



Air quality prediction using CNN+LSTM-based hybrid deep learning architecture

Aysenur Gilik¹ · Arif Selcuk Ogrenç¹ · Atilla Ozmen¹

Received: 30 April 2021 / Accepted: 25 August 2021 / Published online: 23 September 2021
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2021

Abstract

Air pollution prediction based on variables in environmental monitoring data gains further importance with increasing concerns about climate change and the sustainability of cities. Modeling of the complex relationships between these variables by sophisticated methods in machine learning is a promising field. The objectives of this work are to develop a supervised model for the prediction of air pollution by using real sensor data and to transfer the model between cities. The combination of a convolutional neural network and a long short-term memory deep neural network model was proposed to predict the concentration of air pollutants in multiple locations of a city by using spatial-temporal relationships. Two approaches have been adopted: the univariate model contains the information of one pollutant whereas the multivariate model contains the information of all pollutants and meteorology data for prediction. The study was carried out for different pollutants which are in the publicly available data of the cities of Barcelona, Kocaeli, and İstanbul. The hyperparameters of the model (filter, frame, and batch sizes; number of convolutional/LSTM layers and hidden units; learning rate; and parameters for sample selection, pooling, and validation) were tuned to determine the architecture that achieved the lowest test error. The proposed model improved the prediction performance (measured by the root mean square error) by 11–53% for particulate matter, 20–31% for ozone, 9–47% for nitrogenoxides, and 18–46% for sulfur dioxide with respect to the 1-hidden layer long short-term memory networks utilized in the literature. The multivariate model without using meteorological data revealed the best results. Regarding transfer learning, the network weights were transferred from the source city to the target city. The model has more accurate prediction performance with the transfer of the network from Kocaeli to İstanbul as those neighbor cities have similar air pollution and meteorological characteristics.

Keywords Deep learning · Air pollution · Prediction · Convolutional neural network · Long short-term memory · Transfer learning

Introduction

Since air pollution is shown to be one of the significant threats to the health of society and the ecosystem, governments have considered clean air as a basic requirement

Responsible Editor: Marcus Schulz

Aysenur Gilik
aysenur.gilik@khas.edu.tr

Arif Selcuk Ogrenç
ogrenç@khas.edu.tr

Atilla Ozmen
aozmen@khas.edu.tr

of health and well-being. Air pollution is one of the most important reasons for serious illnesses causing early deaths, such as heart and lung diseases, stroke, lung cancer (Scovronick 2015; European Environment Agency 2019; WHO 2019). Air pollution adversely damages natural ecosystems and biodiversity by damaging the quality of water and soil that support the ecosystem (European Environment Agency 2019). Air pollution also has negative economic impacts such as a decrease in the lifetime of individuals, an increase in medical costs, and a decrease in productivity. The people living in urban areas are the residents who need better living conditions in terms of environment, transportation, health, and so on. Authorities set air quality standards as an important component of environmental policies and risk management to protect the public health and quality of life. Many countries have taken important steps towards developing smart cities (Santander, Barcelona, Singapore)

¹ Electrical & Electronics Engineering Department,
Kadir Has University, Fatih, İstanbul, Turkey

(Sánchez et al. 2013; Kloeckl et al. 2012; Bakici et al. 2012). European Commission offers EU-wide targets and policy objectives to reduce these impacts for the period from 2021 to 2030, and enable the EU to implement its commitments under the Paris Agreement (European Commission 2017).

Recent works about air quality monitoring and prediction point out the requirements for smart city air quality applications (Park et al. 2017). Air pollution depends not only on the concentration of pollutants (Bashir Shaban et al. 2016; Zhang et al. 2019) but also on the meteorological conditions such as temperature, wind direction, and wind strength (Soh et al. 2018; Chu et al. 2012; Tai et al. 2010). The information about air quality and meteorology gathered from various types of sensors must be combined in a way to achieve a useful representation of real-world environments. Also, changes in city conditions and people's behavior over time negatively affect the performance of the model obtained with old data. Since each city data have a different marginal probability distribution, training and optimizing the model at regular intervals with current data are necessary for an up-to-date prediction system (Zanella et al. 2014).

This study aimed to solve the challenges by developing a CNN+LSTM (convolutional neural network + long short-term memory) neural network model for hourly air pollution prediction. The historical information from various locations in a city is processed to predict the concentration of particles in the same locations in the next hour. In addition, the proposed model is supported by the transfer learning method to obtain a deep learning structure with the optimal set of hyperparameter values to use in various other cities.

CNN extracts the relationship between different features with spatial dependency in sensor locations. LSTM extracts temporal information characteristics from time series data and it consolidates the nonlinear relationship between air pollutants. Various sensor data that are collected at different locations in a city are processed to predict not only the local pollution affecting a small area but also the propagation of pollution factors affecting a wider precinct over time.

Artificial intelligence and machine learning-based techniques consist of computational methods which improve the learning performance of the machine in solving nonlinear problems from complex data. Therefore, these techniques are commonly used for modeling, search, and optimization in various smart city systems (Soh et al. 2018; Khan et al. 2018; Chen et al. 2019; Qin et al. 2019; Tao et al. 2019; Ma et al. 2019). This has led to a tendency of using various machine learning methods in recent air quality studies (Soh et al. 2018; Qin et al. 2019; Tao et al. 2019; Ma et al. 2019; Qi et al. 2018). Deep learning is a multilayer neural network-based machine learning method; it learns data representations with multiple levels of abstraction from

complex structures. Also, deep learning models can consist of more than one machine learning algorithm. As for another advantage of the model, CNN and LSTM work based on parameter sharing to reduce the number and the complexity of the parameters that the model has to learn.

Transfer learning focuses on extracting information from a data set and reapplying it in another data set that has a different distribution from the first one. It provides the transfer of the knowledge/model from a source domain to a target domain. In this study, the proposed deep neural network is supported by transfer learning to transfer weights/features to improve the prediction performance in different cities.

European Innovation Partnership on Smart Cities and Communities supported by the European Commission aims to bring sustainable solutions to city problems in different fields such as energy, transportation, and communication (EU 2021). Especially in developing countries, the health and environmental conditions of big cities are at risk due to transportation and energy sources. The main aim for such cities is to reduce the emission in a certain region and control the air pollution that may become a threat. Intelligent air quality monitoring and prediction systems play a major role in building sustainable and clean cities. This kind of work raises awareness for a clean environment; residents of the city can learn about environmental information, services and precautions instantly. Existing studies are generally limited to monitoring, and there is no application available to integrate large amount of collected data for a consistent prediction model.

The data used by the proposed model include information about both pollutant concentrations and meteorological conditions. The advantage of artificial intelligence (AI) is its capability to extract relevant features from input data automatically to learn the correlation between relevant features. CNN is able to generalize and learn by reducing the size and complexity of data, LSTM has a memory to connect the previous information to the current and generate output according to the sequence of inputs. Another advantage is that AI is capable of running different types of machine learning algorithms together to improve the learning performance of air pollution prediction. Thus, the model can process relevant spatial-temporal features simultaneously where the spatial features are about the location of sensors and temporal features are based on the hourly concentration of the pollutants. One of the important challenges for air pollution prediction studies is to design a model and obtain the optimal hyperparameters that have high prediction accuracy for many regions/cities. Transfer learning enables the model that successfully performs a specific task on the source domain to be used on target data for the same task or for a related task.

The remainder of the paper is organized as follows: the “[Literature survey](#)” section describes prior work in this area, the “[Methodology](#)” section explains proposed air pollution prediction method, the “[Implementation](#)” section describes the implementation, the “[Results](#)” section presents the results of simulation studies investigating the performance, finally, the “[Conclusion and discussion](#)” section concludes the study and evaluates the results.

Literature survey

The studies in the literature consider the effects of spatial and temporal features of prediction performance and compare several machine learning algorithms (Bashir Shaban et al. 2016; Tao et al. 2019; Soh et al. 2018; Zhao et al. 2019; Djalalova et al. 2015; Zeinalnezhad et al. 2020; Schürholz et al. 2020; Qin et al. 2019; Ma et al. 2019; Aceves-Fernandez et al. 2020; Di Antonio et al. 2019; Zhou et al. 2018; Lv et al. 2019; Wei Y et al. 2016; Ma et al. 2019; Li et al. 2016; Zhang et al. 2021; Zhao et al. 2020). Summary of machine learning-based air quality studies in the literature is given in Table 1.

In Tao et al. (2019), the prediction model is designed with 1D-CNN (1-Dimensional CNN) and Bi-GRU (Bi-directional Gated Recurrent Units). They use the UCI machine learning repository database for $PM_{2.5}$ data of Beijing. The model is trained and tested employing different combinations of various features. The best performance is obtained with inputs of pollution, temperature, wind speed, wind direction, and the dew point. Adding other features to the input increases the complexity and causes difficulty in learning. The study concludes that variants of RNN (recurrent neural networks) such as LSTM and GRU have better performance than RNN.

In Soh et al. (2018), a spatial-temporal deep neural network (ST-DNN) model is proposed to predict air quality. They use historical data including temporal information, the concentration of pollutants ($PM_{2.5}$ and PM_{10}), and meteorological characteristics as input; the output is the future $PM_{2.5}$ concentration. Two real-world data sets (76 locations in 23 cities in Taiwan and locations from Beijing) are used in experiments. While the convolutional layer can extract the temporal delay factor from surrounding features by learning spatial information, the proposed model provides a long time frame prediction by LSTM.

In Zeinalnezhad et al. (2020), a regression technique by ANFIS modeling is applied to conduct a time series prediction for air pollution. Input contains the information of O_3 , SO_2 , CO , NO_2 with 24-h time series length. The input layer takes five input variables which represent the contamination rate for consecutive 5 days. The output layer gives a single output variable which is the predicted

contamination rate for the next day. According to the comparison between errors related to different pollutants, the ANFIS model designed for NO_2 has the lowest error for short-term contamination prediction.

In Qin et al. (2019), an air quality prediction model is built by combining CNN and LSTM. The input features contain the concentration of pollutants and meteorological conditions; the output is the predicted sequence of $PM_{2.5}$ for Shanghai. According to the comparison between the three models (CNN, LSTM, CNN+LSTM), the performance of the CNN+LSTM model is the best for long-term sequence prediction. CNN and RNN have the same RMSE value where CNN+LSTM reduces RMSE by 41.45% compared to CNN and RNN.

