Introduction:

Background:

Kathmandu with a population of more than one million is growing rapidly [1]. Growth of a city increases industrialization which generates more factories and more employment, thus attracting more people to migrate towards cities in search of opportunities but they have to trade off quality air especially in cities like Kathmandu where air pollution is massive. Degraded Air quality has shown health problems in people living in city areas which consequently has caused premature deaths and reduced average lifespan [2,3]. In the last century the human population has increased by four times [4], causing migration of people from one place to another. It is

seen that mostly people in underdeveloped countries tend to migrate toward metropolitan areas for employment possibilities [2,3], Since these metropolitan cities in the less developed regions are poorly planned the increase in population will result in poor air quality.
Management of industrial waste is not alone enough to tackle air pollution [5], short term and long term forecasting using models is also an important strategy. Recently machine learning models have gained attention for their application in air-quality forecasting as they are efficient in forecasting temporal sequences [6,7,8]. Simulation based and data driven methods are the two approaches for forecasting air quality, simulation based models use physical and chemical models to simulate emission, transport and chemical transformation of air pollution whereas data driven methods use statistical or machine learning methods to forecast air quality [9,10]. Both statistical model and simulation model can be used for forecasting future hazardous air quality index in order to alarm the general public and to plan a strategy. One disadvantage of simulation based models is, it needs data for all the important parameters. For example the parametrization of aerosol emissions is restricted due to lack of data [11]. Forecasting using machine learning models is done by predicting patterns between dependent and independent variables in temporal sequence [12,13,14,15,16].tant parameters.
Most deep learning models have long term dependency problems. When the desired output depends on the inputs presented at an earlier time, those types of problems are termed as long term dependency problems. Recurrent neural networks(RNNs) are commonly used for those problems involving long term dependency and suited for modeling temporal sequences [17,18,19,20]. Long short term memory networks (LSTMs) models are a special kind of RNNs model to address the issue of long term dependency problems [21,22].
Covid 19, an Infectious disease which became a global pandemic [23,24,25,26] has forced many countries to shut their borders and implement a full or partial lockdown which reduced anthropogenic activity during the lock-down period. There have been many studies regarding the impact of covid-19 on environment and air pollution. Due to less traffic on the street some cases showed reduced air pollution while others where industrial activity is dominant, not much of a difference is seen on air pollution despite the lockdown [27,28]. The other reason for insignificant reduction of air pollution is due to meteorological conditions [27,28].

Effect of air Pollution

Data-driven machine learning technologies allow researchers to examine the influence of different air contaminants on health outcomes at the same time. There is growing evidence that early-life exposure to ambient air pollution affects neurodevelopment in children [29]. A study that showed that air pollution is a potential risk factor for obesity with higher body mass index in adults that warrants further investigation about other health effects[30]. Assessing school children's exposure to air pollution during the daily commute in a systematic review highlighted studies with schoolchildren’s exposure during commutes that are linked with adverse cognitive outcomes and severe wheeze in asthmatic children[31]. Furthermore, Ambient air pollution is associated with reduction in lung functionality and other respiratory conditions among children [32]. Apart from health, air pollution has taken a toll on various other sectors, which includes agriculture. Industrial air pollution has a drastic effect in agricultural production as shown by a

study in China with lower marginal products and further alters the relationships of labor-capital and other factors [33]. Finally, air pollution has a drastic effect on development and the economy. A study on the relationship between air pollution and stock returns further showed that industrial air pollution significantly reduces the technical efficiency of agricultural production [34].

Objective:

We employ unique deep learning algorithms for long-term air-quality prediction in Kathmandu, Nepal, in this work. Our deep learning methods consist of Long Short Term Memory (LSTMs) networks. A multivariate time series approach has been used using 8 predictive variables covering a timespan from 30th dem 2019 to 23rd dem 2021. Hence the main objectives of this paper are (i) to predict and simulate Pm2.5 concentration through the application of LSTM model and (ii) detect the factors which have the maximum association with Pm2.5 concentration during the lockdown phase over Kathmandu.

LSTM network models:

RNNs are an artificial neural network which gained popularity in recent years, the earliest example of simple RNNs used prominently is The Elman RNN [16,35] trained by back propagation through-time (BPTT) algorithm [18]. Studies have also shown that in knowledge representation tasks RNNs perform better than feed-forward networks [36,37,38]. Backpropagation through time (BPTT) [18] features error backpropagation which uses the idea of training RNNs using gradient descent in a way similar to feedforward neural networks. The major difference is that the error is back-propagated for a deeper network architecture that features states defined by time; however, BPTT experiences the problems of vanishing and exploding gradients in case of learning tasks which involve long-term dependencies or deep network architectures. [39].Long Short Term Memory(LSTM) network [19] was developed which was capable of overcoming the fundamental problems of deep learning [16].LSTM networks addressed the problem by using memory cells and gates for temporal sequences to recall long-term dependence considerably better. The memory cells are trained in a supervised fashion using an adaptation of the BPTT algorithm that considers the respective gates [19]. More recently, Adam optimiser which features adaptive learning rate has been prominent in training LSTM models via BPTT [40].

