Aggregated Long Short Term Memory Model of Air Quality Prediction for Environmental Monitoring

S. Sowmyayani
Assistant Professor, Department of Computer Science (SF),
St. Mary's College (Autonomous),
(Affiliated to Manonmaniam Sundaranar University, Tirunelveli)
Thoothukudi, Tamilnadu, India
sowmyayani@gmail.com

Abstract:

One of the main environmental problems facing the industrialized world is air pollution, as it has significant consequences on all living beings. There are numerous organizations that have released alerts regarding the extreme air pollution. When all the negative effects of air pollutants are considered, it is essential to create accurate models to predict air pollution levels in order to calculate future concentrations. Even though there have been numerous attempts to simulate pollution levels in the literature, new advancements in deep learning techniques offer better data integration and prediction outcomes that are more accurate. In this study, a detailed research about modelling with deep learning architectures on air pollution data is given. In particular, the Long Short Term Memory (LSTM) is analysed and aggregation of three LSTM network is designed. The experimental results suggested using this architecture in various big air pollution data.

Keywords: pollutants, forecast, deep learning, air quality index

I. INTRODUCTION

Air pollution is one of the most important environmental issues of our time and it must be taken into account while assessing sustainable urban development.. Several studies have shown that air pollution is a major cause of cancer [1–2]. It significantly affects people's quality of life and is closely related to the social and economic development of cities as well as people's health.

In recent years, researchers have focused on air pollution forecasting [3-4]. Most of these studies [5] use simulation techniques or mathematical equations. These traditional approaches are represented by traditional shallow learning algorithms. Dong et al. presented a technique based on Hidden Semi-Markov Models (HSMMs) [6] for forecasting PM2.5

concentration values. Donnelly et al. developed a model based on the Integrated Parametric and Nonparametric Regression (IPNR) technique for air quality forecasts [7].

All these models fail to adequately collect and predict air pollution since it is greatly influenced by weather, transportation and other factors. In the big data era, various sensors and associated data gathering equipment gather big data including PM2.5, NO2, PM10, weather condition data, traffic data, etc. This is a result of the rapid development and application of sensor technology, the Internet of Things (IoTs) and other breakthroughs. As traditional shallow learning models still struggle to handle massive amounts of data, new air quality forecasting methods need data-driven model support [8-9].

This paper overcomes the challenge of big data air quality prediction by designing an aggregated LSTM model with the aggregation of three LSTM network.

The remaining of the paper is organized as follows: Section 2 discusses various literatures in air pollution prediction. Section 3 elaborates the proposed air pollution methodology. Section 4 demonstrates the proposed method with experimental results. Section 5 concludes the chapter.

II. SURVEY

Currently, deep learning [10] is the most popular technology as it can automatically extract and comprehend the essential features of big data. Since 2012, deep learning has growing in natural language understanding, audio processing, image processing and video processing [11–13]. Researchers are increasingly concentrating on deep learning approaches for time series analysis and air quality prediction also [14–15]. Air quality forecasting is a multivariate time series analysis problem and it is a useful study of learning many features based on hybrid deep learning model [16].

Air pollution prediction can be used to forecast changes and trends in air pollutant concentration at different spatiotemporal scales [17]. The ability to forecast the future distribution of air pollution is critical to prevention, control and treatment of air pollution in the future [18]. There are two types of models used to predict air pollution: deterministic theoretical models based on physical and chemical principles and data-driven statistical prediction models.

Deterministic theoretical models utilize real observed data, such as the chemical composition of pollutants, meteorological elements and emission source characteristics, to establish physical and chemical reaction mechanisms of pollutant emission, diffusion, transmission, secondary reaction and removal for use in pollution simulation and prediction [19-22]. There are still certain limitations on how these models can be used.

Data-driven statistical prediction methods match the quantitative relationship between historical pollutant data and external parameters, such as meteorological features and spatiotemporal characteristics, to estimate the distribution of future air pollution [23-24]. The accuracy of forecasts is limited by the relationship between targets and variables, which is difficult for traditional machine learning techniques [25].

As deep learning algorithms have developed and grown, several researchers have begun to use deep neural network models, such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM), for air pollution prediction. Numerous studies have also shown how resilient LSTM is when handling time series issues, demonstrating that it outperforms general shallow networks [26–27]. A few of the LSTM-based models that have been developed recently are Graph Convolutional LSTM (GC-LSTM), Temporal Sliding LSTME (TS-LSTM), Convolutional LSTME (C-LSTM) and LSTM Extended (LSTME).