In Di Antonio et al. (2019), the performance comparison is done between univariate and multivariate prediction for PM_{10} with LSTM. The PM_{10} concentration data set contains time series measurements from January 2009 to December 2017 in Pescara, Italy. An increase in the number of features causes a decrease in prediction performance because it increases the complexity of the network.

In Zhou et al. (2018), a multi-output LSTM network-based deep learning model built with three deep learning algorithms is proposed to make regional multi-step-ahead prediction (horizons $t + 1$ up to $t + 4$) for variables $PM_{2.5}$, PM_{10} , and NO_X in Taipei City, China, by using the air quality data and meteorological data. The proposed Deep Multi-output LSTM model has smaller RMSE (with improvement rates ranged from 2.18% to 13.91%) than the Selective Multimodal LSTM model.

In Li et al. (2016), a deep learning-based air quality method is proposed for hourly $PM_{2.5}$ concentration. The data set is from 12 monitoring stations in Beijing between the 1st of January 2014 and the 28th of May 2015. A stacked autoencoder model is used to extract representative spatiotemporal features for air pollution data. The study shows that deep learning outperforms the traditional time series models; it is able to predict the air quality with seasonal stability at all monitoring stations.

In Zhang et al. (2021), the cutting-edge bidirectional long short-term memory neural network (BiLSTM) is used for time series air quality prediction. The real-world air quality data set consists of hourly $PM_{2.5}$ concentration collected from 12 observation points in Beijing, China, during 2013–2017. In terms of prediction error, the BiLSTM model outperforms the existing models.

In Zhao et al. (2020), forward neural networks are combined with recurrent neural networks to create a hybrid model (called CERL) in order to improve the prediction accuracy of air quality in two capital cities Xi'an and Lanzhou. Each sample includes the date, time, and 6 air pollutant factors (AQI, the concentration of $PM_{2.5}$, PM_{10} , CO , SO_2 , NO_2 , and O_3). The performance of CERL model

Table 1 Proposed architectures and their performances in literature

Reference	Features	Output	Architecture	Performance
Bashir Shaban et al. (2016)	Concentration of target pollutant	Concentration of target pollutant	ANN	RMSE(O_3)=26.4
	Concentration of target pollutant and meteorological data	Concentration of target pollutant	ANN	RMSE(NO_2)=25.7 RMSE(SO_2)=74.1 RMSE(O_3)=11.6
Tao et al. (2019)	Concentration of $PM_{2.5}$ and meteorological data	Concentration of $PM_{2.5}$	RNN	RMSE=20.94
			LSTM	RMSE=17.31
Soh et al. (2018)	Concentration of pollutants and meteorological data	Concentration of $PM_{2.5}$	ST_DNN with ANN, LSTM and CNN	MAE=3.40
Zeinalnezhad et al. (2020)	contamination rate of target pollutant for consecutive 5 days	the predicted contamination of target pollutant rate for the next day	ANFIS	RMSE(SO_2)=0.25
			Semi-experimental Regression	RMSE(O_3)=0.20 RMSE(NO_2)=0.16 RMSE(SO_2)=0.27
Qin et al. (2019)	Concentration of air pollutants and meteorological data	Concentration of $PM_{2.5}$	CNN	RMSE=30.66
			LSTM	RMSE=17.95
Ma et al. (2019)	Concentration of $PM_{2.5}$	Concentration of $PM_{2.5}$	ANN	RMSE=14.30
			LSTM	RMSE=11.20
Di Antonio et al. (2019)	Concentration of PM_{10} and meteorological data	Concentration of PM_{10}	CNN+LSTM	RMSE=8.98
			LSTM	RMSE=8.40
Zhou et al. (2018)	Concentration of target pollutant and meteorological data	Concentration of target pollutants	SM-LSTM	RMSE($PM_{2.5}$)=4.59
			DM-LSTM	RMSE(PM_{10})=5.55 RMSE(NO_X)=6.75 RMSE($PM_{2.5}$)=4.49 RMSE(PM_{10})=5.43 RMSE(NO_X)=6.77
Li et al. (2016)	Concentration of $PM_{2.5}$	Concentration of $PM_{2.5}$	STDL	RMSE=14.96
Zhang et al. (2021)	Concentration of $PM_{2.5}$	Concentration of $PM_{2.5}$	VMD-BiLSTM	RMSE=5.12
Zhao et al. (2020)	Concentration of target pollutants	Concentration of target pollutants	CERL	RMSE($PM_{2.5}$)=1.21
				RMSE(PM_{10})=3.04 RMSE(SO_2)=2.05 RMSE(NO_2)=2.33 RMSE(O_3)=2.48 RMSE(CO)=0.08

is compared with CFNN, ESN RNN, and Pyrenn in terms of RMSE. It is observed that the proposed model improves the 1-h step prediction performance for only $PM_{2.5}$.

In Al-Janabi et al. (2019), an intelligent predictor called smart air quality prediction model (SAQPM) is proposed for the prediction of the air pollutant concentrations $PM_{2.5}$, PM_{10} , NO_2 , CO , O_3 , SO_2 over the next 2 days. The model is based on LSTM. Particle swarm optimization algorithm is used to determine the optimal structure of the proposed method including weights, bias, number of hidden layers, number of nodes in each hidden layer, and activation function.

In Al-Janabi et al. (2021), a real time intelligent programmable forecaster system is designed which is capable of predicting the pollutant concentrations within the next 48 h. The proposed model consists of two parts. In the first part, air pollution data including $PM_{2.5}$, PM_{10} , NO_2 , CO , O_3 , SO_2 are collected and stored. In the second part, concentrations of pollutants are predicted for the next 48 h using LSTM neural network.

Regarding studies using transfer learning for air quality, in Lv et al. (2019), the transfer learning model is proposed to transfer features from an urban area to a non-urban area. Three types of features (terrain, spatial, and temporal) are used as inputs. Real air quality data (Air Quality Index (AQI), PM_{10} , $PM_{2.5}$, SO_2 , NO_2 , CO, O_3) and meteorology data (weather, temperature, wind speed, humidity) are collected hourly from three cities (Hangzhou, Ningbo, Wuxi) in China. As a result, adding terrain features in the learning process does not make any significant improvement in prediction for urban areas.

In Ma et al. (2019), bidirectional long short-term memory (BLSTM) network and transfer learning are used to predict air quality and their performance is compared with other commonly seen models. A case study is conducted in Guangdong, China, the data are collected from all the monitoring stations in the city for three years for the hourly $PM_{2.5}$ concentration. Transfer learning improves the prediction accuracy of BLSTM at larger temporal resolutions by up to 40%.

Table 2 Explanation of the methods according to the input-output type

Method	Input structure	Input	Output
UNI/UNI	2-dimensional array	Historical information of target pollutant (univariate input)	Predicted value for the concentration of target pollutant (univariate output)
MULTI/UNI	3-dimensional array	Historical information of all pollutants and meteorological data (multivariate input)	Predicted value for the concentration of target pollutant (univariate output)
MULTI/MULTI	3-dimensional array	Historical information of all pollutants and meteorological data (multivariate input)	Predicted values for the concentration of all pollutants (multivariate output)

Methodology

The first objective of this study is to predict the hourly concentration of air pollutants in multiple locations in a city effectively. The second objective is to transfer the model between cities that have different marginal probability distributions for air quality and meteorological data. For data preprocessing, all duplicate data are cleaned in data sets, and linear interpolation is applied to fill in missing values. Then the data are normalized to scale the values into a specific range to reduce the data redundancy/bias effect on learning, which is caused by the wide variation of the value range in the data (Eessaar 2016).

The steps of the study can be summarized as follows.

1. The data are preprocessed; bad data are removed, missing data are corrected by linear interpolation, normalization is applied.
2. Data are separated for training, validation, and test. Three types of sample selection methods are identified.
3. Time series data structure is created with the rolling-window method.
4. Two types of input structure are defined for the neural network.
5. Hybrid CNN+LSTM deep neural network is created.
6. Hyperparameter tuning is applied according to validation and test performances.
7. The results are compared with the previous works in the literature.
8. A 1-hidden layer LSTM network is run on the same data sets and their performances are compared.
9. Transfer learning method is applied for weight transfer between different cities.

The study can be split into three parts according to the input-output relationship (Table 2).

Two types of input structure are defined for the neural network: 2D and 3D array; these inputs hold the spatial and temporal features together. 2D input consists of the historical information about a target pollutant at sensor locations; 3D input consists of both historical information

about the target pollutant and other pollutants related to air quality. In this way, the model can use the interaction between different variables on pollutants in neighboring locations. The summary of data properties used for the methods is given in Table 3.

The first step of the machine learning model design is to determine the layers and their properties. Then, the optimal architecture and hyperparameters are determined by training-testing with different activation functions. Rectified Linear Unit (ReLU) and sigmoid activation functions are used to search for performance improvement. After the optimal model and hyperparameter values are determined, the neural network is transferred from the source city to the target city by changing the last learning layers, which is the fully connected layer in the proposed model.

Data

Barcelona Air Quality Data: Data for air quality in Barcelona is taken from The Open Data BCN that is a service of the Municipal Data Office (Barcelona City Council 2020). Real-time hourly measurement is made for three pollutants (O_3 (tropospheric ozone), NO_2 (Nitrogen dioxide), PM_{10} (Suspended particles)) by the stations throughout Catalonia from 06/13/2018 to 01/31/2019. Data are generated by those three sensors deployed in 7 different locations (Ciutadella, Eixample, Gracia, Palau Reial, Poblenou, Sants, Vall Hebron) where the distance between sensor locations in Barcelona is not more than 5 km.