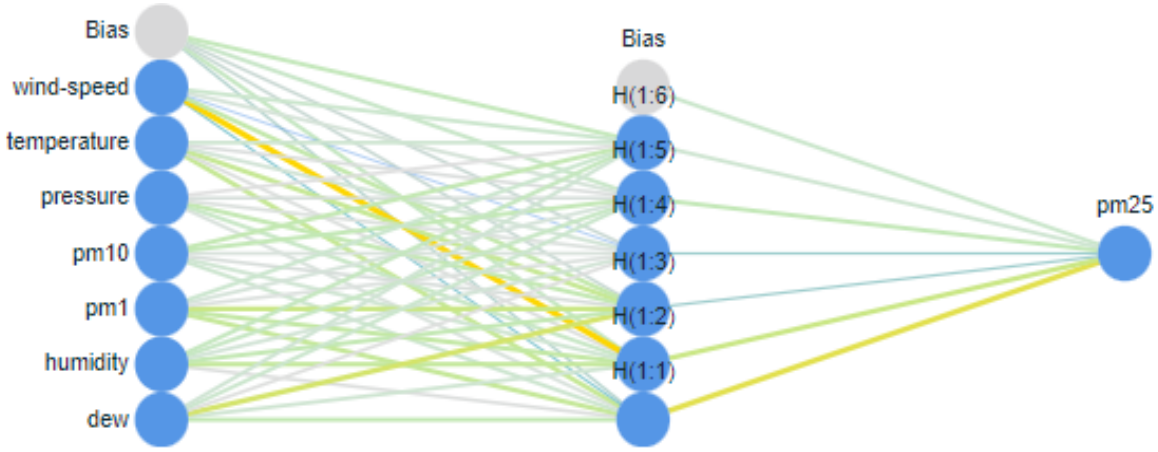
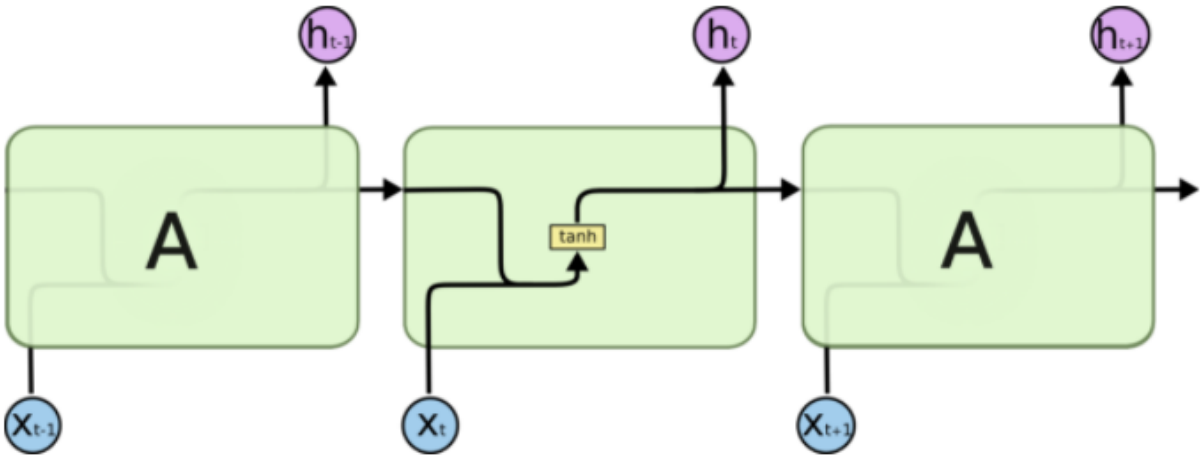


Fig: A neural network model showing dependent and independent variables:

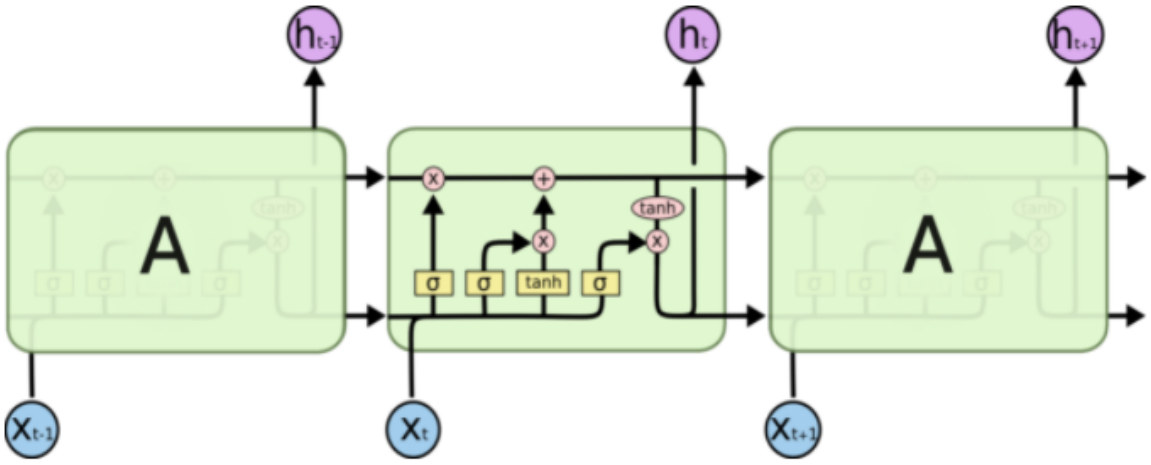
A sequence of repeating neural network modules make up all recurrent neural networks. In conventional RNNs, this repeating module will have a basic structure, such as a single tanh layer.

LSTMs have a chain-like structure as well, but the repeating module is different. Instead of a single neural network layer, there are four, each of which interacts in a unique way. Each line in the figure above transmits a full vector from one node's output to the inputs of others. The pink circles denote pointwise operations, such as vector addition, and the yellow boxes denote learnt neural network layers. Concatenation occurs when lines merge, whereas forking occurs when a line's content is replicated and the copies are sent to various locations.



The repeating module in a standard RNN contains a single layer.

(*Understanding LSTM Networks -- Colah's Blog*, 2015)

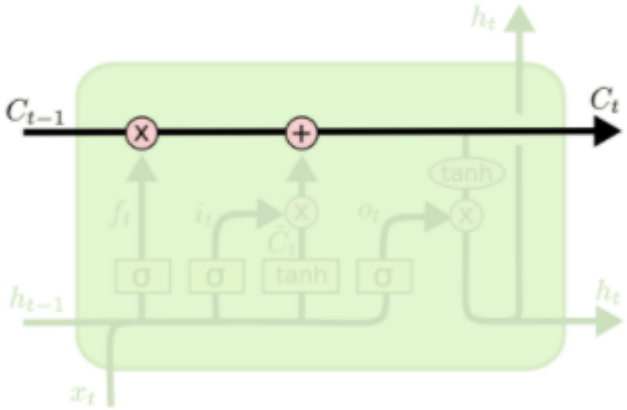


The repeating module in an LSTM contains four interacting layers.

