Ecological Modelling 457 (2021) 109676

Contents lists available at ScienceDirect

Ecological Modelling

journal homepage: www.elsevier.com/locate/ecolmodel

Comparative assessment of modeling deep learning networks for modeling ground-level ozone concentrations of pandemic lock-down period

Ekin Ekinci [a,*], Sevinç İlhan Omurca [b], Bilge Özbay [c]

[a] Sakarya University of Applied Sciences, Faculty of Technology, Department of Computer Engineering, Sakarya, Turkey
[b] Kocaeli University, Faculty of Engineering, Department of Computer Engineering, Kocaeli, Turkey
[c] Kocaeli University, Faculty of Engineering, Department of Environmental Engineering, Kocaeli, Turkey

ARTICLE INFO

Keywords:
Ground-level ozone
Pandemic lock-down
COVID-19
Deep learning
Long short term memory (LSTM)

ABSTRACT

Covid-19 pandemic lock-down has resulted significant differences in air quality levels all over the world. In contrary to decrease seen in primary pollutant species, many of the countries have experienced elevated ground-level ozone levels in this period. Air pollution forecast gains more importance to achieve air quality management and take measures against the risks under such extra-ordinary conditions. Statistical models are indispensable tools for predicting air pollution levels. Considering the complex photochemical reactions involved in tropospheric ozone formation, modeling this pollutant requires efficient non-linear approaches. In this study, deep learning methods were applied to forecast hourly ozone levels during pandemic lock-down for an industrialized region in Turkey. With this aim, different deep learning methods were tested and efficiencies of the models were compared considering the calculated RMSE, MAE, R^2 and loss values.

1. Introduction

Tropospheric ozone (O_3) is a worldwide air pollutant that is well-known as the main index substance of photochemical smog phenomenon. Formation of tropospheric O_3 involves complex photochemical interactions of nitrogen oxides (NOx), volatile organic compounds (VOCs) and carbon monoxide (CO) which are known to be precursors of O_3 (Özbay et al., 2011). So, it is evident that tropospheric ozone levels are correlated with anthropogenic pollution emissions (Alvim-Ferraz et al., 2006). Due to serious harmful effects on human well-being and ecology, tropospheric ozone is of concern to many scientific studies. Chronic exposure to high levels of O_3 is associated with decreased pulmonary functions and cardiovascular diseases (Zhan et al., 2018). Raised ground level O_3 has well-known harmful impacts on vegetation as it causes leaf injury and growth reduction (Hayes et al., 2010). Furthermore, ozone exhibits a remarkable greenhouse potential that contributes to the global climate change (Sharma et al., 2017). Certainly, intensity of the mentioned impacts increases with rising O_3 concentrations.

Air quality parameters indicate significant variations during Covid-19 pandemic period all over the world. Although many of the primary pollutants (...) exhibited decreasing tendencies during lock-down days, an increasing trend was noted for ground level ozone concentrations. Hashim et al. (2020) concluded that O_3 concentrations increased during lock-down periods and the daily O_3 concentration exceeded World Health Organisation (WHO) limit (100 μg/m^3) in mid-July. In another study, Kerimray et al. (2020) investigated the impacts of COVID-19 lock-downs on variations of air quality in Kazakhstan and determined 15% increase in ground level O_3 concentrations when compared to pre-lock-down period. Similarly, O_3 levels exhibited a remarkable increasing tendency in Rio de Janeiro (Dantas et al., 2020), Barcelona (Tobías et al., 2020) and Delhi (Mahato et al., 2020) during lock-down period. The observed increase in O_3 levels has attributed to declining NOx emissions considering the negative correlation between them. Actually, increasing trend of tropospheric O_3 during pandemic period worth to be examined in detail. Undoubtedly, predictive models are useful tools to forecast the forthcoming levels of ground-level O_3 under such extra-ordinary conditions.

In literature, there are different studies about application of statistical and deterministic approaches for modeling tropospheric O_3 concentrations. Statistical methods which can be categorized as linear and non-linear models are based on analyzing monitored datasets in order to obtain qualitative or semi-quantitative results about forthcoming ozone levels. On the other hand, Eulerian, Lagrangian and Gaussian are the deterministic chemistry-transport approaches used to model atmospheric processes. Deterministic models can be used to provide knowledge about formation mechanism of ground-level ozone and also to obtain prognostic time- and spatially-resolved concentrations for

* Corresponding author.
E-mail addresses: ekinekinci@subu.edu.tr (E. Ekinci), silhan@kocaeli.edu.tr (S. İlhan Omurca), bilge.alyuz@kocaeli.edu.tr (B. Özbay).

https://doi.org/10.1016/j.ecolmodel.2021.109676
Received 7 February 2021; Received in revised form 19 July 2021; Accepted 20 July 2021
Available online 5 August 2021

different scenarios (Zhang et al., 2012). There are numerous studies about combined and individual applications of these approaches for O_3 modeling. Among the statistical methods, multivariate regression models (Draxler, 2000; Abdul-Wahab et al., 2005; Kovač-Andrić et al., 2009; Özbay et al., 2011; Lv et al., 2016), fuzzy models (Lin and Cobourn, 2007; Cheng et al., 2011; Carbajal-Hernández et al., 2012) and neural network models (Hadjiiski and Hopke, 2000; Chaloulakou et al., 2003; Pastor-Bárcenas et al., 2005; Coman et al., 2008; Sekar et al., 2016; Gao et al., 2018) have been extensively studied. In some cases, statistical and deterministic models were applied in an integrated manner in order to improve forecasting performance (Kalenderski and Steyn, 2011; Gradišar et al., 2016).

Deep learning technologies are also emerging methods for efficient modeling of different air pollutants (Li et al., 2016; Zhang et al., 2016). When compared to conventional neural networks, the most significant superiorities of deep learning methods can be sorted as demand for less computational units, automatic learning without guidance and higher modeling efficiency (Wu et al. 2015). Hence deep learning provides adequate accuracy in solving complex problems even in presence of large datasets, it is considered to be a promising tool for prediction of ground-level O_3 concentrations (Wang et al., 2020).

The major goal of this work was to evaluate the efficiency of deep learning approach in modeling hourly concentrations of tropospheric O_3 for pandemic lock-down period. Appropriate model architecture was selected to predict relatively higher ozone levels by selected inputs consisted of meteorological parameters (temperature, wind speed and relative humidity) and pollutant parameters ($PM10$, SO_2, NO, NO_2, O_3).

2. Materials and methods

2.1. Studied area and data collection

Data used in this study was collected in Korfez district of Kocaeli located at the northeastern coast of Marmara Sea. There are important industrial establishment in the region which of two is working at petroleum refinery sector. Transportation facilities are also developed with two motorways, railway and special ports in order to meet the requirements of industrial plants.

Unfortunately, ecological structure has been destroyed in the region as a result of industrial development. In particular, local air quality is of concern and is monitored by continuous measurements. $PM10$, $PM2.5$, NO, NO_2, NOx, SO_2, O_3 and also meteorological parameters are being measured in the station monitored with continuous measurements in the station belonging to Air Quality Monitoring Network of Environmental Ministry. In the station NO, NO_2 and NOx were measured by using Teledyne API 200 E model analyzer whereas O_3 and SO_2 were measured by Teledyne API 400 E and Teledyne API 100 E model devices. BAM-1020PM Monitoring System was used for $PM10$ and $PM2.5$ measurements. Temperature (T), humidity (RH), pressure (P), wind direction (WD) and wind speed (WS) were monitored by using Delta OHM model device.

In this study hourly measured values of $PM10$, SO_2, NO, NO_2, T, WS, RH and O_3 were used as input variables in order to predict the O_3 concentrations of next hour (t+1). With this aim, data obtained in May 2020 was preferred as most of the stay-at-home days (total 15 days) were recorded in May during Covid-19 pandemic. Summary of various pollutants and meteorological parameters measured at study area.

2.2. Methodology

2.2.1. LSTM

Long Short Term Memory (LSTM) proposed by Hochreiter and Schmidhuber (1997) is a kind of special Recurrent Neural Network (RNN) architecture that can perform better for long-term dependency problems. The structure of RNN and LSTM are shown in Figs. 1 and 2 respectively.

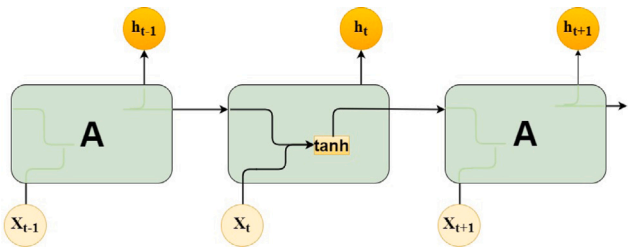

Fig. 1. RNN architecture.