Most LSTM-based models are used to estimate short-term air pollution concentrations and they consider basic temporal dependencies. They usually have limited accuracy in long-term prediction because they are unable to maintain high temporal correlations. In LSTME and C-LSTME models, suitable time delays were used to generate several models to provide long-term prediction, but they did not perform well. While TS-LSTME, were effective in identifying the implicit temporal linkages among the data, they failed to take into account the geographical dependencies of air pollutants.

The C-LSTME model was utilized to simulate the spatial connection of air pollution at the monitoring stations using Convolutional Neural Networks (CNNs). However, as air pollution spreads in non-Euclidean space, conventional CNN only functions in normal Euclidean space and is not applicable to network topologies [23, 27]. In order to overcome this problem, the authors developed the GC-LSTM model, which combines spectral graph convolution, the Chebyshev approximation and superficial LSTM layers to demonstrate that network-based predictions are more valuable and accurate. The Chebyshev approximation reduces complexity and the possibility of lower accuracy.

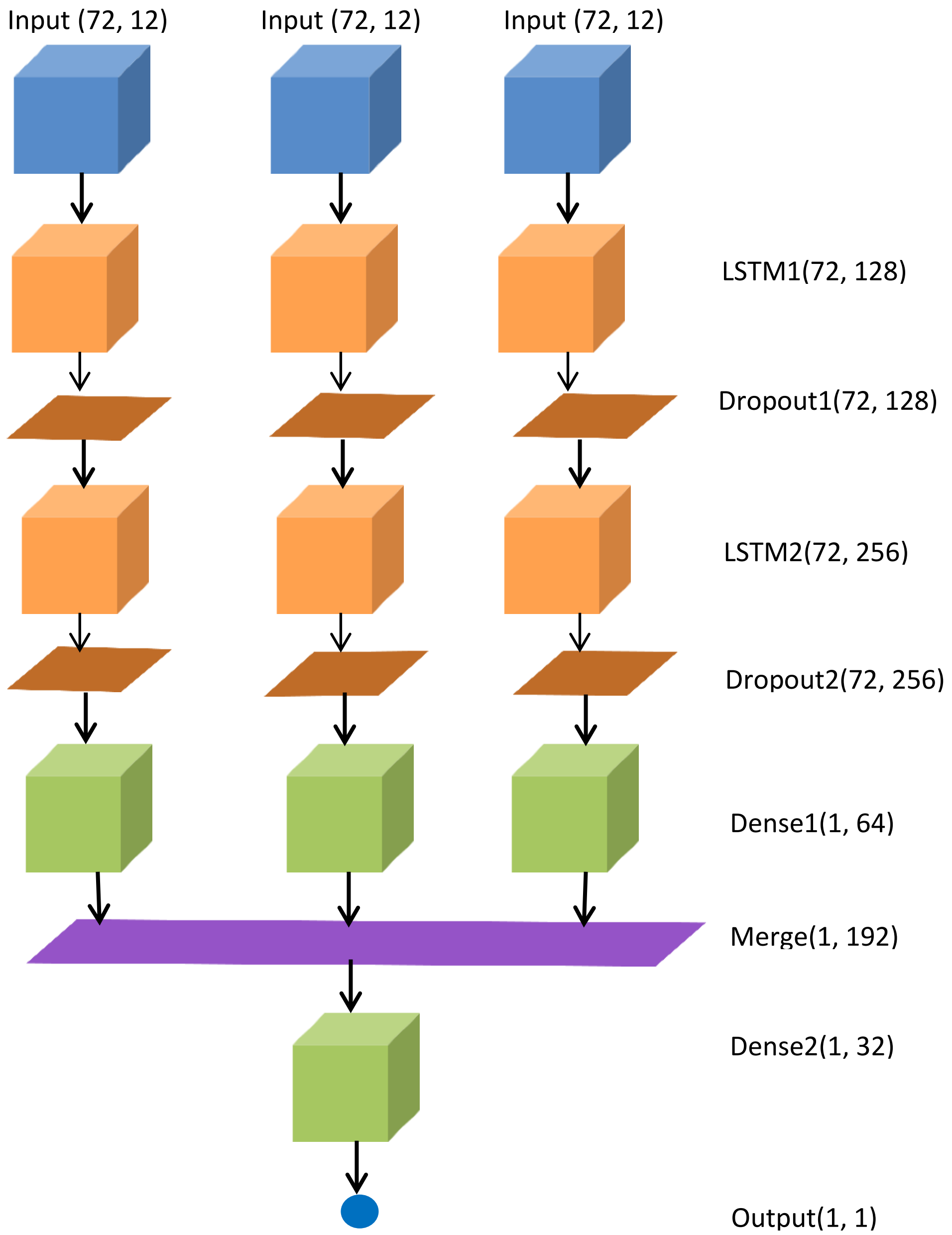
Long-term dependencies were too big for the standard LSTM layers used in temporal modeling. Furthermore, most studies only forecast one air pollutant, making it impossible to assess how well the recommended models predict other pollutants [16, 28–29]. The aforementioned constraints were addressed in the creation of the GT-LSTM model [30], which predicts various air pollutants by merging graph convolution networks with self-loops and temporal sliding LSTM networks.

