Prediction of Air Quality in Smart Cities Using Deep Learning Techniques

Author:

Muhammad Zeeshan

Registration Number: 00000398012

Supervisor:

Assoc. Prof. Dr. Fahim Arif

A thesis submitted to the faculty of Computer Software Engineering Department, Military College of Signals, National University of Sciences and Technology, Islamabad, Pakistan in partial fulfilment of the requirements for the degree of MS in Software Engineering

(Mar 2024)

THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS Thesis written by **Muhammad Zeeshan,** Registration No. **00000398012**, of **Military College of Signals** has been vetted by undersigned, found complete in all respects as per NUST Statutes/Regulations is free of plagiarism, errors, and mistakes and is accepted as partial fulfillment for award of MS degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the said thesis.

Signature: ______________________

Name of Supervisor Assoc Prof Dr. Fahim Arif

Date: ______________________

Brig
Head of Dept of CSE
Mil College of Sigs (NUST)

Signature (HOD): ______________________

Date: 11/4/24

Brig
Dean, MCS (NUST)
(Asif Masood, Phd)

Signature (Dean/Principal) ______________________

Date: 18/3/24

Form TH-4

NATIONAL UNIVERSITY OF SCIENCES & TECHNOLOGY

MASTER THESIS WORK

We hereby recommend that the dissertation prepared under our supervision by **Muhammad Zeeshan,** Regn No **00000398012** Titled: **"Prediction of Air Quality in Smart Cities Using Deep Learning Techniques"** be accepted in partial fulfillment of the requirements for the award of **MS Software Engineering** degree.

Examination Committee Members

1. Name: **Prof Dr. Hammad Afzal** Signature: Hammad Afzal

2. Name: **Asst Prof Dr. Nauman Ali Khan** Signature:

Supervisor's Name: **Assoc Prof Dr Fahim Arif** Signature:

Date:

Brig
Head of Dept of CSE
Mil College of Sigs (NUST)

Head of Department

11/3/24
Date

COUNTERSIGNED

Brig
Dean, MCS (NUST)
(Asif Masood, Phd)

Dean

Date: 14/3/24

CERTIFICATE OF APPROVAL

This is to certify that the research work presented in this thesis, entitled "**Prediction of Air Quality in Smart Cities using Deep Learning Techniques**" was conducted by Mr. Muhammad Zeeshan under the supervision of Assoc. Prof Dr. Fahim Arif PhD. No part of this thesis has been submitted anywhere else for any other degree. This thesis is submitted to the **Military College of Signals, National University of Sciences and Technology, Islamabad (Computer Software Engineering)** in partial fulfillment of the requirements for the degree of Master of Science in Field of **Computer Software Engineering** (Subject Name) **MSSE-28** Department of **Computer Science** National University of Sciences and Technology, Islamabad.

Student Name: Muhammad Zeeshan Signature: ____________

Examination Committee:

a) External Examiner 1:

Name Prof. Hammad Afzal - Signature: ____________

(Designation and Office Address) - CSE Department, MCS.

b) External Examiner 2:

Name Asst. Prof. Nouman Ali Khan - Signature: ____________

(Designation and Office Address) - CSE Department, MCS.

Name of Supervisor: Assoc. Prof Dr. Fahim Arif Signature: ____________ Supervisor's Sign

Name of Dean/HOD: Brig Adnan Ahmed Khan, PhD Signature: ____________ Brig ... Dept of CSE

PLAGIARISM UNDERTAKING

I solemnly declare that research work presented in the thesis titled "**Prediction of Air Quality in Smart Cities using Deep Learning Techniques**." is solely my research work with no significant contribution from any other person. Small contribution/ help wherever taken has been duly acknowledged and that complete thesis has been written by me.

I understand the zero tolerance policy of the HEC and National University of Sciences and Technology (NUST), Islamabad towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MS degree, the University reserves the rights to withdraw/revoke my MS degree and that HEC and NUST, Islamabad has the right to publish my name on the HEC/University website on which names of students are placed who submitted plagiarized thesis.

Student Signature:

Name: Muhammad Zeeshan

Date:

AUTHOR'S DECLARATION

I **Muhammad Zeeshan** here by state that my MS thesis titled **"Prediction of Air Quality in Smart Cities using Deep Learning Techniques"** is my own work and has not been submitted previously by me for taking any degree from the National University of Sciences and Technology, Islamabad or anywhere else in the country/ world.

At any time if my statement is found to be incorrect even after I graduate, the university has the right to withdraw my MS degree.

Student Signature: ____________

Name: Muhammad Zeeshan

Date: ____________

Dedication

“In the name of Allah, the most Beneficent, the most Merciful"

I dedicate this Thesis to my beloved Parents, friends, and fellows, who have all been my endless source of love, encouragement, and strength. Their unwavering belief in my abilities, countless sacrifices, and relentless support have been the foundation upon which I built my academic pursuits. Without their love and support, this research work would not have been possible.

Acknowledgments

All worship and glory be to the All-Magnificent and All-Merciful ALLAH Almighty. I am deeply grateful to Allah Almighty for granting me the capability and determination to pursue and complete this research. His divine blessings and guidance have been instrumental in overcoming obstacles and achieving success. I humbly acknowledge that no words can fully express my gratitude for the countless blessings bestowed upon me. I dedicate this thesis as a humble tribute to Allah Almighty, recognizing His infinite wisdom and benevolence. I pray that my work may benefit others and be pleasing to Him. I am thankful to all my family, particularly my parents, for they have always been my pillar of utmost strength and assistance. All this is an outcome of their love and prayers. I owe this degree of mine to my father, who was the ultimate source of motivation for me to pursue the bigger and better opportunities offered by life. I also want to extend my gratitude to my mother for always taking care of my needs and schedule and always blessing me with lots of prayers.

I would also like to express my heartfelt appreciation to my thesis supervisor, **Brig ® Assoc Prof Dr. Fahim Arif**, for his unwavering support and guidance throughout my thesis. His knowledge, expertise, and dedication to his field have been a source of inspiration to me, and I am grateful for the time and effort he invested in my success. From the beginning of my journey until the end, he has been an embodiment of kindness, motivation, and inspiration towards me. In addition, I extend my gratitude to my GEC committee members, **Professor Dr. Hammad Afzal and Assistant Professor Nouman Ali Khan**, for their continuous availability for assistance and support throughout my degree, both in coursework and thesis. Their expertise and knowledge have been invaluable to me, and I am grateful for their unwavering support and guidance.

Lastly, I express my heartiest thanks and warm wishes to my friends and people around me who have always understood my hectic schedule. Their genuine concern for my studies is exemplary, and their profound contribution cannot be put into words and is deeply heartfelt. I appreciate their efforts from the bottom of my heart.

Abstract

Air pollution has developed as a serious and potentially fatal anxiety in various nations throughout the world in recent decades, owing mostly to human activity, industry, and urbanization. Substantial Particulate Matter having a Diameter of 2.5m (PM2.5) is a particularly dangerous component of air pollution that causes major health hazards, including respiratory and cardiovascular disorders. As a result, precisely forecasting PM2.5 levels is critical in order to protect people from the negative effects of air pollution. PM2.5 levels are impacted by a number of factors, such as climatic conditions and the quantity of other contaminants in metropolitan areas. In this study, we used a DL (Deep Learning) technique, especially (CLARP) CNN-LSTM-Attention Mechanism-Recurrent Mechanism-Pooling Mechanism, to anticipate the hourly PM2.5 concentration in Beijing, China.

Our model includes data fusion approaches that involve the merging of many data sources, such as historical pollutant data, meteorological data, and PM2.5 values, to provide more accurate estimates or projections. We compared the performance of numerous LSTM, Bi-LSTM, GRU, Bi-GRU, PM-GRU, RM-LSTM and a hybrid CLARP model. Based on experimental data, the CLARP Model technique outperformed all standard models tested, giving 99% R2 and improved results for RMSE & MAE emphasizing its expanded predictive capabilities.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

CLARP	-----------------------	CNN-LSTM-Attention Mechanism-Recurrent Mechanism-Pooling Mechanism
CNN	-----------------------	Convolutional Neural Network
LSTM	-----------------------	Long Short-Term Memory
GRU	-----------------------	Gated Recurrent Unit
Bi-GRU	-----------------------	Bidirectional Gated Recurrent unit
Bi-GRU	-----------------------	Bidirectional Long Short-Term
Bi-LSTM	-----------------------	Memory
SVR	-----------------------	Support Vector Regression
ANN	-----------------------	Artificial Neural Network
RNN	-----------------------	Recurrent Neural Network
MLR	-----------------------	Multiple Linear Regression
DL	-----------------------	Deep Learning
ML	-----------------------	Machine Learning
GCN	-----------------------	Graph Convolutional Networks
NFNN	-----------------------	Neuro Fuzzy Neural Network
AQI	-----------------------	Air Quality Index
R2	-----------------------	Coefficient of Determination
MAE	-----------------------	Mean Absolute Error
RMSE	-----------------------	Root Mean Square Error

Chapter 1: Introduction

1.1 Overview

The percentage of people living in cities throughout the world is increasing, indicating a rising trend in urban migration. In 2020, the urban population was projected by the United Nations (UN) to be at 56.15 percent [1]. Furthermore, according to forecasts by 2050, cities will be home to 68% of the world's population. [2]. Logistics, healthcare, and air among the problems brought on by urbanization and industry are those of quality. The idea of "smart cities" has evolved to solve these problems and enhance inhabitants' quality of life. Information and communication technology (ICT) is combined with mobile and stationary sensors that are strategically positioned across a city to track human activity.

Energy consumption has significantly increased in recent decades as a result of the quick industrialization and urbanization processes, which has raised concerns about air pollution [3,4,5]. Chronic obstructive pulmonary disease, Asthma, heart disease and cancer are a some of the illnesses that can be exacerbated by the presence of toxic compounds in such as PM2.5, CO, SO2, and NO2 in the air [6,7]. The World Health Organization (WHO) estimates that air pollution results in over 7 million deaths per year, underscoring the grave threat it represents to people's health. Several models have been created by researchers for forecast changes in air quality, allowing for the prompt deployment of appropriate interventions in order to lessen the negative impacts of air pollution [8, 9]. Among these models, DL models have shown to have the best prediction skills [10]. DL models face difficulties of being "black boxes," which makes it challenging to understand their prediction behaviors. Additionally, time-series information on the atmosphere includes signals at several frequencies and is frequently tainted by unreliable noise signals. It is difficult to accurately detect pollutants because of these entangled signals, which conceal the relationship between atmospheric conditions and pollutant characteristics. In order to extract clearer signals derived from the original data, it is important to detangle the various frequency signals in order to enhance the interpretability and accuracy of forecasts.

Designing a comprehensible network to record these correlation rules also becomes crucial.

It is extremely important for both human health and governmental decision-making to have an efficient system in place to monitor and forecast air pollution in advance. Since the complicated structure to the features, including non-linear characteristics over time and location, PM2.5 generation is a very complex process and mechanism [11]. These intricacies have a substantial influence on forecast accuracy and need for thorough analysis and thought. The significant temporal relationship that air quality data displays further suggests that it may be categorized as a time series with discernible periodic patterns. The timely nature of the data emphasizes the importance of time forecasts, which have become key issues needing careful consideration from researchers and academics. Thus, time series analysis is crucial for there are several uses, including those in astronomy, geology, and other natural sciences as well as economics and medicine. Traditional statistical approaches have been widely utilized to solve the issues of air forecasting. These strategies rely heavily on historical data for teaching purposes. Two prominent statistical approaches used in predicting air quality are the Autoregressive Integrated Moving Average (ARIMA) and the Autoregressive Moving Average (ARMA) [12, 13]. However, because of the lengthy training times these approaches call for, they are no longer able to fully satisfy the practical requirements as the number and complexity of the available data have increased.

Machine learning-based prediction techniques are more and more common as a result of developments in artificial intelligence and the growth of large data. These models have the benefit of not requiring awareness of the chemical and physical properties of air pollutants. Support Vector Regression (SVR), Multiple Linear Regression (MLR), Artificial Neural Networks (ANN), and Random Forest (RF) are the majority well-liked machine learning methods. These methods are capable of capturing complicated and non-linear correlations between meteorological factors and air pollution concentrations. To predict air pollution in diverse regions, researchers have created numerous ANN architectures. The neuro-fuzzy neural network (NFNN) [14] and the Bayesian neural network [15] are two examples. Additionally, a multi-machine learning algorithm ensemble approach has shown reliable and accurate in forecasting levels of pollution in Greater London region [16].

The performance of predictions is enhanced when GCN-LSTM and CNN-LSTM models are merged [17, 18]. These models combine the benefits of CNN/GCN in extracting LSTM and spatial information in capturing temporal dependencies. However, the complexity of these models makes it difficult to interpret their predictions and comprehend how they behave, making it difficult to put effective air pollution mitigation strategies into action. The complex and dynamic structure of the atmospheric environment also causes air pollution data to be made up of intertwined signals of various instances, frequently complemented by arbitrary sound, which reduces forecast precision even more.