Kocaeli and İstanbul Air Quality Data: Data for air quality and meteorology are provided by Republic of Turkey Ministry of Environment and Urbanization (Republic of Turkey Ministry of Environment and Urbanization 2019). The data contain the information about the concentration of various pollutants (Sulfur Dioxide (SO_2)), Nitrogen Oxides (NO_X), Tropospheric Ozone (O_3), Particulate Matters (PM), etc.) and meteorological conditions (temperature, relative humidity, air pressure, wind speed, and direction) for Kocaeli and İstanbul which are two large cities in Turkey. The hourly measurement data for Kocaeli are dated between 11/14/2017 17.00 and 4/11/2020

23.00 (total of 21,103 h) and for İstanbul data are dated between 01/01/2015 00.00 and 04/11/2020 23.00 (total of 46,272 h).

Sensors are deployed at 14 locations in Kocaeli and at 43 locations in İstanbul. Three locations are chosen for each city (Alikahya, Gebze, and Körfez for Kocaeli; Silivri, Esenyurt, and Sultangazi for İstanbul) in this study. The amount of missing data and the availability of the information for a variable were taken into consideration while selecting these locations. In Kocaeli, the distance between locations are 25 km and 18 km for Gebze-Körfez and Gebze-Alikahya respectively. In İstanbul, the distance between locations are 37 km and 17 km for Silivri-Esenyurt and Esenyurt-Sultangazi respectively.

Time series data set preparation

Rolling-window

“Rolling-window” (Zivot and Wang 2003) method is used to create time series input for neural networks. One sample for time record t is created by using the values $[t_0 - d, t_0]$ as the features of the target value at time step t_0 . Figure 1 displays how the rolling-window method works. d and s represent the frame size and the step size respectively. The values of the past d time record points are taken as features and the value at the time step t_0 is taken as the target value. Then, the frame is slid s steps for the next sample.

Input structure of the CNN

Time series sequences created by the rolling-window model are combined to construct a 2D sample model (Fig. 2a) as the input for convolutional neural network. 2D array in Fig. 2a is input to the model with Method-1 (UNI/UNI) to obtain output of size $m \times 1$ for the predicted values of the pollutant concentration in each location.

The input is rearranged (Fig. 2b) to take advantage of CNN’s ability to process 3D structures. m , n , and k represent the number of locations, the number of the time step, and the number of pollutants. 3D input is used by Method-2 (MULTI/UNI) for the prediction of one pollutant

Table 3 Explanation of the data properties used for the methods

Properties		UNI/UNI	MULTI/UNI	MULTI/MULTI
Input structure	2D input	Yes	No	No
	3D input	No	Yes	Yes
	Non-normalized data	Yes	No	No
Data normalization	Normalized data	Yes	Yes	Yes
Meteorological data		No	Yes	Yes
Transfer learning		No	Yes	Yes

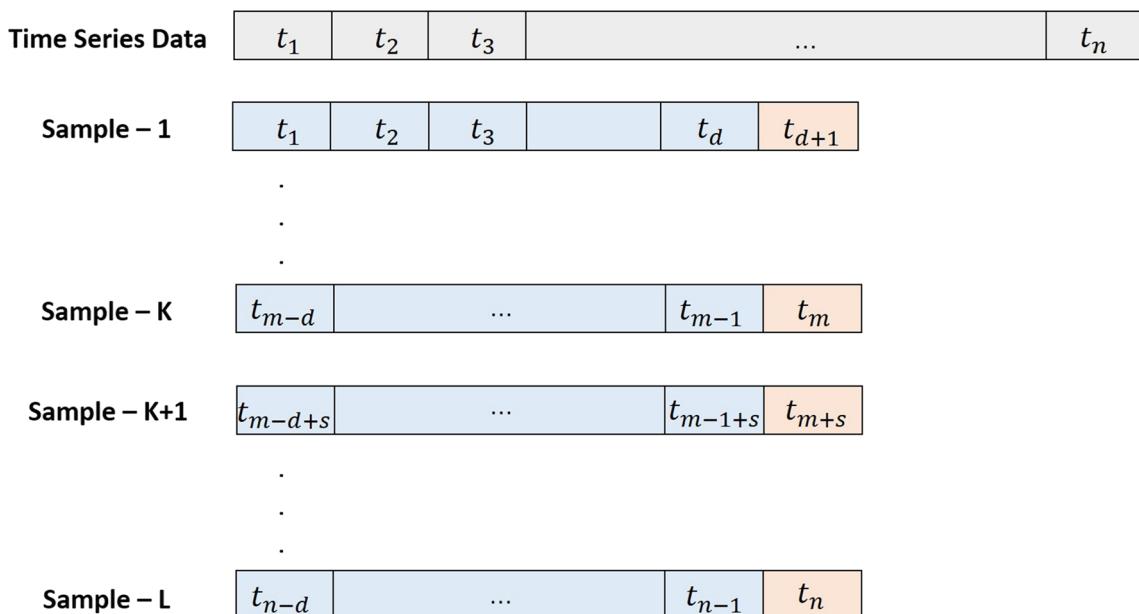


Fig. 1 Rolling window method for the time series data

and Method-3 (MULTI/MULTI) for prediction of multiple pollutants.

Figure 3 illustrates an example of the CNN+LSTM deep neural network where the feature maps obtained by CNN are input to the LSTM.

Convolutional neural network (CNN)

Convolutional neural network is initially used to process image data. A deep convolutional neural network structure is a multilayer perceptron network with more than one hidden layers. A typical CNN consists of convolutional layers and pooling layers. The convolutional layer is a set

of artificial neurons that represent convolutional filters to generate feature maps. Input is split into small blocks to be convolved with a specific set of weights. Different sets of features are obtained by sliding convolutional filters on input with the same weights.

The pooling layer is employed to reduce the spatial size of the input representation and the number of parameters. Similar information in the local region is identified and the dominant response is output.

The activation function is a nonlinear function used for learning complicated nonlinear patterns. In this study, the Rectified Linear Unit (ReLU) and sigmoid are used in learning layers (Nwankpa et al. 2018). The fully connected

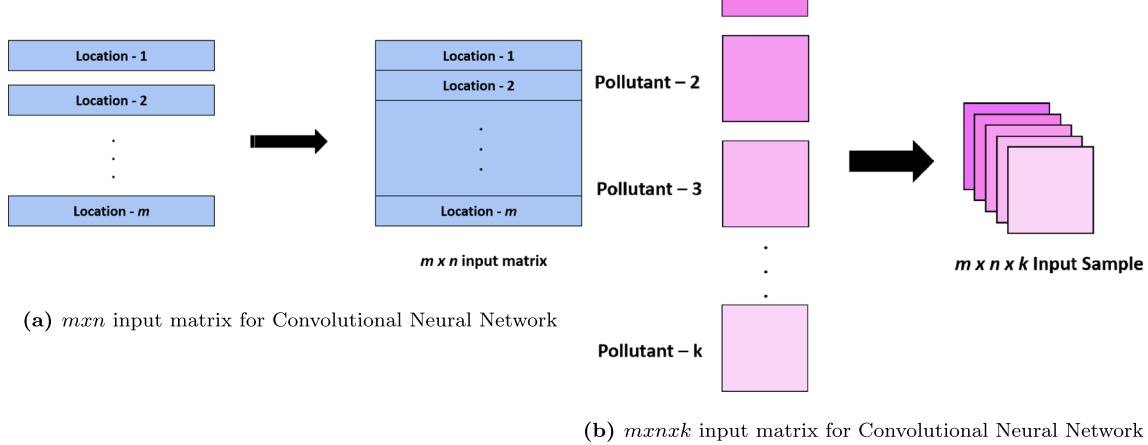


Fig. 2 Input structures for proposed CNN+LSTM neural network

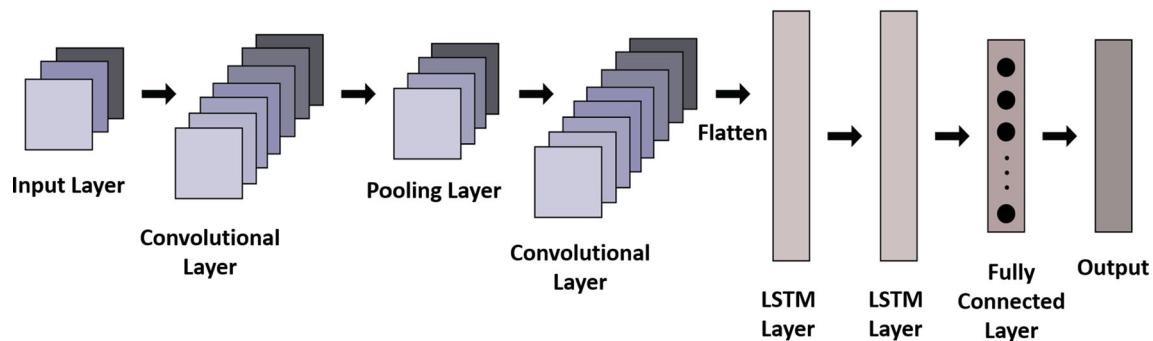


Fig. 3 CNN+LSTM deep learning structure

learning layer is implemented at the end of the neural network, it combines the features by globally analyzing the outputs of previous layers.

Long short-term memory

Recurrent neural networks (RNNs) use previous information with the current information to generate the output depending on the sequence input, the network produces different outputs by the same input. Long short-term memory (LSTM) is a special type of RNNs that is used to capture the long-term dependencies in a sequence data. There are four interacting layers (called as “gate”) in the hidden state of LSTM (Fig. 4) connected in a special way. Information is received from outside and it is stored, written to the memory cell, and read via the gates.

The memory unit controls the flow of information to decide the influence of previous information on output, a

copy of predictions is held on by this unit. The weights and information in memory are multiplied to decide which data and how much of it will be used, some of them are then added back to the prediction. A few predictions are selected as the prediction for that moment, then the information that is not relevant immediately is set apart to block its effect on the predictions that are going forward.