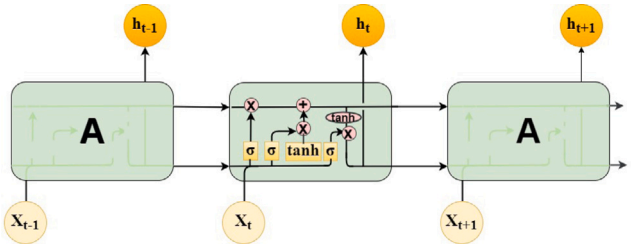

Fig. 2. LSTM architecture.

x_t and h_t represent the input and the recurrent information of the cell at time t respectively.

While RNN is a single layer neural network with a feedback loop, in LSTM, the repeating module has four neural network layers interacting in a special way as shown in Fig. 2. An LSTM network consists of three gates that controls the cell states: the input gate, the forget gate, and the output gate.

The forget gate decides which information are going to be discarded from the cell state. The output of the forget gate is obtained by a sigmoid function that takes the information from the previous cell and the current cell as inputs. It is mathematically expressed as follows based on :

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

In Eq. (1) above, while f_t represents the forget gate, W_f and b_f represent the weight and the bias of the forget gate respectively.

The input gate decides what new information is going to be added in the cell. To achieve this, a sigmoid layer first decides which values are going to be updated. Then a tanh layer generates a vector of candidates that could be added to the state. The mathematical expression is given with Eq. (2). Finally the outputs of the tanh function and the sigmoid function are multiplied as in Eq. (3).

$$\begin{aligned} i_t &= \sigma(W_i.[h_{t-1}, x_t] + b_i) \\ \tilde{C}_t &= tanh(W_c.[h_{t-1}, x_t] + b_c) \end{aligned} \tag{2}$$

$$C_t = f_t * C_{t-1} + (1 - f_t) * \tilde{C}_t. \tag{3}$$

When the i_t represents the input gate, W_i and b_i represent the weight and the bias of the input gate respectively. C_t denotes the state of the LSTM cell.

The output gate decides what information to output from the current cell state. To achieve this, first the previous and the current state are transferred to the sigmoid function. Then the new state is put through tanh function and multiply it by the sigmoid unit output. The new state is transferred to the next state. The mathematical expressions for output gate is given with Eq. (4) and Eq. (5) as follows:

$$\begin{aligned} o_t &= \sigma(W_o[h_{t-1}, x_t] + b_o) \\ h_t &= o_t * tanh(C_t) \end{aligned} \tag{4}$$

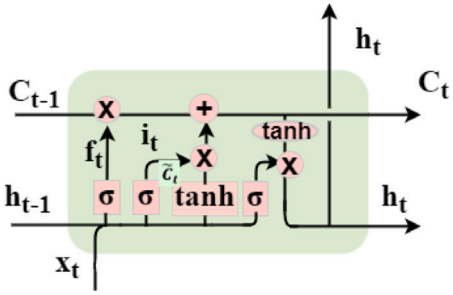

Fig. 3. Inner structure of LSTM unit.

$$\begin{aligned} f_t &= \sigma(W_f[C_{t-1}, h_{t-1}, x_t] + b_f) \\ i_t &= \sigma(W_i[C_{t-1}, h_{t-1}, x_t] + b_i) \\ o_t &= \sigma(W_o[C_{t-1}, h_{t-1}, x_t] + b_o) \end{aligned} \tag{5}$$

where o_t, W_o and b_o represent the output gate, weight and the bias of the output gate respectively.

Complete view of An LSTM unit is given in Fig. 3.

2.2.2. BiLSTM

Bidirectional LSTM (BiLSTM) is an improved LSTM network proposed by Graves and Schmidhuber and it is also used in occasions where the learning problem is sequential (Graves and Schmidhuber, 2005). Bidirectional LSTM composed of two LSTM units, one of them is built and optimized in a forward direction, and the other is built and optimized in a backward direction. Applying the LSTM units in two direction improves the ability of learning long term-dependencies and consequently this improves the accuracy of the model. The hidden state of the forward LSTM process can be represented as $\vec{h}_t = LSTM(x_t, \vec{h}_{t-1})$; the hidden state of the backward LSTM process can be represented as $\overleftarrow{h}_t = LSTM(x_t, \overleftarrow{h}_{t+1})$. The eventual output of BiLSTM unit are the sum of the forward and the backward hidden states and represented as $h_t = \vec{h}_t + \overleftarrow{h}_t$.

2.2.3. Stacked LSTM

Stacked LSTM network which has a simple and efficient structure, provide higher capacity and depth for the LSTM network by stacking the LSTM layers. Three recurrent layers LSTM is illustrated in Fig. 4. The unrolled stacked LSTM along the time dimension is presented in Fig. 5 where we assume that the sequence length is 4.

The output of the $(L-1)$th LSTM layer at time t is represented as h_t^{L-1}. This output is also the input, x_t^L, of the Lth layer. When the model is examined it is observed that the recurrent connections are only within one layer. The mathematical formulas of the Lth LSTM layer are as follows:

$$\begin{aligned} f_t^L &= \sigma(W_{fh}^L h_{t-1}^L + W_{fx}^L h_t^{L-1} + b_f^L) \\ i_t^L &= \sigma(W_{ih}^L h_{t-1}^L + W_{ix}^L h_t^{L-1} + b_i^L) \\ \tilde{c}_t^L &= tanh(W_{\tilde{c}h}^L h_{t-1}^L + W_{\tilde{c}x}^L h_t^{L-1} + b_{\tilde{c}}^L) \\ c_t^L &= f_t^L c_{t-1}^L + i_t^L \tilde{c}_t^L. \end{aligned} \tag{6}$$

2.2.4. CNN-LSTM

Convolutional neural network (CNN) is a special Multilayer perceptron (MLP) which composes of one or more convolutional and max pooling layers followed by one or more fully connected layers with a rectified linear activation function (ReLU).

On the other hand, CNN-LSTM combines both the advantages of CNN and LSTM networks. In this respect, it can be regarded as a hybrid network which can be proposed with the various combinations of CNN and LSTM networks. This hybrid network basically includes four layers namely Convolutional layer, Pooling layer, LSTM layer and Dense layer.

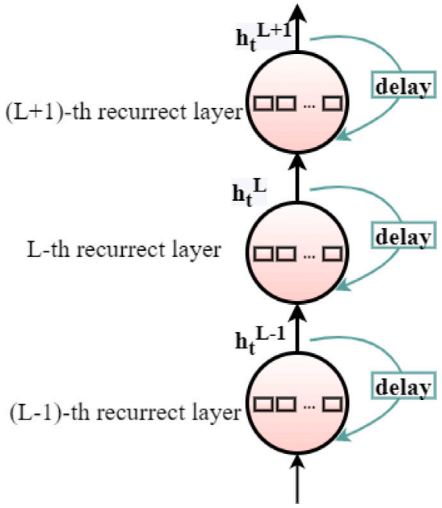

Fig. 4. Stacked LSTM.

The way this architecture works is as follows: at first the input is given to the network and features are extracted from CNN layers. The output of the CNN is fed to LSTM to complete learning.

2.2.5. Conv-LSTM

Convolutional-LSTM (Conv-LSTM) is a special LSTM architecture that contains a convolution operation inside the LSTM units. Namely, it changes the matrix multiplication in the LSTM unit with a convolutional operation. The Conv-LSTM model defines the next state of a cell by the inputs and the past states of its neighbors (Shi et al., 2015). In Conv-LSTM, the states, cell memory, gates and parameters are encoded as high dimensional tensors. Let the inputs are $X_1, \ldots, X_t$, the cell outputs are $C_1, \ldots, C_t$, the hidden states are $H_1, \ldots, H_t$. The gates which are 3D tensors are i_t, f_t, o_t. The inner structure and the equations of Conv-LSTM are shown in Figs. 6 and 7, respectively. The convolution operation is denoted as * and Hadamard product is denoted as ∘ in the equations.

$$\begin{aligned} i_t &= \sigma(W_{xi} * X_t + W_{hi} * H_{t-1} + W_{ci} \circ Ct - 1 + b_i) \\ f_t &= \sigma(W_{xf} * X_t + W_{hf} * H_{t-1} + W_{cf} \circ Ct - 1 + b_f) \\ C_t &= f_t \circ C_{t-1} + i_t \circ tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c) \\ o_t &= \sigma(W_{xo} * X_t + W_{ho} * H_{t-1} + W_{co} \circ C_t + b_o) \\ H_t &= o_t \circ tanh(C_t). \end{aligned} \tag{7}$$

3. Case study

3.1. Data preparation

In the input data used in the experiments both the range within attributes and among attributes are considerably different. For example, while RH values range from 27.93 to 100.0, WS values range from 0.15 to 4.02. Differences on ranges of attributes cause the model to learn incorrectly. To deal with this problem normalization is recommended to brings the range to certain values. Consequently, variables in our dataset were transformed into $[0, 1]$ range by using min–max normalization. This can provide equalization of the impact of attributes on prediction and more accurate predictions. The formula of min–max normalization is as follows (8):

$$y_i = \frac{x_i - x_{min}}{x_{max} - x_{min}}. \tag{8}$$

where x_i is the i^th data point; while x_{min} minimum-valued data point, x_{max} maximum-valued data point in the dataset. y_i represents the normalized form of x_i and a value in the range $[0, 1]$.