1.2 Research Motivation

Despite the development of various approaches for prediction of air quality, like Bi-LSTM, LSTM, CNN, GRU, CNN-GRU and CNN-LSTM, this article will perform a research study. Comparing and analyzing the results produced by various methods is the main goal in order to determine how well they forecast PM2.5 concentration. To provide precise forecasts with high accuracy, the research also objectives to create a forecasting system for PM2.5 that integrates meteorological data with the concentration data from neighboring stations. In this study, we used cutting-edge deep neural networks to create a system for PM2.5 prediction. To be more precise, we presented a crossbreed CNN-LSTM predicting model.

1.3 Research Contribution

We proposed a new DL Model (CLARP) CNN-LSTM-Attention Mechanism-Recurrent Mechanism-Pooling Mechanism, to anticipate the hourly PM2.5 concentration. Major contributions of this study are: -

- The study uses harmful substances, weather information, and nearby stations across various time periods as input factors. The data is preprocessed by handling missing values, handling outliers, standardized the data, encoding categorical values, feature scaling and feature engineering it. In addition, the association between the features and PM2.5 concentration is examined to help choose the best features. In our study, frame the data as supervised learning. In comparison to other techniques, the findings show that the CLARP is more effective at extracting spatiotemporal information and obtaining greater prediction accuracy for PM2.5.
- Our intended model is capable of extracting the spatiotemporal properties of the data. The model excels at collecting spatial properties, such as the interactions between different pollution components, weather patterns, and nearby stations. In addition, information is extracted using our proposed model.
- We have effectively proved the applicability and viability of the proposed model for the PM2.5 concentration forecast by contrasting the performance of the seven widely used DL algorithms in forecasting air contamination. We have verified its performance by comparing metrics across various batch sizes and latency. Additionally, the outcomes of this investigation are comparable to those of other sophisticated DL techniques that have been described in prior literature.
- We propose a completely lightweight model with efficient processing speed, can perform quickly. To improve the model's effectiveness, a variety of optimization strategies are applied.

1.4 Research Objectives

The fundamental tenets for this research thesis are summarized in the following broad range of objectives:

- **RO1**: To review studies related to air quality prediction.
- **RO2**: To apply multiple deep learning techniques and propose a technique that gives the most accurate air quality index (AQI) results.
- **RO3**: To evaluate quality parameters like the performance and accuracy of the proposed model.

1.5 Relevance to National Needs

- Air quality forecast and assessment systems help decision makers to improve air quality, mitigate the occurrence of acute air pollution episodes, particularly in urban areas, and reduce the associated impacts on agriculture, economy, ecosystems, and climate.
- Accurate prediction of air quality will help government health departments to take early preventive measures for any disease and save many lives.
- Pakistan is a country prone to various natural disasters, which can disrupt software development projects. Accurately predicting the air quality index (AQI) can help National Disaster Management Authority (NDMA) to anticipate and mitigate these disasters by taking preventive actions.
- It will also help the National Highways Authority (NHA) to plan ahead the traffic accordingly as well.

1.6 Area of Application

Application of this research will aid these industries in minimizing potential risks and enhancing project outcomes: -

- Weather forecasting.
- Health industry.
- Traffic control management.
- Supply Chain Management.

- Disaster Management Authority.

1.7 Thesis Outline

This thesis is divided into six chapters:

- **Chapter 1**: This chapter includes the basic introduction, establishes the research motivation and research contribution.
- **Chapter 2**: This chapter describes the literature survey of articles related to this research.
- **Chapter 3:** This chapter describes the different deep learning models.
- **Chapter 4**: This chapter describes the proposed Model.
- **Chapter 5**: This chapter describes the experiment results and compares our work with state-of-the-art research.
- **Chapter 6**: This chapter presents the discussion on the overall research and highlights the direction for future work.

Chapter 2: Literature Review

2.1 Overview

In this chapter, related studies to air quality prediction are referred. As we know that Air quality prediction is an emerging topic in the environmental field, and the common prediction methods are numerical simulation, statistical methods, and Machine Learning. Earlier studies on air quality prediction mostly used numerical simulation. But these simulation methods place high demands on the dataset and assume that the pollution emission is constant, which is not true since pollutants are emitted randomly in fact. And in the case of numerical simulation methods, these produce complex calculations, which are not user-friendly. Therefore, Machine learning has been a popular choice for air quality forecasting because it is good at dealing with nonlinear problems.

Overall, this chapter demonstrates the great potential of deep learning models for prediction of air quality, providing hope for more precise and effective analysis in air pollution / environmental field.

2.2 Related Work

PM2.5 is a term used to describe particulate matter (PM) that is less than 2.5 micrometers in size, or about 3% of the diameter of a human hair [19]. These are incredibly light and small particles. PM2.5 particles are so small that they tend to float in the air representing sustained periods of instance than bigger elements, which expands the potential that humans and mammals will breathe them in. Additionally, due to their tiny size, certain particles can reach the circulatory system and the lungs, as well as other parts of the respiratory system [20]. Forecasting PM2.5 levels is crucial for the development of smart cities due to the pressing need to manage PM2.5 air pollution in metropolitan areas. Predicting PM2.5 concentrations, however, is extremely difficult because of the impact of climatic variables such wind speed and direction. These factors regularly change across various time periods and show a high level of unpredictability [21, 22].

Different PM2.5 prediction strategies have been researchers created utilizing arithmetical simulations and machine learning approaches. DNN has lately gained favor in the academic world as a method for predicting pollution concentrations. DL approaches make use of several layers and big datasets, enabling simultaneous processing across all levels to provide extremely accurate results [23]. DL is ideal for air contamination modelling and predicting due to its beneficial properties, for instance its capacity for handling difficult issues and making precise forecasts.

Many different types of models may be used for this. Utilizing four models FBProphet (Facebook prophet), —ARIMA, CNN—, LSTM, and the authors of a study by [24] evaluate and analyze PM2.5 level forecasts for 12 stations in Beijing. LSTM beat all other models for the majority of stations, attaining an MAE of 13.2 and RMSE of 20.8 by using chronological air superiority data, atmospheric data, and climate prediction data. A forecast model for PM awareness at 25 monitoring sites in South Korea (Seoul), is proposed by the authors of a different research [25] by using chronological PM2.5 awareness and atmospheric data. The models that were evaluated were the DAE (Denoising AutoEncoders) and LSTM, and the findings demonstrated that the LSTM prediction model had higher accuracy than the model (DAE).

Bi-LSTM model for PM2.5 concentration prediction in China is presented by the authors in [26]. Hourly data from the US Embassy in Beijing, including the weather and PM2.5 intensity, are utilized as input in the model. The intended model's RMSE was 9.86, SMAPE was 0.1664 and MAE was 7.53, indicating outstanding accuracy.

Similar to this, other researchers have concentrated on foretelling Beijing's PM2.5 pollution levels using other models. They used an ANN, LSTM-fully connected, LSTM, and chronological air attribute, atmospheric, forecast, and day of the week data. With an MAE of 23.97 and an RMSE of 35.82 during a time period of 1-6 hours, the LSTM-FC model outperformed the ANN and LSTM models among these [27]. However, none of these models included information on pollution concentrations from nearby places. It is essential to take geographical information into account in the modeling process since modifications in contaminants are affected not only by time but also by spatial variables.

A CNN is made up of several convolutional layers that are used to take spatial data out of neural networks. It is a well-known area of research for interpreting environmental conditions from digital photographs because of its outstanding performance in handling multi-dimensional spatial arrays. The authors of [28] suggest utilizing an ensemble of deep neural networks to predict PM2.5 concentrations from photos taken outside. As the basic learners, the ensemble uses three CNN: Inception-v3, Resnet50 and VGG-16. The experimental findings illustrated that the suggested combination performs excellent in predicting PM2.5 focuses than each specific DL network.

CNN has demonstrated to be an effective approach for handling spatial data. Additionally, it has been utilized to calculate contamination awareness in municipal zones, usually by analyzing satellite pictures [29, 30]. Just separated examining data, for instance wind speed, temperature, and position may be supplied in situations where picture data is not accessible.

Researchers suggested using ConvLSTM (Convolutional Long Short-Term Memory), a hybrid model combining CNN and LSTM, to focus the problem of air contamination in Korea (Seoul). This method successfully captures the data's geographical and temporal characteristics [31]. This study's spatiotemporal model takes into account a number of variables, including information on air pollution, weather, indices of outdoor air pollution, traffic volume, and average driving speed locations. The proposed model has proven to be better than competing models.

The authors of a different research [32] looked at the viability and usefulness of utilizing CNN-LSTM to forecast the PM2.5 intensity in Beijing for the next period. The model also included the total amount of hours of rain and wind speed during the previous 24 hours. The model CNN-LSTM fared superior to other models, as seen by its MAE of 14.6344 and RMSE of 24.22874.

2.3 Summary

In this chapter different Machine Learning Models related to air quality prediction have been studied and referred. As we know that Machine learning has been a popular choice for air quality forecasting because it is good at dealing with nonlinear problems. Deep learning is a branch of machine learning. Among the many algorithms for deep learning, the long short-term memory network (LSTM) is often used to predict air quality due to its effectiveness in solving long- distance dependence. But our proposed model has improved results as compared to all these models.

Chapter 3: Deep Learning Models

3.1 Overview

In this chapter our goal is to assess how well different DL models predict PM2.5 concentrations. As a result, we have determined to use the LSTM, Bi-LSTM, GRU, Bi-GRU, CNN, PM-GRU, RM-LSTM models that were in the past described. We give a succinct summary of each network.

3.2 LSTM

RNN, which first appeared in 1980 [33, 34], is the basis for LSTM. An effective artificial neural network type frequently used for time-series forecasting problems is the RNN. They have the capacity for internal memory retention, which enables them to recall details from the past and anticipate the course of the future. RNNs frequently run into problems with disappearing and expanding gradients, which can slow down or stop the learning process completely. LSTMs were first developed in 1997 to resolve these challenges [35]. Standard RNNs have several limitations, but LSTMs have larger memory spans and can acquire from inputs that are temporally estranged from one another. Three essential gates make up an LSTM architecture: an input gate, a forget gate, and an output gate. The output gate identifies the data to be outputted, the forget gate chooses the irrelevant information to delete, and the input gate decides whether to integrate fresh data. These gates act analogously, taking their cue from the way logic gates work. Computing the hidden states in model are depicted in **Figure 3.1**. LSTM formulae are listed:

The input gate (i) is defined as:

$$i(t) = (Wi * [h(t-1), x(t)] + bi) \quad (1)$$

*Ign*ore Gate (f), which is defined as:

$$f(t) = (Wf * [h(t-1), x(t)] + bf) \quad (2)$$

The output gate (o) is as follows:

$$o(t) = (Wo * [h(t-1), x(t)] + bo) \quad (3)$$

Candidate Cell State:

$$(t): Wc * [h(t-1), x(t)] + bc \quad (4)$$

The formula for cell state:

$$(C) is: C(t) = f(t) * C(t-1) + i(t) * (t) \quad (5)$$

The formula for Hidden State:

$$(h) is\ h(t) = o(t) * tanh\big(C(t)\big) \quad (6)$$

In the formulae above: T stands for the most recent time step. The sigmoid activation function is indicated. The matrix multiplication symbol is *. [h(t-1), x(t)] denotes the concatenation of the current input (x(t)) with the prior concealed state (h(t-1)). The weight matrices Wi, Wf, Wo, and Wc, as well as the bias vectors bi, bf, bo, and bc, are unique to each gate.

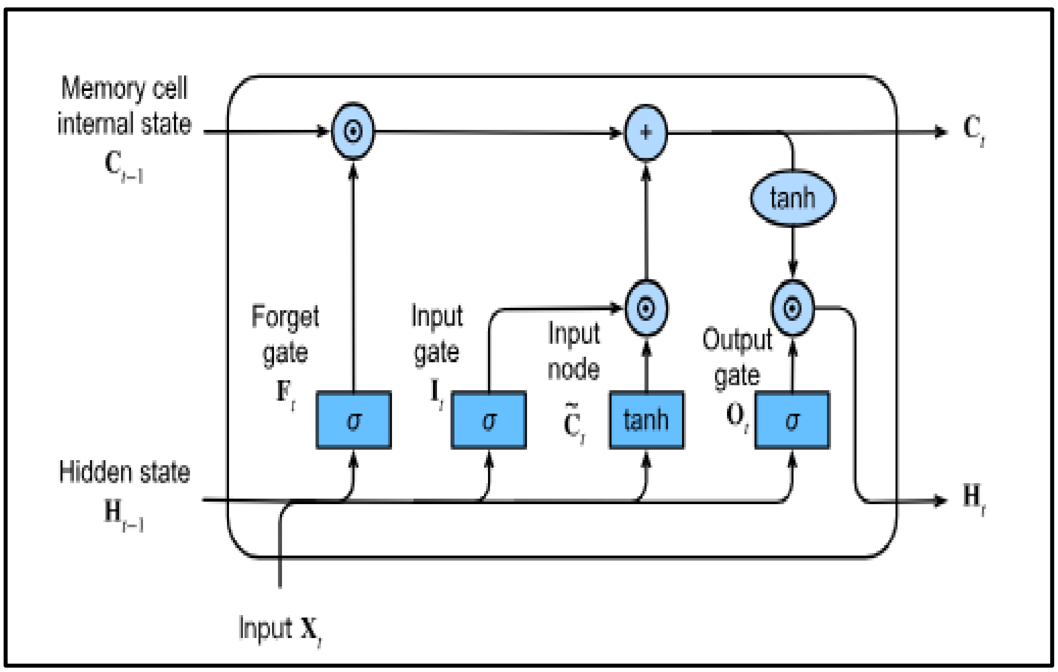

Figure 3.1 Computation of Hidden States of LSTM

3.3 GRU

The GRU, also known as a Gated Recurrent Unit, is a developed RNN [36]. It is a kind of recurrent unit with LSTM-like characteristics. The reset gate and the update gate are the two primary parts of the GRU unit. The GRU model's design is shown in **Figure 3.2**. The reset gate gives the model the ability to overlook the earlier condition relating the first candidate initiation and the second activation. The update gate, on the other hand, chooses the proposed activation that changes the state of the cell. GRU formulae are provided as following:

Reset Gate (r) is defined as:

$$r(t) = (Wr * [h(t-1), x(t)] + br) \qquad (7)$$

Update Gate:

$$(z): (Wz * [h(t-1), x(t)] + bz)z(t) \qquad (8)$$

Candidate Activation:

$$(h): W * [r(t)\, h(t-1), x(t)] + b) = h(t) \qquad (9)$$

Hidden State:

$$(h): h(t) = (z(t) - 1)Z(t) + h(t-1) = h(t) \qquad (10)$$

In the formulae T stands for the most recent time step. The sigmoid activation function is indicated. The hyperbolic tangent activation function is represented by tanh. The symbol for element-wise multiplication is "."[h(t-1), x(t)] denotes the concatenation of the current input (x(t)) with the prior concealed state (h(t-1)). The bias vectors for each gate are br, bz, and b, while the weight matrices are wr, wz, and w.