III. AGGREGATED LONG SHORT TERM MEMORY MODEL

The aggregated LSTM combines all of the predictive features for the final predicted value. All forecast features have identical weights. However, the causes of air pollution need to vary at different times or in different locations of the station. Thus, the idea of aggregation serves as the foundation for this study. For every feature produced by three separate models, different weights are given. Therefore, the purpose of this paper is to optimize our air pollutant forecast model through the use of aggregation model.

Three prediction features are generated using aggregation learning for different kinds of stations. Predicted functional data from the fully connected neural network layer is used to construct the data. The reverse propagation adjusts weights after each batch and the system trains data instantaneously [31-32]. In the end, the best outcomes are possible to achieve. Figure 4.5 depicts the suggested design of the aggregated LSTM. It comprises three subnetwork layers and three input layers. The output layers added together yield a single forecast.

Each time point has data with 12 dimensions. Hence, in the input layer of the proposed model, the size of the data is 12 dimensions and 72 hours (three days are used for prediction). The hidden layer will receive the hourly 12-dimension data supplied from the input layer. Three gates, each coupled by four Fully Connected (FC) layers, are present inside the hidden layer. The output of the LSTM cells at that particular moment in time will then be collected, together with a matrix of cell states. This technique has preserved and transferred the memory data to the LSTM cells at the final time point.



In the forget gate of LSTM, the following calculation is made.

$$f_t = \sigma(W_f[x_t, h_{t-1}] + b_f) \tag{1}$$

Where the size of two input matrices x_t and h_{t-1} are 17 x 1 and 128 x 1 respectively. b_f is the forget gate bias. The sigmoid function is used to convert to the sequence 0-1.

The input gate is the same as the forget gate, where input matrices x_t, $h_{t-1}$and $x_t + h_{t-1}$ are connected and entered two different FC layers. The first FC layer will determine how much information will be added to the state of the cell. Then the matrix

$x_t + h_{t-1}$ is multiplied with the weight matrix W_l of the forget gate and the input gate bias b_i is added to get i_t. Then we reduce the limit of i_t to the sequence of 0–1 by the Sigmoid function. The second FC layer controls how much information will be added to the cell state. The matrix $x_t + h_{t-1}$ is multiplied with the weight matrix W_c of the input gate and the input gate bias b_c is added. Then we reduce the limit to the sequence from -1 to1 by the $tanh$ function. The structure is used to save the state of the cell that needs to be stored for a long time. The output of the forget gate is multiplied by f_t and c_{t-1}.

When the value of f_t is closer to 0, it means that the state of the neuron is to be completely forgotten, and the value closer to 1 means to completely remember the state of the neuron. And then we use the output i_t of the input gate to choose the amount of information to be memorized to multiply the information that needs to be memorized to get the information we need to memorize.

In the output gate, there are three inputs and the current cell state c_t, which will connect x_t and h_{t-1} matrices to obtain $x_t + h_{t-1}$. And by multipling the matrix with the weight matrix W_o of the output gate and adding the output gate b_o, o_t is obtained which is then converted to a sequence from 0 to 1 by sigmoid function and limit c_t to the sequence from 1 to -1 by tanh function. We multiply c_t with the amount of information of the cell status to become the output in the output layer and in the LSTM cells at the next time.

In order to avoid over-fitting, we have added two dropout layers between our LSTM layers. In the FC layer, the 72 neurons of the second LSTM network layer are connected to all 64 neurons to produce the weight and activation function for the next fusion layer. In the Merge layer, the features from the three sub-neural networks are connected and the output is given to next FC layer. This method collects all predicted features and outputs for the next layer and gives each predicted feature a different weight.