(*Understanding LSTM Networks -- Colah's Blog*, 2015)

The Core Idea Behind LSTMs:

The cell state, the horizontal line going through the top of the diagram, is the key to LSTMs. The state of the cell is similar to that of a conveyor belt. With only a few tiny linear interactions, it flows straight down the entire chain. It's incredibly easy for data to simply



travel along it unaltered.

(*Understanding LSTM Networks -- Colah's Blog*, 2015)

The LSTM can delete or add information to the cell state, which is carefully controlled by structures called gates. Gates are a mechanism to selectively allow information to pass

through. A sigmoid neural net layer plus a pointwise multiplication operation make them up.

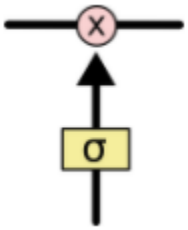


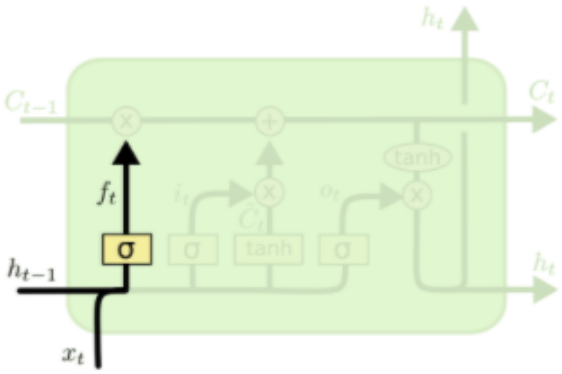
fig: pointwise multiplication operation

(*Understanding LSTM Networks -- Colah's Blog*, 2015)

The sigmoid layer produces integers ranging from zero to one, indicating how much of each component should be allowed to pass. A value of zero indicates that "nothing should be allowed through," whereas a value of one indicates that "everything should be allowed through!" Three of these gates are present in an LSTM to protect and govern the cell state.

Step-by-Step LSTM Walk Through

The first stage in our LSTM is to decide which information from the cell state will be discarded. The "forget gate layer," a sigmoid layer, makes this judgment. It examines ht-1 and xt and returns a value between 0 and 1 for each number in the Ct-1 cell state. A1 indicates "totally keep this," while a 0 indicates "entirely discard this."

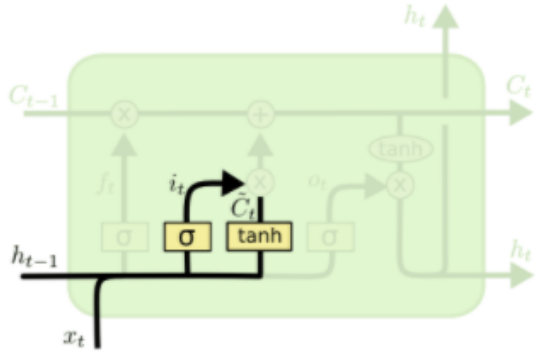


$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

(*Understanding LSTM Networks -- Colah's Blog*, 2015)

The next stage is to figure out what new data we'll store in the cell state. There are two components to this. The "input gate layer," a sigmoid layer, chooses which values we'll update first. A tanh layer then generates a C-t vector of new candidate values that could be

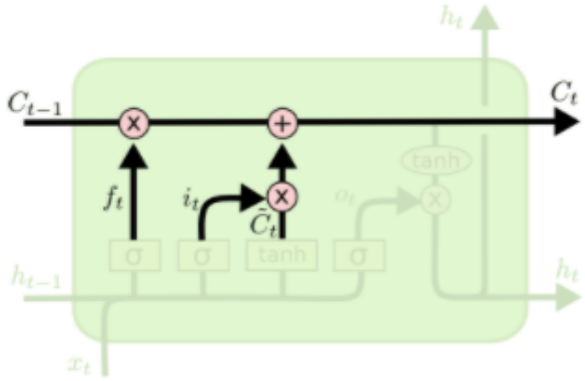
added to the state. We'll combine these two in the next step to make a state update.



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

(*Understanding LSTM Networks -- Colah's Blog*, 2015)

It's time to switch from the old cell state **Ct-1** to the new cell state **Ct**. We already know what to do because of the previous steps; now we just have to perform it. We magnify the previous state by foot, forgetting the items we had previously agreed to forget. Then we add **it*Ct** to the equation. This is the new set of candidate values, scaled by how much each state value was updated.

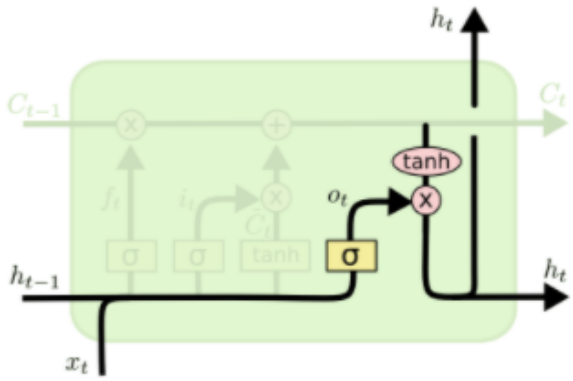


$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

(*Understanding LSTM Networks -- Colah's Blog*, 2015)

Finally, we must determine what we will produce. This output will be based on the state of our cells, but it will be filtered. First, we run a sigmoid layer to determine which aspects of the cell state will be output. The cell state is then passed through tanh (to force the values to be between 1 and 1) and multiplied by the output of the sigmoid gate, resulting in only the

parts we choose to output.