These formulae show how a GRU cell's calculations were done. The update gate (z) decides how widely latest data will be integrated, while the reset gate (r) governs how widely data from the prior concealed state is used. The reset gate, the prior hidden state, and the current input are all combined in the candidate activation (h). Finally, the prior hidden state and the new candidate activation are combined to update the hidden state (h) depending on the update gate.

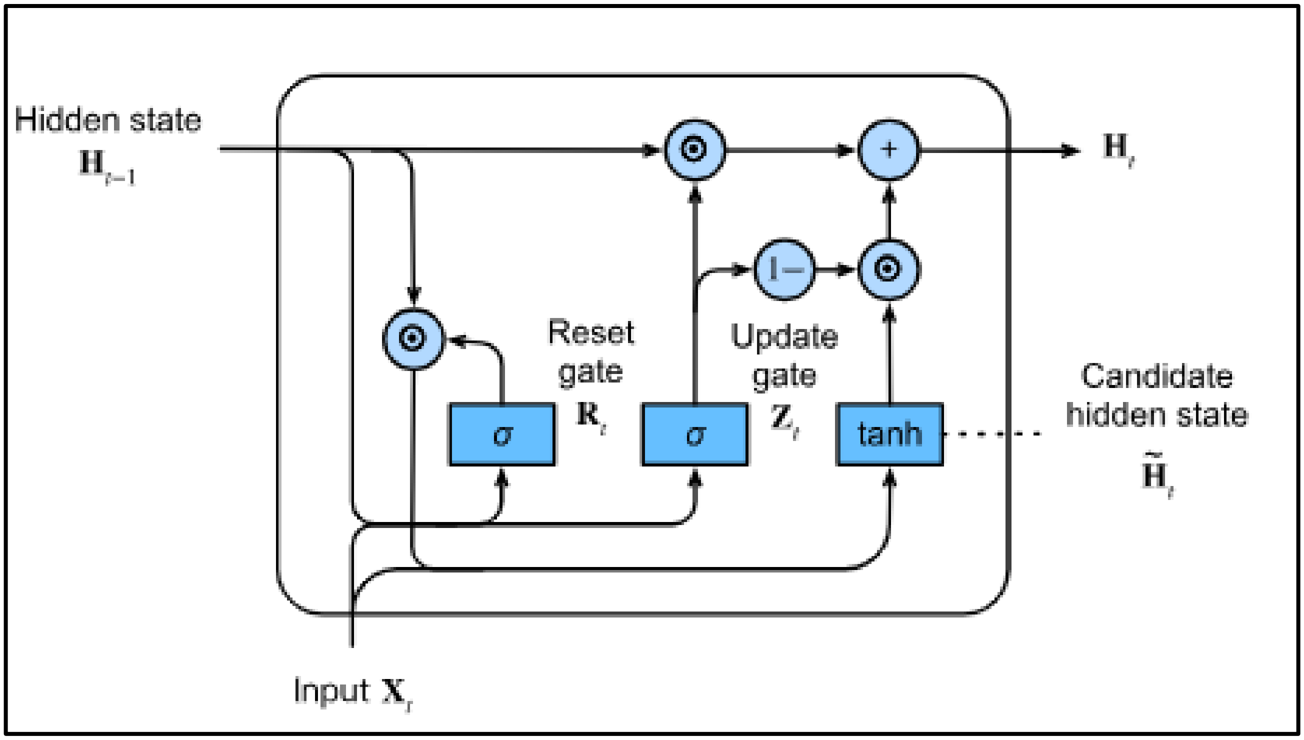

Figure 3.2 Computation of hidden states in GRU

3.4 Bi-LSTM

Bidirectional LSTM (Bi-LSTM) is meant to take use of such future information, in contrast to traditional RNNs and LSTMs that have a tendency to ignore it during time-processing tasks. For each training sequence, Bi-LSTM uses two LSTM networks, one of which analyses the sequence one going forward and the other going backward. There are input and output layers connecting these LSTM networks. By combining data from each point through the bidirectional arrangement, this structure enables the output layer to receive historical information from every entry in the input sequence and to also catch forthcoming data. Bi-LSTM formulae are listed below:

The input gate (i) is classified as:

$$i(t) = (Wi * [h(t-1), x(t)] + bi) \qquad (11)$$

Ignore Gate (f), which is defined as:

$$f(t) = (Wf * [h(t-1), x(t)] + bf) \qquad (12)$$

Output gate (o) is as follows:

$$o(t) = (Wo * [h(t-1), x(t)] + bo) \quad (13)$$

Candidate Cell State:

$$(t): Wc * [h(t-1), x(t)] + bc \quad (14)$$

Cell state (C) is:

$$C(t) = f(t) * C(t-1) + i(t) * (t) \quad (15)$$

Hidden State (h) is:

$$h(t) = o(t) * tanh\big(C(t)\big) \quad (16)$$

The calculations made in a bidirectional LSTM (Bi-LSTM) cell are represented by above mentioned formulae. The input sequence is processed both forwardly (from the beginning to the end) and backwardly (from the end to the beginning) in the Bi-LSTM architecture. Concatenation is used to combine the LSTMs' outputs from both directions. Architectural flow is shown in **Figure 3.3** as below:

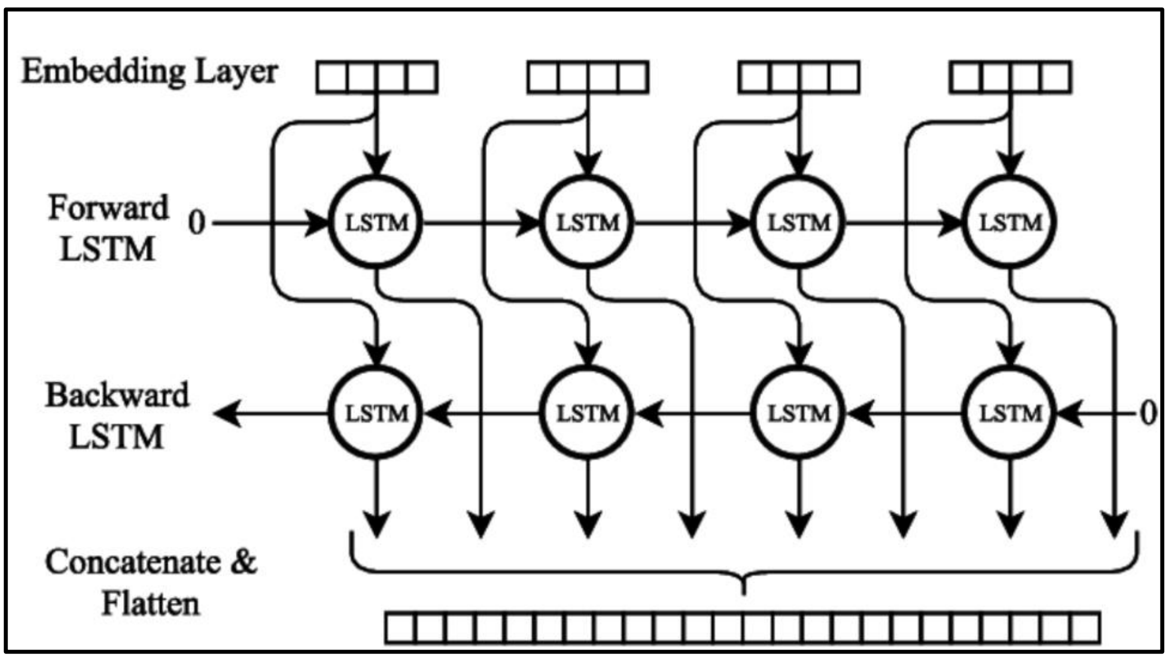

Figure 3.3 Architectural flow of Bi-LSTM Model

3.5 Bi-GRU

Despite the Recurrent neural network (RNN) architectures can input sequences that are processed concurrently in both the directions both forward and backward. One such design is the Bidirectional Gated Recurrent Unit (Bi-GRU). It combines bidirectional processing with the benefits of Gated Recurrent Units (GRU) to collect data from both past and future contexts. Bi-GRU formulas are listed:

Reset Gate (r) is defined as:

$$r(t) = (Wr * [h(t-1), x(t)] + br) \qquad (17)$$

Update Gate:

$$(z): (Wz * [h(t-1), x(t)] + bz)z(t) \qquad (18)$$

Candidate Activation:

$$(h): W * [r(t)\, h(t-1), x(t)] + b) = h(t) \qquad (19)$$

Hidden State:

$$(h): h(t) = (z(t) - 1)Z(t) + h(t-1) = h(t) \qquad (20)$$

In the formulae above, T stands for the most recent time step. The sigmoid activation function is indicated. The hyperbolic tangent activation function is represented by tanh. The symbol for element-wise multiplication is "." For both the forward and backward GRU networks, [h(t-1), x(t)] denotes the concatenation of the prior hidden state with the current input. The bias vectors for each gate are br, bz, and b, while the weight matrices are wr, wz, and w. Structure of Bi-GRU is depicted in **Figure 3.4**.

The Bi-GRU may capture dependencies from both earlier and later sequence components by processing the input order in equal directions. This enables the model to comprehend the temporal relationships in the data more thoroughly. Bi-GRU is frequently used in sequential data applications including time series analysis, speech recognition, and natural language processing.

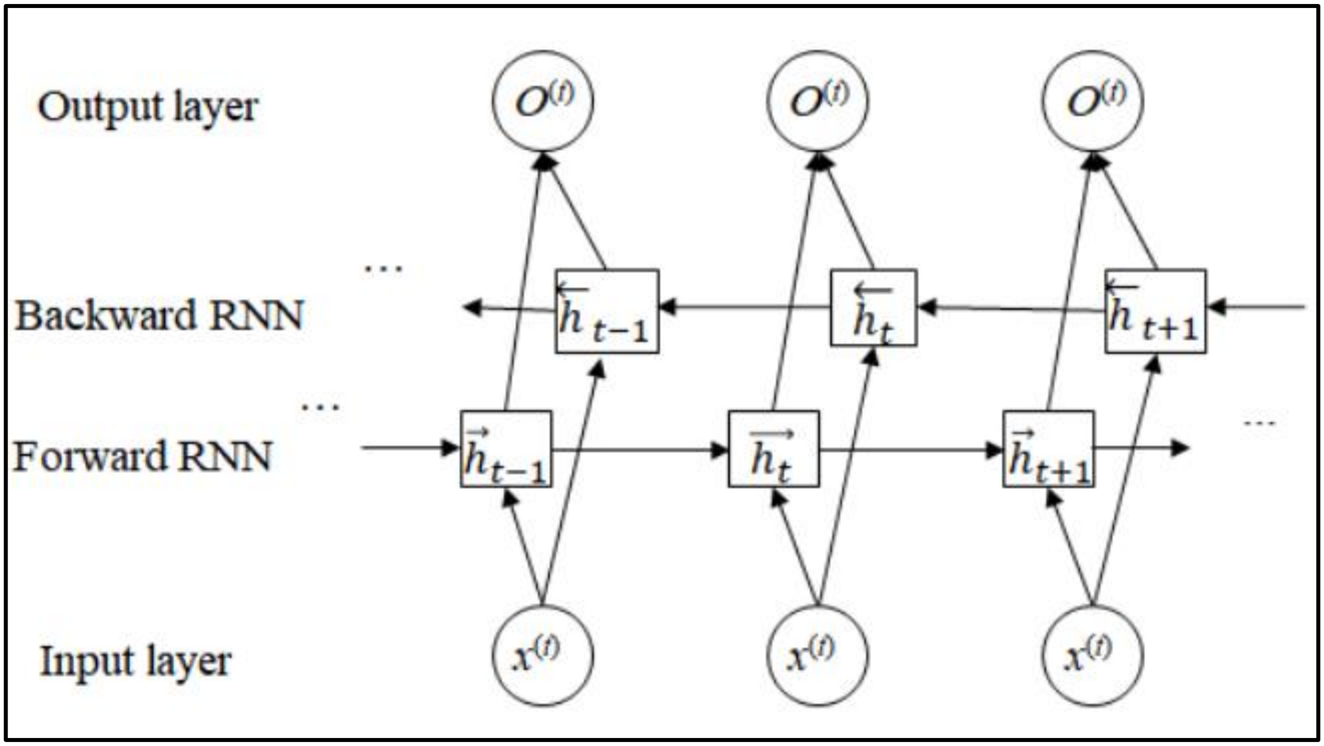

Figure 3.4 Structure of Bi-GRU

3.6 PM-GRU

Convolutional neural networks (CNNs) employ the pooling mechanism as a method to scale down the spatial dimensionality of feature maps. It works with the feature maps' local neighborhoods and compiles data to create a compressed representation. The feature maps are divided into distinct, non-overlapping areas for the pooling procedure, and each zone is given a different pooling function, such as max pooling or average pooling. The goal of pooling is to minimize the size of the feature maps while retaining the most important data. For instance, maximum pooling chooses the highest value possible within each pooling zone, whereas average pooling determines the average value. Pooling allows the system to save the highly important portions while rejecting the less important ones, which helps to cut down on computation and control overfitting.