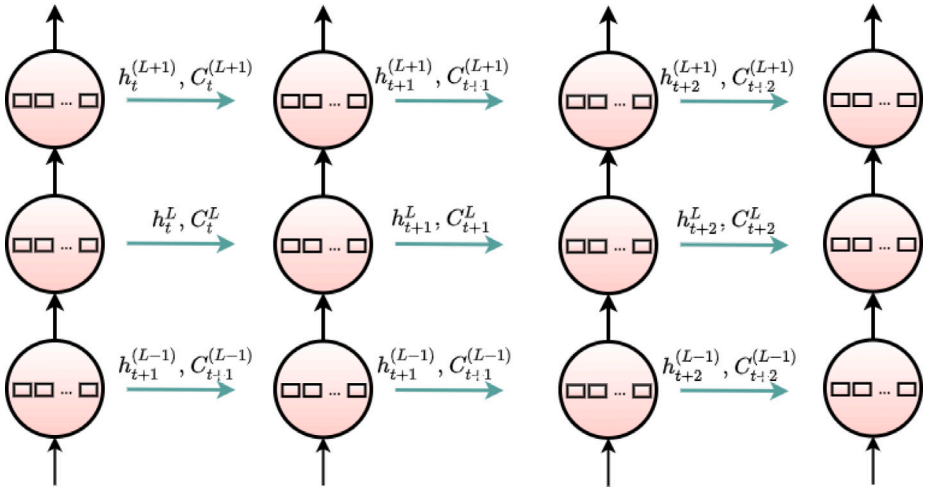

Fig. 5. Unrolled LSTM.

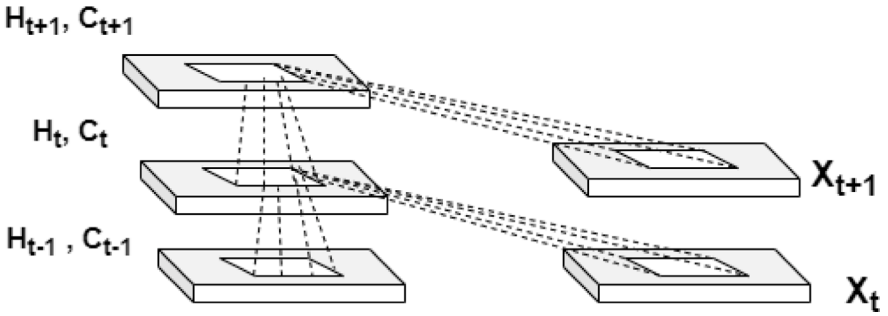

Fig. 6. Inner structure of Conv-LSTM.

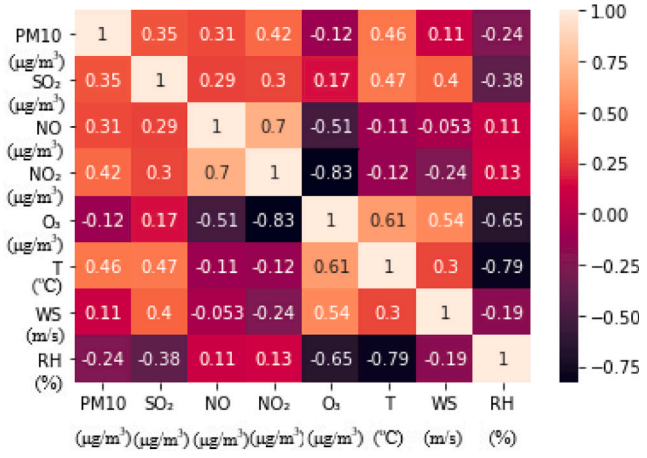

Fig. 7. Correlations between model input variables.

In order for the models to be evaluated correctly training and test sets should be suitable for air pollution. On the other hand, there is no certain rule how to divide dataset into training and test set. For comparison, experimental dataset containing 470 hourly observation values in total is divided into two, training set and test set. The different data splits for testing the models is proposed as given in Table 1. When performing splitting training and test sets should have the same distribution and our splitting provides this.

Table 1
The different data splits for testing the LSTM models.

Train split	Test split	# Data points train set	# Data points train set
90	10	424	46
80	20	376	94
75	25	357	113

3.2. Error metrics

In order to evaluate performance of the proposed models we used Root Mean Square Error (RMSE), Mean Absolute Error (MAE), R-squared (R^2) and loss value error metrics in these experiments. In these experiments, the (RMSE) is used as the main evaluation metric. In addition, while evaluating the network structures of the models, Akaike information criterion (AIC) and Bayesian information criterion (BIC) criteria were used.

RMSE is the standard deviation of the error between expected and predicted value. RMSE is sensitive to the outliers and is used to measure magnitude of extreme errors and outliers and degree of dispersion; therefore, we use the RMSE indicator. While lower RMSE indicates better central tendency and less extreme errors, higher RMSE indicates worse central tendency and a large dispersion.

MAE is the average of the error between each pair of expected and predicted values. Just as RMSE is sensitive to outliers, so is the MAE. The deviation is high in data with high dispersion because of outliers, so the error obtained with MAE is also high. The lower error reflects the better model performance.

R^2 is the measure of linear relation between two random variables. If the variation between expected and predicted value is high, little R^2 is obtained and little R^2 shows that expected and predicted have low

relation with each other. Otherwise, the relation between these values is high, so the R^2 value is high. In general, R^2 which is equal or greater than 0.8 is accepted as a successful prediction.

The loss value is one of the basic error metric for deep learning algorithms. The closer the loss value is to zero, the better the prediction performance of the model is. Hence, the aim is to minimize the loss value.

Formulas of these three error metrics are presented in Eqs. (9)–(11).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} y_i - \hat{y}_i}, \tag{9}$$

$$MAE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i|}, \tag{10}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y}_i)^2}, \tag{11}$$

where y_i is the expected value of ith observation; $\hat{y}_i$ is the predicted value of the same observation; $\overline{y}_i$ is the mean value of observation; n is the number of observations in the test set.

AIC and BIC are the most commonly used model selection methods used to make selection amongst the models of different network structures. The reason for using AIC and BIC is that while R^2 considers the model only on fit, AIC and BIC considers the model complexity Xiong et al. (2019).

AIC was suggested by Akaike Akaike (1998) which is used to evaluate the fitness of the model. The AIC is computed as follows:

$$AIC = L \times \log(\frac{1}{n}\sum_{i=1}^{n} y_i - \hat{y}_i) + 2 \times F. \tag{12}$$

For data prediction, the efficiency of the model is determined by using BIC. The BIC is computed as follows:

$$BIC = L \times \log(\frac{1}{n}\sum_{i=1}^{n} y_i - \hat{y}_i) + F \times \log(L). \tag{13}$$

In the Eqs. (12)–(13) above, while L is the length of the dataset, F represents the number of features. When both of the AIC and BIC are the smallest, the model is accepted as the optimal. The main difference between these two models is that BIC shows that it chooses simpler models, whereas AIC generally tends to choose a more elaborate model (Agiakloglou and Tsimpanos, 2021).

3.3. Building models

In this study we have proposed to compare LSTM models to decide which model and which parameters are better making forecast for air pollution for O_3 value. We use Keras[1] a high-level Python library developed for deep learning in the implementation of LSTM models namely LSTM, BiLSTM, Stacked LSTM, CNN-LSTM and Conv-LSTM. In order to build architectures for models to be used, several model parameters including the number of the neurons in the input layer, the number of hidden layers, the number of in hidden neurons in each hidden layer, size of fully connected and output layer should be determined primarily. In addition to these, batch size and number of epoch that need to be adjusted for all models which have effect on working of the models. These parameters are common to all models.