Transfer learning

The representation of transfer learning is given in Fig. 5. Transfer learning is classified into three categories (Pan and Yang 2010): Inductive transfer learning, unsupervised transfer learning, transductive transfer learning. The source task and target task in inductive transfer learning are different. Unsupervised transfer learning is similar to inductive transfer learning, but it focuses on unsupervised learning tasks in the target domain. Source and target tasks

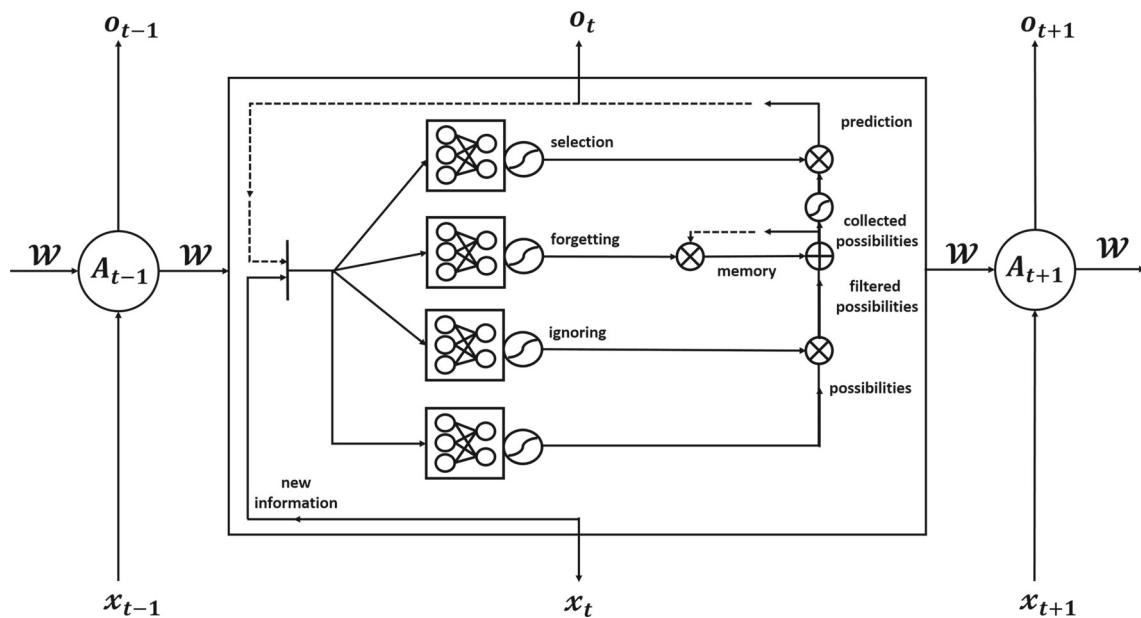


Fig. 4 The structure of the long short-term memory (LSTM) neural network

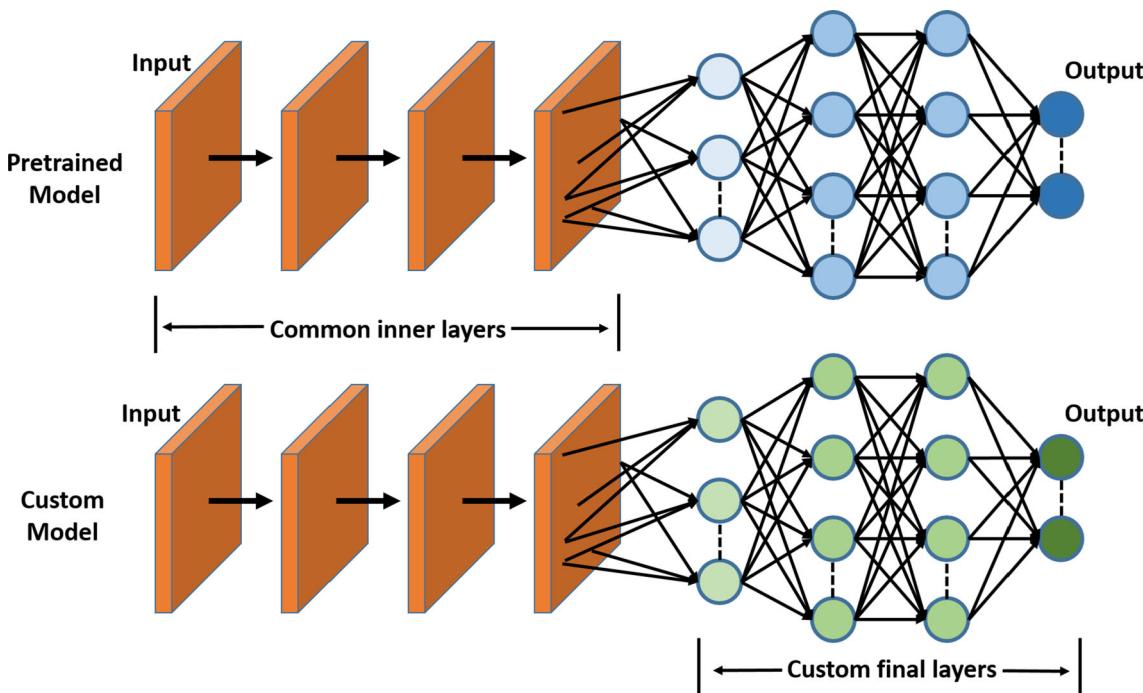


Fig. 5 Representation of transfer learning between two neural network models

in transductive transfer learning are the same, the target domain is different but related to the source domain.

Aim of transfer learning is to improve the learning of the target prediction function, $f_T(\cdot)$, in target domain, D_T , using the knowledge in source domain, D_S , and source task, T_S , where $D_S \neq D_T$ or $T_S \neq T_T$ with the target task given as T_T .

There are two cases in transfer learning according to the relations between source domain and target domain. Firstly, the feature space of the source domain can be different from the target domain, ($\chi_S \neq \chi_T$). Secondly, there can be no difference in source and target domains, but the marginal probability distributions of input data are different, ($\chi_S = \chi_T$, $P_S(X) \neq P_T(X)$). It is called homogeneous transfer learning when $\chi_S = \chi_T$, and heterogeneous transfer learning when $\chi_S \neq \chi_T$ (Weiss et al. 2016).

Implementation

In the initial step of data preprocessing, data were cleaned and interpolated if there were missing values for at most five consecutive time points, then data were normalized by min-max normalization method. Interpolated time series data set was created with various frame sizes and different data separation methods. After 2D and 3D input structures were defined, CNN+LSTM deep learning-based time series prediction model has been implemented by using Deep Network Designer Tool on MATLAB version R2020a.

Experiments were run on a computer with an Intel (R) Core (TM) i3-4005U CPU processor, running Windows version 10. Hyperparameters were tuned to increase the prediction performance of the model. The model was trained on three cities' data to obtain the optimal hyperparameter values. We have no overfit issue as our training includes validation such that the training stops if validation error increases in certain number of training epochs (e.g., 6) in MATLAB. At the end, the pre-trained network, that had been trained with the source city's data, was run with transfer learning on the target city's data where the probability distributions are different from each other. Performance metrics (RMSE and correlation coefficient) were measured to evaluate the prediction performance of the proposed system.

The hyperparameters were optimized during the training and test runs. The model was run 15 times for each method, the neural network structure with the lowest validation RMSE is selected to be used on test data. If the test RMSE value is smaller than the reference level (smaller one of the two values: the results of the studies in the literature or the result of 1-hidden layer LSTM run for comparison in this study), the results are accepted. Otherwise, the neural network was redesigned and trained by changing network properties and hyperparameters (such as hidden layer type and number, neuron number, activation function).

Method-1 (UNI/UNI) was firstly used for non-normalized time series data of Barcelona to predict the actual concentration value of the target pollutant. Then, it was employed for the normalized data to compare the

Table 4 Descriptive statistics of air pollution data in three target cities

Descriptive statistics of air pollution data												
Pollutant	Barcelona				Kocaeli				İstanbul			
	Min	Max	Mean	Standard deviation	Min	Max	Mean	Standard deviation	Min	Max	Mean	Standard deviation
O_3	1	351	42.09	28.70					1	925	41.75	29.28
NO_2	1	197	36.73	23.39					1	272	27.94	19.77
PM_{10}	2	109	21.41	8.83	1	604	41.32	30.14	1	985	56.13	57.39
SO_2					1	312	8.39	14.82	1	220	5.04	5.67
NO_x					1	1950	78.03	106.14	1	272	27.93	19.77

performance with non-normalized data. Then, all three methods (Method-1, Method-2, Method-3) were performed with normalized time series data for each city (Barcelona, Kocaeli, İstanbul).

Meteorology data available in Kocaeli and İstanbul were added to the input with Method-2 (MULTI/UNI) and Method-3 (MULTI/MULTI) because of the need for 3D input structure. Temperature, relative humidity, air pressure, wind speed, and wind direction were used along with the concentration of pollutants.

Deep neural networks that have high prediction accuracy in Barcelona and Kocaeli data were transferred to İstanbul. Also, transfer learning was performed in cases where different pollutants' information exist in the target domain and source domain, for example, between Barcelona and Kocaeli.

Descriptive statistics of pollution data for the three cities (Barcelona, Kocaeli, İstanbul) is given in Table 4 where all pollutant concentrations are given in $\mu\text{g}/\text{m}^3$.

Hyperparameter tuning

Hyperparameters can be related to the model selection tasks (topology and size of the network) or to the optimization and training process (such as learning rate and mini-batch

size). The following set of hyperparameters were tuned in this study:

Frame Size: Frame size was varied as integer values in the interval [8, 15].

Step Size: The step size of the time series sample was taken as 1 in this study.

Data Separation: Data have been split by 70%-15%-15% and 80%-10%-10% for training set, validation set, and test set respectively.

Sample Selection: There are three sample selection methods: random, sequential, and consecutive. The random selection method is that when the training, validation, and test samples are selected randomly from the data. Figure 6 shows the consecutive and sequential selection methods. Sequential sample selection refers to the selection of validation samples sequentially from the training data set by using a validation frequency where the data are split into training and test sets consecutively. Consecutive selection is that the data are split into three pieces for training, validation, and test, respectively.

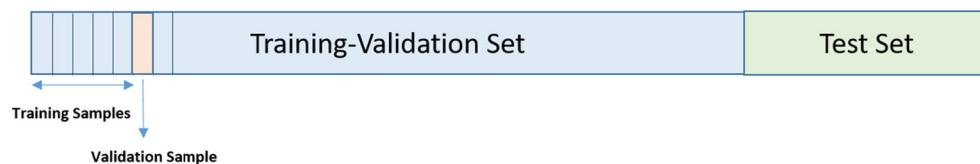
Validation Frequency: Validation frequency refers to the number of iterations between evaluations of validation metrics, it was taken as 10, 15, and 20, respectively. It also refers to the period of selecting the validation sample in the sequential sample selection method.