$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

(*Understanding LSTM Networks -- Colah's Blog*, 2015)

Methods:

Data source and data acquisition:

https://aqicn.org/data-platform/covid19/ was used to obtain the information.It's an open data platform for air quality. From January 2020 to the present, the data sets include around 380 significant cities throughout the world. The average (median) of numerous stations is used to create the data for each large city. For each of the air pollutant species (Pm2.5,Pm10,Ozone), as well as meteorological data (Wind, Temperature,...), the data set contains min, max, median, and standard deviation.

In the United States, all air pollutant species are translated to EPA requirements (i.e no raw concentrations). All dates are in UTC (Universal Time). The count column shows how many samples were utilized to compute the median and standard deviation.

	Date	Country	City	Specie	count	min	max	median	variance
329551	2019-12-30	NP	Kathmandu	pm1	74	55.0	223.0	148.0	16086.80
329323	2019-12-30	NP	Kathmandu	humidity	10	60.0	100.0	93.0	1467.67
329152	2019-12-30	NP	Kathmandu	wind-speed	4	0.4	0.9	0.5	0.47
329119	2019-12-30	NP	Kathmandu	pressure	10	1020.0	1024.0	1020.5	30.00
329330	2019-12-30	NP	Kathmandu	wind speed	4	0.4	0.9	0.5	0.47
...	...	...	...	...	...	...	...	...	...
15645	2021-12-23	NP	Kathmandu	humidity	52	31.0	100.0	88.8	4986.29
17046	2021-12-23	NP	Kathmandu	temperature	52	4.0	17.0	6.0	216.26
15351	2021-12-23	NP	Kathmandu	dew	36	0.0	5.0	4.0	24.83
15976	2021-12-23	NP	Kathmandu	pm25	56	117.0	257.0	159.0	5335.43
17687	2021-12-23	NP	Kathmandu	pm10	56	42.0	165.0	60.0	5819.26

5576 rows × 9 columns

Table: Data set acquired from https://aqicn.org/data-platform/covid19/

(The World Air Quality Index project, 2021)

Data Set Preparation:

Attribute Information:

The min value of the following attributes from several stations were used for inputs in our model.

•Date : Date is used as an index in our data frame.
•Pm 2.5
•Pm 1
•Pm 10
•Dew
•Humidity
•Temperature
•Pressure
•wind-speed

Date	pm25	pm1	pm10	dew	humidity	temperature	pressure	wind-speed
2019-12-30	55.0	55.0	1.0	2.0	60.0	1.1	1020.0	0.4
2019-12-31	74.0	69.0	33.0	0.0	0.0	0.0	0.0	0.0
2020-01-01	62.0	60.0	25.0	0.0	0.0	0.0	0.0	0.0
2020-01-02	74.0	70.0	27.0	0.0	0.0	0.0	0.0	0.0
2020-01-03	68.0	67.0	21.0	5.0	81.0	6.0	1020.0	0.4
...	...	...	...	...	...	...	...	...
2021-12-19	42.0	50.0	19.0	3.0	40.1	4.0	1018.0	0.3
2021-12-20	54.0	52.0	18.0	2.0	37.0	3.5	1019.0	0.3
2021-12-21	70.0	71.0	29.0	0.0	27.4	3.5	1017.0	0.3
2021-12-22	77.0	65.0	37.0	-1.5	27.0	3.0	1015.0	0.3
2021-12-23	117.0	104.0	42.0	0.0	31.0	4.0	1016.5	0.3

699 rows × 8 columns

Table: Data sets used in our model:
(The World Air Quality Index project, 2021)

LSTM Training data preparation:

Before machine learning can be used, time series forecasting issues must be reframed as supervised learning problems. From a series to a pair of input and output sequences. series to supervised() accepts univariate or multivariate time series data and converts it to a supervised learning dataset in Python.
The new dataset is organized as a DataFrame, with each column labeled according to the variable number and time step. You can generate a variety of different time step sequence type forecasting problems from a given univariate or multivariate time series. Once the DataFrame is returned, you can partition the rows of the supplied DataFrame into X and y components for supervised learning in any way you like. If you run the function with simply your data, it will create a DataFrame with t-1 as X and t as y because it is built with default parameters.

	var1(t-1)	var2(t-1)	var3(t-1)	var4(t-1)	var5(t-1)	var6(t-1)	var7(t-1)	var8(t-1)	var1(t)
1	0.312139	0.333333	0.009259	0.344262	0.986044	0.050000	0.997555	0.133333	0.421965
2	0.421965	0.418182	0.305556	0.278689	0.965111	0.000000	0.000000	0.000000	0.352601
3	0.352601	0.363636	0.231481	0.278689	0.965111	0.000000	0.000000	0.000000	0.421965
4	0.421965	0.424242	0.250000	0.278689	0.965111	0.000000	0.000000	0.000000	0.387283
5	0.387283	0.406061	0.194444	0.442623	0.993371	0.272727	0.997555	0.133333	0.393064
...	...	...	...	...	...	...	...	...	...
694	0.410405	0.400000	0.250000	0.278689	0.975508	0.136364	0.994621	0.100000	0.236994
695	0.236994	0.303030	0.175926	0.377049	0.979101	0.181818	0.995599	0.100000	0.306358
696	0.306358	0.315152	0.166667	0.344262	0.978020	0.159091	0.996577	0.100000	0.398844
697	0.398844	0.430303	0.268519	0.278689	0.974670	0.159091	0.994621	0.100000	0.439306
698	0.439306	0.393939	0.342593	0.229508	0.974531	0.136364	0.992665	0.100000	0.670520

698 rows × 9 columns

Table: Supervised learning data set

Split data into train and test sets:

The dataset is split into train set and test set, then the train set and test set are split into input and output variables. Finally, the inputs (X) are reshaped into the 3D format expected by LSTM, [samples, timesteps, features]. Seventy percent data is used for the training set and thirty percent data is used for the testing set. After the inputs are reshaped into the 3D format the corresponding shape for training inputs and testing inputs are (490,1,8) and (208,1,8).