The number of neurons in the input layer depends on the number of attributes in the dataset so the total number of neurons in this layer is equal to 8. There is no formula for determination of the hidden neuron size. In the network there is one hidden layer number of hidden neurons are selected from the set 10, 20, 50, 64 and a fully connected layer with one neuron. The one-hour prediction is used to predict (t+1)-hour air pollution that is why the size of the output layer is 1 with one neuron. Furthermore, the number of epochs is also important. In our experiment, epoch size is selected from the set 1000, 2000 and batch size is set to 72. Grid search or heuristic search methods can be used to provide better parameter configuration; but, realizing these methods are almost impossible and computationally expensive because of the large search space. For these reasons, the experiments are conducted by using fixed set of parameters.

In addition to parameter settings, loss function optimization is crucial for building deep learning models. Especially in recent years, optimization of the loss function has become very important. In the training phase of the models we used Adam loss function optimizer. Adam, a stochastic gradient descent algorithm computes first and second order moment with exponential moving average. In this regard, Adam provides fast convergence which is also the main reason why it is used in this study.

To figure out performance of the deep learning model it is necessary to use an activation function (Misra, 2019). While LSTM, BiLSTM and Stacked LSTM tanh is used as activation function, for CNN-LSTM and Conv-LSTM Rectified Linear Unit (ReLU) is used.

At first, LSTM and BiLSTM models are designed based on the parameter settings above. Our stacked LSTM which consists 2 stacked LSTM layers is composed adding a new LSTM layer to LSTM architecture. The neurons of the first LSTM layer are fed with input attribute vector. Then, the first LSTM layer feeds into the second LSTM layer. At the end, the second LSTM layer feeds into a fully connected layer.

The CNN-LSTM differs from LSTM with it layers. It has convolutional layer and in it number of filters, kernel size and activation function needs to be determined. These parameters play a very important role in the performance of the model. In this study only one convolutional layer is used. For this layer number of filters is selected as 64. As an activation function we use relu to accelerate learning convergence. To minimize the information loss, we use 2 kernel. Another layer that differs from LSTM is the pooling layer. We have one pooling layer and set 1 to pooling size. The input data at first feeds into convolutional layer, the output of this layer feeds into pooling layer, finally passes to LSTM layer.

Our Conv-LSTM consists of one convolutional layer, one flatten layer and one dense layer. The number of filters is selected from the set 10, 20, 60, 64 and kernel is set to 2×1 in the convolutional layer.

The summary of the models is given with Table 2.

4. Results and discussions

4.1. Evaluation of the input variables

As mentioned previously although pandemic lock-downs implementations were started in April, May was the prevailing month of the period. In this work, the impacts of the lock-downs on air quality have been evaluated considering the monthly average values of the pollution parameters. Tables 3 and 4 represent the monthly average values of the pollution and meteorological parameters, respectively. Data of pre- and post-lockdown periods (March and June months) were also presented in the tables.

As seen from Table 3 $PM10$, NO and NO_2 concentrations exhibited gradually declining trends from March to May as a result of lock-down implementations. Conversely, 46.98% increase was observed for ground-level O_3 levels in this period. This increase can be attributed to decreasing NO and NO_2 levels which are negatively correlated with ground-level O_3 concentrations. Increasing temperature values (Table 4) may also promote the photochemical reactions of ozone formation. Average temperature values increased remarkably in the region (from 10.88 to 18.50 °C) in March–May period.

Bivariate correlation analysis was performed in order to evaluate the interactions between the input variables. Obtained Pearson's correlation coefficients (r) were given in Fig. 7.

[1] https://pypi.org/project/Keras/

Table 2
Summary of the Models.

Model name	Optimizer	Activation function	# LSTM layer	# Hidden neurons (layer)/Filter	# Epoch	# Kernel
LSTM	Adam	tanh	1	10, 20, 50, 64	1000, 2000	–
BiLSTM	Adam	tanh	1	10, 20, 50, 64	1000, 2000	–
Stacked LSTM	Adam	tanh	2	10, 20, 50, 64	1000, 2000	–
CNN-LSTM	Adam	relu	1	10, 20, 50, 64	1000, 2000	2
Conv-LSTM	Adam	relu	1	10, 20, 50, 64	1000, 2000	(2,1)

Table 3
Monthly average values of the pollution parameters during March–June 2020.

Months	$PM10$ (μg/m^3)		SO_2 (μg/m^3)		NO (μg/m^3)		NO_2 (μg/m^3)		O_3 (μg/m^3)	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
March	47.14	21.37	8.76	7.36	16.50	17.24	30.88	13.78	39.76	16.90
April	35.39	13.04	16.70	16.50	6.11	5.34	28.55	15.99	55.01	16.76
May	**30.05**	**17.30**	**14.29**	**12.06**	**3.93**	**3.03**	**27.76**	**12.66**	**58.43**	**9.44**
June	32.93	7.67	5.29	4.53	5.38	3.40	34.40	9.95	53.66	10.50

Table 4
Monthly average values of the meteorological parameters during March-June 2020.

Months	T °C		WS (m/s)		RH (%)	
	Mean	SD	Mean	SD	Mean	SD
March	10.88	2.83	1.26	0.48	76.72	9.80
April	12.83	2.80	1.24	0.23	67.57	11.75
May	**18.50**	**4.95**	**1.38**	**0.63**	**70.84**	**18.46**
June	23.02	2.51	0.99	0.22	71.78	8.01

Considering the relationships between O_3 and other pollutant species, it is clearly seen that NO and NO_2 have strong negative correlations with O_3 (r=-0.51 and −0.83, respectively). Additionally, ground-level O_3 levels exhibited strong positive correlation with temperature (r=0.61) and strong negative correlation with relative humidity (r=-0.65). Obtained correlations which are generally coherent with the literature provide informative explanations for increasing O_3 levels in the studied region.

4.2. Effect of parameter settings and LSTM models

In this study, we aim to investigate effect of different epoch size, number of hidden neurons and dataset splitting on the prediction performance of ground-level ozone concentrations of pandemic lock-down period obtained for each LSTM model at hourly.

Although there is no systematic way to determine the network structure in deep learning models, the information gain to be obtained from the models with the differentiation of the number of neurons is taken into account. For this purpose, firstly, the information gain to be obtained in different neuron numbers for different dataset splits is calculated and the results are given in Tables 5–9.

When the tables showing the information gain of the models are examined, two important conclusions were reached: firstly, it is observed that the change in the network structure has no effect on the information gain; secondly, the information gain increased with the increase in the training samples. Obviously, the number of data in the training set increases the information gain of the models will improve. The model with the highest information gain is the Stacked-LSTM model performed with 2000 epochs with 64 neurons in which the dataset is divided as 90:10 (AIC= 251.79, BIC= 266.42).

It is seen that results based on RMSE given in the Tables 10–12 are supported by AIC and BIC results. When all results are evaluated together, it is clearly seen that chain-like Stacked LSTM has the powerful modeling capability of air pollution data. Among Stacked LSTM models 64 hidden neurons with 2000 epoch and data split 90:10 is the superior. The reason this model is most successful is that it is more appropriate model for time sequence data and has deeper structure.

In detailed analysis, for epoch size=1000 and number of hidden neuron/filters is equal to 10, RMSE value varies from 13.25 to 14.35, MAE value varies from 7.55 to 9.34 and R^2 value varies from 0.92 to 0.95. For epoch size=2000, RMSE value varies from 13.27 to 14.22, MAE value varies from 7.8 to 8.96 and R^2 value varies from 0.92 to 0.94. When these two analyzes are compared in terms of the number of iterations, although the average performance of the models has increased in 2000 iterations based on RMSE, the error values for all three metrics are very close to each other. When the number of hidden neurons/filters is taken 20 for the cases where the number of iterations is equal to 1000 and 2000, analyzes have been made. RMSE value varies from 13.33 to 13.77, MAE value varies from 7.59 to 9.64 and R^2 value varies from 0.93 to 0.94 for 1000 epoch. In 2000 epoch results of RMSE takes a value between 13.3 and 14.3, MAE takes a value between 7.7 and 8.87 and R^2 takes a value between 0.92 and 0.93. When the iteration numbers for 20 hidden neurons are compared, it is seen that the best success is achieved with 1000 iterations based on RMSE.

For hidden neurons/filters=50 and epoch size=1000, RMSE takes values in the range of 11.07 and 14.52, MAE takes values in the range of 7.51 and 8.85 and R^2 takes values in the range of 0.92 and 0.94. When the results are examined in terms of 2000 epochs, it is seen that RMSE takes values in the range of 10.75 and 13.88, MAE takes values in the range of 7.69 and 8.87 and R^2 takes values in the range of 0.9 and 0.93. When the number of hidden neurons/filters is taken 64, RMSE value varies from 10.95 to 13.77, MAE value varies from 7.68 to 9.59 and R^2 value varies from 0.93 to 0.94 for 1000 epoch. The training is done with 2000 epoch, RMSE takes values in the range of 10.68 and 13.6, MAE takes values in the range of 7.73 and 8.5 and R^2 takes values in the range of 0.91 and 0.94.