Finally, the final FC layer is connected with the prediction feature of the merge layer. The final layer will give different weights to its hidden layer through back propagation. The data will be merged to get the final output. By giving different weights to the predicted features, the final layer can represent the correlation of local station data for each predicted feature.

IV. Experimental Results

The experimental evaluation of the proposed model is performed by using Air Quality Data in India (2015-2020), which is openly available on Kaggle [33]. The dataset contains the data starting from January 1, 2015 to February 28, 2021. The dataset contains Air Quality Index (AQI) and air quality data in terms of daily and hourly levels of several stations across multiple cities in India.

It encompasses the concentrations of different pollutants for each hour of the day along with the data of the environmental conditions. Pollutants include PM2.5, PM10, SO2, NO2, CO, O3 etc. The dataset consists of 5 sub datasets: city_day, station_day, city_hour, station_hour and stations. We concentrated on station_day dataset. It consists of the attributes listed in Table 1.

Table 1 Attributes of Station_day Dataset

S. No.	**Attribute**	**Description**
1.	StationId	ID given to each station of a city
2.	Date	Date on which the concentrations are obtained
3.	PM2.5	particulate matter of a diameter equal or smaller than 10 μm
4.	PM10	particulate matter of a diameter equal or smaller than 2.5 μm
5.	NO	Nitrogon Oxide
6.	NO2	Nitrogen dioxide
7.	NOx	Total Nitrogen Oxides
8.	NH3	Atmospheric Ammonia
9.	CO	Carbon Monoxide True hourly averaged concentration CO in mg/m^3 (reference analyzer)
10.	SO2	Sulphur dioxide
11.	O3	Ozone
12.	Benzene	Benzene
13.	Toluene	Toluene
14.	Xylene	Xylene

15.	AQI	Air Quality Index
16.	AQI_Bucket	AQI classification

In the above table, Station Id, Date, AQI and AQI_Bucket are excluded for prediction. The remaining 12 features are predicted individually using the proposed model. The statistical information about the dataset is shown in Fig. 5.1.

