objectives

The objectives of this project is to train a neural network model to predict the PM2.5 concentration using the data set collected by web-scraping for Kolkata. Further, the weights of the model are extracted layer by layer and is used to create a prediction function in C++ to be fed into *Xilinx Vivado* - a high level synthesis(HLS) tool which generates hardware requirements and configuration details for the model synthesis.

1.4 Research work flow

According to the research objectives, the report will describe the work flow as below:

Step 1: We collected the daily values of the input features like average temperature, maximum temperature, wind speed, wind direction, etc., using web-scrapping from https://en.tutiempo.net for years 2018 to 2022 for Kolkata. We downloaded the data of PM2.5 concentration for Kolkata for above mentioned years from Central Pollution Control Board (CPCB) and combined these data together to compile our data set.

Step 2: We performed the data pre-processing on our combined dataset. First we begin with dropping the rows with any column having null values. We then plotted a pair-plot between all the features(input/output) of the dataset.

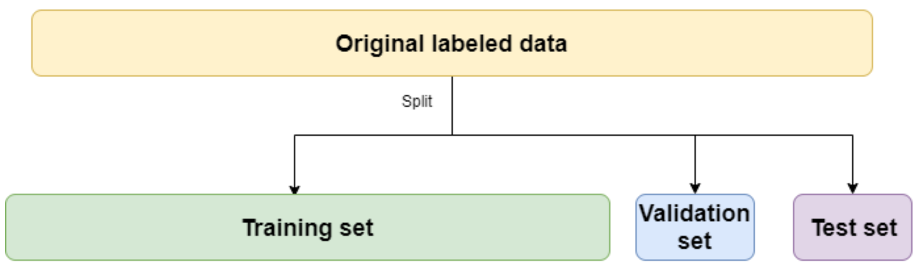

Figure 1.1: Schematic representation of splitting dataset

Step 3: We then calculated the correlation coefficient between all the features and plotted the heat-map between the features to find out how the features are related to each other or to the target variable which is PM2.5 in this case.

Step 4: Further in the data pre-processing step we moved to the **feature importance and selection** stage. We calculated the feature importance indices for the input features. After testing our model with various combinations of input features, we found that the model gives best result with the all the features included.

Step 5: The next step is to perform train-test split. We split the whole data set into two subsets. The training set is the 70% size of the whole data set and the testing dataset is 30% the size of the whole data set.

Step 6: We then created our model using **tensorflow**. Our model has 3 hidden dense layers apart from one input dense layer and another output dense layer. We used *relu* activation function in the hidden dense layers.

Step 7: After the training, we evaluate out trained model on the test set, with RMSE value as 26.15 µg per cubic-meter.

float

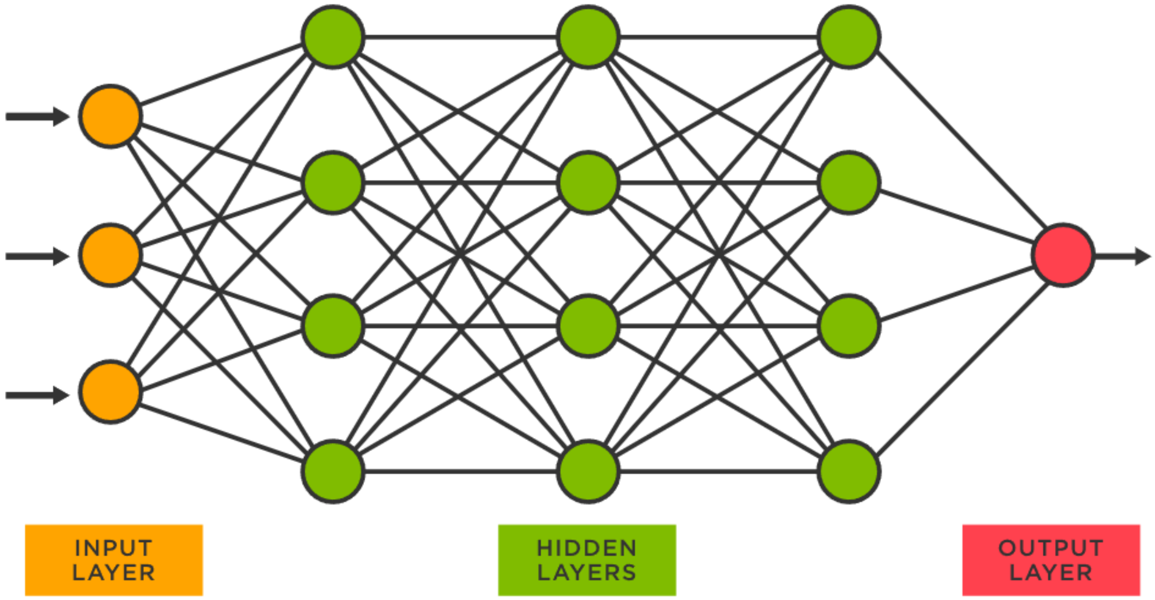

Figure 1.2: Artificial Neural Network

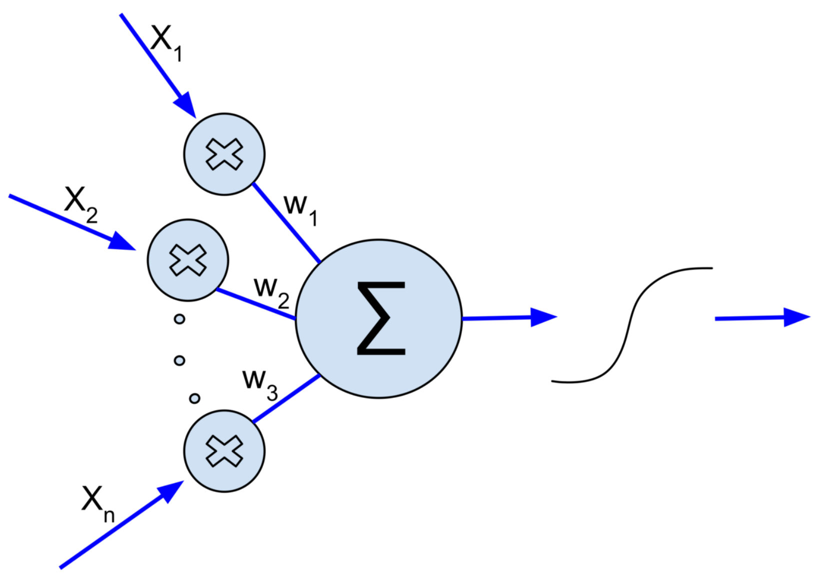

Figure 1.3: Neuron Structure

Chapter 2

Literature review

2.1 Background

Air pollution is considered to occur whenever harmful or excessive quantities of defined substances such as gases, particulates, and biological molecules are introduced into the atmosphere. These excessive emissions have obvious consequences, causing diseases and death of populations and other living organisms and impairing crops. Air pollutants can either be solid particles, liquid droplets, or gases.