When the methods are compared for the number of epoch, it is observed that there are improvements in terms of RMSE values with the increase of epoch.

For 64 hidden neurons/filters with 2000 epoch R^2 graphics are given Fig. 8.

Performance of the developed models are evaluated considering the coefficient of determination (R^2) between the predicted and expected O_3 levels. R^2 values of the models are almost closer to each other.

For 64 hidden neurons/filters with 2000 epoch graphics of loss value are given with Fig. 9.

The train and test set errors of the deep learning models which are depicted in Fig. 9, allows us to answer an important question – are train set and test set errors tend to decrease over time? – In all of the figures, it is seen that, both train and test errors tend to decrease over the iterations. When these figures are examined in more detail, it is realized that, the train and test set error of the Stacked LSTM model are much closer to each other. Besides, There is no over-fitting in any of the models.

Based on 64 hidden neurons/filters with 2000 epoch, expected and predicted values of O_3 are represented with Fig. 10.

When the results are examined, although the obtained values converge to the observed values for all models, it is seen that the Stacked

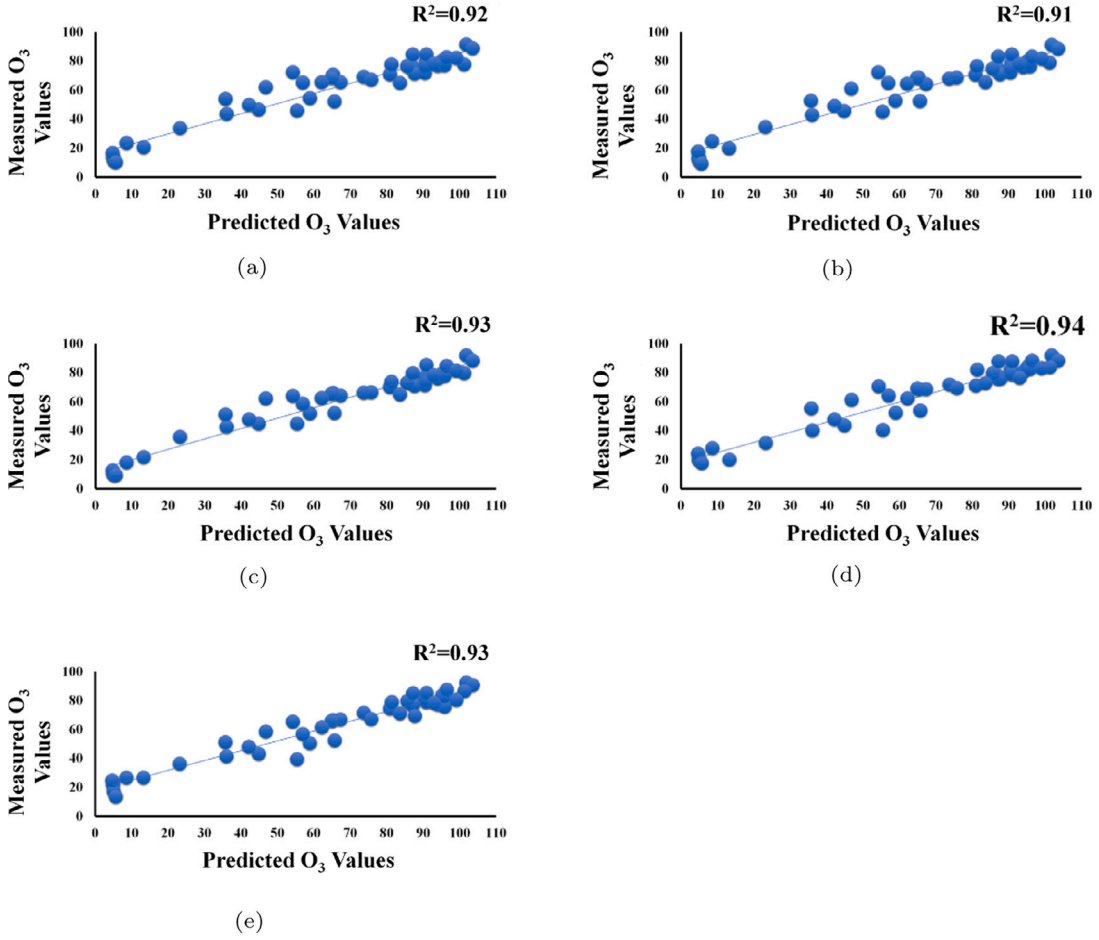

Fig. 8. Predicted O_3 values versus measured O_3 values for (a) LSTM (b) BiLSTM (c)Stacked LSTM (d) CNN-LSTM (e) Conv-LSTM.

Table 5
Information gain of LSTM for different dataset split.

# Hidden neurons (layer)	# Epoch	75:25		80:20		90:10	
		AIC	BIC	AIC	BIC	AIC	BIC
10	1000	552.13	573.95	473.75	494.1	253.75	268.38
20	1000	554.99	576.81	471.9	492.24	254.61	269.24
50	1000	551.18	573.0	471.85	492.2	254.29	268.92
64	1000	554.42	576.24	471.64	491.99	253.88	268.51
10	2000	556.87	578.69	470.64	490.99	253.85	268.48
20	2000	553.21	575.03	470.77	491.12	254.66	269.29
50	2000	553.3	575.12	471.2	491.54	254.74	269.37
64	2000	553.8	575.61	470.1	490.45	254.89	269.51

Table 6
Information gain of BiLSTM for different dataset split.

# Hidden neurons (layer)	# Epoch	75:25		80:20		90:10	
		AIC	BIC	AIC	BIC	AIC	BIC
10	1000	551.8	573.62	472.03	492.38	254.39	269.02
20	1000	552.99	574.81	472.03	492.38	254.29	268.92
50	1000	551.75	573.57	472.23	492.58	254.22	268.85
64	1000	551.72	573.54	471.5	491.85	255.45	270.08
10	2000	552.38	574.2	473.47	493.82	254.61	269.24
20	2000	553.21	575.03	472.05	492.4	255.26	269.89
50	2000	555.33	577.15	471.33	491.67	253.23	267.86
64	2000	556.67	578.49	470.84	491.18	255.3	269.93