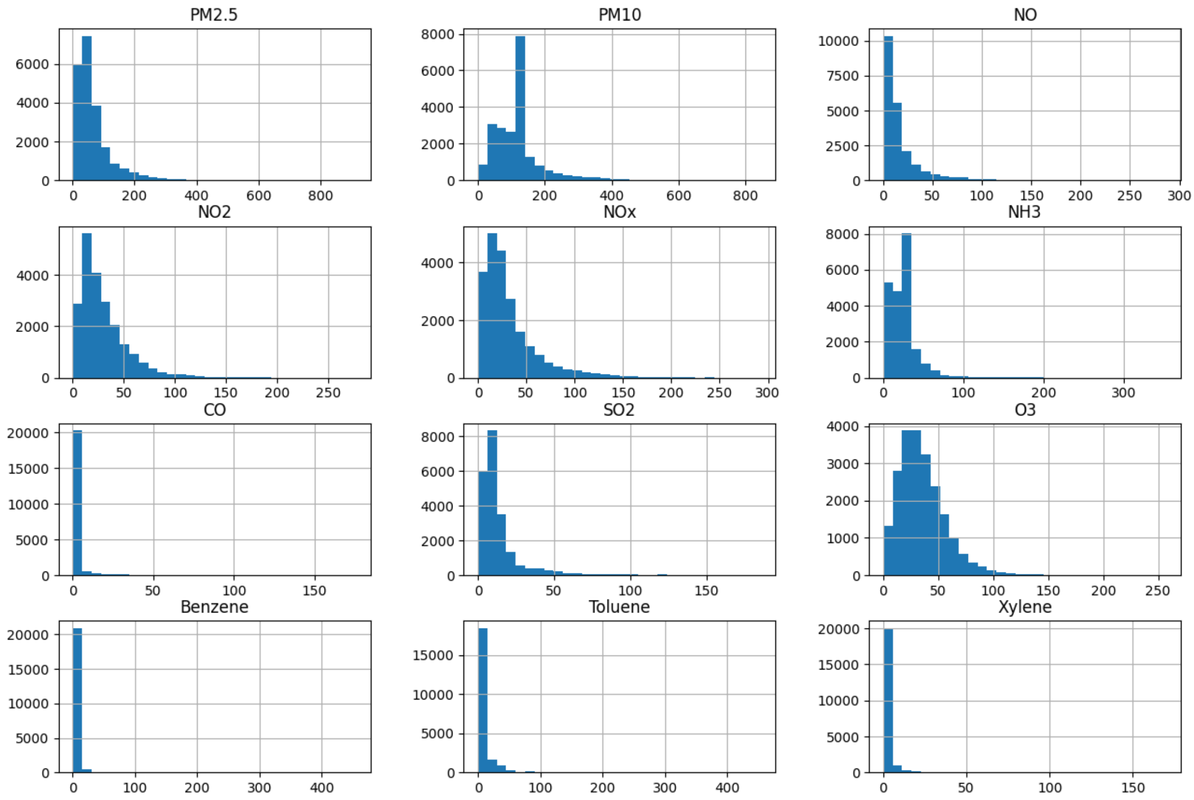


Fig. 5.1 Statistical Information about Attributes

All pollutants values are displayed in feature graph. Finally, the Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Coefficient of determination (R^2) values were selected to evaluate the effectiveness of the proposed method. We assume that $\boldsymbol{O_i}$ and $\boldsymbol{P_i}$ are the observed and predicted values, respectively. $\boldsymbol{\bar{O}_i}$Is the average value of n observed samples. These indicators can be formulated as follows.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(P_i - Q_i)^2} \tag{2}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|P_i - Q_i| \quad (3)$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|P_i - Q_i|}{Q_i} \quad (4)$$

$$R^2 = 1 - \frac{\sum_{i=}^{n}(P_i - \bar{Q}_i)^2}{\sum_{i=}^{n}(O_i - \bar{Q}_i)^2} \quad (5)$$

The following Table 2 provides various attributes of the proposed model.

Table 2 Model Settings

Parameter	Value
Epochs	30
Batch Size	1
Learning Rate	0.001
Dropout Layer (%)	0.1
Activation Function	ReLU
Loss Function	MAE
Optimizer	Adam

The proposed model works on few epochs only with batch size is set to 1. Only 10% dropout and Adam optimizer are used. The learning rate is set to 10^{-3}. The prediction results obtained by BiLSTM model for each pollutant are shown in Table 3.

Table 3 Results Obtained by Proposed Model for each Pollutant

Pollutant/ Measure	RMSE	MAE	MAPE	R2
PM2.5	1.08	6.82	0.25	0.72
PM10	0.37	10.36	0.18	0.68
NO	1.18	1.32	0.39	0.28
NO2	1.75	2.6	0.47	0.31
NOx	3.55	3.76	0.34	0.26
NH3	1.55	1.98	0.14	0.68
CO	0.04	0.09	0.33	0.28
SO2	2.81	3.99	0.2	0.35
O3	6.32	3.32	0.14	0.18

Benzene	0.22	0.26	3.15	0.28
Toluene	0.68	1.01	0.52	0.33
Xylene	8.78	9.9	1.02	0.35

The error values such as RMSE, MAE and MAPE should be as less as it could be. But the R2 value should be nearer to 1 as it ranges from 0 – 1. In the above table, the RMSE values are lesser than 6.32 (obtained for O3). All the MAE values are lesser than 5 except for PM 2.5 and PM 10. The MAPE values are very encouraging as it is very lesser than 1 except for Benzene pollutant. But the R2 values did not reach even 0.7. It indicates the correlation between the prediction and the original data are different from each other.

The importance of aggregation is studied by comparing the proposed model with single LSTM and BiLSTM. Table 4 compares only PM2.5 and PM10 concentration on different models.

Table 4 Comparison of Different Type of Models on PM2.5 and PM10

Model	Pollutant	RMSE	MAE	MAPE	R2
LSTM	**PM2.5**	1.81	7.17	0.43	0.59
	PM10	3.87	10.66	0.28	0.52
BiLSTM	**PM2.5**	2.8	1.75	0.34	0.43
	PM10	4.26	3.42	0.53	0.49
ALSTM	**PM2.5**	1.08	6.82	0.25	0.72
	PM10	0.37	10.36	0.18	0.68

From Table 4, it is very clear that the proposed aggregated LSTM model outperforms other models in all the metrics. Another study is carried out for aggregation type: Linear (carrying different weights) and Uniform (carrying same weight). Table 5 shows the results obtained by linear and uniform aggregation.