Fig. 6 Consecutive and sequential sample selection

Consecutive Selection



Sequential Selection



Mini-batch Size: According to the amount of the time series data set, mini-batch size per iteration was taken as 30, 70, 100, 150, and 200 for training progress.

Number of the epoch: The early stopping method was used in this study to determine when to stop training the model.

Number of Convolutional Layer: The CNN part was constructed with one, two, and three convolutional layers.

Pooling Layer: The CNN part was firstly built without a pooling layer, then an average pooling layer, and a max-pooling layer were used.

Filter Size: In the CNN part, the filter size was set as 2×2 and 3×3 .

Number of Filters: In the CNN part, the number of filters was taken as 4, 5, and 10.

Number of LSTM Layers: The LSTM part was constructed with one and two LSTM layers.

Number of Hidden Units in LSTM Layer: The number of hidden units in LSTM layers was taken as 25, 50, 75, 100, 150, and 200.

Activation Function: Rectified Linear Unit (ReLU) and sigmoid were used as activation functions.

Learning Rate: Learning rate was set as 0.001, 0.005, 0.01.

Results

The structure of the proposed CNN+LSTM deep learning prediction model and its properties are given in Table 5 with the following parameters:

- m : number of locations
- n : number of time steps
- k : number of features
- n_k : number of convolutional filters
- n_l : number of LSTM units

- n_o : number of neurons in fully connected layer.

In our study, adding a pooling layer could not improve the success of the model. To the contrary, it caused unstable training performance with high test RMSE values. As the pooling layer reduces the information in the huge input data, it caused loss of the main features of input with smaller size.

The model was run 15 times for all combinations of different hyperparameter values (described above) to predict the concentration of pollutants by each method. The best test results were chosen according to the lowest RMSE values, then, test RMSE and the correlation coefficients were calculated. A learning rate of 0.005 provided the lowest RMSE during the training process. Also, the random sample selection method reached a lower test RMSE value than consecutive and sequential sample selection methods.

Table 6 gives the data properties and methods for the best prediction results for different cities. Test RMSE values were used as the prediction performance metric, and the best prediction performance was obtained for PM_{10} for all three cities. 3D samples created from the normalized time series data were used as input to the model. For Kocaeli, Method-3 (multi-input/multi-output) provided the lowest test RMSE while Method-2 (multi-input/uni-output) provided lowest RMSE for İstanbul. For Barcelona, both methods reached the lowest test RMSE for PM_{10} prediction (Table 7).

Proposed architectures and their performances in the literature are given in Table 8. In Zeinalnezhad et al. (2020), the proposed ANFIS model reached RMSE values ($\mu\text{g}/\text{m}^3$) of 0.25 for SO_2 , 0.20 for O_3 , and 0.16 for NO_2 . The architecture proposed in this paper reached lower RMSE values for these pollutants in three cities than the study in Zeinalnezhad et al. (2020). Results of Method-2 and Method-3 based on multivariate input, which is also called “multivariate prediction” (Di Antonio et al. 2019), were

Table 5 Proposed deep learning model’s layers and their properties

Layer number	Layer type	Properties
1	Sequence input	Sequence input with $mxnxk$ dimensions
2	Convolution	n_k 2x2 convolutions with stride [1 1] and padding ‘same’
3	Activation	ReLU
4	Convolution	n_k 2x2 convolutions with stride [1 1] and padding ‘same’
5	Activation	ReLU
6	Flatten	Flatten
7	LSTM	LSTM with n_l hidden units
8	Activation	ReLU
9	Fully connected	n_o fully connected layer
10	Activation	ReLU, Sigmoid
11	Regression output	Linear Unit

Table 6 Data properties and methods that give the best prediction results

Properties	Barcelona	Kocaeli	İstanbul
Pollutant	PM_{10}	PM_{10}	PM_{10}
Test RMSE $\mu g/m^3$	0.07	0.07	0.07
Method	Method-2 & Method-3	Method-3	Method-2
Input structure	3-dimensional array	3-dimensional array	3-dimensional array
Data normalization	Yes	Yes	Yes
Meteorological data	-	No	Yes
Features	Historical hourly concentration of pollutants (O_3 , NO_2 , PM_{10})	Historical hourly concentration of pollutants (PM_{10} , SO_2 , NO_X)	Historical hourly concentration of pollutants (PM_{10} , SO_2 , NO_X) and meteorological conditions
Input type	Multivariate	Multivariate	Multivariate
Output type	Univariate & multivariate	Multivariate	Univariate

compared with other studies in the literature (Soh et al. 2018; Di Antonio et al. 2019). The RMSE values in related studies are computed by normalization for comparison purposes. The RMSE for ANN-based prediction model in Soh et al. (2018) was measured as 0.47 for O_3 and 0.11 for SO_2 by the prediction with univariate input; 0.21 for O_3 , and 0.15 for SO_2 by the prediction with multivariate input. The RMSE of our CNN+LSTM neural network is lower for O_3 in Barcelona and İstanbul, and for SO_2 in Kocaeli and İstanbul than the RMSE of the study in Soh et al. (2018). The study in Di Antonio et al. (2019) measured RMSE as 0.08 with both univariate and multivariate input. In our proposed model, the RMSE was measured as 0.07 in Barcelona and Kocaeli while it was 0.08 in İstanbul.

In Zhou et al. (2018), the multi-output-based DM-LSTM model has smaller RMSE than the SM-LSTM model has; the improvement rates are reported to range from 2.18% to 13.91%. Table 7 shows the test RMSE and improvement rates of the models for three cities (Barcelona, Kocaeli, İstanbul). The CNN+LSTM model has improvement rates between 11 and 53% for PM_{10} , 20 and 31% for O_3 , 8 and 15% for NO_2 , 18 and 46% for SO_2 , and 9 and 47% for NO_X with respect to the LSTM model.

Method-1 (UNI/UNI)

2D samples created from non-normalized Barcelona data contain historical information of one target pollutant which are used to predict the concentration in $\mu g/m^3$. The network with the lowest validation RMSE was tested and performance metrics were measured. In the second stage, the model was trained for normalized Barcelona data to compare the performance with non-normalized data. The activation function in the fully connected layer was set as ReLU and sigmoid, and their performances are compared for the normalized case. Scatter plots of the prediction results are given in Fig. 7. As can be seen in the Fig. 7a, b,

and c, although the predicted values produced by the model are above the actual values, the model generally captured the behavior of the non-normalized time series data. Looking at the results obtained with the normalized data in Fig. 7d, e, and f, it is seen that the use of ReLU in the last layer leads to the prediction of extremely high values from the actual values. Using sigmoid instead of ReLU in the last layer provides to capture the behavior of normalized time series data.

The test RMSE for non-normalized data was measured as 37.29 for O_3 , 29.58 for NO_2 , and 13.87 for PM_{10} ; the correlation coefficient between target and predicted values was measured as 0.87 for O_3 , 0.79 for NO_2 , and 0.96 for PM_{10} . The test RMSE for normalized data with sigmoid was measured as 0.06 for O_3 , 0.09 for NO_2 , and 0.06 for PM_{10} ; the test RMSE with ReLU was measured as 0.28 for O_3 , 0.39 for NO_2 , and 0.42 for PM_{10} .

Method-2 (MULTI/UNI)

3D input samples were created from normalized data to combine the information about various variables that have different ranges of values. The activation functions in hidden layers were set to ReLU while sigmoid is only used at the fully connected layer. Scatter plots of the prediction results are given in Fig. 8.

In Barcelona, test RMSE was measured as 0.12 for O_3 , 0.11 for NO_2 , and 0.07 for PM_{10} ; correlation coefficient was measured as 0.87 for O_3 , 0.83 for NO_2 , and 0.95 for PM_{10} .

In Kocaeli, test RMSE was measured as 0.08 for PM_{10} , 0.08 for SO_2 , and 0.08 for NO_X ; correlation coefficient was measured as 0.85 for PM_{10} , 0.74 for SO_2 , and 0.77 for NO_X .

İstanbul data contain the information of five pollutants (O_3 , NO_2 , PM_{10} , SO_2 , NO_X); therefore, two different data sets were created to compare prediction results with

Table 7 Comparison of the prediction performance between 1-hidden layer LSTM model and CNN+LSTM model
Test RMSE results ($\mu\text{g}/\text{m}^3$) by different methods

Pollutant	Method	Barcelona			Kocaeli			İstanbul		
		LSTM	CNN+LSTM	Improvement rate	LSTM	CNN+LSTM	Improvement rate	LSTM	CNN+LSTM	Improvement rate
O_3	Method-2	RMSE=0.16	RMSE=0.12	25%				RMSE=0.13	RMSE=0.11	31%
	Method-3	RMSE=0.15	RMSE=0.12	20%				RMSE=0.15	RMSE=0.12	20%
NO_2	Method-2	RMSE=0.13	RMSE=0.11	15%				RMSE=0.10	RMSE=0.10	0%
	Method-3	RMSE=0.12	RMSE=0.11	8%				RMSE=0.10	RMSE=0.11	-10%
PM_{10}	Method-2	RMSE=0.11	RMSE=0.07	36%	RMSE=0.12	RMSE=0.07	42%	RMSE=0.09	RMSE=0.08	11%
	Method-3	RMSE=0.14	RMSE=0.07	50%	RMSE=0.17	RMSE=0.08	52%	RMSE=0.10	RMSE=0.08	20%
SO_2	Method-2				RMSE=0.12	RMSE=0.09	25%	RMSE=0.10	RMSE=0.10	0%
	Method-3				RMSE=0.13	RMSE=0.07	46%	RMSE=0.11	RMSE=0.09	18%
NO_X	Method-2				RMSE=0.12	RMSE=0.09	25%	RMSE=0.11	RMSE=0.10	9%
	Method-3				RMSE=0.13	RMSE=0.08	39%	RMSE=0.17	RMSE=0.09	47%

Table 8 Proposed architectures and their performances in the literature

Machine learning models and performance results in literature

Reference	Soh et al. (2018)				Zeinalnezhad et al. (2020)				Di Antonio et al. (2019)	
Model	ST-DNN				ANFIS				LSTM	
Input type	Univariate		Multivariate		Multivariate				Multivariate	
Pollution	O_3	SO_2	O_3	SO_2	O_3	SO_2	NO_2	PM_{10}		
RMSE	0.47	0.11	0.21	0.15	0.20	0.25	0.16	0.08		

other cities. The first data set (Dataset-1) contains the information of the same pollutants with Barcelona (O_3 , NO_2 , and PM_{10}); the second data set (Dataset-2) contains information of the same pollutants with Kocaeli (PM_{10} , SO_2 , NO_X).