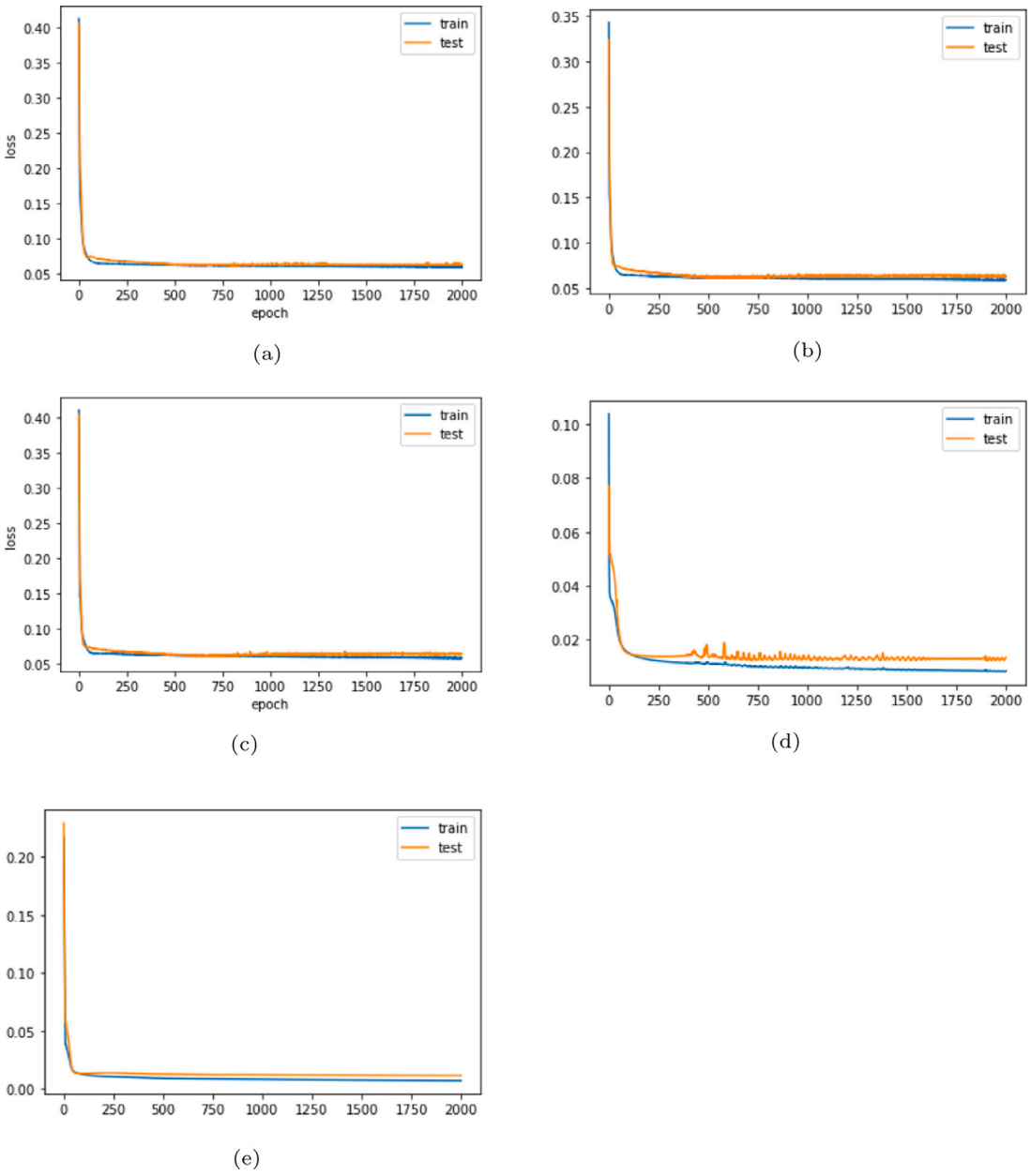

Fig. 9. Loss values for (a) LSTM (b) BiLSTM (c)Stacked LSTM (d) CNN-LSTM (e) Conv-LSTM.

Table 7
Information Gain of Stacked LSTM for Different Dataset Split.

# Hidden neurons (layer)	# Epoch	75:25		80:20		90:10	
		AIC	BIC	AIC	BIC	AIC	BIC
10	1000	550.25	572.07	474.12	494.46	256.11	270.74
20	1000	551.18	572.99	471.46	491.81	255.37	270.0
50	1000	553.95	575.77	471.61	491.96	254.36	268.99
64	1000	551.54	573.36	471.76	492.11	255.14	269.77
10	2000	550.96	572.78	470.23	490.57	255.83	270.45
20	2000	554.43	576.25	469.78	490.13	254.98	269.61
50	2000	557.25	579.07	469.43	489.78	254.39	269.02
64	2000	551.93	573.75	465.86	486.21	251.79	266.42

Table 8
Information Gain of CNN-LSTM for Different Dataset Split.

# Hidden neurons (layer)	# Epoch	75:25		80:20		90:10	
		AIC	BIC	AIC	BIC	AIC	BIC
10	1000	558.95	580.77	482.22	502.57	261.08	275.71
20	1000	565.34	587.16	482.97	503.32	255.51	270.14
50	1000	565.72	587.54	493.27	513.62	259.33	273.96
64	1000	558.72	580.54	480.53	500.88	257.83	272.46
10	2000	558.95	580.77	598.42	618.77	260.22	274.85
20	2000	559.32	581.14	477.96	498.31	260.76	275.39
50	2000	584.5	606.32	476.96	497.31	261.14	275.77
64	2000	566.0	587.82	483.22	503.57	258.41	273.04

Table 9
Information Gain of Conv-LSTM for Different Dataset Split.

#Filter	# Epoch	75:25		80:20		90:10	
		AIC	BIC	AIC	BIC	AIC	BIC
10	1000	558.92	580.74	489.65	509.99	257.55	272.18
20	1000	559.38	581.2	480.94	501.29	257.28	271.91
50	1000	563.92	585.74	476.21	496.55	255.39	270.01
64	1000	570.1	591.92	479.06	499.41	255.09	269.71
10	2000	559.52	581.34	474.95	495.3	256.47	271.10
20	2000	571.24	593.06	482.73	503.08	254.05	268.68
50	2000	569.08	590.9	485.66	506.01	262.79	277.42
64	2000	578.0	599.818	486.22	506.57	256.57	271.2

Table 10
Experimental Results for Dataset Split 75:25.

Model name	# Hidden neurons (layer)/Filter	# Epoch	RMSE	MAE	R^2	Model name	# Hidden neurons (layer)/Filter	# Epoch	RMSE	MAE	R^2
LSTM	10	1000	10.72	7.4	0.91	Stacked LSTM	10	2000	10.67	7.59	0.89
LSTM	20	1000	10.86	7.65	0.92	Stacked LSTM	20	2000	10.83	7.88	0.9
LSTM	50	1000	10.78	7.52	0.91	Stacked LSTM	50	2000	13.38	7.89	0.91
LSTM	64	1000	10.74	7.42	0.91	Stacked LSTM	64	2000	13.25	7.71	0.89
LSTM	10	2000	10.95	7.85	0.89	CNN-LSTM	10	1000	11.31	7.69	0.88
LSTM	20	2000	10.78	7.62	0.91	CNN-LSTM	20	1000	11.37	7.94	0.88
LSTM	50	2000	10.89	7.82	0.9	CNN-LSTM	50	1000	10.97	7.53	0.87
LSTM	64	2000	10.77	7.67	0.9	CNN-LSTM	64	1000	11.33	7.57	0.89
BiLSTM	10	1000	10.71	7.36	0.92	CNN-LSTM	10	2000	11.05	7.61	0.87
BiLSTM	20	1000	10.76	7.56	0.92	CNN-LSTM	20	2000	11.07	7.07	0.9
BiLSTM	50	1000	10.76	7.55	0.92	CNN-LSTM	50	2000	11.8	8.83	0.84
BiLSTM	64	1000	10.8	7.59	0.91	CNN-LSTM	64	2000	10.96	7.71	0.88
BiLSTM	10	2000	10.73	7.65	0.89	Conv-LSTM	10	1000	11.05	7.91	0.87
BiLSTM	20	2000	10.77	7.68	0.89	Conv-LSTM	20	1000	11.07	7.89	0.87
BiLSTM	50	2000	10.8	7.66	0.89	Conv-LSTM	50	1000	11.6	8.75	0.88
BiLSTM	64	2000	11.0	7.98	0.9	Conv-LSTM	64	1000	10.96	7.68	0.85
Stacked LSTM	10	1000	10.63	7.16	0.91	Conv-LSTM	10	2000	11.08	8.17	0.86
Stacked LSTM	20	1000	10.68	7.3	0.92	Conv-LSTM	20	2000	11.67	8.94	0.81
Stacked LSTM	50	1000	13.38	7.89	0.91	Conv-LSTM	50	2000	12.09	9.31	0.8
Stacked LSTM	64	1000	13.3	7.86	0.92	Conv-LSTM	64	2000	12.26	9.44	0.84

Table 11
Experimental Results for Dataset Split (80:20).

Model name	# Hidden neurons (layer)/Filter	# Epoch	RMSE	MAE	R^2	Model name	# Hidden neurons (layer)/Filter	# Epoch	RMSE	MAE	R^2
LSTM	10	1000	11.37	7.58	0.93	Stacked LSTM	10	2000	11.24	7.31	0.92
LSTM	20	1000	11.3	7.45	0.93	Stacked LSTM	20	2000	11.2	7.5	0.91
LSTM	50	1000	11.26	7.46	0.92	Stacked LSTM	50	2000	10.91	7.44	0.9
LSTM	64	1000	11.33	7.6	0.92	Stacked LSTM	64	2000	10.95	7.42	0.91
LSTM	10	2000	11.29	7.5	0.90	CNN-LSTM	10	1000	12.19	8.99	0.9
LSTM	20	2000	11.27	7.66	0.91	CNN-LSTM	20	1000	12.81	9.6	0.87
LSTM	50	2000	11.21	7.54	0.9	CNN-LSTM	50	1000	12.34	8.59	0.78
LSTM	64	2000	11.3	7.63	0.9	CNN-LSTM	64	1000	11.99	8.39	0.83
BiLSTM	10	1000	11.3	7.53	0.92	CNN-LSTM	10	2000	11.81	7.62	0.89
BiLSTM	20	1000	11.26	7.39	0.92	CNN-LSTM	20	2000	11.8	8.33	0.87
BiLSTM	50	1000	11.32	7.4	0.92	CNN-LSTM	50	2000	11.8	8.59	0.85
BiLSTM	64	1000	11.27	7.53	0.92	CNN-LSTM	64	2000	13.25	10.22	0.7
BiLSTM	10	2000	11.23	7.48	0.91	Conv-LSTM	10	1000	11.68	8.06	0.9
BiLSTM	20	2000	11.28	7.58	0.91	Conv-LSTM	20	1000	11.64	7.97	0.9
BiLSTM	50	2000	11.25	7.66	0.89	Conv-LSTM	50	1000	11.53	7.97	0.88
BiLSTM	64	2000	11.27	7.63	0.9	Conv-LSTM	64	1000	11.64	7.96	0.9
Stacked LSTM	10	1000	11.27	7.35	0.93	Conv-LSTM	10	2000	11.48	7.95	0.89
Stacked LSTM	20	1000	11.27	7.41	0.93	Conv-LSTM	20	2000	11.71	8.26	0.88
Stacked LSTM	50	1000	11.19	7.62	0.93	Conv-LSTM	50	2000	12.03	8.62	0.85
Stacked LSTM	64	1000	11.20	7.40	0.92	Conv-LSTM	64	2000	12.49	9.22	0.82

Table 12
Experimental Results for Dataset Split 90:10.