Table 5 Comparison of Aggregation Type on PM2.5 and PM10

Aggregation Type	Pollutant	RMSE	MAE	MAPE	R2

Uniform	**PM2.5**	2.91	9.11	0.3	0.57
	PM10	4.69	10.86	0.18	0.43
Linear	**PM2.5**	1.08	6.82	0.25	0.72
	PM10	0.37	10.36	0.18	0.68

The RMSE value of linear aggregation (0.37) is very lesser for PM10 than uniform aggregation (4.69). The R2 value is greater than 0.6 in linear aggregation. From the results, it is substantially proved the linear aggregation is better than uniform aggregation.

5. Conclusion

The air and water pollution has a significant impact on human life. The technology development should prevent human life in all the means. One such model is designed for air quality prediction using LSTM network. An aggregated LSTM model is proposed with linear aggregation. The proposed ALSTM model is experimented on all pollutants of Indian dataset individually. As PM2.5 and PM10 has a greater impact on air pollution, these two pollutants are analysed with other models such as LSTM and BiLSTM. The linear aggregation is also compared with uniform aggregation. From all the analyses, it is proved that ALSTM with linear aggregation is better for air quality prediction. In future, it can be further extended to include more than one pollutant for prediction. Also, the huge variety of datasets can also be tested considering the feature correlation.

References:

[1] Zhang Y., Bouquet, M., Mallet, V., Seigneur, C., and Baklanov, A., “Real-time air quality forecasting, Part I: History, techniques, and current status,“ Atmospheric Environment, vol. 60, pp. 632-655, 2012.

[2] Zhang, Y., Bouquet, M., Mallet, V., Seigneur, C., and Baklanov, A., “Real-time air quality forecasting, Part II: State of the science, current research needs, and future prospects,” Atmospheric Environment, vol. 60, pp. 656-676, 2012.

[3] Vardoulakis S, Fisher BE A, Pericleous K, et al, “Modelling air quality in street canyons: a review,” Atmospheric Environment, vol. 37, no. 2, pp. 155-182, 2003.

[4] M. Dong, D. Yang, Y. Kuang, D. He, S. Erdal, and D. Kenski, “PM2.5 concentration prediction using hidden semi-Markov model-based times series data mining,” Expert Systems with Applications, vol. 36, no. 5, pp. 9046-9055, 2009

[5] Donnelly, A., Misstear, B., and Broderick, B, "Real-Time air quality forecasting using integrated parametric and nonparametric regression techniques," Atmospheric Environment , vol. 103, pp. 53-65, 2015.

[6] Y. Zheng, F. Liu, and H.-P. Hsieh, "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 (KDD), pp. 1436-1444, 2013.

[7] H.-P. Hsieh, S.-D. Lin, and Y. Zheng, "Inferring air quality for station location recommendation based on urban big data," in Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp. 437-446, 2015.

[8] Schmidhuber J, "Deep learning in neural networks: An overview. Neural networks, " vol. 61, pp. 85-117, 2015.

[9] Krizhevsky A, Sutskever I, and Hinton G E, "Imagenet classification with deep convolutional neural networks," in Proceedings of Advances in Neural Information Processing Systems (NIPS), pp. 1097- 1105, 2012.

[10] Karpathy A, and Li F F, "Deep visual-semantic alignments for generating image descriptions," in Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3128-3137, 2015.

[11] Sowmyayani. S, 2022, Machine Learning. In Kumar, A., Sagar, S., Kumar, T.G. and Kumar, K.S. eds., 2022. Prediction and Analysis for Knowledge Representation and Machine Learning. CRC Press.

[12] Gamboa J C B, "Deep learning for time-series analysis". arXiv preprint arXiv:1701.01887, 2017

[13] Qi Z, Wang T, Song G, et al, "Deep air learning: interpolation, prediction, and feature analysis of fine-grained air quality". IEEE Transactions on Knowledge and Data Engineering, vol. 30, no. 12, pp. 2285- 2297, 2018

[14] Zhuang, Dennis EH, Gary CL Li, and Andrew KC Wong, "Discovery of temporal associations in multivariate time series," IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 12, pp. 2969- 2982, 2014.

[15] Zhang, K., J. Thé, G. Xie, and H. Yu. 2020a. "Multi-step Ahead Forecasting of Regional Air Quality Using Spatial-temporal Deep Neural Networks: A Case Study of Huaihai Economic Zone." Journal of Cleaner Production 277: 123231.