For Dataset-1 in İstanbul, test RMSE was measured as 0.12 for O_3 , 0.11 for NO_2 , and 0.08 for PM_{10} ; correlation coefficient was measured as 0.89 for O_3 , 0.82 for NO_2 , and 0.75 for PM_{10} .

For Dataset-2 in İstanbul, test RMSE was measured as 0.08 for PM_{10} , 0.09 for SO_2 , and 0.09 for NO_X ; correlation coefficient was measured as 0.79 for PM_{10} , 0.79 for SO_2 , and 0.86 for NO_X .

Method-3 (MULTI/MULTI)

Method-3 was performed to predict the concentration of multiple variables at the same time for a city by using 3D input that contains the information of all pollutants. The target is the concentration of three pollutants in multiple locations in the next hour. The network was built with ReLU activation function in hidden layers and sigmoid activation function in the fully connected layer. Scatter plots of the prediction results are given in Fig. 8.

In Barcelona, test RMSE was measured as 0.12 for O_3 , 0.11 for NO_2 , and 0.07 for PM_{10} ; correlation coefficient

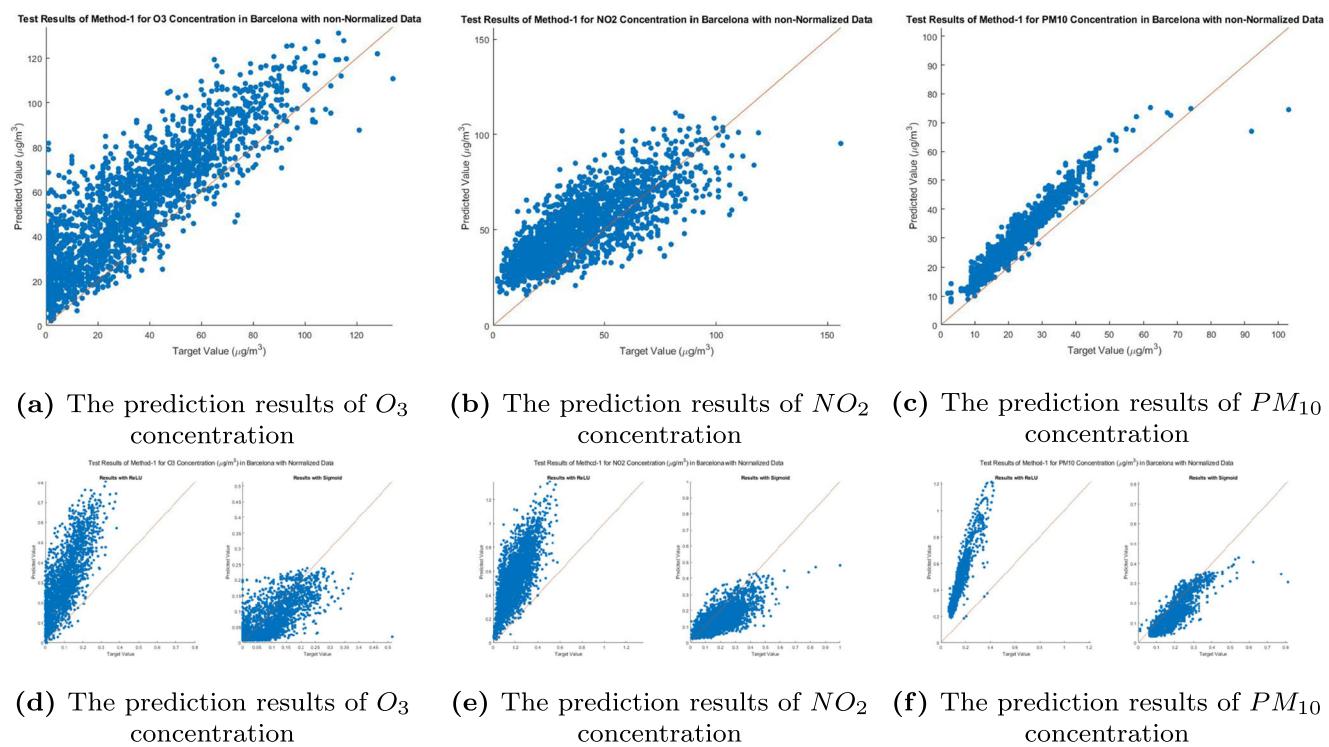


Fig. 7 Results of Method-1 for the prediction in Barcelona for non-normalized (a–c) and normalized (d–f) air pollution data

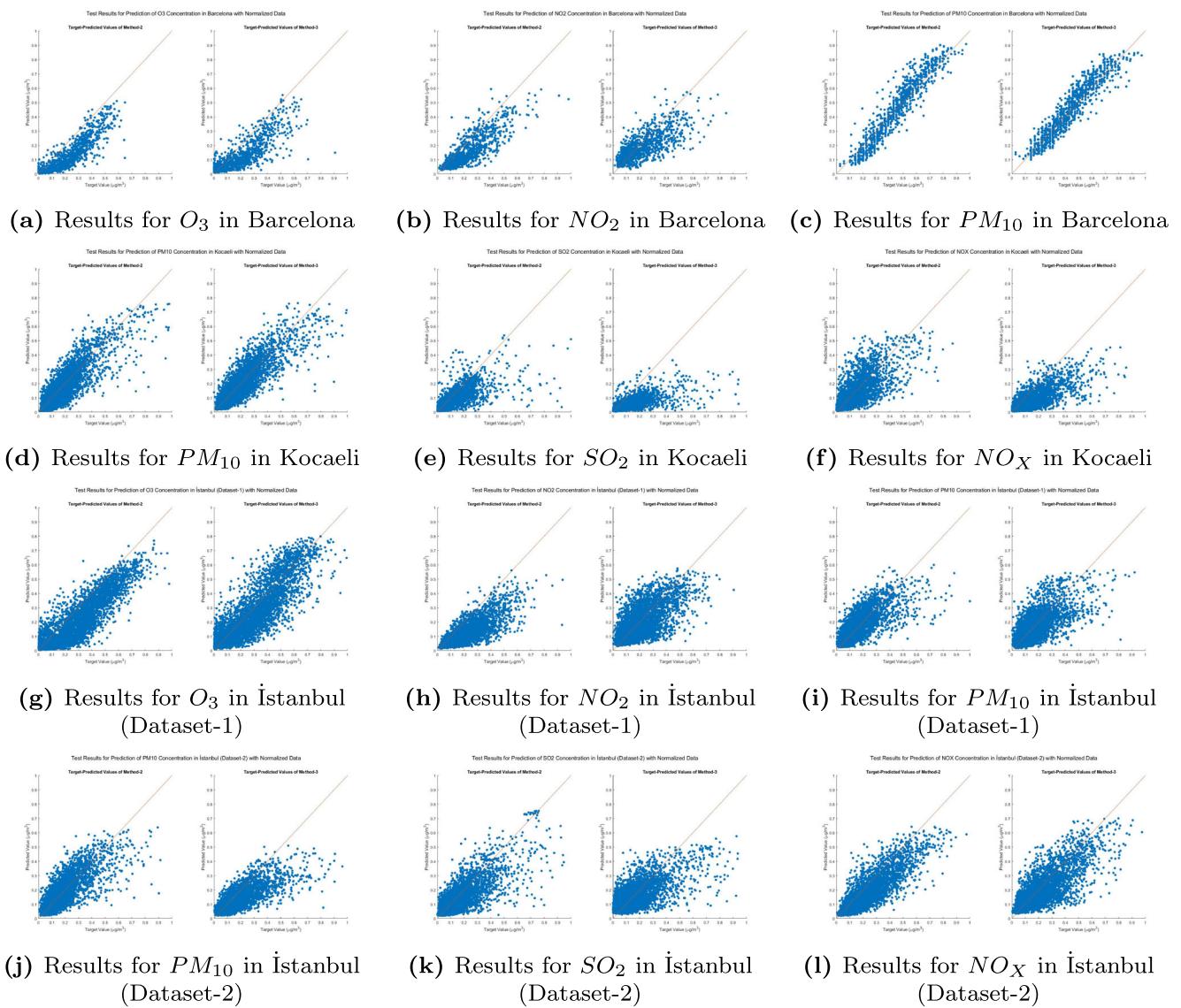


Fig. 8 Scatter plots of target and predicted concentrations by Method-2 and Method-3

was measured as 0.82 for O_3 , 0.73 for NO_2 , and 0.94 for PM_{10} .

In Kocaeli, test RMSE was measured as 0.07 for PM_{10} , 0.09 for SO_2 , and 0.09 for NO_X ; correlation coefficient was measured as 0.86 for PM_{10} , 0.74 for SO_2 , and 0.79 for NO_X .

For Dataset-1 in İstanbul, test RMSE was measured as 0.11 for O_3 , 0.10 for NO_2 , and 0.08 for PM_{10} ; correlation coefficient was measured as 0.86 for O_3 , 0.69 for NO_2 , and 0.70 for PM_{10} .

For Dataset-2 in İstanbul, test RMSE was measured as 0.09 for PM_{10} , 0.10 for SO_2 , and 0.10 for NO_X ; correlation coefficient was measured as 0.77 for PM_{10} , 0.68 for SO_2 , and 0.78 for NO_X .