Model name	# Hidden neurons (layer)/Filter	# Epoch	RMSE	MAE	R^2	Model name	# Hidden neurons (layer)/Filter	# Epoch	RMSE	MAE	R^2
LSTM	10	1000	13.25	7.684	0.94	Stacked LSTM	10	2000	13.56	8.06	0.94
LSTM	20	1000	13.38	7.672	0.94	Stacked LSTM	20	2000	13.43	7.8	0.93
LSTM	50	1000	13.4	7.508	0.94	Stacked LSTM	50	2000	10.75	7.69	0.93
LSTM	64	1000	13.34	7.706	0.94	Stacked LSTM	64	2000	10.68	7.73	0.93
LSTM	10	2000	13.27	7.799	0.93	CNN-LSTM	10	1000	14.35	9.34	0.95
LSTM	20	2000	13.39	7.821	0.93	CNN-LSTM	20	1000	13.51	9.64	0.94
LSTM	50	2000	13.32	7.847	0.92	CNN-LSTM	50	1000	14.52	8.85	0.94
LSTM	64	2000	13.31	7.849	0.92	CNN-LSTM	64	1000	13.77	9.59	0.93
BiLSTM	10	1000	13.35	7.546	0.94	CNN-LSTM	10	2000	14.22	8.89	0.92
BiLSTM	20	1000	13.33	7.729	0.93	CNN-LSTM	20	2000	14.3	8.87	0.92
BiLSTM	50	1000	13.31	7.816	0.93	CNN-LSTM	50	2000	13.88	8.87	0.92
BiLSTM	64	1000	13.28	7.871	0.93	CNN-LSTM	64	2000	13.61	8.50	0.94
BiLSTM	10	2000	13.38	7.804	0.92	Conv-LSTM	10	1000	13.81	8.78	0.92
BiLSTM	20	2000	13.47	7.698	0.92	Conv-LSTM	20	1000	13.77	8.69	0.94
BiLSTM	50	2000	13.35	7.947	0.93	Conv-LSTM	50	1000	13.35	8.29	0.92
BiLSTM	64	2000	13.49	8.017	0.91	Conv-LSTM	64	1000	10.96	7.68	0.93
Stacked LSTM	10	1000	13.6	7.9	0.94	Conv-LSTM	10	2000	13.65	8.96	0.94
Stacked LSTM	20	1000	13.49	7.59	0.94	Conv-LSTM	20	2000	13.3	8.84	0.93
Stacked LSTM	50	1000	11.07	8.07	0.94	Conv-LSTM	50	2000	13.12	8.15	0.9
Stacked LSTM	64	1000	10.95	7.81	0.93	Conv-LSTM	64	2000	13.33	8.18	0.93

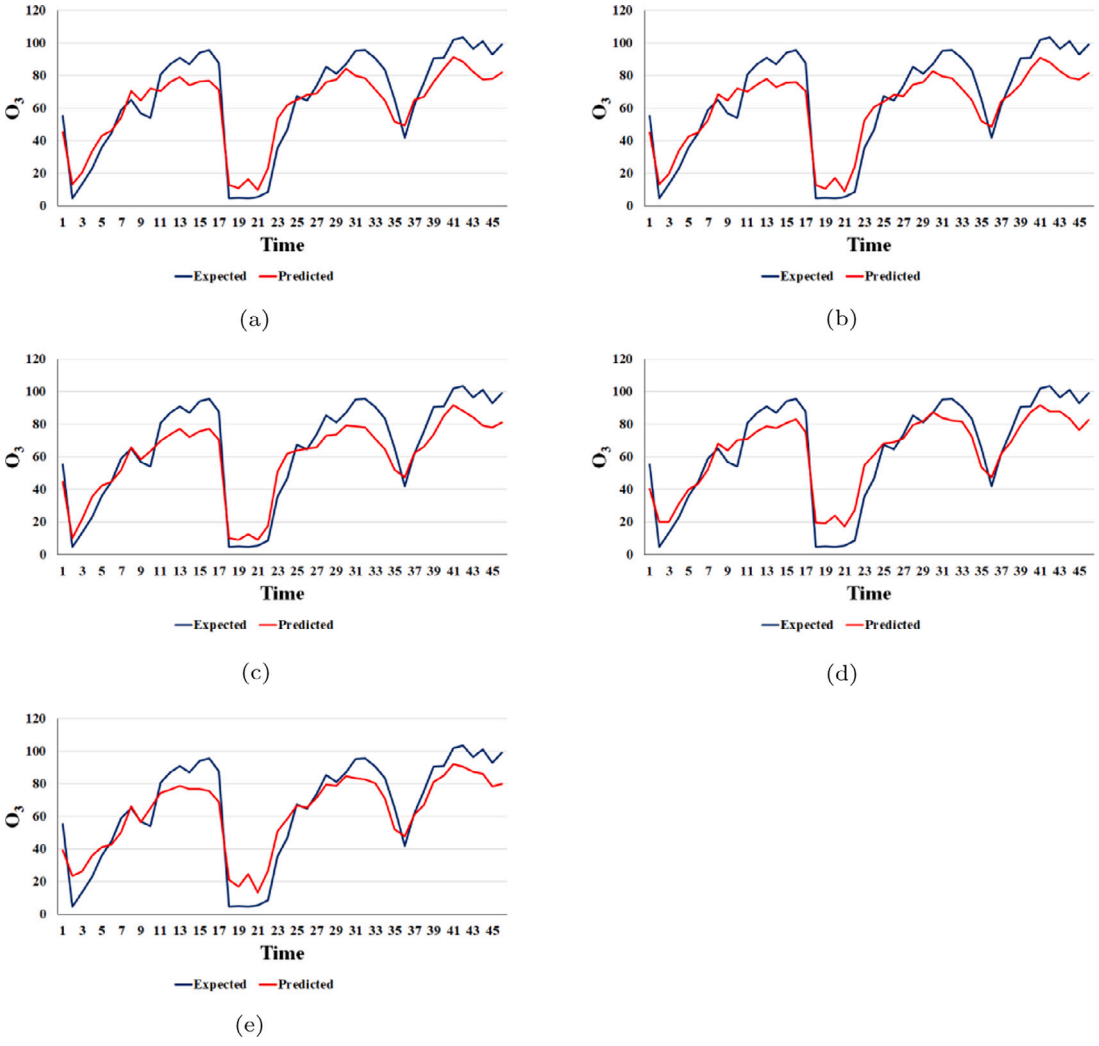

Fig. 10. Comparison between the expected and predicted O_3 values for (a) LSTM (b) BiLSTM (c)Stacked LSTM (d) CNN-LSTM (e) Conv-LSTM.

LSTM model converges better especially in the 18–22 time interval. Therefore, Stacked-LSTM outperforms thanks to this time interval.

5. Conclusion

As cited in the recent literature, ground-level O_3 concentrations have exhibited increasing tendency in many countries during pandemic lock-down period. During this unusual period, concentrations of ozone precursors (NO_x, VOCs, etc.) which are negatively correlated with O_3, declined remarkably especially in developed countries as a result of decreased traffic and industrial activities. Therefore, investigation of ground-level O_3 levels during pandemic period is of concern especially for developed regions.