Primary pollutants, which are emitted from the source directly to the atmosphere. The sources can be either natural processes, such as sandstorms or human-related, such as industry and vehicle emissions. The most common primary pollutants are sulfur dioxide (SO2), particulate matter (PM), nitrogen dioxide (NOx), and carbon monoxide (CO).

Secondary pollutants, which are air pollutants formed in the atmosphere, resulting from the chemical or physical interactions between primary pollutants. Photochemical oxidants and secondary particulate matter are the major examples of secondary pollutants.

CPCB, EPA, EU, and many other national environmental agencies have set standards and air quality guidelines regarding allowable levels for these pollutants. The air quality index (AQI) is an indicator created to report air quality, measuring how clean or unhealthy the air is and what as- sociated health effects might be a concern, especially for risk groups. It focuses on health effects that can be experienced within a few hours or days after being exposed to polluted air. It is calculated based on the maximum individual AQI registered for the criteria pollutants mentioned above.

Building a forecasting system, based on the levels of concentration of PM 2.5 pollutant, that can predict air quality hourly, will make the AQI more flexible and useful for the population's health. Systems that can generate warnings based on air quality are therefore needed and important for the populations. They may play an important role in health alerts when air pollution levels might exceed the specified levels; also, they may integrate existing emission control programs, for instance, by allowing environmental regulators the option of "on-demand" emission reductions, operational planning, or even emergency response [6].

2.2 Key related research

One of the most significant and frequently employed models for predicting time series is the autoregressive integrated moving average model (ARIMA). It was first proposed in[2] and quickly gained popularity because of its statistical features [9], adaptability to describe a variety of processes, and extensibility. Since there has been an increase in interest in urban air quality and quality of life, statistical techniques like ARIMA have been popular for predicting air pollution levels and air quality. For instance, [7] examined the ability of ARIMA to estimate the monthly values for the air pollution index and found that it was capable of producing forecasts with a lower level of confidence than 95%. In order to predict the daily values of the AQI, ARIMA's performance was more recently contrasted with that of a Holt exponential smoothing model [11].

Machine learning (ML) [4] models have attracted attention as a solution that can take the place of the more traditional statistical models in time-series forecasting due to the growing amount of historical data available for analysis and the need to perform more precise forecasts in various scientific fields and domains. Particularly, ML algorithms have found extensive usage in the forecasting of air quality.

It is exceedingly challenging to create a model that can predict events of this nature because of the high nonlinear processes that involve pollution concentrations and their imperfectly understood dynamics [13]. A nonparametric and nonlinear model that solely uses historical data to uncover latent relationships between data is the machine learning (ML) model [14].

Artificial neural networks (ANNs), genetic programming (GP), and support vector machines (SVMs), among other ML techniques, have generally been demonstrated to outperform ARIMA when predicting time series (TS) with a high degree of nonlinearity. As an illustration, Sharda and Patil [8] contrasted the outcomes obtained by an ANN and ARIMA. Later, a paper contrasted artificial neural networks (ANNs) with conventional multivariate regression, coming to the conclusion that ANNs perform better than the conventional statistical approaches when the dataset exhibits more volatile conditions.

2.3 Problem formulation

In this study, an artificial neural network will be used to predict PM 2.5 concentration (in µg/m^3) for kolkata city trained using daily data of kolkata city from 2018 to 2022.

Chapter 3

Methodology

This section introduces the hypothesis and the analytical validation of the proposed solution.

3.1 Proposed hypothesis

Artificial Neural Network model should produce better results for PM 2.5 concentration prediction on our MISCA dataset. Furthermore, we intend to evaluate the hardware requirements using high level synthesis of the model in **Xilinx Vivado**.

3.2 Artificial Neural Networks

The idea of biological neural networks in human brains gave rise to the Artificial Neural Network (ANN), a deep learning method. The development of ANN was the result of an effort to replicate how the human brain functions. Although it differs slightly from biological neural networks in operation, ANN functions fairly similarly to them.

3.2.1 Structure of Artificial Neurons and Functions

Perceptrons are neural networks with just one layer. Artificial Neural Networks are multi-layer perceptrons. Any number of layers are possible in a neural network. There may be one or more neurons or units in each layer. Each and every neuron is related to every other neuron. Additionally, each layer might serve a separate activation function. Back-propagation and Forward-propagation are the two phases of an ANN. As part of the forward propagation, weights are multiplied, bias is added, and the activation function is applied to the inputs before being propagated forward.

The most crucial step is back-propagation, which typically entails finding the model's ideal parameters by propagating backward through the layers of neural networks. The back-propagation needs an optimization function to determine the model's ideal weights. By adjusting the output layers' activation functions appropriately, ANN can be used for both classification and regression applications.

The ANN makes an effort to incrementally and hierarchically learn. It is not necessary to carry out feature engineering directly because of this. ANN outperforms traditional Machine Learning algorithms when the data size is large. Traditional Machine Learning algorithms typically perform at the same level as the data size increases.