Results with meteorological information

Meteorology information were added to the input data to observe the contribution of meteorological conditions to the prediction accuracy. Hourly information about temperature, air pressure, relative humidity, wind speed, and wind direction were added to the 3D input. The results were observed by Method-2 and Method-3 with the same deep learning model structure. Since meteorology data are available for only Kocaeli and İstanbul, this part of the study was done for these two cities.

Table 9 gives the results observed when ReLU is used in hidden layers and sigmoid is used in the fully connected layer as activation functions. For the prediction of PM_{10}

Table 9 Prediction performance comparison between studies with and without meteorological data for Kocaeli and İstanbul, bold entries indicate the lowest RMSE value for the pollutant

Pollutant	Study	Kocaeli		İstanbul	
		with AQ Data	with AQ+MI Data	with AQ Data	with AQ+MI Data
PM_{10}	Method-2	0.08	0.09	0.08	0.07
	Method-3	0.07	0.09	0.09	0.08
	Method-2	0.07	0.08	0.09	0.09
SO_2	Method-3	0.09	0.08	0.10	0.10
	Method-2	0.08	0.10	0.09	0.13
NO_X	Method-3	0.09	0.12	0.10	0.11

concentration, using meteorological information (MI) and air quality (AQ) data together reduced the RMSE by 10% with Method-2 and Method-3 in İstanbul. However, it increased RMSE by 12% with Method-2 and by 30% with Method-3 in Kocaeli. For the prediction of SO_2 concentration, the use of MI data with AQ data caused a 15% increase of RMSE with Method-2 for Kocaeli, on the other hand, there was no change in the test RMSE in İstanbul. For the prediction of NO_X concentration, adding MI data to the input caused an increase of RMSE in both cities. The test RMSE increased by about 45% in İstanbul and 25% in Kocaeli with Method-2; it increased by 33% in Kocaeli and by 10% in İstanbul with Method-3. When the meteorological data are added to the model along with air quality data, the complexity in the input increases. As the complexity in the input increases, the model has difficulty in capturing the relation between spatial-temporal features. Also in the study (Di Antonio et al. 2019), a decrease in the prediction performance caused by the increase in the number of features is observed.

Transfer learning

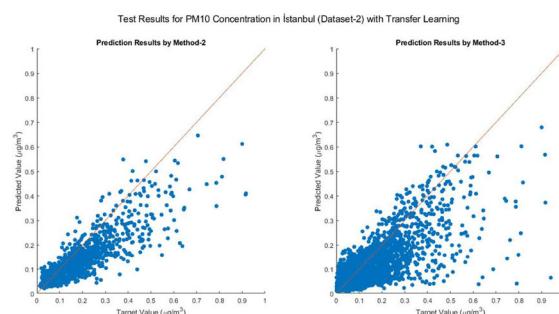
The transductive transfer learning method was applied to transfer network weights where the source task and target

task are the same, whereas the source domain and target domain are different but related. Barcelona and Kocaeli were selected as source city and İstanbul was selected as the target city. The pre-trained model was firstly tested directly on the target data. Then, the last learning layer (fully connected layer) was changed using Deep Network Designer Tool, a short training process was performed with learning rate 0.001 for the training with target city's data.

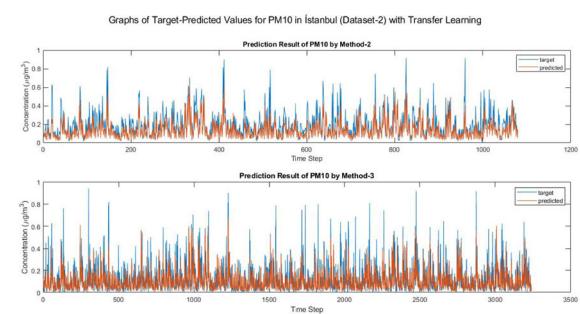
Transfer learning was performed for the prediction of PM_{10} concentration in İstanbul. Figure 9a shows the relation between target and predicted values; the graphs in Fig. 9b are for the predicted and target values in İstanbul. These are for the transfer learning results where Kocaeli is the source domain and İstanbul is the target domain. Test RMSE were measured as 0.12 and 0.09 for Method-2 and Method-3, respectively.

Conclusion and discussion

In this study, a CNN+LSTM-based deep neural network model was proposed to predict the hourly concentration of air pollutants in three cities (Barcelona, Kocaeli, İstanbul) based on spatial-temporal features. There are three different methods used according to the input-output relationship.



(a) Scatter plot for PM_{10} concentration in İstanbul by Method-2 and Method-3



(b) Graph of the transfer learning for PM_{10} concentration in İstanbul by Method-2 and Method-3

Fig. 9 Transfer learning results for PM_{10} concentration in İstanbul by pre-trained neural network with Kocaeli data

Method-1 is based on univariate input and univariate output; Method-2 is based on multivariate input and univariate output; Method-3 is based on multivariate input and multivariate output. All methods are performed for prediction of air pollutants in multiple locations in the city.

The first objective of the study is to develop a supervised model for air pollution prediction from real sensor data obtained in different locations, and to set the parameters that give the optimal level of learning based on the best performance metric values achieved for the most number of pollutants and cities. The CNN+LSTM neural network predicts the future hourly concentration of air pollutants in certain environmental factors (air pollution and meteorological conditions). Convolutional layers extract the relation between locations for spatial features while LSTM layers extract temporal feature characteristics from time series data. The nonlinear relationship between multi-variable time series and air pollutants were combined, and the effects of air pollution and meteorological data on prediction performance were observed. Although the model selection task-related hyperparameters generally did not change among different pollutants, the training process-related hyperparameters changed among the pollutants and the cities.

The use of different activation functions at the last layer of the network provided different prediction performance. ReLU in the fully connected layer causes exceeding predicted values for each pollutant while working with normalized data. Sigmoid limits the prediction values between 0 and 1 and it improves the prediction performance for normalized data. However, it produces the prediction result at lower points than the target values at higher pollution levels. The optimal case comes out to use different activation functions: ReLU in each hidden layer and sigmoid in the last learning layer which is the fully connected layer.

While O_3 , NO_2 , and PM_{10} were selected as target pollutants in Barcelona, PM_{10} , SO_2 , and NO_X were selected in Kocaeli. There are two data sets in Istanbul since there are five pollutants' data for Istanbul: The first data set (Data Set-1) contains the same pollutants with Barcelona (O_3 , NO_2 and PM_{10}); the second data set (Data Set-2) contains the same pollutants with Kocaeli (PM_{10} , SO_2 and NO_X).

Considering the test RMSE, the performances of Method-2 and Method-3 are better than Method-1 for time series prediction in each city. In Barcelona, test RMSE values are the same as Method-2 and Method-3. On the other hand, Method-2 has higher correlation coefficients than Method-3. Test RMSE for PM_{10} prediction in Barcelona and Kocaeli is generally the same. The RMSE values for O_3 and NO_2 in Barcelona are 5% higher than the RMSE value for PM_{10} . The difference between RMSE

for different pollutants is 0.01 in Kocaeli. The RMSE for Nitrogen-based pollutants (NO_2 and NO_X) is 0.04 higher than the RMSE for PM_{10} in Istanbul. Istanbul data set is larger than Barcelona and Kocaeli data sets. Nevertheless, RMSE for O_3 , NO_X , and NO_2 are approximately the same as in other cities while RMSE for PM_{10} and SO_2 are higher than the other two cities.

3D samples were created with information on pollutants and meteorology to be used for Method-2 and Method-3. The meteorological features are temperature, relative humidity, air pressure, wind speed, and wind direction. As a result, adding meteorological features to the input generally increased the test RMSE by 10 to 40% in both Kocaeli and Istanbul, using only air pollution information as features caused the model to be more successful. Use of meteorology data in Istanbul only improved the prediction performance for PM_{10} concentration with both Method-2 and Method-3 by a rate of 12%.

The second objective of the study is to transfer the models between cities and to examine prediction ability of different models on weight transfer and to identify whether the parameters that give an optimal level of learning are the same for different cities. The training process with the target city's data set did not improve the prediction performance, the lowest test RMSE was observed when the model was applied directly to the target domain.

The transductive transfer learning success was achieved in weight transfer for Method-2 and Method-3 from Kocaeli to Istanbul. Weight transfer between Barcelona-Istanbul or Barcelona-Kocaeli has higher test RMSE than the transfer between Kocaeli-Istanbul has. Kocaeli and Istanbul are neighbor cities and they have similar air pollution and meteorological characteristics, on the other hand, the characteristics of Barcelona are not related to the other two cities.

Looking at the optimal hyperparameters, the model selection task-related hyperparameters are the same for different cities, however, the training process-related hyperparameters change among cities. Although a common deep learning network structure was determined, the transfer of the model did not increase the prediction performance in the target city. For an air pollution prediction model with high accuracy, it would be better to train the model with the data of each city and to set training-related hyperparameters as given in our work. Further work using data from other cities will enhance the prediction capability of the proposed model.

The major outcomes of this study are as follows: different input types for CNN+LSTM hybrid machine learning algorithm were defined and their effects on the performance of the model were examined. The relationship between the concentration of various pollutants was observed. Besides the concentration information of air pollutants, the effect of

meteorological conditions was observed for the prediction of air pollution. The efficiency and performance of transfer learning between cities were examined. Both spatial and temporal features were used as inputs to the algorithm, the concentration of pollutants for the next hour were predicted at multiple locations. Also, as the output of the model, the estimated concentration of more than one pollutant was obtained at the same time.

However, some limitations should also be noted for this study. There is a large amount of missing and/or bad data in the publicly available data sources for the target cities selected for this study. In this case, the size of the data set used for training the model is limited due to the lack of usable data collected from all sensors located in cities. Also, due to this limitation of available/reliable data, small sized samples were created as model input. This makes it difficult for the model to extract meaningful relationships from the data. In addition, since the behavior and characteristics of the data for the target cities will be specific to each, the lack of previous studies on this type data has also affected the data analysis and the selection of the appropriate algorithm.

Author contribution Aysenur Gilik has contributed to conceptual design, data processing, algorithm development, implementation, writing and review. Arif Selcuk Ogrenci has contributed to conceptual design, algorithm development, writing and review. Atilla Ozmen has contributed to conceptual design, algorithm development, writing and review.