LSTM Model:

The LSTM Model with **50** neurons in the first hidden layer and **1** neuron in the output layer for prediction is defined. The input shape includes 8 features and the time step is 1. Mean Absolute Error(MAE) loss function and efficient Adam version of stochastic gradient descent is used.The model will be fit for **50** training epochs with a batch size of **72**.

Model Validation:

There are three different methods which have been used here for the validation of the models. They are:

- ☐ Root Mean Square Error (RMSE)

The standard deviation of the residuals (prediction mistakes) is called the Root Mean Square Error (RMSE). The residuals are a measure of how distant the data points are from the regression line. The RMSE is a measure of how evenly distributed the residuals are.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (Predicted_i - Actual_i)^2}{N}}$$

In other words, it indicates how tightly the data is clustered around the line of best fit. Here,predicted i is the predicted value and actual i is the actual value. N refers to the total length of data.

- ☐ Pearson correlation coefficient.

The Pearson product-moment correlation coefficient (abbreviated as r) is a measure of the strength of a linear relationship between two variables.

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

Where,

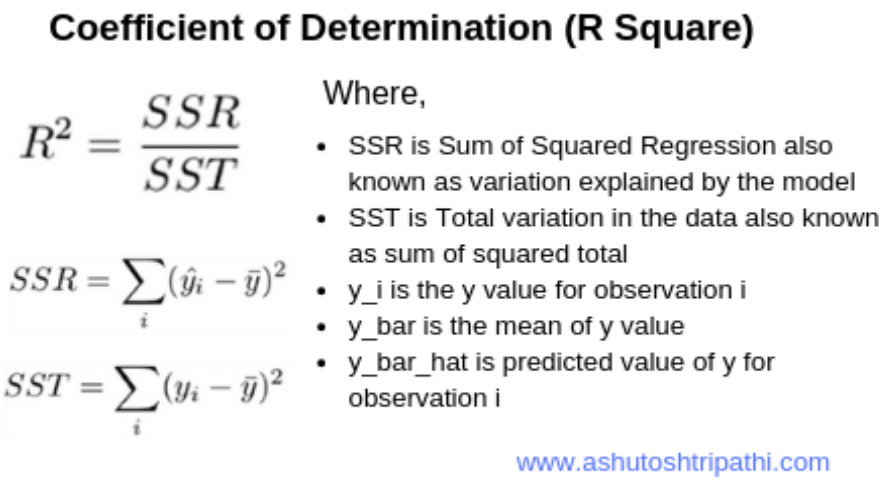
r = Pearson Correlation Coefficient

x_i = x variable samples y_i = y variable sample

$\bar{x}$ = mean of values in x variable $\bar{y}$ =mean of values in y variable

- ☐ 3)R squared:

The coefficient of determination is a metric for determining how much variability in one component can be attributed to its relationship with another.



Results:

The performance of our LSTM models is tested in our experiment for multivariate prediction models. Impacts of Covid-19 lockdown are evaluated and then prediction is made on the test data.

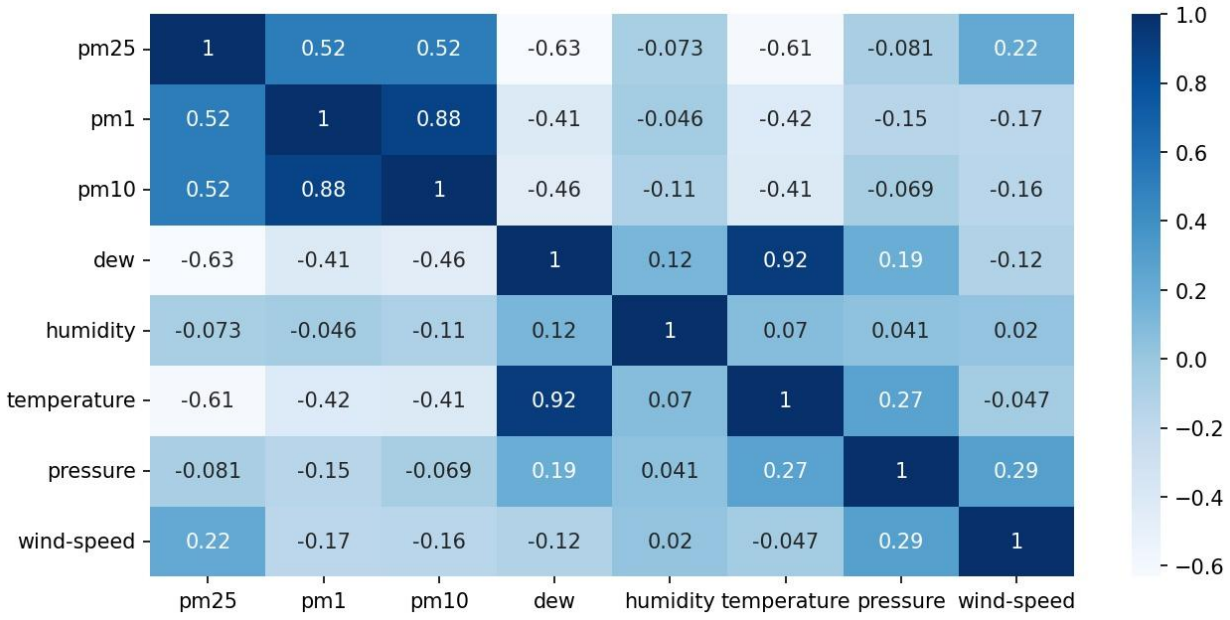


Fig: Correlation between different features used in the model.