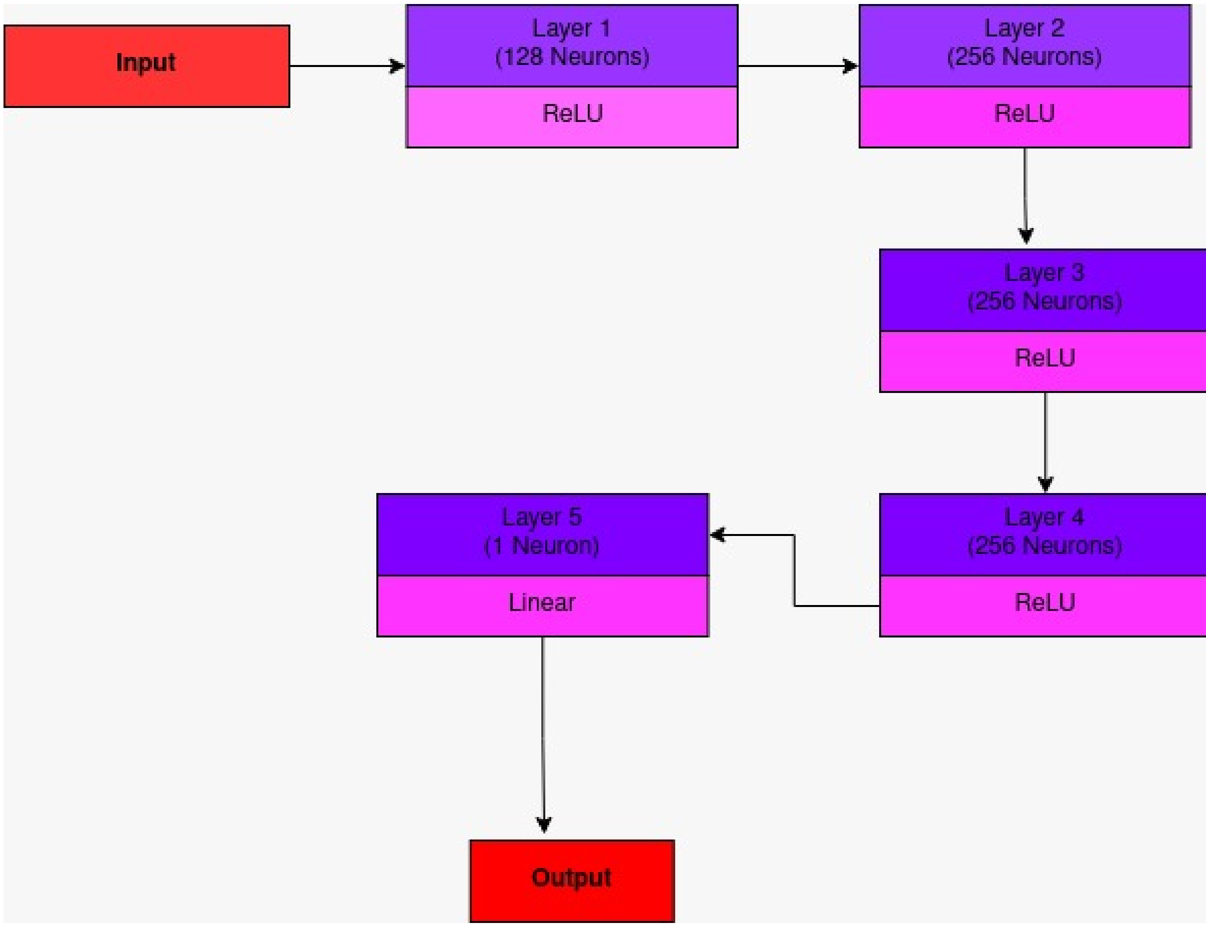

Figure 3.1: Architecture of implemented Artificial Neural Network

3.2.2 Working of ANN

The working of ANN can be broken down into following phases:

- Forward Propagation
- Back Propagation

3.2.3 Forward Propagation

Forward propagation entails multiplying feature values with weights, adding bias, and then applying an activation function to every neuron in a neural network. By multiplying feature values with weights and adding bias to each neuron, essentially linear regression is applied.

3.2.4 Activation Function

The objective of an activation function is to introduce non-linearity into the data. Introducing non-linearity facilitates the identification of complicated underlying patterns. Additionally, it is used to scale the value to a specific interval. The sigmoid activation function, for example, adjusts the value between 0 and 1. The ReLU activation function gives *max(0, x)* for any data point.

3.2.5 ReLU Activation Function

ReLU (Rectified Linear Unit) outputs the same number if x¿0 and outputs 0 if x¡0.

It prevents the vanishing gradient problem but introduces an exploding gradient problem during backpropagation. The exploding gradient problem can be prevented by capping gradients.For our model ReLU function produces best results.

3.2.6 Backpropagation

Backpropagation is used to determine the optimal model parameters by iteratively updating parameters by partially differentiating gradients of the loss function with respect to parameters. During backpropagation, an optimization function is utilised. The purpose of an optimization function is to determine the optimal parameter values.

The optimization functions available are,

- Gradient Descent
- Adam optimizer
- Gradient Descent with momentum
- RMS Prop (Root Mean Square Prop)

In our project we have used Adam optimizer, as it produces the best results among the tried optimizers.

3.2.7 Terminologies

Epoch: An epoch is a single run through the training data. Mini-batches of training data are fed to the model, and after all mini-batches have been fed to the model, this defines an epoch.

Metrics: A metric is used to evaluate the model's performance. Metric functions are identical to cost functions, with the exception that the results of evaluating a metric are not utilised during model training. Note that any cost function may be used as a metric. Root Mean Squared Error metrics has been utilised as a measure and cost function.

float

Chapter 4

Experiments and results

This section discusses the various experiments pertaining to the proposed hypothesis and their findings.

4.1 Experiment design

4.1.1 Dataset

For the prediction of air quality in the city of Kolkata, we ourselves have compiled the dataset. We took the input feature datapoints using webscraping in python from . The PM2.5 data points are collected from the CPCB website. We have utilized the daily data points for various features. The data frame is shown in the following figure.

All the files contain hourly data, separated by pollutant or parameter that is being measured - average temperature, maximum temperature, minimum temperature, atmospheric pressure, humidity, precipitation, wind speed, maximum sustainable wind speed, and PM2.5 concentration.

4.1.2 Data Collection and Prepossessing

GitHub URL: https://github.com/sadityakumar9211/air-quality-prediction-using-ann

4.1.2.1 Data Collection and Combination

The daily data is being is being collected for the duration from october 2018 to september 2022. The data initially downloaded was in form of monthly HTML files. These files are individually parsed and then the data present in the tables are extracted using python HTML parsing library *BeautifulSoup*. These monthly data files(.CSV) are then combined to form yearly files and these are then combined to make a single data file from october 2018 to september 2022.

The data point for PM 2.5 concentration is taken from the CPCB website and this file and the above mentioned file is again combined to form a single file containing the data set. The file name is Real-Combine.csv.

The web-scrapping code used for data collection is also present in the specified GitHub URL.