Data availability The data used in this paper are publicly available at Ministry of Environment and Urbanization Continuous Monitoring Center (URL: <https://sim.csb.gov.tr/>) and Ajuntament de Barcelona's open data service (URL: <https://opendata-ajuntament.barcelona.cat/en/>). The codes are not packaged into a library however the codes can be shared with interested readers.

Declarations

Ethics approval and consent to participate Not applicable.

Consent for publication Not applicable.

Competing interests The authors declare no competing interests.

References

- Aceves-Fernandez M, Domínguez-Guevara R, Pedraza Ortega JC, Vargas-Soto J (2020) Evaluation of key parameters using deep convolutional neural networks for airborne pollution (pm10) prediction. *Discret Dyn Nat Soc* 2020:1–14. <https://doi.org/10.1155/2020/2792481>
- Al-Janabi S, Mohammad M, Al-Sultan A (2019) A new method for prediction of air pollution based on intelligent computation. *Soft Comput* 24:661–680. <https://doi.org/10.1007/s00500-019-04495-1>
- Al-Janabi S, Alkaim A, Al-Janabi E, A Aljeboree MM (2021) Intelligent forecaster of concentrations (pm2.5, pm10, no2, co, o3, so2) caused air pollution (ifcsap). *Neural Computing and Applications*. <https://doi.org/10.1007/s00521-021-06067-7>
- Bakici T, Almirall E, Wareham J (2012) A smart city initiative: The case of barcelona. *J Knowl Econ*:4. <https://doi.org/10.1007/s13132-012-0084-9>
- Barcelona City Council (2020) Open data bcn. <https://opendata-ajuntament.barcelona.cat/en/>, (last accessed: 15.04.2021)
- Bashir Shaban K, Kadri A, Rezk E (2016) Urban air pollution monitoring system with forecasting models. *IEEE Sens J* 16(8):2598–2606. <https://doi.org/10.1109/JSEN.2016.2514378>
- Chen Q, Wang W, Wu F, De S, Wang R, Zhang B, Huang X (2019) A survey on an emerging area: Deep learning for smart city data. *IEEE Trans Emerging Top Comput Intell* 3(5):392–410. <https://doi.org/10.1109/TETCI.2019.2907718>
- Chu HJ, Lin CY, Cj Liau, Kuo YM (2012) Identifying controlling factors of ground-level ozone levels over southwestern Taiwan using a decision tree. *Atmos Environ* 60:142–152. <https://doi.org/10.1016/j.atmosenv.2012.06.032>
- Di Antonio L, Rosato A, Colaiuda V, Lombardi A, Tomassetti B, Panella M (2019) Multivariate prediction of pm 10 concentration by lstm neural networks. pp 423–431. <https://doi.org/10.1109/PIERS-Fall48861.2019.9021929>
- Djalalova I, Delle Monache L, Wilczak J (2015) Pm2.5 analog forecast and kalman filter post-processing for the community multiscale air quality (cmaq) model. *Atmos Environ*:108. <https://doi.org/10.1016/j.atmosenv.2015.02.021>
- Eessaar E (2016) The database normalization theory and the theory of normalized systems: finding a common ground. *Baltic J Modern Comput* 4:5–33
- EU (2021) Explore. <https://eu-smartcities.eu/>, (last accessed: 15.04.2021)
- European Environment Agency (2019) Air quality in europe — 2019 report. Tech. Rep. EEA Report 10/2019
- European Commission (2017) 2030 climate and energy framework. https://ec.europa.eu/clima/policies/strategies/2030_en, (last accessed: 15.04.2021)
- Khan S, Paul D, Momtahan P, Aloqaily M (2018) Artificial intelligence framework for smart city microgrids: state of the art, challenges, and opportunities. In: 2018 Third International Conference on Fog and Mobile Edge Computing (FMEC), pp 283–288
- Kloeckl K, Senn O, Ratti C (2012) Enabling the real-time city: live singapore! *J Urban Technol*:19. <https://doi.org/10.1080/10630732.2012.698068>
- Li X, Peng L, Shao J, Chi T (2016) Deep learning architecture for air quality predictions. *Environ Sci Pollut Res*:23. <https://doi.org/10.1007/s11356-016-7812-9>
- Lv M, Li Y, Chen L, Chen T (2019) Air quality estimation by exploiting terrain features and multi-view transfer semi-supervised regression. *Inf Sci*:483. <https://doi.org/10.1016/j.ins.2019.01.038>
- Ma J, Cheng J, Lin C, Tan Y, Zhang J (2019) Improving air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques. *Atmos Environ* 214:116885. <https://doi.org/10.1016/j.atmosenv.2019.116885>
- Ma J, Ding Y, Gan VJL, Lin C, Wan Z (2019) Spatiotemporal prediction of pm2.5 concentrations at different time granularities using idw-blstm. *IEEE Access* 7:107897–107907
- Nwankpa C, Ijomah W, Gachagan A, Marshall S (2018) Activation functions: Comparison of trends in practice and research for deep learning. *arXiv*:1811.03378

- Pan SJ, Yang Q (2010) A survey on transfer learning. *IEEE Trans Knowl Data Eng* 22(10):1345–1359. <https://doi.org/10.1109/TKDE.2009.191>
- Park S, Kim M, Kim M, Namgung HG, Kim KT, Cho K, Kwon SB (2017) Predicting pm 10 concentration in seoul metropolitan subway stations using artificial neural network (ann). *J Hazard Mater*:341. <https://doi.org/10.1016/j.jhazmat.2017.07.050>
- Qi Z, Wang T, Song G, Hu W, Li X, Zhang Z (2018) Deep air learning: interpolation, prediction, and feature analysis of fine-grained air quality. *IEEE Trans Knowl Data Eng* 30(12):2285–2297. <https://doi.org/10.1109/TKDE.2018.2823740>
- Qin D, Yu J, Zou G, Yong R, Zhao Q, Zhang B (2019) A novel combined prediction scheme based on cnn and lstm for urban pm2.5 concentration. *IEEE Access* 7:20050–20059. <https://doi.org/10.1109/ACCESS.2019.2897028>
- Republic of Turkey Ministry of Environment and Urbanization (2019) National air quality monitoring network (in turkish). <https://sim.csb.gov.tr/>, (last accessed 15.04.2021)
- Schürholz D, Kubler S, Zaslavsky A (2020) Artificial intelligence-enabled context-aware air quality prediction for smart cities. *J Cleaner Prod*:121941. <https://doi.org/10.1016/j.jclepro.2020.121941>
- Scovronick N (2015) Reducing global health risks through mitigation of short-lived climate pollutants
- Sánchez L, Muñoz L, Galache J, Sotres P, Santana J, Gutierrez V, Ramdhany R, Gluhak A, Krco S, Theodoridis E, Pfisterer D (2013) Smartsantander: Iot experimentation over a smart city testbed. *Computer Networks*. <https://doi.org/10.1016/j.bjp.2013.12.020>
- Soh P, Chang J, Huang J (2018) Adaptive deep learning-based air quality prediction model using the most relevant spatial-temporal relations. *IEEE Access* 6:38186–38199. <https://doi.org/10.1109/ACCESS.2018.2849820>
- Tai A, Mickley L, Jacob D (2010) Correlations between fine particulate matter (pm2.5) and meteorological variables in the united states: implications for the sensitivity of pm2.5 to climate change. *Atmos Environ* 44:3976–3984. <https://doi.org/10.1016/j.atmosenv.2010.06.060>
- Tao Q, Liu F, Li Y, Sidorov D (2019) Air pollution forecasting using a deep learning model based on 1d convnets and bidirectional gru. *IEEE Access* 7:76690–76698. <https://doi.org/10.1109/ACCESS.2019.2921578>
- Wei Y, Zheng Y, Yang Q (2016) Transfer knowledge between cities. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16. Association for Computing Machinery, New York, pp 1905–1914. <https://doi.org/10.1145/2939672.2939830>
- Weiss K, Khoshgoftaar T, Wang D (2016) A survey of transfer learning. *J Big Data*:3. <https://doi.org/10.1186/s40537-016-0043-6>
- WHO (2019) Healthy environments for healthier populations: why do they matter, and what can we do?
- Zanella A, Bui N, Castellani A, Vangelista L, Zorzi M (2014) Internet of things for smart cities. *IEEE Internet Things J* 1(1):22–32. <https://doi.org/10.1109/JIOT.2014.2306328>
- Zeinalnezhad M, Gholamzadeh Chofreh A, Goni F, Klemes J (2020) Air pollution prediction using semi-experimental regression model and adaptive neuro-fuzzy inference system. *J Cleaner Prod*:121218. <https://doi.org/10.1016/j.jclepro.2020.121218>
- Zhang Y, Wang Y, Gao M, Ma Q, Zhao J, Zhang R, Wang Q, Huang L (2019) A predictive data feature exploration-based air quality prediction approach. *IEEE Access* 7:30732–30743. <https://doi.org/10.1109/ACCESS.2019.2897754>
- Zhang Z, Zeng Y, Yan K (2021) A hybrid deep learning technology for pm2.5 air quality forecasting. *Environmental Science and Pollution Research*. <https://doi.org/10.1007/s11356-021-12657-8>
- Zhao G, Huang G, He H, Wang Q (2019) Innovative spatial-temporal network modeling and analysis method of air quality. *IEEE Access* 7:26241–26254. <https://doi.org/10.1109/ACCESS.2019.2900997>
- Zhao Z, Qin J, He Z, Li H, Yang Y, Zhang R (2020) Combining forward with recurrent neural networks for hourly air quality prediction in northwest of china. *Environ Sci Pollution Res Int* 27(23):28931–28948. <https://doi.org/10.1007/s11356-020-08948-1>
- Zhou Y, Chang FJ, Chang LC, Kao IF, Wang YS (2018) Explore a deep learning multi-output neural network for regional multi-step-ahead air quality forecasts. *J Clean Prod*:209. <https://doi.org/10.1016/j.jclepro.2018.10.243>
- Zivot E, Wang J (2003) Rolling Analysis of Time Series, pp 299–346. https://doi.org/10.1007/978-0-387-21763-5_9

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.