A recurrent neural network (RNN) design known as a gated recurrent unit (GRU) overcomes the drawbacks of conventional RNNs, such as disappearing or exploding gradients. In order to improve learning and identify long-term relationships in sequential input, GRU incorporates gating techniques that let the network selectively keep or update knowledge over time.

The network can successfully reduce the spatial dimensionality of feature maps while capturing dependencies in sequential data by combining the pooling strategy with GRU. In many different applications, CNNs with GRU layers are used in combination to process input sequences and extract useful information.

3.7 RM-LSTM

RNNs are based on the core idea of the recurrent mechanism, which allows networks to process sequential data by retaining a hidden state that stores the network's memory of prior inputs. RNNs include connections that allow input to be sent back into the network, providing a loop-like structure, in contrast to feedforward neural networks, which process data strictly sequentially.

Long Short-Term Memory (LSTM), an RNN architecture, was explained to focus the limitations of standard RNNs in acquiring long-term dependencies. Processing data sequences with temporal delays or dependencies over longer time periods is where LSTMs excel most.

The network can successfully simulate temporal dependencies and capture long-term relationships in sequential data by integrating the recurrent mechanism with LSTM. The LSTM design, with its memory cells and gating methods, enables the network to choose add or forget information, addressing the disappearing or expanding gradient problem. The recurrent mechanism allows the network to keep recollection of previous inputs. Because of this, LSTMs are very good at tasks requiring sequential data, such time series analysis, speech recognition, and natural language processing.

3.8 Summary

In this chapter, different models' working has been described in detail. LSTMs overcomes the problems faced in RNN as LSTMs have larger memory spans and can acquire from inputs that are temporally estranged from one another. Three essential gates make up an LSTM architecture: an input gate, a forget gate, and an output gate. GRU is a kind of recurrent unit with LSTM-like characteristics. It has 2 gates i.e. The Reset gate and the Update gate. Bidirectional LSTM (Bi-LSTM), (in contrast to traditional RNNs and LSTMs that tend to ignore future information during time-processing tasks) is meant to make use of such future information. Bi-GRU combines bidirectional processing with the benefits of GRUs to collect data from both past and future contexts. PM-GRU has a network that can successfully reduce the spatial dimensionality of feature maps while capturing dependencies in sequential data by combining the pooling strategy with GRU. In many different applications, CNNs with GRU layers are used in combination to process input sequences and extract useful information. RM-LSTM has a network which can successfully simulate temporal dependencies and capture long-term relationships in sequential data by integrating the recurrent mechanism with LSTM.

Chapter 4: Proposed Methodology & Framework

4.1 Overview

In this chapter, we have proposed a methodology which produces better results as compared to other models. LSTMs overcomes the problems faced in RNN as LSTMs have larger memory spans and can acquire from inputs that are temporally estranged from one another. GRU is a kind of recurrent unit with LSTM-like characteristics. Bidirectional LSTM (Bi-LSTM) combines bidirectional processing with the benefits of GRUs to collect data from both past and future contexts. PM-GRU has a network that can successfully reduce the spatial dimensionality of feature maps while capturing dependencies in sequential data by combining the pooling strategy with GRU. RM-LSTM has a network which can successfully simulate temporal dependencies and capture long-term relationships in sequential data by integrating the recurrent mechanism with LSTM. Developed methodology is combination of these models.

4.2 Proposed Approach

An effective hybrid DL model for forecasting PM2.5 air contamination in smart cities. In order to make use of each component's capabilities in identifying spatial and temporal patterns in air quality data, the CLARP model integrates CNNs, LSTM, Attention Mechanism, Recurrent Mechanism, and Pooling Mechanism. The goal is to construct a reliable and precise model that can accurately forecast PM2.5 pollution levels and enable proactive air quality control strategies in smart cities.

The architecture of the hybrid model is described, which combines a number of elements to identify spatial and temporal patterns in the data. The architecture consists of a CNN for extracting spatial features, a Long Short-Term Memory (LSTM) for capturing temporal dependencies, an Attention Mechanism for concentrating on crucial portions of the input, and optional recurrent layers for capturing additional temporal dependencies. In order to process and convert the pooled data for the final prediction, the model also includes fully linked layers and a pooling mechanism to collect significant characteristics. The preprocessed data is then used to assemble and train the model. Mean Absolute Error

(MAE), Mean Squared Error (MSE), and Measure, and the Adam optimizer are described. With a predetermined batch size and number of epochs (iterations), the model is fitted to the training data. Using the testing data, the model's performance is assessed after training, and metrics such as mean absolute error (MAE) and loss (MSE) are calculated. On the base of the analyzing findings, the model is then utilized to establish predictions.

4.2.1 Dataset Acquisition and Pre-Processing

We acquired the data from the Beijing Municipal Environmental Monitoring Centre, which offers an openly accessible data site for air attribute statistics, to obtain the air pollution forecast dataset for Beijing, China as shown in **Figure 4.1**. The dataset contains elements including the concentration of PM2.5, meteorological information (temperature, humidity, wind speed, wind direction, and atmospheric pressure), timestamps, and location data for the monitoring places spread out over Beijing. In addition to obtaining authorization to use the data for study, we abided by ethical standards and data protection laws. After obtaining the raw dataset, we preprocessed it by cleaning the data to eliminate outliers and missing values, standardizing variable units, addressing temporal inconsistencies, and aggregating the data into hourly or daily intervals. Additionally, we made sure the dataset was properly documented by outlining its sources, variables, data gathering techniques, and preparation methods used. During the whole gathering and preparation procedure, ethics and compliance with data privacy laws were upheld.

To assure its quality and appropriateness for analysis, the obtained Beijing, China, air pollution prediction dataset underwent a number of preprocessing stages. The subsequent actions were taken:

4.2.2 Handling of Missing Values

The collection is made up of 35,064 records from various stations, each of which has numerous characteristics. The recordings were made between March 1st, 2013 and February 28th, 2017. Date, PM2.5 concentration, PM10 concentration, Sulphur dioxide (SO2) levels, Nitrogen dioxide (NO2) levels, carbon monoxide (CO) levels, ozone (O3) levels, dew point, temperature, atmospheric pressure, combined wind direction, cumulated wind speed, cumulated hours of snow and rain are all included in the data. Nonetheless, data loss due to machine failure or other uncontrolled circumstances is a risk in the collecting of air quality and meteorological data. These missing value cases might have an influence on data mining. Mean or median imputation are often used techniques to substitute missing field values when working with time-independent (non-chronological) data. However, this method is not appropriate for time series data. In these situations, several imputation approaches are used to solve the problems with partial data. The linear interpolation approach outperformed other methods across all percentages of missing values, according to research on the estimate of hourly monitoring data for PM10 in the presence of simulated missing values [37]. There are less than 4% missing values in the processed dataset, which were resolved using linear spline imputation. Local anomalies can be accounted for using the SL(y) equation while keeping interruption consequences at other data peaks. The following equation represents the SL(y) linear spline interruption function. This is how it is defined:

$$SL(y) = f(yi-1) * \frac{(y-yi)}{}(yi-1 - yi) + f(yi) * \frac{(y-yi-1)}{}(yi - yi-1) \tag{21}$$

where y belongs to the interval [yi-1, yi] and i takes values from 1 to n. The function calculates the interpolated value at a given point y based on the neighboring data points yi-1 and yi, using linear interpolation.

The presented equation denotes the linear spline interpolation function, which is used to estimate values between two known data points. Here's how the equation works: SL(y): This is the interpolated value at position y. f(yi-1): This is the known function value at data point yi-1, which is the data point immediately preceding x. (y-yi) / (yi-1-yi): Based on the distance between y and the two neighboring data points, this term computes the weight for the function value at yi-1. It specifies how much the value at yi-1 influences the interpolated value. f(yi): This is the known function value at data point yi, which is the data point after y. (y - yi-1) / (yi - yi-1): This term determines the weight for the function value at yi depending on the distance between y and the two adjacent data points. It specifies how much weight the value at yi has on the interpolated value.

To estimate the value at point y, the equation effectively combines the weighted contributions of the function values at yi-1 and yi. The linear spline interpolation function adjusts to the local behavior of the data by computing these weights depending on the distances, allowing for more accurate approximations between the provided data points.

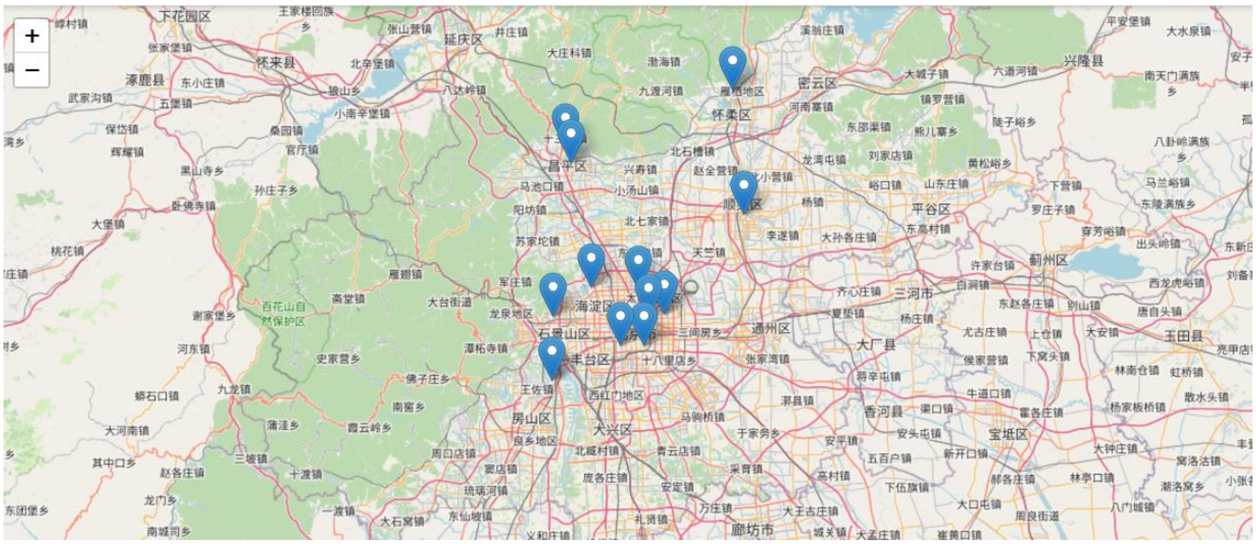

Figure 4.1 Beijing's Monitoring Stations distribution

4.2.3 Dealing with Outliers

Several ways may be used to deal with outliers in the Beijing, China air pollution forecast dataset. To begin, domain-specific criteria based on regulatory requirements or expert knowledge can be set to detect observations exceeding permitted air pollution levels. To detect values that deviate considerably from the predicted range, statistical approaches such as z-scores or modified z-scores can be used. By studying the geographical distribution of air pollution levels throughout Beijing's monitoring stations, spatial analytic techniques may be used to find outliers. Temporal analysis can assist in identifying abrupt and large shifts or spikes in air pollution concentrations that deviate from regular trends. Consulting with air quality professionals or topic expertise might help you validate and evaluate any anomalies. Outliers can be dealt with by eliminating them, replacing them with estimated values, or grouping them individually. The unique qualities of the data and the analytic needs should dictate the approach selection.

4.2.4 Categorical Values Encoding

The wind component is critical in understanding atmospheric activity in this analysis. Pollutant concentrations are influenced by wind speed [21], and wind path is especially important in influencing PM2.5 concentrations [22]. The wind direction property is made up of 16 categorical values: N, NNE, NE, ENE, E, ESE, SE, SSE, S, SSW, SW, WSW, W, WNW, NW, and NNW. We split the compass into 16 sectors, each spanning 22.5 degrees, to translate each cardinal wind direction into degrees of azimuth. North was given a value of zero, and as we proceed clockwise, the value grows by 22.5 for each segment, with each segment representing a 22.5-degree direction.