[16] Wang, S., Y. Li, J. Zhang, Q. Meng, L. Meng, and F. Gao, 2020. "PM2.5-GNN: A Domain Knowledge Enhanced Graph Neural Network For PM2.5 Forecasting." arXiv:2002.12898.

[17] Jeong, J. I., and R. J. Park. 2018. "Efficacy of Dust Aerosol Forecasts for East Asia Using the Adjoint of GEOS-Chem with Ground-based Observations." Environmental Pollution 234: 885–893.

[18] Yang, X., Q. Wu, R. Zhao, H. Cheng, H. He, Q. Ma, L. Wang, and H. Luo. 2019. "New Method for Evaluating Winter Air Quality: PM2.5 Assessment Using Community Multi-Scale Air Quality Modeling (CMAQ) in Xi'an." Atmospheric Environment 211: 18–28

[19] Wang, Q., Q. Zeng, J. Tao, L. Sun, L. Zhang, T. Gu, Z. Wang, and L. Chen. 2019a. "Estimating PM2.5 Concentrations Based on MODIS AOD and NAQPMS Data over Beijing–Tianjin–Hebei." Sensors 19 (5): 1207.

[20] Wang, P., X. Qiao, and H. Zhang. 2020. "Modeling PM2.5 And O3 with Aerosol Feedbacks Using WRF/Chem over the Sichuan Basin, Southwestern China." Chemosphere 254: 126735.

[21] Gariazzo, C., G. Carlino, C. Silibello, M. Renzi, S. Finardi, N. Pepe, P. Radice, et al. 2020. "A Multi-city Air Pollution Population Exposure Study: Combined Use of Chemical-transport and random-Forest Models with Dynamic Population Data." Science of the Total Environment 724: 138102.

[22] Hu, J., B. Ostro, H. Zhang, Q. Ying, and M. J. Kleeman. 2019. "Using Chemical Transport Model Predictions to Improve Exposure Assessment of PM2.5 Constituents." Environmental Science 6(8): 456–461.

[23] Zhang, Y., T. Cheng, Y. Ren, and K. Xie. 2020b. "A Novel Residual Graph Convolution Deep Learning Model for Short-term Network-based Traffic Forecasting." International Journal of Geographical Information Science 34 (5): 969–995.

[24] Hochreiter, S., and J. Schmidhuber. 1997. "Long Short-Term Memory." Neural Computation 9 (8): 1735–1780.

[25] Ma, J., Y. Ding, J. C. P. Cheng, F. Jiang, V. J. L. Gan, and Z. Xu. 2020. "A Lag-FLSTM Deep Learning Network Based on Bayesian Optimization for Multi-sequential-variant PM2.5 Prediction." Sustainable Cities and Society 60: 102237.

[26] Mao, W., W. Wang, L. Jiao, S. Zhao, and A. Liu. 2021. "Modeling Air Quality Prediction Using a Deep Learning Approach: Method Optimization and Evaluation." Sustainable Cities and Society 65: 102567.

[27] Qi, Y., Q. Li, H. Karimian, and D. Liu. 2019. "A Hybrid Model for Spatiotemporal Forecasting of PM2.5 Based on Graph Convolutional Neural Network and Long Short-term Memory." Science of the Total Environment 664: 1–10.

[28] Wen, C., S. Liu, X. Yao, L. Peng, X. Li, Y. Hu, and T. Chi. 2019. "A Novel Spatiotemporal Convolutional Long Short-term Neural Network for Air Pollution Prediction." Science of the Total Environment 654: 1091–1099.

[29] Zhao, J., F. Deng, Y. Cai, and J. Chen. 2019. "Long Short-term Memory - Fully Connected (LSTM-FC) Neural Network for PM2.5." Concentration Prediction. Chemosphere 220: 486–492.

[30] Wenjing Mao, Limin Jiao, Weilin Wang, Jianlong Wang, Xueli Tong & Suli Zhao (2021) A hybrid integrated deep learning model for predicting various air pollutants, GIScience & Remote Sensing, 58:8, 1395-1412, DOI: 10.1080/15481603.2021.1988429

[31] Pineda FJ (1987) Generalization of back-propagation to recurrent neural networks. Phys Rev Lett 59:2229–2232

[32] Rumelhart, D.E., Hinton, G.E. and Williams, R.J., 1985. Learning internal representations by error propagation.

[33] https://www.kaggle.com/datasets/rohanrao/air-quality-data-in-india