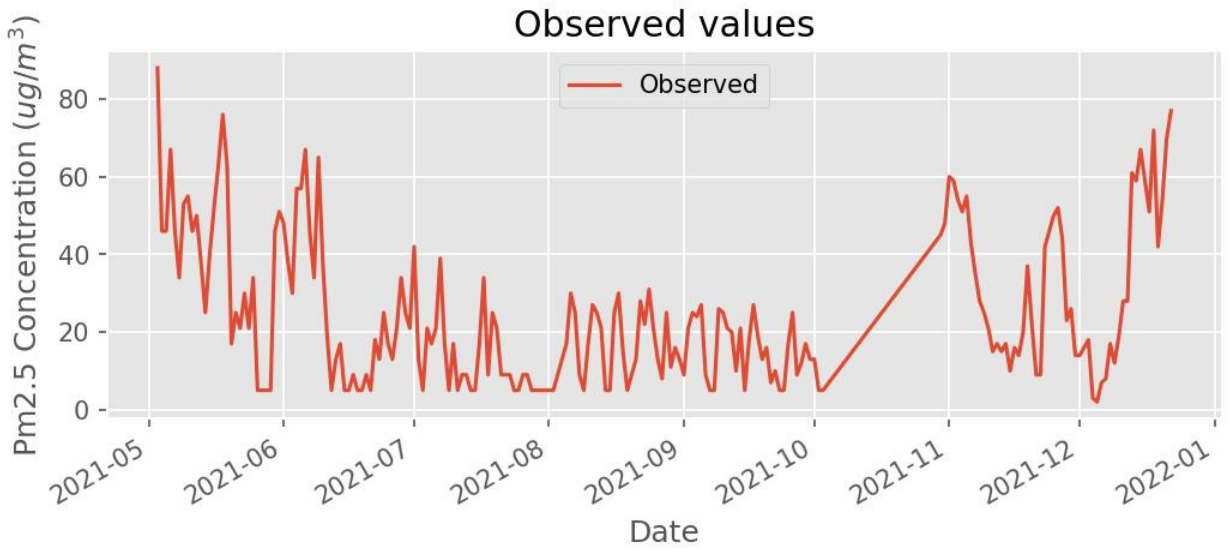


Fig: Observed graph of Test data

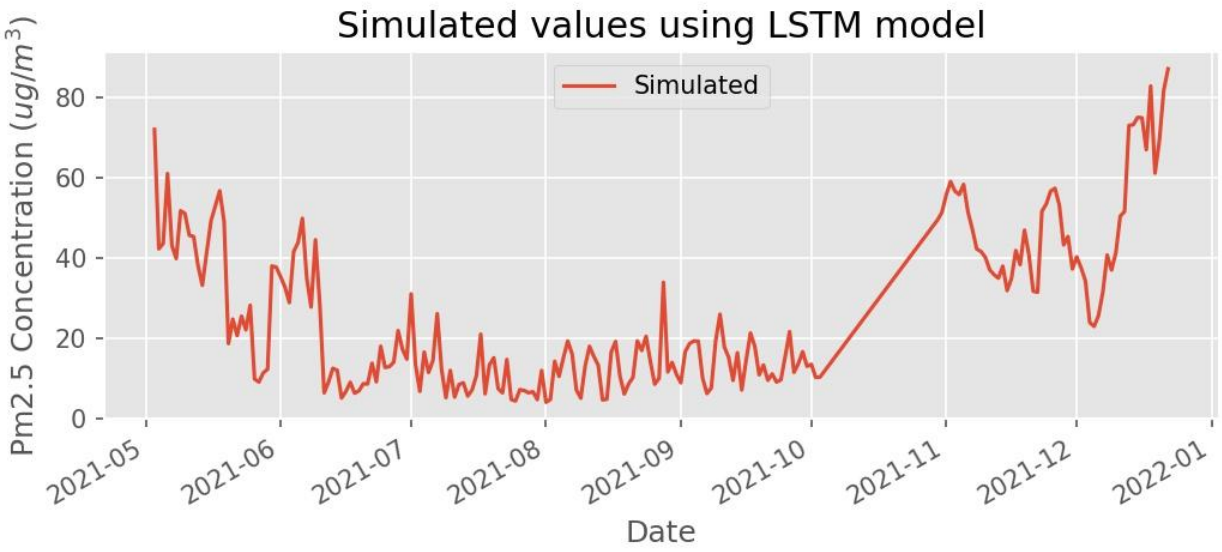


Fig: Simulated graph of test data

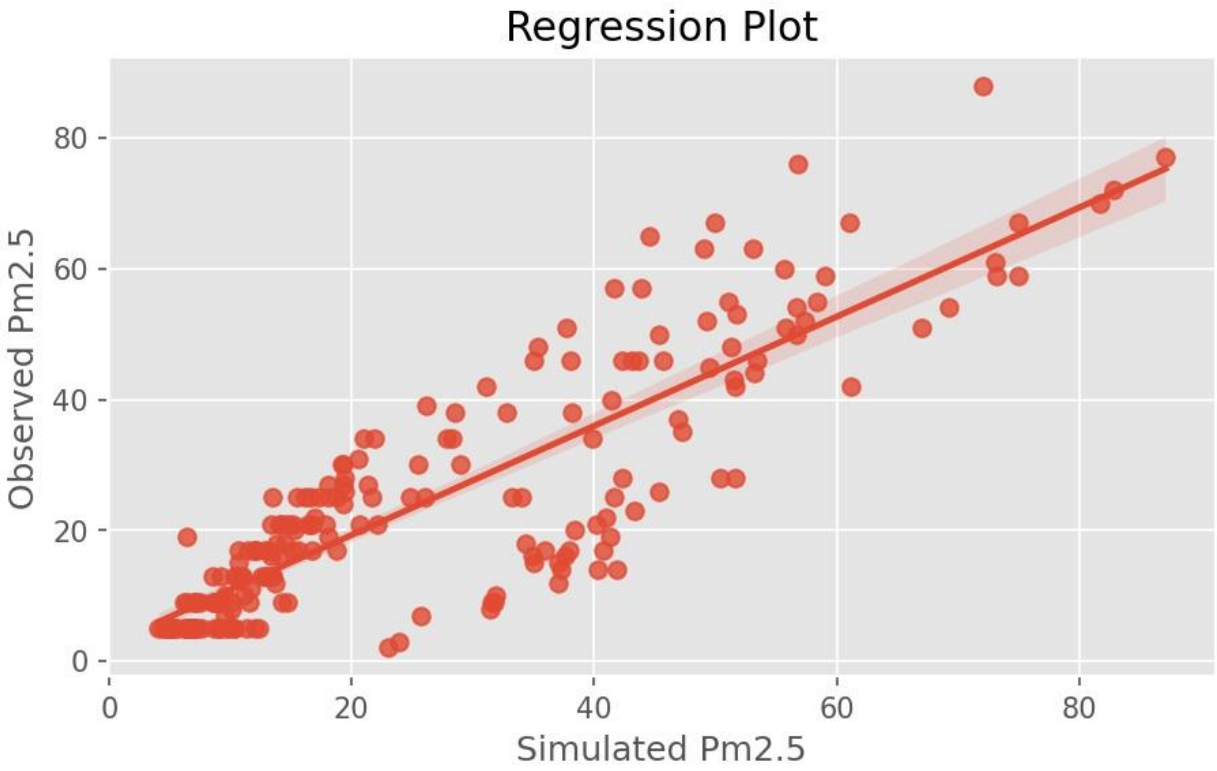


Fig: Regression plot between Observed and Simulated values for Pm2.5.

We used three methods to validate our results, Root mean square error (RMSE), Pearson's Correlation Coefficient and Coefficient of Determination (R squared).The validation results are as follow:

- The root mean square error(RMSE) is 10.45 ug/m^3
- Pearson's correlation coefficient (R) is 0.85.
- The Coefficient of Determination (Rsquared) is 0.70.

Conclusion:

During the implementation of a long-term environmental management strategy, the LSTM model can be used to estimate the spatiotemporal concentration of PM2.5 in any city throughout the world. We also discovered that COVID-19 had a major impact on air quality during the whole lockdown, which lasted a few months, and that there was an extraordinary increase in poor air quality following the lockdown, which has a seasonal effect compared to previous years.The implementation of a lockout system is not a long-term solution to the pollution concern. As a result, a sustainable management strategy should be used to preserve environmental purity.

Reference:

[1] Wikipedia contributors. (2021b, December 25). *Kathmandu*. Wikipedia.

https://en.wikipedia.org/wiki/Kathmandu

[2] “7 million premature deaths annually linked to air pollution,” August 2020, [Online; accessed 22-August-2020]. [Online]. Available: https://www.who.int/mediacentre/news/releases/2014/air-pollution/en/

[3] J. S. Apte, M. Brauer, A. J. Cohen, M. Ezzati, and C. A. Pope III, “Ambient pm2. 5 reduces global and regional life expectancy,” Environmental Science & Technology Letters, vol. 5, no. 9, pp. 546–551, 2018.

[4] H. R. Max Roser and E. Ortiz-Ospina, “World population growth,” Our World in Data, 2013, https://ourworldindata.org/world-populationgrowth.

[5] W. Hogland and J. Stenis, “Assessment and system analysis of industrial waste management,” Waste Management, vol. 20, no. 7, pp. 537–543, 2000.