4.1.2.2 Data Preprocessing

We have utilized various methods of data preprocessing including but not limited to the removal of null values and removal of outliers. We have also utilized the feature importance and selection techniques to narrow out the data set to train our model better. Various techniques and visualizations like feature importance and selection, correlation coefficient matrix, heatmap, pairplot are used to better analyse the dataset. All these are annotated with code in our Jupyter Notebook file present in the above provided GitHub Link.

Variable Name	Description
T	Daily average temperature(in ℃)
Tm	Minimum daily temperature(in ℃)
TM	Maximum daily temperature(in ℃)
SLP	Atmospheric pressure at sea level (hPa)
H	Average relative humidity (%)
VV	Average visibility (Km)
VM	Maximum sustained wind speed (Km/h)
PM2.5	PM2.5 concentration (µg/m^3)

Table 4.1: Dataset Variable Description

	T	TM	Tm	SLP	H	VV	V	VM	PM 2.5
count	775.000000	775.000000	775.000000	775.000000	775.000000	775.000000	775.000000	775.000000	775.000000
mean	26.855871	30.927613	23.047097	1007.657161	75.286452	3.481677	3.510194	7.188129	51.528594
std	4.226625	3.994206	4.912739	6.148475	10.849357	0.751295	2.301665	3.455447	46.208104
min	15.800000	17.800000	11.200000	991.200000	39.000000	0.600000	0.200000	1.900000	4.410000
25%	23.900000	28.400000	19.250000	1002.500000	68.000000	3.100000	1.900000	5.400000	17.465000
50%	28.500000	31.800000	25.100000	1007.800000	76.000000	4.000000	3.000000	7.600000	31.940000
75%	29.900000	34.000000	26.900000	1013.250000	84.000000	4.000000	4.800000	9.400000	76.850000
max	32.800000	39.500000	29.300000	1020.000000	98.000000	5.100000	14.100000	31.700000	297.540000

Figure 4.1: Dataset Descriptive Statistics

4.1.3 Experiment - Artificial Neural Network Model

4.1.3.1 Parameter settings

Number of ANN layers: 5

Number of neurons in various ANN layers: [128, 256, 256, 256, 1]

Number of Hidden Dense Layers: 3

Number of neurons in Hidden Dense layers: [256, 256, 256]