4.2.5 Normalization

Normalization is a data processing technique used on the Beijing air pollution dataset for PM2.5. In this context, normalization is used to scale the PM2.5 measurements to a standardized range. This improves comparability between measurements and eliminates biases caused by magnitude fluctuations. Various normalization procedures, such as Min-Max scaling or Z-score standardization, can be employed to guarantee that normalized PM2.5 measurements fall within a certain range. Normalization is a critical step in analyzing air pollution data because it ensures fair comparisons and correct interpretations.

4.2.6 Feature scaling

Feature scaling is a technique for transforming the values of features in a dataset, such as the Beijing PM2.5 air pollution data. The goal of feature scaling is to guarantee that all characteristics are on a similar scale, preventing any single element from taking over the analysis owing to magnitude disparities. Feature scaling methods such as Min-Max scaling or Z-score standardization can be used with PM2.5 data. The values are rescaled to a certain range, usually between 0 and 1. Subtracting the minimum value of the PM2.5 data and dividing it by the range (highest value minus minimum value) yields this transformation.

To standardize the Z-score, remove the mean value of the PM2.5 data and divide it by the normal deviation. The data is transformed using this approach to have a mean of 0 and a standard deviation of 1. When feature scaling is applied to PM2.5 data, all values are scaled to a comparable scale, allowing for fair comparisons, and avoiding bias due to variable magnitudes. To ensure accurate and relevant interpretations of the dataset, feature scaling is a prevalent practice in data analysis, particularly in machine learning algorithms.

4.2.7 Feature Selection

Selecting the relevant features is a critical stage in machine learning tasks, and there are several ways for doing so. Previous research has frequently used mathematical correlation approaches to detect correlations between input and output variables [38-41]. When dealing with a large number of characteristics to be included in the training process, creating a connection between the goal output value and these attributes can assist in simplifying the training complexity and improving performance [38]. The Pearson correlation coefficient is commonly used to quantify the relationship between two variables. It can be computed using the following equation:

$$r = \Sigma[(yi - \overline{y})(zi - \overline{z})] / \sqrt{[\Sigma(yi - \overline{y})^2 * \Sigma(zi - \overline{z})^2]} \qquad (22)$$

Subtract the mean value of y (y) from the individual value of y (yi) for each observation. Subtract the mean value of y (z) from the individual value of y (zi) for each observation. For each observation, double the differences obtained in steps 1 and 2. Add the products from step 3 for all observations. Divide the product of two sums: the sum of squared differences of y from its mean (ni=1(yi y)2) and the total of y's squared deviations from its mean (ni=1(zi z)2). The Pearson correlation coefficient (r) shows the degree and direction of the linear relationship between the variables y and z. The coefficient ranges from -1 to 1, with -1 indicating a perfectly negative linear relationship, 0 indicating no linear link, and 1 indicating a perfectly positive linear relationship.

4.2.8 Air Quality Feature

Various contaminants have been found in the atmosphere, and increased concentrations of these pollutants have a negative influence on air quality. We used correlation calculations to analyze the links between the air quality variables. Notably, we discovered a high link between PM2.5, PM10, and CO, as seen in Table 1.

Table 4.1 The air quality characteristic correlation matrix

	PM2.5	**PM10**	**CO**	**O3**	**NO2**	**CO2**
PM2.5	1.00	0.85	0.72	0.43	0.67	0.52
PM10	0.85	1.00	0.68	0.39	0.62	0.48
CO	0.72	0.68	1.00	0.55	0.79	0.61
O3	0.43	0.39	0.55	1.00	0.32	0.24
NO2	0.67	0.62	0.79	0.32	1.00	0.71
CO2	0.52	0.48	0.61	0.24	0.71	1.00

The extra air quality variables NO2 and CO2 are included in this enlarged correlation matrix. Each cell, like the previous explanation, indicates the correlation coefficient between two variables. Closer to 1 indicates a high positive connection, whereas closer to -1 indicates a strong negative correlation. Values close to 0 imply a weak or non-existent association. The analysis of this correlation matrix reveals information on the links between PM2.5, PM10, CO, O3, NO2, and CO2. It aids in understanding the interconnection and possible influence of these factors on air pollution levels by identifying which air quality parameters are closely associated with one other.

4.2.9 Meteorological Feature

Meteorological characteristics such as atmospheric temperature, pressure, wind speed, wind direction, and relative humidity all have an influence on air quality. Different meteorological conditions have an impact on several components of air pollution. High wind speed, for example, tends to reduce PM2.5 concentrations, whilst high humidity tends to increase air pollution levels. Furthermore, higher air pressure is often associated with improved air quality [40, 41]. As a result, meteorological characteristics are crucial in accurately predicting air quality, as shown in **Figure 4.2**.

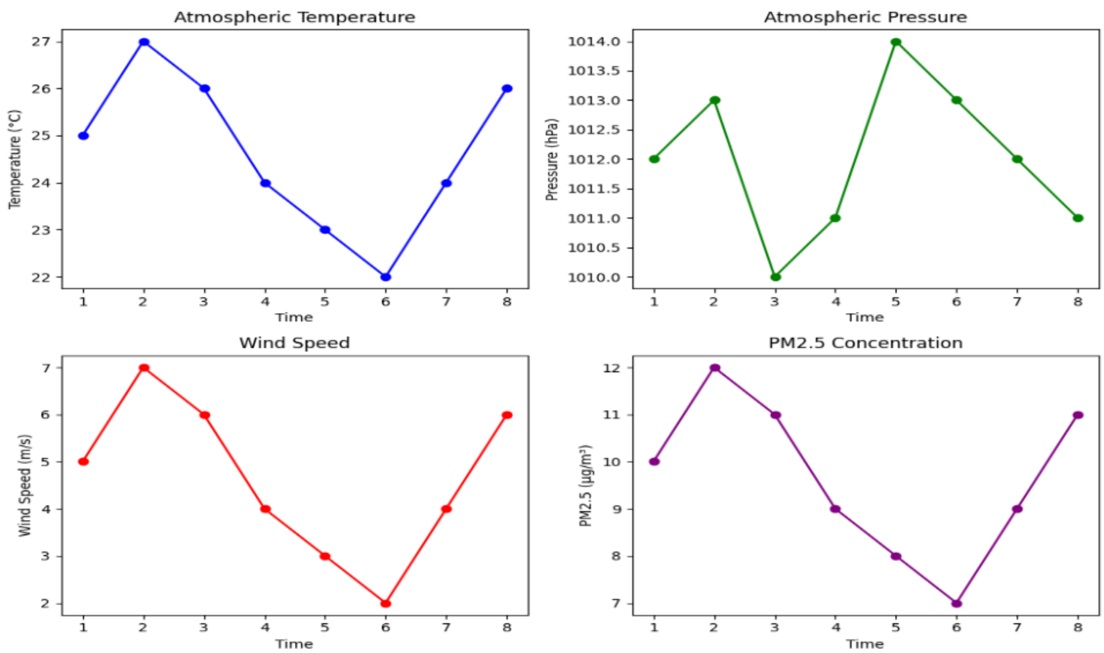

Figure 4.2 Meteorological data and PM2.5 weather characteristics (atmospheric temperature, atmospheric pressure, wind speed, PM2.5 Concentration).

4.2.10 Spatial Analysis

We conducted a spatial correlation analysis between the Aotizhongxin station (target) and neighboring stations. The Pearson correlation coefficient was utilized to identify the PM2.5 monitoring stations that exhibited strong correlations with the target station. The correlation results are presented in **Figure 4.3**, where all correlation values exceeding 0.80 indicate a robust spatial correlation among the selected stations. To facilitate model training and accuracy evaluation, the dataset was divided into a training set and a test set. The training set consisted of 80% (28,052 hours) of the data, while the remaining 20%

(7,012 hours) was allocated as the test set to assess the model's performance and analyze its accuracy.

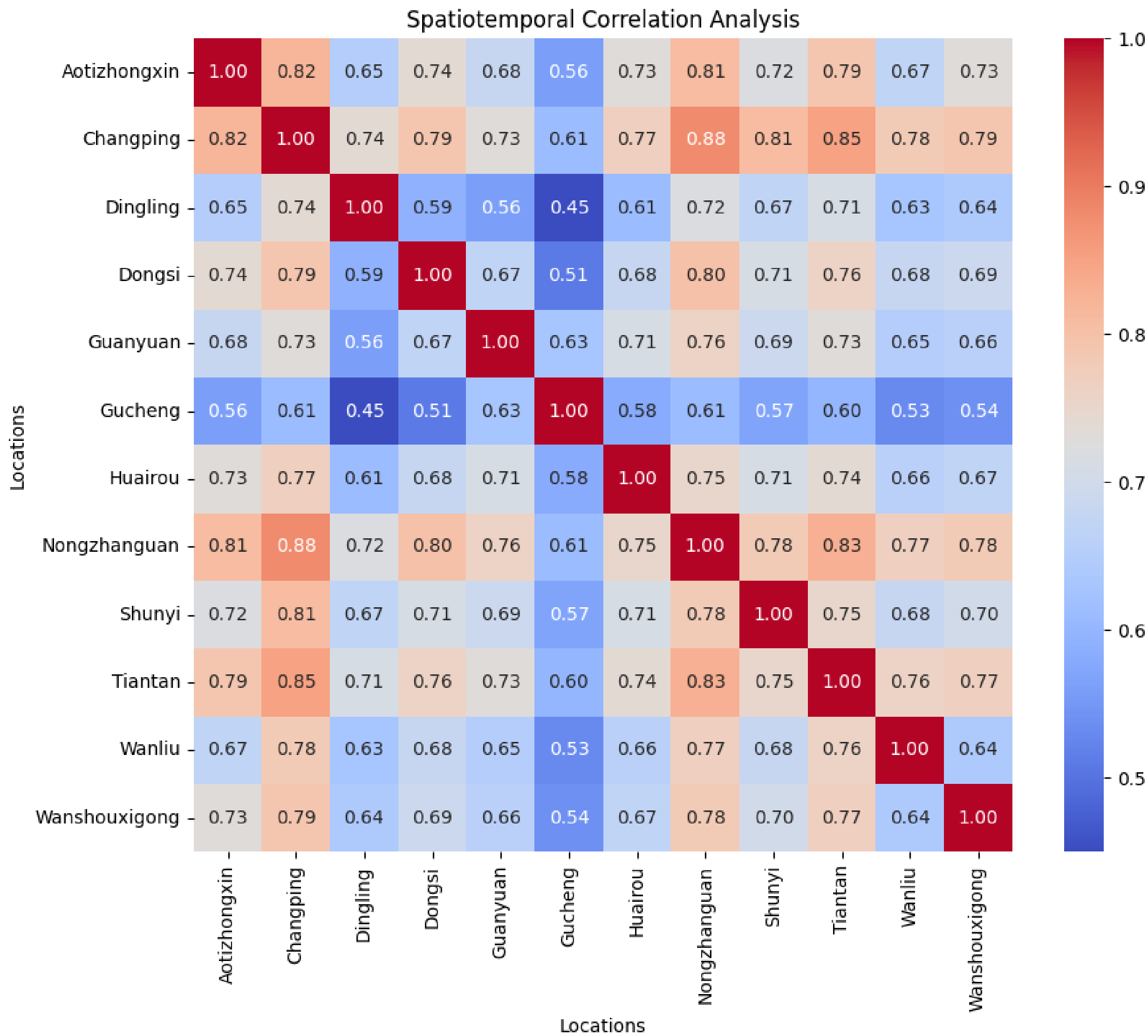

Figure 4.3. The Analysis of Spatiotemporal Correlation

4.3 Working of CLARP Model

In this CLARP model, we investigate a range of recurrent neural network combinations and specialized strategies to optimize learning and modelling capabilities. Among them are

LSTM + Bi-LSTM, GRU + Bi-GRU, PM + GRU, and RM + LSTM. We merge LSTM and Bi-LSTM networks in LSTM + Bi-LSTM, using LSTM units in several hidden layers to capture long-term dependencies and Bi-LSTM units to incorporate bidirectional processing. This combination enables the model to capture temporal patterns and dependencies well. To reduce overfitting, a 10% dropout rate is used between the layers in all of these networks, and the ReLU activation function is used for the hidden layers. Similarly, with the GRU + Bi-GRU combination, GRU and Bi-GRU networks are combined. GRU layers containing GRU units record temporal relationships, however, the Bi-GRU layers' bidirectional nature allows for the integration of input from both the past and future contexts. This combination improves comprehension of data context and dependencies. A prediction mechanism (PM) is also combined with the GRU network in the PM + GRU configuration. GRU layers record temporal relationships, whereas the PM is concerned with producing accurate forecasts or predictions. By using the taught patterns and dynamics, this combination provides exact forecasting. We combine a representation mechanism (RM) with an LSTM network in the RM + LSTM combo. The LSTM layers record long-term dependencies, with LSTM units in several hidden layers, while the RM provides meaningful and useful data representations. The model's capacity to acquire effective representations and catch sequential patterns is improved as a result of this combination. Each combination is suited to unique model architectures and requirements, with many hidden layers and adjustable units. These various combinations take advantage of the capabilities of each network and approach to capture and model temporal relationships, include bidirectional context, make accurate predictions, and provide meaningful data representations. The workflow diagram of the proposed model is shown in **Figure 4.4** and **Figure 4.5** shows the architecture diagram of CLARP.