[6] V.-D. Le, T.-C. Bui, and S.-K. Cha, “Spatiotemporal deep learning model for citywide air pollution interpolation and prediction,” in 2020 IEEE International Conference on Big Data and Smart Computing (BigComp). IEEE, 2020, pp. 55–62.

[7] X. Li, L. Peng, Y. Hu, J. Shao, and T. Chi, “Deep learning architecture for air quality predictions,” Environmental Science and Pollution Research, vol. 23, no. 22, pp. 22 408–22 417, 2016.

[8] T.-C. Bui, V.-D. Le, and S.-K. Cha, “A deep learning approach for forecasting air pollution in South korea using lstm,” arXiv preprint arXiv:1804.07891, 2018.

[9] G. Grell, S. Peckham, R. Schmitz, S. McKeen, G. Frost, W. Skamarock, and B. Eder, “Fully coupled “online” chemistry in the wrf model,” Atmospheric Environment, vol. 39, pp. 6957–6975, 12 2005.

[10] L. Emmons, S. Walters, P. Hess, J.-F. Lamarque, G. Pfister, F. D, C. Granier, A. Guenther, K. D, T. Laepple, O. J, X. Tie, T. G, C. Wiedinmyer, S. Baughcum, and S. Kloster, “Description and evaluation of the model for ozone and related chemical tracers, version 4 (mozart-4),” Geoscientific Model Development Discussions, vol. 3, 01 2009.

[11] H. Karimian, Q. Li, C. LI, L. Jin, J. Fan, and Y. Li, “An improved method for monitoring fine particulate matter mass concentrations via satellite remote sensing,” Aerosol and Air Quality Research, vol. 16, pp. 1081– 1092, 01 2016.

[12] Z. Yuan, X. Zhou, T. Yang, and J. Tamerius, “Predicting traffic accidents through heterogeneous urban data : A case study,” 2017.

[13] R. Collobert and J. Weston, “A unified architecture for natural language processing: Deep neural networks with multitask learning,” in Proceedings of the 25th International Conference on Machine Learning, ser. ICML ’08. New York, NY, USA: Association for Computing Machinery, 2008, p. 160–167. [Online]. Available: https://doi.org/10. 1145/1390156.1390177

[14] J. Fan, Y. Gao, and H. Luo, “Integrating concept ontology and multitask learning to achieve more effective classifier training for multilevel image annotation,” IEEE transactions on image processing : a publication of the IEEE Signal Processing Society, vol. 17, pp. 407–26, 04 2008.

[15] C. Widmer, J. Leiva, Y. Altun, and G. Ratsch, “Leveraging sequence ¨ classification by taxonomy-based multitask learning,” vol. 6044, 04 2010, pp. 522–534.

