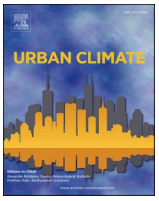
Urban Climate 48 (2023) 101418

Contents lists available at ScienceDirect

Urban Climate

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



Predicting next hour fine particulate matter ($PM_{2.5}$) in the Istanbul Metropolitan City using deep learning algorithms with time windowing strategy

Beytullah Eren [a,b,*], İpek Aksangür [a], Caner Erden [c,d]

[a] Department of Environmental Engineering, Faculty of Engineering, Sakarya University, Esentepe, Sakarya, Türkiye
[b] Halfeti Vocational School, Harran University, Halfeti, Şanlıurfa, Türkiye
[c] Department of International Trade and Finance, Faculty of Applied Science, Sakarya University of Applied Science, Sakarya, Türkiye
[d] AI Research and Application Center, Sakarya University of Applied Sciences, Sakarya, Türkiye

ARTICLE INFO

Keywords:
Deep learning
Fine particulate matter ($PM_{2.5}$)
Gated recurrent unit (GRU)
Long-short term memory (LSTM)
Recurrent neural network (RNN)
Time windowing

ABSTRACT

Poor air quality has various detrimental physical and mental effects on human health and quality of life. In particular, $PM_{2.5}$ air pollution has been associated with cardiovascular and respiratory problems. Therefore, air quality management is an essential issue for densely populated cities to reduce or prevent the adverse effects of air pollution. Considering this, reliable models for predicting pollution levels for pollutants like $PM_{2.5}$ are critical tools for decision-making. For this purpose, this study presents three kinds of deep learning (DL) algorithms (LSTM, RNN, and GRU) that utilize a time-windowing strategy to predict the hourly concentration of $PM_{2.5}$ in the Istanbul metropolitan. The models were trained and tested using large data sets that envelope air quality parameters ($PM_{2.5}$, SO_2, NO, NO_2, NO_X, and O_3) and meteorological factors (temperature, wind speed, relative humidity, and air pressure) for about five years. The experimental results demonstrate that the LSTM+LSTM model performs significantly better with an R^2 of 0.98 and 0.97 at the significance level ($p < 0.05$) for training and test sets compared to other deep learning algorithms. In addition, data for one year from several stations located in nine different districts of Istanbul were used to evaluate the proposed model's generalization ability. As a result, the proposed LSTM+LSTM model has a good generalization ability with an R^2 accuracy rate of 0.90 ($p < 0.05$) and above for all stations and can be used for non-linear, non-stationary multidimensional time series data. Furthermore, the results were compared to other studies in the literature; it was found that the proposed LSTM+LSTM model performed better in predicting $PM_{2.5}$ concentrations.

1. Introduction

Air pollution has become a critical and costly environmental problem associated with global industrialization and urbanization. (US EPA O, 2014). Fine particulate matters that are defined as complex mixtures of solids and aerosols with an aerodynamic diameter of $\leq$2.5 μm ($PM_{2.5}$) have been categorized as the most health-damaging effect, including respiratory and cardiovascular morbidity and mortality (Yu et al., 2022). $PM_{2.5}$ commonly originates from outdoor sources, including transportation systems, heating systems,

* Corresponding author at: Department of Environmental Engineering, Faculty of Engineering, Sakarya University, Esentepe, Sakarya, Türkiye.
E-mail address: beren@sakarya.edu.tr (B. Eren).

https://doi.org/10.1016/j.uclim.2023.101418
Received 15 June 2022; Received in revised form 6 December 2022; Accepted 8 January 2023
Available online 11 January 2023
2212-0955/© 2023 Elsevier B.V. All rights reserved.

wildfire smoke, volcanic eruptions, power plants, and indoor activities, including tobacco smoke, cooking, operating fireplaces, and burning candles or oil lamps (Tucker, 2000). Several studies confirm associations between $PM_{2.5}$ exposure and adverse health effects such as total mortality, cardiovascular mortality, respiratory mortality, hypertension, lung cancer, and influenza (Miller and Xu, 2018; Guan et al., 2021; Yu et al., 2022). According to the World Health Organization, approximately 7 million deaths per year (one-eighth of the total annual global deaths) are expected via outdoor and indoor air pollution (WHO, 2022a). A secondary adverse effect of $PM_{2.5}$ is that it reduces atmospheric visibility by contributing to smog formation. Hence, low visibility causes a decrease in the photosynthetic activity of plants, while the physicochemical attributes of soils change to lead to the accumulation of minerals and metals in soil (Mukherjee and Agrawal, 2017).