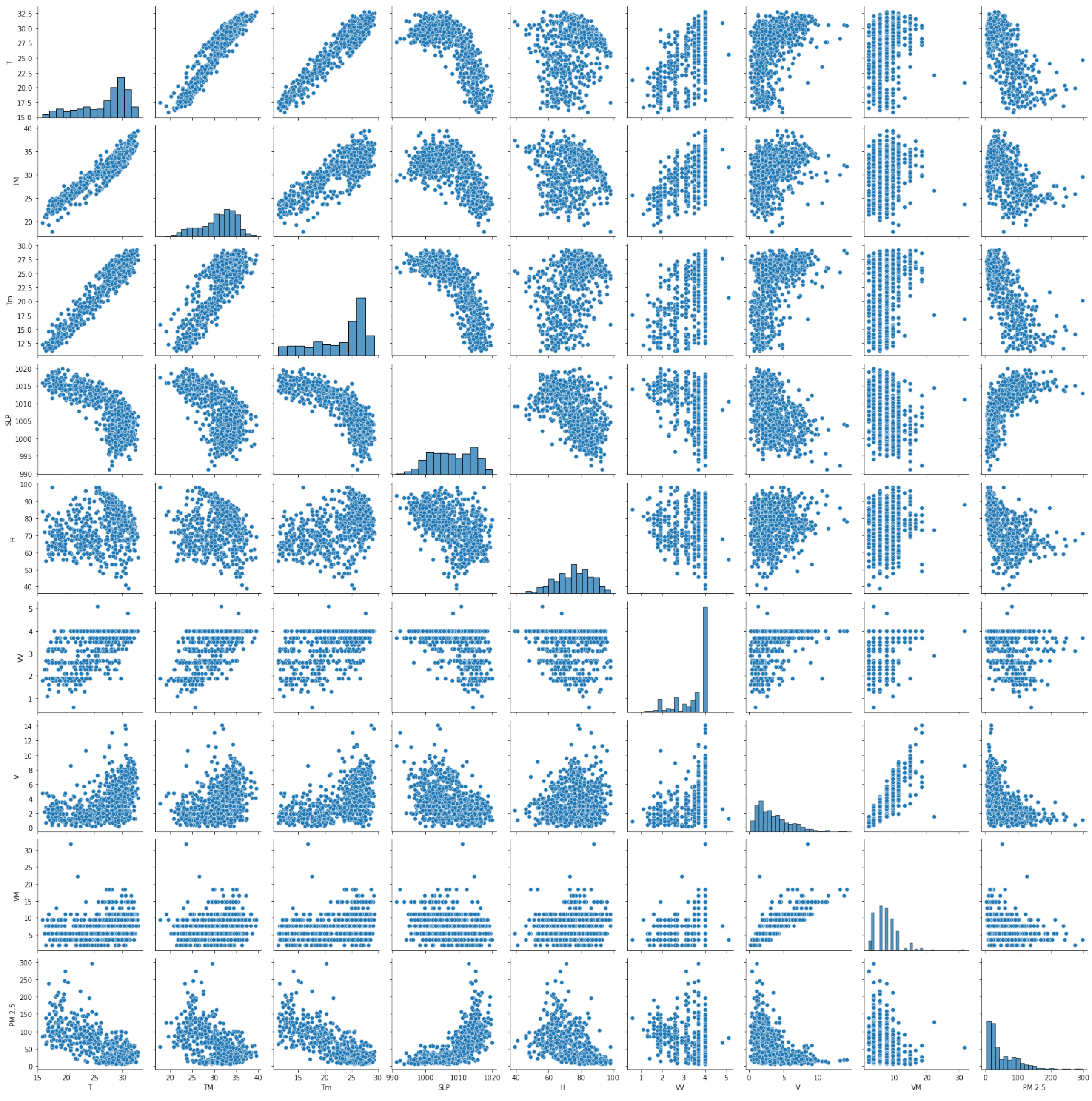

Figure 4.2: Pair-plot description

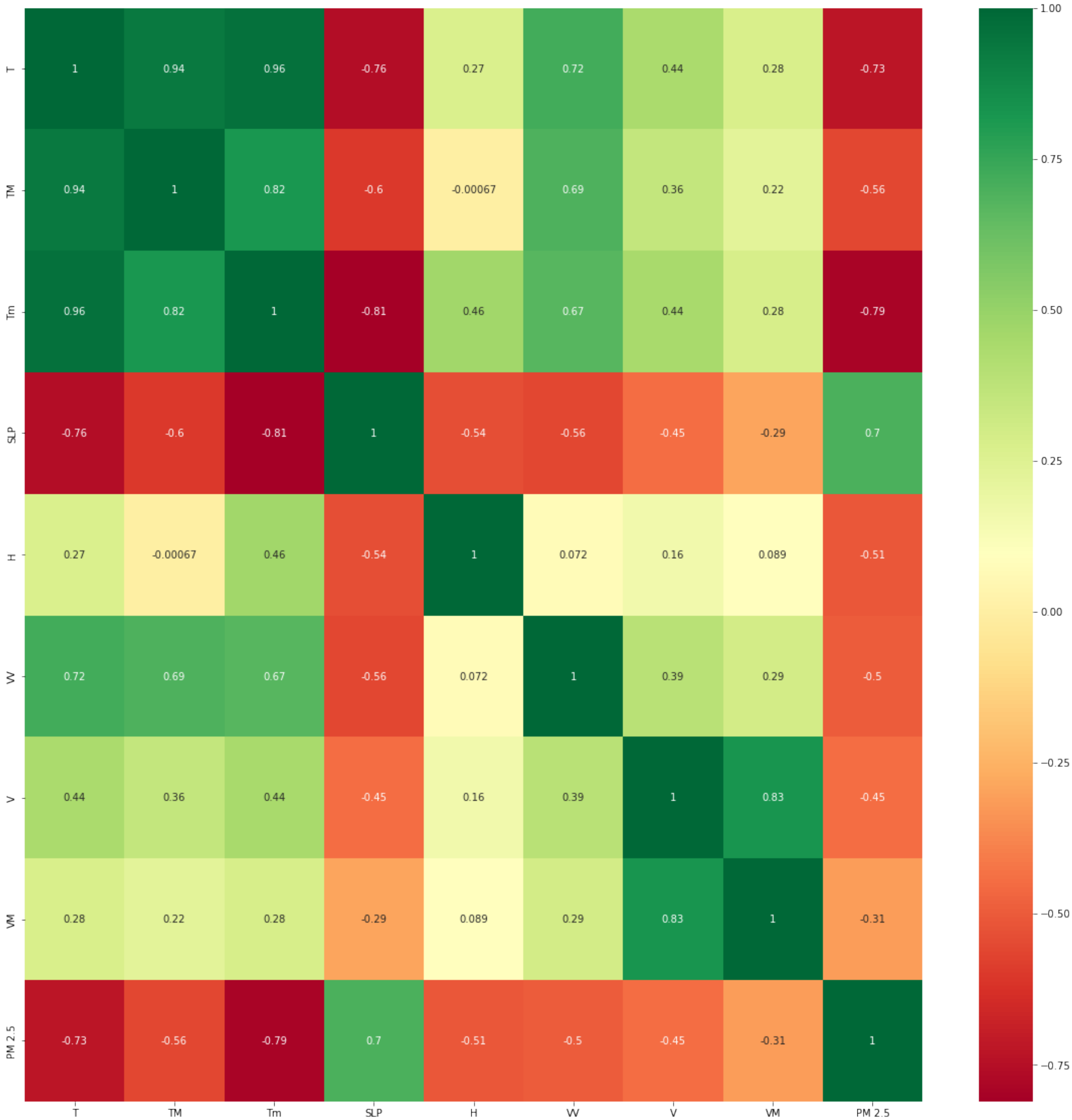

Figure 4.3: Heat-map of correlation

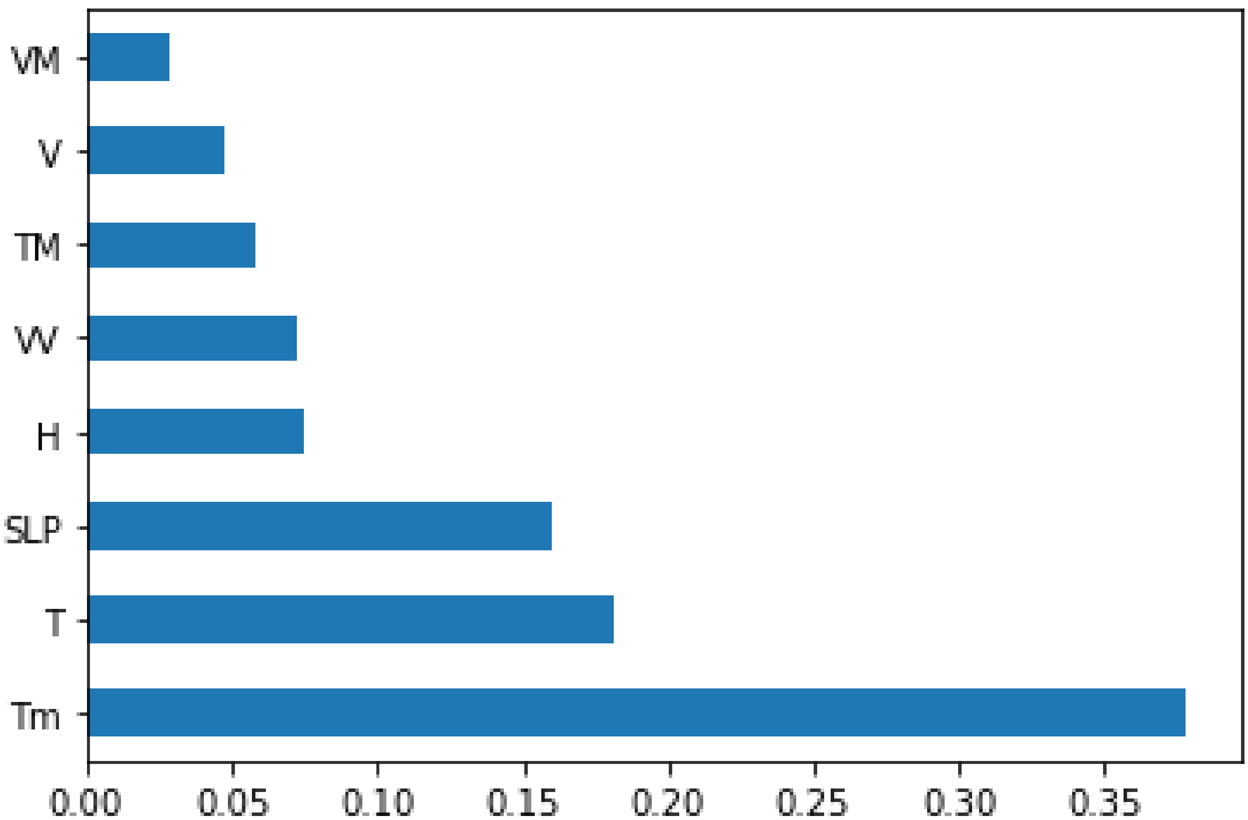

Figure 4.4: Feature importance realtion

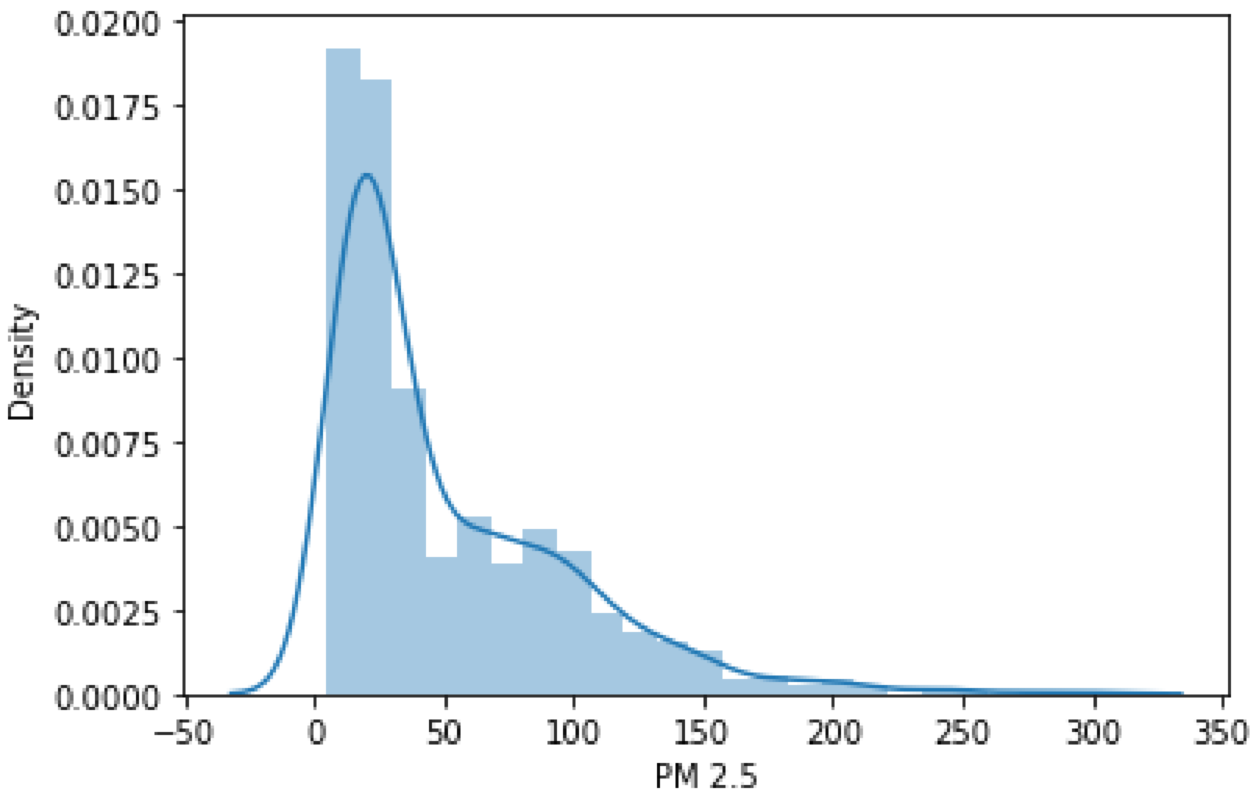

Figure 4.5: Feature importance realtion

Kernel Initializer in each layers: [normal, normal, normal, normal, normal]

Activation Function for Hidden Dense layers: ReLU

Output Layer: A Dense layer with 1 neuron and linear activation function All the layers including input layer are densely connected layers.

4.1.3.2 Experiment description

Here we have used an ANN model with one input layer consisting of 128 neurons, 3 consecutive hidden dense layers consisting of 256 neurons each. The last layer is a single neuron layer utilizing linear activation function.

We first did the pre-processing of the MISCA dataset and found that including all the features in the feature selection phases gave better results in the end than removing some of the least important features. So, we kept all of the features in the dataset.

In the train-test split phase, we split the dataset into training, validation, and testing sets, where the size of teh test set is 30% of the original dataset. The validation data-subset is 0.33 fraction of the training data-subset.