[16] A. Lindbeck and D. J. Snower, “Multitask learning and the reorganization of work: from tayloristic to holistic organization,” Journal of labor economics, vol. 18, no. 3, pp. 353–376, 2000.

[17] J. L. Elman and D. Zipser, “Learning the hidden structure of speech,” The Journal of the Acoustical Society of America, vol. 83, no. 4, pp. 1615–1626, 1988. [Online]. Available: http://link.aip.org/link/?JAS/83/ 1615/1

[18] P. J. Werbos, “Backpropagation through time: what it does and how to do it,” Proceedings of the IEEE, vol. 78, no. 10, pp. 1550–1560, 1990.

[19] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[20] R. Chandra, “Competition and collaboration in cooperative co-evolution of Elman recurrent neural networks for time-series prediction,” Neural Networks and Learning Systems, IEEE Transactions on, vol. 26, pp. 3123–3136, 2015.

[21] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” arXiv preprint arXiv:1412.3555, 2014.

[22] K. Cho, B. Van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, ¨ H. Schwenk, and Y. Bengio, “Learning phrase representations using rnn encoder-decoder for statistical machine translation,” arXiv preprint arXiv:1406.1078, 2014.

[23] A. E. Gorbalenya, S. C. Baker, R. S. Baric, R. J. de Groot, C. Drosten, A. A. Gulyaeva, B. L. Haagmans, C. Lauber, A. M. Leontovich, B. W. Neuman, D. Penzar, L. L. M. P. Stanley Perlman10, D. V. Samborskiy, I. A. Sidorov, I. Sola, and J. Ziebuhr, “The species severe acute respiratory syndrome-related coronavirus: classifying 2019-ncov and naming it sars-cov-2,” Nature Microbiology, vol. 5, no. 4, p. 536, 2020.

[24] V. Monteil, H. Kwon, P. Prado, A. Hagelkruys, R. A. Wimmer, M. Stahl, ¨ A. Leopoldi, E. Garreta, C. H. Del Pozo, F. Prosper et al., “Inhibition of sars-cov-2 infections in engineered human tissues using clinical-grade soluble human ace2,” Cell, 2020. [

25] W. H. Organization et al., “Coronavirus disease 2019 (COVID-19): situation report, 72,” 2020.

[26] D. Cucinotta and M. Vanelli, “WHO declares COVID-19 a pandemic,” Acta bio-medica: Atenei Parmensis, vol. 91, no. 1, pp. 157–160, 2020.

[27] G. Dantas, B. Siciliano, B. B. Franc¸a, C. M. da Silva, and G. Arbilla, “The impact of COVID-19 partial lockdown on the air quality of the city of rio de janeiro, brazil,” Science of the Total Environment, vol. 729, p. 139085, 2020

28] P. Wang, K. Chen, S. Zhu, P. Wang, and H. Zhang, “Severe air pollution events not avoided by reduced anthropogenic activities during COVID-19 outbreak,” Resources, Conservation and Recycling, vol. 158, p. 104814, 2020.

[29] E. Kim, H. Park, Y.-C. Hong, M. Ha, Y. Kim, B.-N. Kim, Y. Kim, Y.- M. Roh, B.-E. Lee, J.-M. Ryu et al., “Prenatal exposure to pm10 and no2 and children’s neurodevelopment from birth to 24 months of age: Mothers and children’s environmental health (moceh) study,” Science of the Total Environment, vol. 481, pp. 439–445, 2014.

[30] Y. Huang, Y. Wu, and W. Zhang, “Comprehensive identification and isolation policies have effectively suppressed the spread of COVID-19,” Chaos, Solitons& Fractals, vol. 139, p. 110041, 2020.

[31] X. Ma, I. Longley, J. Gao, and J. Salmond, “Assessing schoolchildren’s exposure to air pollution during the daily commute - a systematic review,” Science of The Total Environment, vol. 737, p. 140389, 2020.

[32] K. Liu, B.-Y. Yang, Y. Guo, M. S. Bloom, S. C. Dharmage, L. D. Knibbs, J. Heinrich, A. Leskinen, S. Lin, L. Morawska, B. Jalaludin, I. Markevych, P. Jalava, M. Komppula, Y. Yu, M. Gao, Y. Zhou, H.- Y. Yu, L.-W. Hu, X.-W. Zeng, and G.-H. Dong, “The role of influenza vaccination in mitigating the adverse impact of ambient air pollution on lung function in children: New insights from the seven northeastern cities

[33] Z. Wang, W. Wei, and F. Zheng, “Effects of industrial air pollution on the technical efficiency of agricultural production: Evidence from china,” Environmental Impact Assessment Review, vol. 83, p. 106407, 2020.

[34] M. Xu, Y. Wang, and Y. Tu, “Uncovering the invisible effect of air pollution on stock returns: A moderation and mediation analysis,” Finance Research Letters, p. 101646, 2020.

[35] J. L. Elman, “Finding structure in time,” Cognitive Science, vol. 14, pp. 179–211, 1990.

[36] C. W. Omlin, K. K. Thornber, and C. L. Giles, “Fuzzy finite state automata can be deterministically encoded into recurrent neural networks,” IEEE Trans. Fuzzy Syst., vol. 6, pp. 76–89, 1998.

[37] C. W. Omlin and C. L. Giles, “Training second-order recurrent neural networks using hints,” in Proceedings of the Ninth International Conference on Machine Learning. Morgan Kaufmann, 1992, pp. 363–368.

[38] C. L. Giles, C. Omlin, and K. K. Thornber, “Equivalence in knowledge representation: Automata, recurrent neural networks, and dynamical fuzzy systems,” Proceedings of the IEEE, vol. 87, no. 9, pp. 1623–1640, 1999.

[39] S. Hochreiter, “The vanishing gradient problem during learning recurrent neural nets and problem solutions,” Int. J. Uncertain. Fuzziness Knowl.-Based Syst., vol. 6, no. 2, pp. 107–116, 1998

[40] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015