The major objective of the present study was to model hourly concentrations of tropospheric O_3 for pandemic lock-down period in a prominent industrial area of Turkey, Korfez. With this aim, deep learning approach, which is known to be superior in modeling complex non-linear systems, was applied. During modeling studies, pollutant parameters of $PM10$, SO_2, NO, NO_2, O_3 and meteorologic factors of temperature, wind speed and relative humidity were attained as input data to forecast the next hour's O_3 levels. Among the tested deep learning methods (LSTM, BiLSTM, Stacked LSTM, CNN-LSTM and Conv-LSTM) the Stacked LSTM has the most powerful modeling capability to classify the present data. This is because in the Stacked-LSTM model the hidden layers of the classic LSTM are stacked and the network becomes deeper and the success increases in parallel with the deepening of the network. The effect of this was seen on RMSE values. Stacked-LSTM had two hidden layers with 64 neurons each running 2000 epochs predicted the most successful method with 10.68 RMSE value for data split 90:10.

Statistical models are useful tools of monitoring and improving air quality in the industrialized sites especially for such unusual conditions. Results of this study may be informative to forecast tropospheric O_3 levels under conditions of declining precursor levels.

CRediT authorship contribution statement

Ekin Ekinci: Conceptualization, Methodology, Software, Writing – original draft. **Sevinç İlhan Omurca:** Conceptualization, Methodology, Supervision, Writing – review & editing. **Bilge Özbay:** Conceptualization, Methodology, Supervision, Writing – review & editing.

References

Abdul-Wahab, S.A., Bakheit, C.S., Al-Alawi, S.M., 2005. Principal component and multiple regression analysis in modelling of ground-level ozone and factors affecting its concentrations. Environ. Model. Softw. 20 (10), 1263–1271.

Agiakloglou, C., Tsimpanos, A., 2021. Evaluating information criteria for selecting spatial processes. Ann. Reg. Sci. 66 (3), 677–697.

Akaike, H., 1998. Information theory and an extension of the maximum likelihood principle. In: Selected Papers of Hirotugu Akaike. Springer, pp. 199–213.

Alvim-Ferraz, M., Sousa, S., Pereira, M., Martins, F., 2006. Contribution of anthropogenic pollutants to the increase of tropospheric ozone levels in the Oporto Metropolitan Area, Portugal since the 19th century. Environ. Pollut. 140 (3), 516–524.

Carbajal-Hernández, J.J., Sánchez-Fernández, L.P., Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F., 2012. Assessment and prediction of air quality using fuzzy logic and autoregressive models. Atmos. Environ. 60, 37–50.

Chaloulakou, A., Grivas, G., Spyrellis, N., 2003. Neural network and multiple regression models for PM10 prediction in athens: a comparative assessment. J. Air Waste Manage. Assoc. 53 (10), 1183–1190.

Cheng, C.-H., Huang, S.-F., Teoh, H.-J., 2011. Predicting daily ozone concentration maxima using fuzzy time series based on a two-stage linguistic partition method. Comput. Math. Appl. 62 (4), 2016–2028.

Coman, A., Ionescu, A., Candau, Y., 2008. Hourly ozone prediction for a 24-h horizon using neural networks. Environ. Model. Softw. 23 (12), 1407–1421.

Dantas, G., Siciliano, B., França, B.B., da Silva, C.M., Arbilla, G., 2020. The impact of COVID-19 partial lockdown on the air quality of the city of Rio de Janeiro, Brazil. Sci. Total Environ. 729, 139085.

Draxler, R.R., 2000. Meteorological factors of ozone predictability at houston, texas. J. Air Waste Manage. Assoc. 50 (2), 259–271.

Gao, M., Yin, L., Ning, J., 2018. Artificial neural network model for ozone concentration estimation and Monte Carlo analysis. Atmos. Environ. 184, 129–139.

Gradišar, D., Grašič, B., Božnar, M.Z., Mlakar, P., Kocijan, J., 2016. Improving of local ozone forecasting by integrated models. Environ. Sci. Pollut. Res. 23 (18), 18439–18450.

Graves, A., Schmidhuber, J., 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural Netw. 18 (5–6), 602–610.

Hadjiiski, L., Hopke, P., 2000. Application of artificial neural networks to modeling and prediction of ambient ozone concentrations. J. Air Waste Manage. Assoc. 50 (5), 894–901.

Hashim, B.M., Al-Naseri, S.K., Al-Maliki, A., Al-Ansari, N., 2020. Impact of COVID-19 lockdown on NO2, O3, PM2. 5 and PM10 concentrations and assessing air quality changes in Baghdad, Iraq. Sci. Total Environ. 754, 141978.

Hayes, F., Mills, G., Jones, L., Ashmore, M., 2010. Does a simulated upland grassland community respond to increasing background, peak or accumulated exposure of ozone? Atmos. Environ. 44 (34), 4155–4164.

Kalenderski, S., Steyn, D.G., 2011. Mixed deterministic statistical modelling of regional ozone air pollution. Environmetrics 22 (4), 572–586.

Kerimray, A., Baimatova, N., Ibragimova, O.P., Bukenov, B., Kenessov, B., Plotitsyn, P., Karaca, F., 2020. Assessing air quality changes in large cities during COVID-19 lockdowns: The impacts of traffic-free urban conditions in Almaty, Kazakhstan. Sci. Total Environ. 139179.

Kovač-Andrić, E., Brana, J., Gvozdić, V., 2009. Impact of meteorological factors on ozone concentrations modelled by time series analysis and multivariate statistical methods. Ecol. Inform. 4 (2), 117–122.

Li, X., Peng, L., Hu, Y., Shao, J., Chi, T., 2016. Deep learning architecture for air quality predictions. Environ. Sci. Pollut. Res. 23 (22), 22408–22417.

Lin, Y., Cobourn, W.G., 2007. Fuzzy system models combined with nonlinear regression for daily ground-level ozone predictions. Atmos. Environ. 41 (16), 3502–3513.

Lv, B., Cobourn, W.G., Bai, Y., 2016. Development of nonlinear empirical models to forecast daily PM2. 5 and ozone levels in three large Chinese cities. Atmos. Environ. 147, 209–223.

Mahato, S., Pal, S., Ghosh, K.G., 2020. Effect of lockdown amid COVID-19 pandemic on air quality of the megacity Delhi, India. Sci. Total Environ. 139086.

Misra, D., 2019. Mish: A self regularized non-monotonic neural activation function. ArXiv Preprint ArXiv:1908.08681.

Özbay, B., Keskin, G.A., Doğruparmak, c.c., Ayberk, S., 2011. Predicting tropospheric ozone concentrations in different temporal scales by using multilayer perceptron models. Ecol. Inform. 6 (3–4), 242–247.

Pastor-Bárcenas, O., Soria-Olivas, E., Martín-Guerrero, J.D., Camps-Valls, G., Carrasco-Rodríguez, J.L., del Valle-Tascón, S., 2005. Unbiased sensitivity analysis and pruning techniques in neural networks for surface ozone modelling. Ecol. Model. 182 (2), 149–158.

Sekar, C., Ojha, C., Gurjar, B., Goyal, M.K., 2016. Modeling and prediction of hourly ambient ozone (o3) and oxides of nitrogen (NOx) concentrations using artificial neural network and decision tree algorithms for an urban intersection in India. J. Hazard. Toxic Radioact. Waste 20 (4), A4015001.

Sharma, S., Sharma, P., Khare, M., 2017. Photo-chemical transport modelling of tropospheric ozone: A review. Atmos. Environ. 159, 34–54.

Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K., Woo, W.-c., 2015. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. Adv. Neural Inf. Process. Syst. 28, 802–810.

Tobías, A., Carnerero, C., Reche, C., Massagué, J., Via, M., Minguillón, M.C., Alastuey, A., Querol, X., 2020. Changes in air quality during the lockdown in Barcelona (Spain) one month into the SARS-CoV-2 epidemic. Sci. Total Environ. 138540.

Wang, H.-W., Li, X.-B., Wang, D., Zhao, J., Peng, Z.-R., et al., 2020. Regional prediction of ground-level ozone using a hybrid sequence-to-sequence deep learning approach. J. Cleaner Prod. 253, 119841.

Xiong, B., Newton, S., Li, V., Skitmore, M., Xia, B., 2019. Hybrid approach to reducing estimating overfitting and collinearity. Eng. Constr. Archit. Manag..

Zhan, Y., Luo, Y., Deng, X., Grieneisen, M.L., Zhang, M., Di, B., 2018. Spatiotemporal prediction of daily ambient ozone levels across China using random forest for human exposure assessment. Environ. Pollut. 233, 464–473.

Zhang, Y., Bocquet, M., Mallet, V., Seigneur, C., Baklanov, A., 2012. Real-time air quality forecasting, part I: History, techniques, and current status. Atmos. Environ. 60, 632–655.

Zhang, C., Yan, J., Li, C., Rui, X., Liu, L., Bie, R., 2016. On estimating air pollution from photos using convolutional neural network. In: Proceedings of the 24th ACM International Conference on Multimedia, pp. 297–301.