We train our ANN model in TENSORFLOW environment using the training and validation sets and then use the hidden test set to evaluate the performance of our dataset. We used the sklearn library of python to randomly select various test sets in various training runs. This makes sure that the test set is not chosen from on specific section of our dataset. Lastly we analyzed the results using various metrics and visualizations.

4.1.3.3 Results and discussion

We used MAE, MSE, and RMSE value to evaluate our trained model. The results are as follows:

- MAE: 16.895
- MSE: 698.133

- RMSE: 26.422

The trained ANN model predicts the new values in our data set with an RMSE value of 26.422 which is acceptable as it lies well below the level jump of 50 specified by Environmental Pollution Agency(EPA).

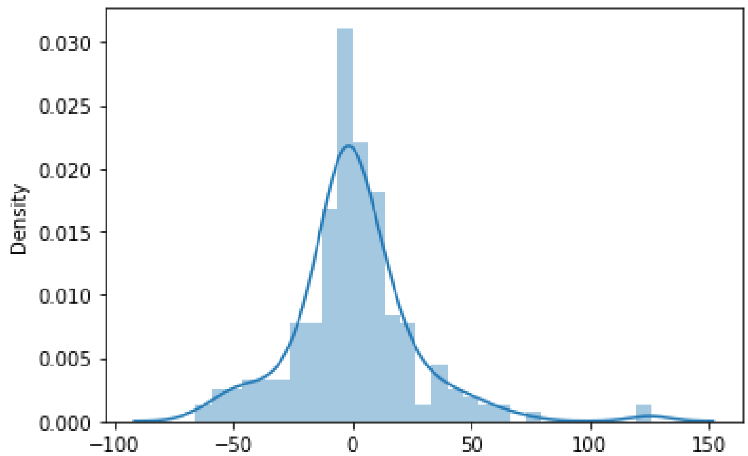

Figure 4.6: Error distribution plot

4.1.3.4 Conclusion

Predicting the air quality is a complex task due to the dynamic nature, volatility, and high variability in space and time of pollutants and particulates. At the same time, being able to model, predict, and monitor air quality is becoming more and more important, especially in urban areas, due to the observed critical impacts of air pollution for populations and the environment.