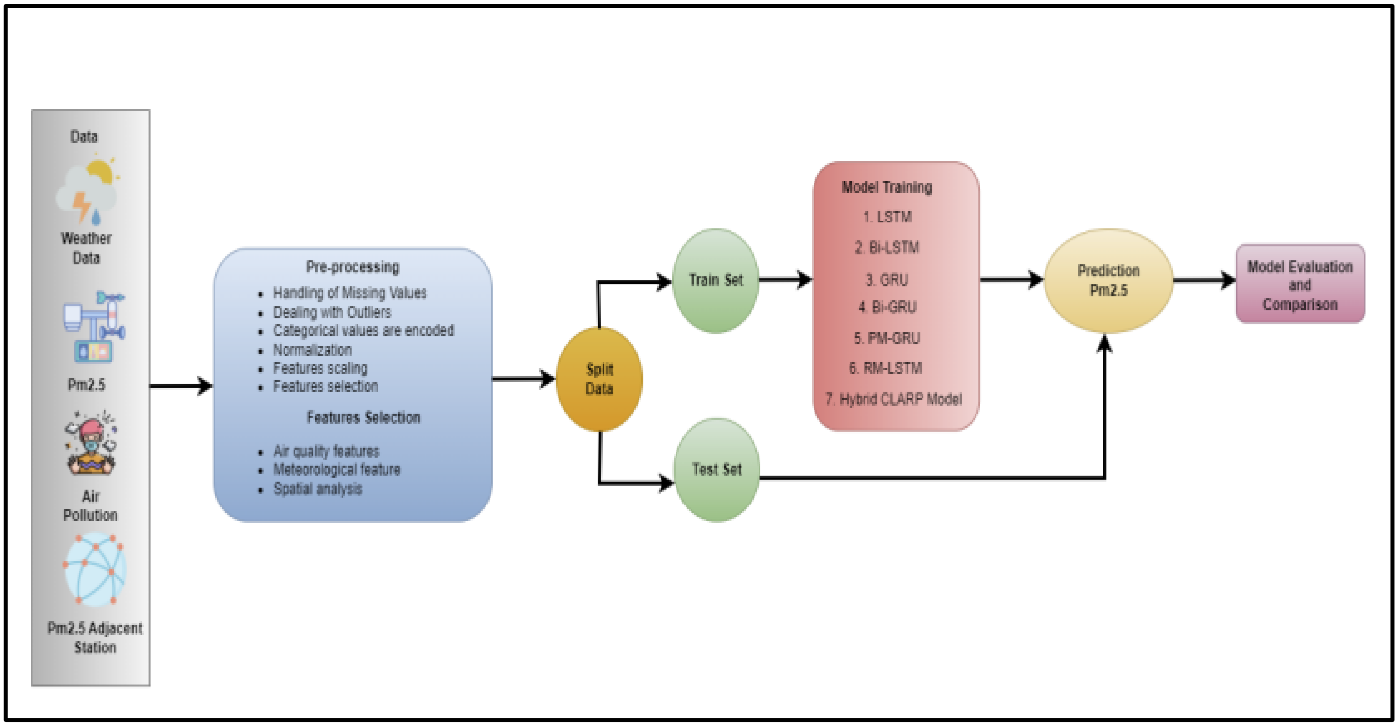

Figure 4.4 Workflow for predicting PM2.5 concentration.

.

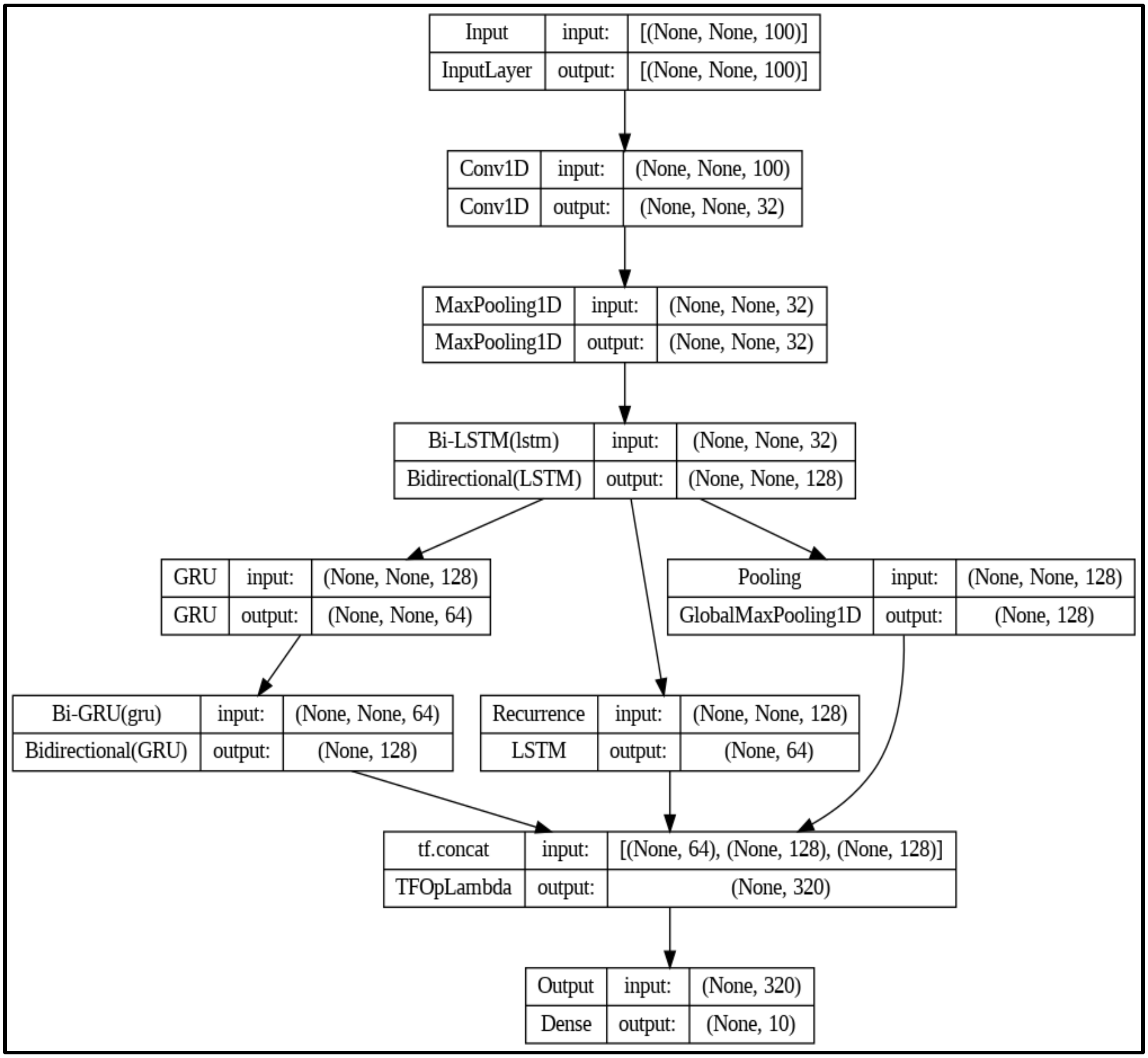

Figure 4.5 The Architecture of the proposed CLARP.

4.4 Summary

In this chapter, we have deliberately discussed our proposed model and its working. To make use of each component's capabilities in identifying spatial and temporal patterns in air quality data, the CLARP model integrates CNNs, LSTM, Attention Mechanism, Recurrent Mechanism, and Pooling Mechanism. The goal is to construct a reliable and precise model that can accurately forecast PM2.5 pollution levels and enable proactive air quality control strategies in smart cities.

Chapter 5: Results and Analysis

5.1 Overview

In this chapter, we first define different metrics to evaluate the efficiency of the model. Then defining these we compare results of different models with our proposed CLARP model. Our models were developed using a range of Python packages, such as Scikit-Learn, Keras, and native TensorFlow. To handle more demanding tasks, we utilized Google Colab, a platform that provided access to NVIDIA's Tesla T4 GPU, enabling us to execute computationally intensive workloads efficiently.

5.2 Evaluation Index of the Models

Once the model structure has been chosen, the training set is utilized to train the network until it approaches convergence. This article evaluates the model's efficacy using three metrics: mean absolute error (MAE), root mean squared error (RMSE), and coefficient of determination (R2). These indicators provide data on the model's accuracy and performance.

5.2.1 MAE

The Mean Absolute Error (MAE) is a statistic used to calculate the average size of the deviations between real data and model projections. It is computed by taking the arithmetic mean of the absolute differences across all samples, resulting in a more accurate representation of the prediction errors. MAE can be calculated using the following formula:

$$MAE = (1/n) * \Sigma |yi - \hat{}yi| \qquad 22$$

To compute MAE, we consider the absolute difference between the true (yi) and anticipated (yi) values for each sample. These absolute differences are then added together for all samples and divided by the total number of samples (n) to yield the average absolute difference, or MAE.

5.2.2 RMSE

The Root Mean Square Error (RMSE) is a metric that measures the square root of the mean of the squared errors. It provides a more comprehensive assessment of the prediction accuracy. The calculation formula for RMSE is as follows:

$$RMSE = sqrt((1/n) * \Sigma (yi - \hat{}yi)^2) \qquad 23$$

To compute RMSE, we first compute the squared difference between the true (yi) and anticipated (yi) values for each sample. These squared differences are then added together for all samples and divided by the total number of samples (n). The RMSE is calculated by taking the square root of the average squared difference between the true and projected values.

5.2.3 R2

The measurement of fortitude is a statistic that shows how much of the total modification in the supported variable can be described by the private variable via the regression relationship. A higher R2 value indicates that the individual variable has a greater capability to justify fluctuations in the dependent variable. The R2 computation formula is as follows:

$$MAE = (1/n) * \Sigma |yi - \hat{}yi| \qquad 24$$

To compute R2, we first compute the computation of established distinctions between the true (yi) and predicted (yi) values for each sample. This total is then multiplied by the sum of the squared discrepancies between the true values (yi) and their mean value (yi). The resultant number is the measurement of determination, which suggests how much of the total variance in the supported variable can be described by the standalone variable via the regression connection. A higher R2 value suggests that the independent and dependent variables have a stronger connection.

5.3 Results Comparison

In this study, we used multiple deep-learning models to simulate PM2.5 concentration prediction. In this part, we compared observed PM2.5 data from the past with calculated PM2.5 values using several artificial neural networks, including LSTM, Bi-LSTM, GRU, Bi-GRU, PM-GRU, RM-LSTM, and CLARP models. These models were evaluated with one to seven days of lag. The procedure for forecasting PM2.5 values is depicted in **Figure 4.1**. Each network strives for accurate predictions, which are quantified by a cost function that penalizes the network for any errors. The objective is to minimize the expense function in succession to complete the best outcome. We utilized Mean Squared Error (MSE) as the cost function for all networks in this investigation. During the training phase, the data is split into batches, with the number of samples per batch chosen by experiment and error. Based on the outcomes of this investigation, the batch sizes for all models were 24, 32, 64, and 128.

The cost function is calculated in each training iteration as the mean MSE between the observed and projected PM2.5 concentration samples within the batch. Epochs are the iteration phases for neural networks that entail running the network once to replicate the streamflow time series. The number of neurons or layers in recurrent networks can be varied. In this study, however, we built all recurrent network models with similar architecture to ease model comparison.

The following configurations are used for each of the LSTM, GRU, Bi-LSTM, and Bi-GRU networks:

- **LSTM:** There are four covert layers in the network. 200 LSTM units make up the first layer, followed by 100 in the second layer and 50 in the next two layers. The output of the last layer is connected to a dense layer that only has one output neuron.
- **GRU:** The GRU network is built up of four secret levels similarly. 200 GRU units are present in the first layer, followed by 100 in the second layer and 50 in the next two tiers. The output of the last layer is connected to a dense layer that only has one output neuron.
- **Bi-LSTM:** There are four more hidden layers in the Bi-LSTM network. 200 Bi-LSTM units make up the top layer, followed by 100 in the next layer and 50 in the next two layers. The output of the last layer is connected to a dense layer that only has one output neuron.
- **Bi-GRU:** The Bi-GRU network also has four covert layers. 200 Bi-GRU units are present in the first layer, followed by 100 in the second layer and 50 in the next two

layers. The output of the last layer is connected to a dense layer that only has one output neuron.

- **PM-GRU:** There are no units or hidden levels in the pooling method. It is a strategy for summarizing the output of the GRU layers and reducing data dimensionality. The GRU network is made up of several hidden layers, each of which contains GRU units. The number of hidden layers and units might vary depending on the model's design. There might be four secret levels in the GRU network. The first concealed layer may be made up of 200 GRU troops. The second concealed layer might contain up to 100 GRU units. Each of the third and fourth concealed tiers could house 50 GRU units. Each GRU unit in a hidden layer has its own set of parameters that enable the network to recognize and learn temporal patterns in data.

  After each GRU layer, the pooling mechanism is used to summarize the output and reduce its dimensionality before transferring it to future GRU levels or other elements of the network for further processing. This pooling phase aids in the compression of information and the extraction of essential characteristics from the GRU output.