In this project we modelled and implemented an artificial neural network to predict concentration of particulate matter with size below 2.5 microns (PM2.5) with good evaluation metrics.

4.2 High Level Synthesis - Xilinx Vivado

After modelling and training our artificial neural network, we extracted weights of every neuron of each layer in files prefixed with *dense*. We modified these files to incorporate for the biases of

```
In [35]: from sklearn import metrics
         print('MAE:', metrics.mean_absolute_error(y_test, prediction))
         print('MSE:', metrics.mean_squared_error(y_test, prediction))
         print('RMSE:', np.sqrt(metrics.mean_squared_error(y_test, prediction)))

         MAE: 17.256978376003808
         MSE: 652.3202964224939
         RMSE: 25.540561787527185
```

Figure 4.7: Model Evalution Results

each layer by putting the biases of the previous layer before the weights of the next layer. We copied the weights of various layers (and the bias values) and used it in our *core.cpp* file to be fed into *Xilinx Vivado* tool. In *core.cpp* file we created a *predict* function to mimic the behaviour of ANN model for predicting the PM 2.5 concentration using the extracted weights. We then performed the activation function on the values. This operation is performed once for each layer with their corresponding activation functions.

We also created *test.cpp* utilizing the predict function defined in the *core.cpp* file and passed the values of input parameters.

Upon *CSimulation* phase in the *Vivado* tool, we got a result differing by a value of 3 µg/m^3.

Further, we proceeded with the *C Synthesis* to generate hardware requirements for our generated model. The results of *C Synthesis* are presented in the table below.

Table 4.2: Timing results generated upon *C Synthesis* in *Vivado*

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	7.796	1.25

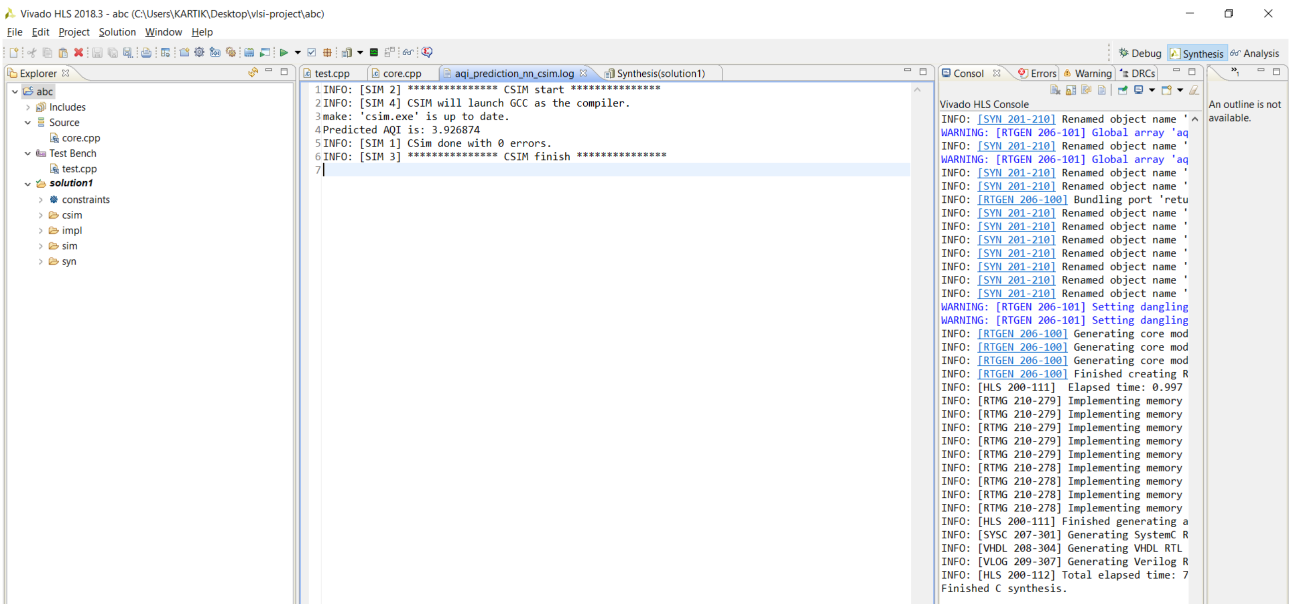

Figure 4.8: Results upon C simulation in *Vivado*

Table 4.3: Latency results generated upon *C Synthesis* in *Vivado*

Latency		Interval		Type
min	max	min	max	
2160157	2160157	2160157	2160157	none

Table 4.4: Hardware requirements auto-generated upon *C Synthesis* in *Vivado*

Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	-	-	-
Expression	-	-	0	546
FIFO	-	-	-	-
Instance	4	15	1709	3638
Memory	1285	-	0	0
Multiplexer	-	-	-	590
Register	-	-	558	-
Total	1289	15	2267	4774
Available	40	40	16000	8000
Utilization (%)	3222	37	14	59

Table 4.5: Hardware Interfaces

RTL Ports	Dir	Bits	Protocol	Source Object	C Type
s_axi_CRTL_BUS_AWVALID	in	1	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_AWREADY	out	1	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_AWADDR	in	5	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_WVALID	in	1	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_WREADY	out	1	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_WDATA	in	32	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_WSTRB	in	4	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_ARVALID	in	1	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_ARREADY	out	1	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_ARADDR	in	5	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_RVALID	out	1	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_RREADY	in	1	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_RDATA	out	32	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_RRESP	out	2	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_BVALID	out	1	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_BREADY	in	1	s_axi	CRTL_BUS	scalar
s_axi_CRTL_BUS_BRESP	out	2	s_axi	CRTL_BUS	scalar
ap_clk	in	1	ap_ctrl_hs	aqi_prediction_nn	return value
ap_rst_n	in	1	ap_ctrl_hs	aqi_prediction_nn	return value
interrupt	out	1	ap_ctrl_hs	aqi_prediction_nn	return value
X_Addr_A	out	32	bram	X	array
X_EN_A	out	1	bram	X	array
X_WEN_A	out	4	bram	X	array
X_Din_A	out	32	bram	X	array
X_Dout_A	in	32	bram	X	array
X_Clk_A	out	1	bram	X	array
X_Rst_A	out	1	bram	X	array

Chapter 5

Discussions and conclusion

In this chapter, the work is concluded and future plan is presented.