- **RM-LSTM:** The recurrent mechanism is an important part of LSTM. It enables information to be transmitted and maintained over a sequence time steps. This method allows the network to detect and exploit temporal relationships in data. The LSTM network is made up of several hidden layers, each of which contains LSTM units. The number of hidden layers and units might differ depending on the model design and needs. There might be four hidden layers in the LSTM network. The first concealed layer might have up to 200 LSTM devices. The second concealed layer might include 100 LSTM units. The third and fourth hidden layers might be made up of 50 LSTM units each. Within a hidden layer, each LSTM unit has its own set of parameters, including the input, forget, and output gates, including the cell state. These methods enable the LSTM network to capture and model temporal relationships efficiently.

  Each LSTM layer's output is given back as an input to the next time step, allowing the recurrent connection and information flow over time. This recurrent method allows the model to incorporate long-term relationships and learn patterns in sequential data effectively. The model can capture and utilize the temporal structure of the data by merging the recurrent mechanism with LSTM, making it suited for applications such as time series analysis, sequence prediction, and language modeling.

- **CLARP Model:** In this CLARP model, we investigate a range of recurrent neural network combinations and specialized strategies to optimize learning and modeling capabilities. Among them are LSTM + Bi-LSTM, GRU + Bi-GRU, PM + GRU, and RM + LSTM. We merge LSTM and Bi-LSTM networks in LSTM + Bi-LSTM, using LSTM units in several hidden layers to capture long-term dependencies and Bi-LSTM units to incorporate bidirectional processing. This combination enables the model to capture temporal patterns and dependencies well. To reduce overfitting, a 10% dropout rate is used between the layers in all of these networks, and the ReLU activation function is used for the hidden layers. Similarly, with the GRU + Bi-GRU combination, GRU and Bi-GRU networks are combined. GRU layers containing GRU units record temporal relationships, however, the Bi-GRU layers' bidirectional nature allows for the integration of input from both the past and future contexts. This combination improves comprehension of data context and dependencies. A prediction mechanism (PM) is also combined with the GRU network in the PM + GRU configuration. GRU layers record temporal relationships, whereas the PM is concerned with producing accurate forecasts or predictions. By using the taught patterns and dynamics, this combination provides exact forecasting. We combine a representation mechanism (RM) with an LSTM network in the RM + LSTM combo. The LSTM layers record long-term dependencies, with LSTM units in several hidden layers, while the RM provides meaningful and useful data representations. The model's capacity to acquire effective representations and catch sequential patterns is improved as a result of this combination. Each combination is suited to unique model architectures and requirements, with many hidden layers and adjustable units. These various combinations take advantage of the capabilities of each network and approach to capture and model temporal relationships, include bidirectional context, make accurate predictions, and provide meaningful data representations. By experimenting with different combinations, we improve the model's skills for a wide range of sequential data jobs.

The usage of ReLU provides a significant benefit in terms of a stable derivative for inputs higher than 0, which speeds up the neural network's learning process. Each method runs for 220 epochs and uses Early Stopping (min_delta = 1e-3, patience = 100). In all models, different batch sizes are used, with **Figures 5.1, 5.2 & 5.3** illustrating the major influence of batch size as an influencing component. The Adam optimizer is used, with a learning rate of 0.001 and a learning rate decay of 0.0001. **Table 5.1 and Figures 5.1, 5.2 & 5.3** give a detailed summary of the assessment criteria, comparing seven various prediction approaches for PM2.5 concentration in terms of MAE, RMSE, and R2 values. The RMSE values for 1-day delays were determined to be the lowest among the 65 models. However, under the same conditions and with different batch sizes, the Hybrid CLARP Model beats other models in one-hour predictions, as shown in **Figures 5.1, 5.2 & 5.3**. Furthermore, with a batch size of 64, the CLARP Model demonstrates improved accuracy across different delays, notably with an advantage in the 1-day lag case. **Figures 4.5, 5.1, 5.2 & 5.3** show the MAE, RMSE, and R2 values for the seven models with a batch size of 64, in 1-day and 7-day delays, respectively. These figures indicate the difference between expected and actual PM2.5 concentrations.

Table 5.1. Results of several models with 1 and 7-day lags (bold results indicate the best outcomes).

Model	Batch	1DayMAE	1DayRMSE	1DayR2	7DayMAE	7DayRMSE	7R2
CNN	64	9.960	16.571	0.979	12.868	21.526	0.953
Bi-LSTM	64	9.541	16.609	0.978	11.055	18.122	0.974
RM	64	10.021	17.220	0.977	12.258	19.797	0.965
PM	64	9.842	16.904	0.978	11.899	19.560	0.970
GRU	64	9.102	15.999	0.980	11.916	19.255	0.969

Bi-GRU	64	9.503	16.217	0.980	11.623	19.154	0.972
CLARP	**64**	**6.218**	**11.044**	**0.989**	**8.874**	**16.254**	**0.989**

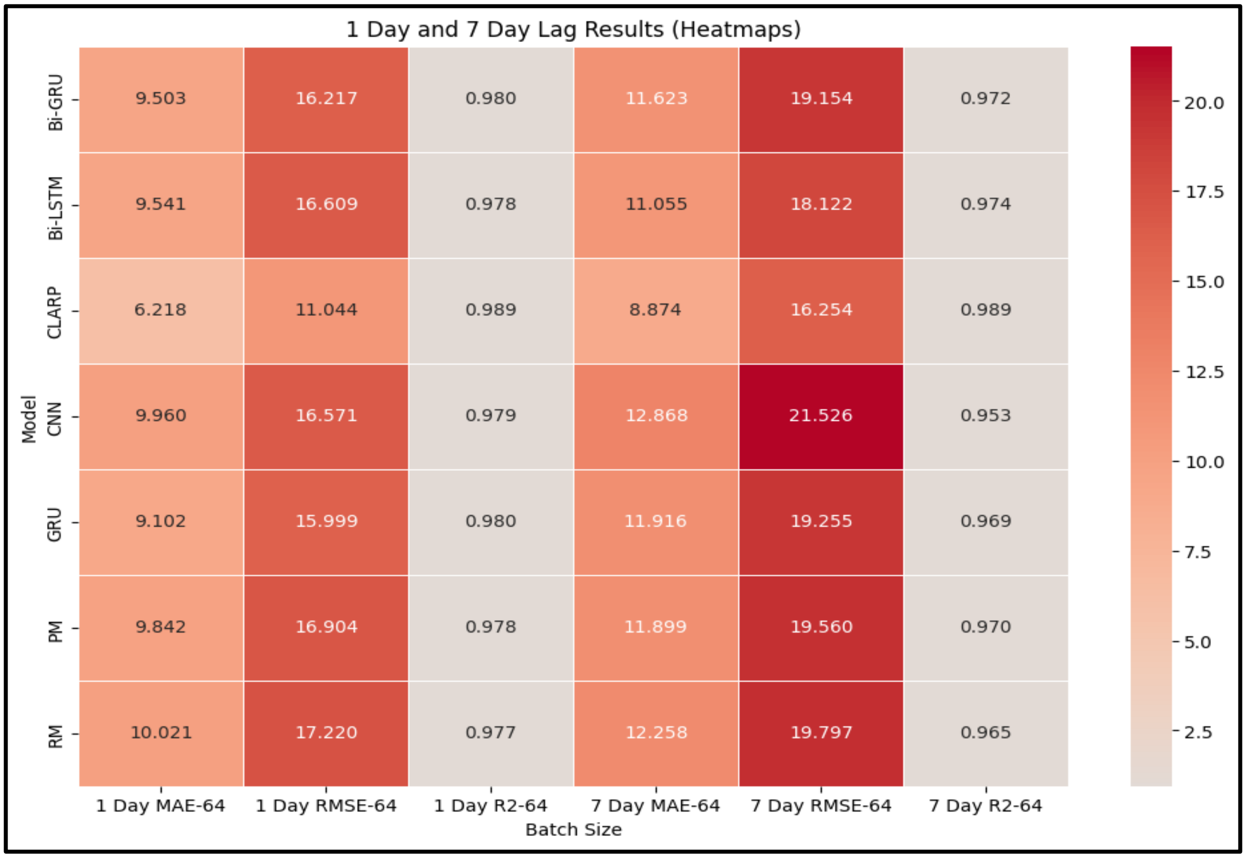

Figure 5.1 Results using Heatmap.

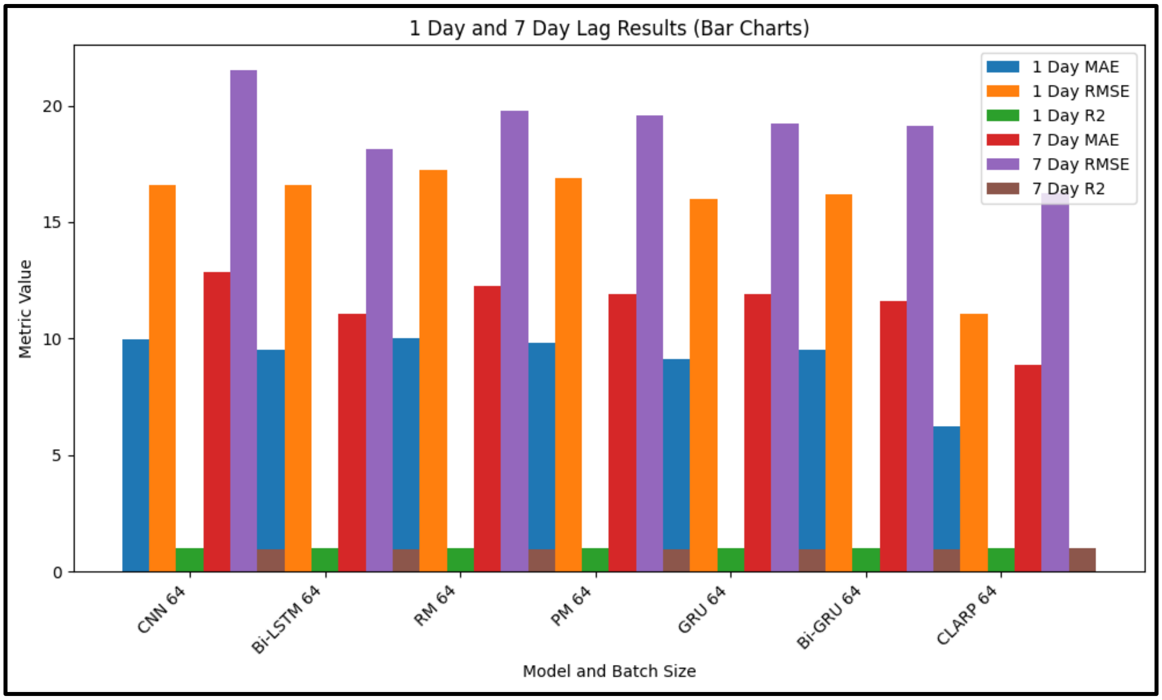

Figure 5.2 Results using Bar Charts.

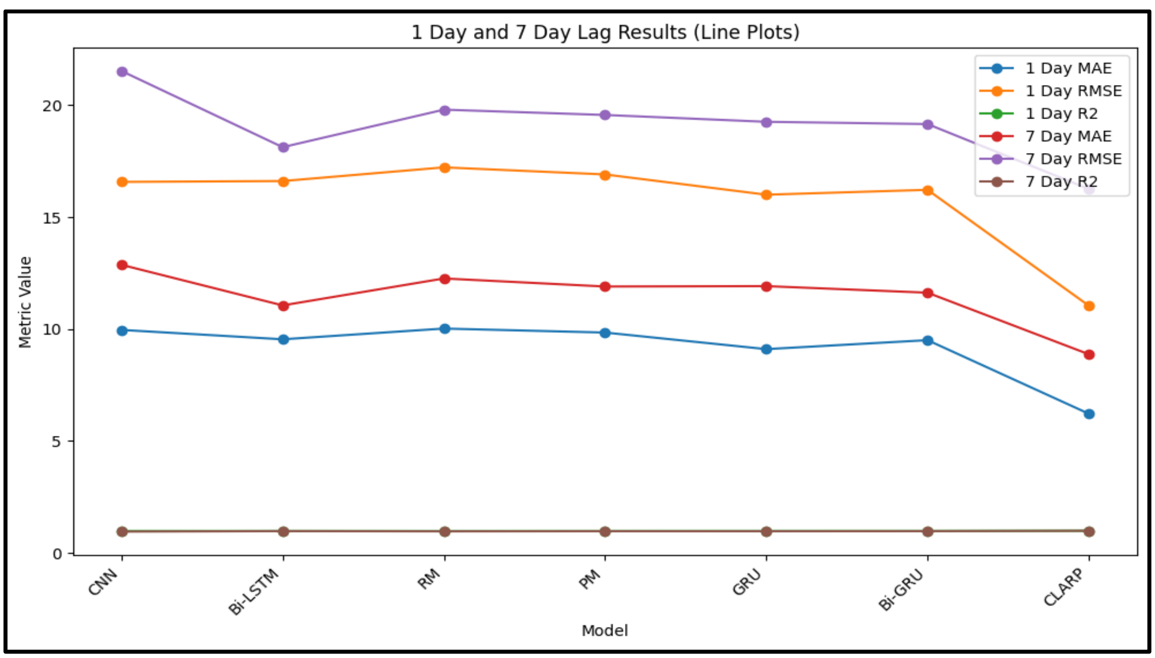

Figure 5.3 Results using Line Plots.