Predicting air quality is difficult due to the dynamic, volatile, and spatially and temporally variable nature of contaminants and particles. In addition, the ability to analyse, predict, and monitor air quality is becoming increasingly crucial, particularly in metropolitan areas, due to the observed detrimental effects of air pollution on populations and the environment.

This work presented a study of artificial neural network (ANN) to forecast concentration of PM 2.5 pollutants and particulates. Our presented model predicts the PM 2.5 with 26 µg/m^3 which is well under perceived error range specified by EPA.

5.1 Contributions

We developed an artificial neural network for predicting PM2.5 concentration for city Kolkata from using the dataset collected for october 2018 to september 2022. Further we generated hardware configuration details for our model using *Xilinx Vivado*. The results are presented in the provided GitHub URL.

5.2 Limitations

Air quality index value range	Levels of health concern	Description
0 to 50	Good	Air quality is considered satisfactory.
51 to 100	Moderate	Air quality is acceptable; however, for some pollutants, there is a moderate health concern for a small number of people, namely, those that experience respiratory problems.
101 to 150	Unhealthy for sensitive groups	Although for most of the people, the health concern is moderate, for groups with lung diseases, the elderly, and children, there is a great risk of exposure to some pollutants and particulates.
151 to 200	Unhealthy	Health side effects for all the affected area population. Sensitive groups may experience more serious effects.
201 to 300	Very unhealthy	Health alerts would be triggered as all the affected area population would experience serious health effects.
301 to 500	Hazardous	Health alerts with emergency warnings would be triggered. The entire area population would be severely affected.

Figure 5.1: Artificial Neural Network

5.3 Future scope

We could try to improve the model by using other popular machine learning models like Support Vector Regression (SVR) and compare the results. We can also try to increase the model metrics by using better pre-processing techniques using more number of parameters and try to capture the relationships and variations with PM 2.5 concentration. We could also try to use larger datasets and better features to improve the model.

Bibliography

[1] Fabio Caiazzo, Akshay Ashok, Ian A Waitz, Steve HL Yim, and Steven RH Barrett. Air pollution and early deaths in the united states. part i: Quantifying the impact of major sectors in 2005. Atmospheric Environment, 79:198–208, 2013.

[2] Ivan Nunes Da Silva, Danilo Hernane Spatti, Rogerio Andrade Flauzino, Luisa Helena Bartocci Liboni, and Silas Franco dos Reis Alves. Artificial neural networks. Cham: Springer International Publishing, 39, 2017.

[3] Xi Gong, Yan Lin, and F Benjamin Zhan. Industrial air pollution and low birth weight: a case-control study in texas, usa. Environmental Science and Pollution Research, 25(30):30375–30389, 2018.

[4] Yolanda Gonzalez, Claudia Carranza, Marco Iniguez, Martha Torres, Raul Quintana, Alvaro R Osornio-Vargas, Carol Gardner, Srijata Sarkar, and Stephan Schwander. Inhaled air pollution particulate matter in alveolar macrophages alters local pro-inflammatory cytokine and peripheral ifn production in response to mycobacterium tuberculosis. In B17. MYCOBACTERIAL HOST DEFENSES, pages A2901–A2901. American Thoracic Society, 2017.

[5] Anjum Hajat, Charlene Hsia, and Marie S O'Neill. Socioeconomic disparities and air pollution exposure: a global review. Current environmental health reports, 2(4):440–450, 2015.

[6] Bonnie Holmes-Gen and Will Barrett. Clean air future. Health and Climate Benefits of Zero Emission Vehicles, American Lung Association, Chicago, IL, USA, 2016.

[7] Ulla Arthur Hvidtfeldt, Matthias Ketzel, Mette Sørensen, Ole Hertel, Jibran Khan, Jørgen Brandt, and Ole Raaschou-Nielsen. Evaluation of the danish airgis air pollution modeling system against measured concentrations of pm2. 5, pm10, and black carbon. Environmental Epidemiology, 2(2):e014, 2018.

[8] Dave Ketul. Air pollutant forecasting using deep learning. PhD thesis, Laurentian University of Sudbury, 2021.

[9] Elisa Marino, Massimo Caruso, Davide Campagna, and Riccardo Polosa. Impact of air quality on lung health: myth or reality? Therapeutic Advances in Chronic Disease, 6(5):286–298, 2015.

[10] Jennifer D Parker, Lara J Akinbami, and Tracey J Woodruff. Air pollution and childhood respiratory allergies in the united states. Environmental health perspectives, 117(1):140–147, 2009.

[11] Devon C Payne-Sturges, Melanie A Marty, Frederica Perera, Mark D Miller, Maureen Swanson, Kristie Ellickson, Deborah A Cory-Slechta, Beate Ritz, John Balmes, Laura Anderko, et al. Healthy air, healthy brains: advancing air pollution policy to protect children's health. American journal of public health, 109(4):550–554, 2019.

[12] Laura Pimpin, Lise Retat, Daniela Fecht, Laure de Preux, Franco Sassi, John Gulliver, Annalisa Belloni, Brian Ferguson, Emily Corbould, Abbygail Jaccard, et al. Estimating the costs of air pollution to the national health service and social care: An assessment and forecast up to 2035. PLoS medicine, 15(7):e1002602, 2018.

[13] Nannuri Suresh and Eswar Poluri. Modeling and forecasting of daily air quality index using machine learning algorithm. NVEO-NATURAL VOLATILES & ESSENTIAL OILS Journal—NVEO, pages 4227–4236, 2021.

[14] Xin Yao. Evolving artificial neural networks. Proceedings of the IEEE, 87(9):1423–1447, 1999.

[15] Jie Zhu, Ruoling Zhang, Binbin Fu, and Renhao Jin. Comparison of arima model and exponential smoothing model on 2014 air quality index in yanqing county, beijing, china. Appl. Comput. Math, 4(6):456–461, 2015.