5.4 State-of-the-Art Comparison

In this section, focus is made on the recent developments in this area. A comparison of a well-known model—Hybrid CNN-LSTM and the proposed CLARP Model is carried out. A comparison of MAE, RMSE and R2 results with Hybrid CNN-LSTM [44] and CLARP was carried out and our proposed model has better results. Results are shown in **Table 5.2**.

Table 5.2. Comparison of MAE, RMSE and R2 results with Hybrid CNN-LSTM and CLARP

Model	Batch	1DayMAE	1DayRMSE	1DayR2	7DayMAE	7DayRMSE	7R2
Hybrid CNN-LSTM [43]	64	6.742	12.921	0.989	9.034	16.625	0.979
CLARP	**64**	**6.218**	**11.044**	**0.989**	**8.874**	**16.254**	**0.989**

Consequently, in comparison, the developed Proposed CLARP Model has produced better results, with 99%, R2 with 1 Day and 7 days. MAE for 1 day is 6.218 which is better than Hybrid CNN-LSTM model results as 6.742 and in the case of 7 Days MAE, it gave improved results. For the case of RMSE, 1 Day RMSE is 11.044 which is better as compared to 12.921 of Hybrid CNN-LSTM model and 7 Days RMSE is 16.254 which is also better as compared to 16.625.

5.5 Summary

The study recommends four published models: AC-LSTM [42], LSTM-FC [45], XGBoost [46], CNN-LSTM [43] and hybrid CNN-LSTM [44]. These models were assessed in order to compare their performance to that of the suggested model. To anticipate pollution particles PM2.5, all five models, including the proposed one, were employed. The comparison was carried out using two commonly used measures, MAE and RMSE. After comparing MAE and RMSE, as shown in **Figures 5.1, 5.2 & 5.3**. It is clear that the proposed model not only has the lowest mean absolute error but also the lowest root mean square error of all the tested models. We also did a state-of-the-art comparison as well and we found out that our proposed model has better results in terms of RMSE, MAE, and R2.

Chapter 6: Conclusion and Future Work

6.1 Conclusion

This research presents a hybrid model for forecasting PM2.5 levels of air contamination in Beijing's metropolitan locality that combines LSTM, Bi-LSTM, GRU, Bi-GRU, PM-GRU, and RM-LSTM. To begin, historical data from monitoring sites was analyzed to find relationships. Following experimental comparisons, a characteristic with a better correlation coefficient with PM2.5 was chosen, as well as meteorological data and correlations with other stations. Following that, the suggested hybrid model efficiently extracted spatial characteristics and internal linkages between distinct qualities, while LSTM captured time-related information, resulting in a further precise and durable estimate conclusion. The execution assessment and result association revealed the following significant judgments: the model extracts temporal and spatial information successfully using LSTM, Bi-LSTM, GRU, Bi-GRU, PM-GRU, and RM-LSTM and achieves eminent precision and strength. A 24-hour time window was used as the input values to accommodate for the periodicity in air quality data.

6.2 Limitations of the Model

Some of the limitations are:-

- High computational requirements.
- Computationally expensive as well.
- Training takes long time so tend to be slow.
- Large, labelled dataset is required to train model.

6.3 Future Work

However, our analysis can be improved as it does not take into consideration the amounts of pollutants from outside sources that impact Beijing's air quality. For instance, air contamination from further Chinese cities is passed by the wind and can have an influence on Beijing's air attribute. This feature can be studied, and future work can be done.

References

[1] Urban population (% of total population). https://data.worldbank.org/indicator/SP.URB.TOTL.IN.ZS Accessed 20 Oct 2021.

[2] Department of Economic and Social Affairs: Urban Population Change; 2018. https://www.un.org/development/ desa/en/news/population/2018-revision-of-world-urbanization-prospects.html. Accessed 20 Oct 2021.

[3] Chen, H.; Deng, G.; Liu, Y. Monitoring the Influence of Industrialization and Urbanization on Spatiotemporal Variations of AQI and PM2.5 in Three Provinces, China. Atmosphere 2022, 13, 1377. [Google Scholar] [CrossRef]

[4] Li, G.; Fang, C.; Wang, S.; Sun, S. The Effect of Economic Growth, Urbanization, and Industrialization on Fine Particulate Matter (PM2.5) Concentrations in China. Environ. Sci. Technol. 2016, 50, 11452–11459. [Google Scholar] [CrossRef] [PubMed]

[5] Xu, L.; Dong, T.; Zhang, X. Research on the Impact of Industrialization and Urbanization on Carbon Emission Intensity of Energy Consumption: Evidence from China. Pol. J. Environ. Stud. 2022, 31, 4413–4425. [Google Scholar] [CrossRef]

[6] Kim, D.; Chen, Z.; Zhou, L.-F.; Huang, S.-X. Air pollutants and early origins of respiratory diseases. Chronic Dis. Transl. Med. 2018, 4, 75–94. [Google Scholar] [CrossRef] [PubMed]

[7] Yang, W.; Omaye, S.T. Air pollutants, oxidative stress and human health. Mutat. Res.-Genet. Toxicol. Environ. Mutagen. 2009, 674, 45–54. [Google Scholar] [CrossRef] [PubMed]

[8] Xu, R.; Liu, X.; Wan, H.; Pan, X.; Li, J. A Feature Extraction and Classification Method to Forecast the PM2.5 Variation Trend Using Candlestick and Visual Geometry Group Model. Atmosphere 2021, 12, 570. [Google Scholar] [CrossRef]

[9] Xu, R.; Deng, X.; Wan, H.; Cai, Y.; Pan, X. A DL method to repair atmospheric environmental quality data based on Gaussian diffusion. Journal of Cleaner Production 2021, 308. [Google Scholar] [CrossRef]

[10] Masood, A.; Ahmad, K. A review on emerging artificial intelligence (AI) techniques for air pollution forecasting: Fundamentals, application and performance. J. Clean. Prod. 2021, 322, 129072. [Google Scholar] [CrossRef]

[11] Lu D, Mao W, Xiao W, Zhang L. Non-linear response of pm2.5 pollution to land use change in China. Remote Sens. 2021;13(9):1612.

[12] Bartholomew DJ. Time series analysis forecasting and control. J Oper Res Soc. 1971;22(2):199–201. https://doi.org/ 10.1057/jors.1971.52.

[13] Kumar U, Jain V. Arima forecasting of ambient air pollutants (o 3, no, no 2 and co). Stochastic Environ Res Risk Assess. 2010;24(5):751–60.

[14] Mishra D, Goyal P. Neuro-fuzzy approach to forecasting ozone episodes over the urban area of Delhi, India. Environ Technol Innov. 2016;5:83–94.

[15] Zaidan MA, Dada L, Alghamdi MA, Al-Jeelani H, Lihavainen H, Hyvärinen A, Hussein T. Mutual information input selector and probabilistic machine learning utilisation for air pollution proxies. Appl Sci. 2019;9(20):4475.

[16] Danesh Yazdi M, Kuang Z, Dimakopoulou K, Barratt B, Suel E, Amini H, Lyapustin A, Katsouyanni K, Schwartz J. Pre- dicting fine particulate matter (pm2. 5) in the greater London area: an ensemble approach using machine learning methods. Remote Sens. 2020;12(6):914.

[17] Huang, C.-J.; Kuo, P.-H. A Deep CNN-LSTM Model for Particulate Matter (PM2.5) Forecasting in Smart Cities. Sensors 2018, 18, 2220. [Google Scholar] [CrossRef][Green Version]

[18] Qi, Y.; Li, Q.; Karimian, H.; Liu, D. A hybrid model for spatiotemporal forecasting of PM(2.5) based on graph convolutional neural network and long short-term memory. Sci. Total Environ. 2019, 664, 1–10. [Google Scholar] [CrossRef].

[19] G. K. Kang, J. Z. Gao, S. Chiao, S. Lu and G. Xie, "Air Quality Prediction: Big Data and Machine Learning Approaches", International Journal of Environmental Science and Development, vol. 9, no. 1, pp. 8-16, 2018.

[20] Z. Qin, C. Cen and X. Guo, "Prediction of Air Quality Based on KNNLSTM", Journal of Physics: Conference Series, vol. 1237, no. 4, 2019.

[21] Shi P, Zhang G, Kong F, Chen D, Azorin-Molina C, Guijarro JA. Variability of winter haze over the Beijing-Tianjin-Hebei region tied to wind speed in the lower troposphere and particulate sources. Atmos Res. 2019;215:1–11.

[22] Pohjola MA, Kousa A, Kukkonen J, Härkönen J, Karppinen A, Aarnio P, Koskentalo T. The spatial and temporal variation of measured urban pm 10 and pm 2.5 in the Helsinki metropolitan area. Water Air Soil Pollut Focus. 002;2(5):189–201.

[23] LeCun Y, Bengio Y, Hinton G. DL. Nature. 2015;521(7553):436–44.

[24] Garg S, Jindal H. Evaluation of time series forecasting models for estimation of pm2.5 levels in air. In: 2021 6th international conference for convergence in technology (I2CT). IEEE; 2021, p. 1–8.

[25] Xayasouk T, Lee H, Lee G. Air pollution prediction using long short-term memory (lstm) and deep autoencoder (dae) models. Sustainability. 2020;12(6):2570.

[26] Danesh Yazdi M, Kuang Z, Dimakopoulou K, Barratt B, Suel E, Amini H, Lyapustin A, Katsouyanni K, Schwartz J. Predicting fine particulate matter (pm2. 5) in the greater London area: an ensemble approach using machine learning methods. Remote Sens. 2020;12(6):914.

[27] Salman AG, Heryadi Y, Abdurahman E, Suparta W. Single layer & multi-layer long short-term memory (lstm) model with intermediate variables for weather forecasting. Procedia Comput Sci. 2018;135:89–98.

[28] Rijal N, Gutta RT, Cao T, Lin J, Bo Q, Zhang J. Ensemble of deep neural networks for estimating particulate matter from images. In: 2018 IEEE 3rd international conference on image, vision and computing (ICIVC). IEEE; 2018, p. 733–8.

[29] Zhang L, Li D, Guo Q. DL from spatio-temporal data using orthogonal regularizaion residual cnn for air prediction. IEEE Access. 2020;8:66037–47.

[30] Li J, Jin M, Li H. Exploring spatial influence of remotely sensed pm2.5 concentration using a developed deep convolutional neural network model. Int J Environ Res Public Health. 2019;16(3):454.

[31] Le V-D, Bui T-C, Cha S-K. Spatiotemporal DL model for citywide air pollution interpolation and prediction. In: 2020 IEEE international conference on big data and smart computing (BigComp); 2020, p. 55–62.

[32] Huang C-J, Kuo P-H. A deep cnn-lstm model for particulate matter (pm2.5) forecasting in smart cities. Sensors. 2018;18(7):2220.

[33] Werbos PJ. Generalization of backpropagation with application to a recurrent gas market model. Neural Netw. 1988;1(4):339–56.

[34] Robinson A, Fallside F. The utility driven dynamic error propagation network. Cambridge: University of Cambridge Department of Engineering; 1987.

[35] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997;9(8):1735–80.

[36] Li J, Jin M, Li H. Exploring spatial influence of remotely sensed pm2.5 concentration using a developed deep convolutional neural network model. Int J Environ Res Public Health. 2019;16(3):454.

[37] Norazian M, Al Bakri AMM, Shukri YA, Azam RN. Estimation of missing values for air pollution ata using interpolationtechnique. Simulation. 2006;75:94.

[38] Tao Q, Liu F, Li Y, Sidorov D. Air pollution forecasting using a DL model based on 1d convnets and bidirectional gru. IEEE Access. 2019;7:76690–8.

[39] Freeman BS, Taylor G, Gharabaghi B, Thé J. Forecasting air quality time series using DL. J Air Waste Manage Assoc. 2018;68(8):866–86. https:// doi. org/ 10. 1080/ 10962 247. 2018. 14599 56 (PMID: 29652217).

[40] Zheng Y, Yi X, Li M, Li R, Shan Z, Chang E, Li T. Forecasting fine-grained air quality based on big data. In: Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining; 2015, p. 2267–76.

[41] Zheng Y, Liu F, Hsieh H-P. U-air: when urban air quality inference meets big data. In: Proceedings of the 19th ACM SIGKDD international conference on knowledge discovery and data mining; 2013, p. 1436–44.

[42] Li S, Xie G, Ren J, Guo L, Yang Y, Xu X. Urban pm2.5 concentration prediction via attention-based cnn-lstm. Appl Sci. 2020;10(6):1953.

[43] Wardana I, Gardner JW, Fahmy SA. Optimising DL at the edge for accurate hourly air quality prediction Sensors. 2021;21(4):1064.

[44] Bekkar, A., Hssina, B., Douzi, S. et al. Air-pollution prediction in smart city, deep learning approach. J Big Data 8, 161 (2021). https://doi.org/10.1186/s40537-021-00548-1.

[45] Zhao J, Deng F, Cai Y, Chen J. Long short-term memory-fully connected (lstm-fc) neural network for pm2.5 concentration prediction. Chemosphere. 2019;220:486–92.

[46] Pan B. Application of xgboost algorithm in hourly pm2. 5 concentration prediction. In: IOP conference series: and environmental science, vol. 113. IOP publishing; 2018, p. 012127.