An adequate air pollution management strategy is essential to improve air quality and air-related public health. Therefore, air quality management tools are crucial to controlling air pollution's adverse outcomes, especially in densely populated metropolitan areas. One of the most effective ways to reduce air pollution-related health risks and economic losses is to develop an early warning system based on the prediction of air quality parameters. In this sense, current air pollution prediction models widely use numerical and statistical methods for their predictions. Numerical models consider atmospheric physics and chemistry while using meteorological principles and mathematical methods to model air quality on a large scale (Yan et al., 2021). Examples of numerical models include nested air quality prediction modeling system (NAQPMS) (Wang et al., 2001), weather research and forecasting model with chemistry (WRF-Chem) (Wang et al., 2016; Zakarin et al., 2021; Nerobelov et al., 2021), community multiscale air quality (CMAQ) (Zhou et al., 2019), the environment-high resolution limited area model (Baklanov et al., 2017) and chemical transport models (CTMs) (Srivastava and Blond, 2022). These models rely upon comprehensive pollutant data sets, complete knowledge of pollutant sources, the chemical composition of emissions, relatively complex calculations, and carry high uncertainty in forecasting results (Bai et al., 2018; Yan et al., 2021). Unlike numerical models, statistical models do not consider atmospheric processes and do not require an understanding of air quality processes. Statistical forecast models depend on a data-driven perspective (Bai et al., 2018). These models were first developed using traditional multiple linear regression methods. Traditional regression methods evolved into statistical forecasting methods along with advancements in computer science and technology. Accordingly, artificial intelligence (AI)-based statistical methods have emerged for non-linear air pollution modeling in a high-dimensional space. Currently, there are many studies on the use of AI for predicting air pollutants. For example, the $PM_{2.5}$ and PM_{10} forecasting system in the Polish agglomerations (Czernecki et al., 2021), the $PM_{2.5}$ hourly forecast in Santiago de Chile (Perez and Menares, 2018), the $PM_{2.5}$ daily forecast in Bishkek (Isaev et al., 2022), the $PM_{2.5}$ forecasting system in Tehran (Karimian et al., 2019a), and in Shanghai, China, ML technology is used to improve $PM_{2.5}$ forecasts using WRF-Chem model simulations (Ma et al., 2020). Moreover, the most commonly used AI-based methods for air pollution prediction include artificial neural network (ANN) (Cabaneros et al., 2019), support vector machine (SVM) (Leong et al., 2020), random forest (RF) (Rubal, 2018), and deep learning (DL) (Bui et al., 2018). Among these methods, the use of DL for air pollution prediction has become widespread due to advantages such as using more layers and more comprehensive datasets and obtaining more accurate results by processing all layers simultaneously (Bekkar et al., 2021).

Deep learning algorithms (LSTM, CNN, RNN, etc.) are particularly suitable for modeling multi-parameter, complex, and non-linear processes. For example, the LSTM is the most commonly utilized DL algorithm in air pollution modeling because it accurately represents non-linear real-world situations, considers the effect of long history data, and solves multiple inputs or multivariate series (Lu et al., 2021; Bekkar et al., 2021). In a recent study, Karimian et al. (2019b) compared the estimation performances of models based on multiple additive regression trees (MART), a deep feed-forward neural network (DFFNN), and LSTM to predict $PM_{2.5}$ concentrations at different time intervals. The study noted that the LSTM model captured the temporal dependencies in the time series data compared to the other two models and gave the best estimate for $PM_{2.5}$ concentrations with 80% of the variability ($R^2 = 0.8$). In another study, Pak et al. (2020) applied a hybrid convolutional neural network and long short-term memory (CNN-LSTM) mode to 3-year air quality and meteorological data obtained from 384 stations to predict the next day's average daily $PM_{2.5}$ concentrations considering the spatio-temporal correlations between the input variables. They concluded that this model has better stability and prediction performance than the multilayer perceptron (MLP) and LSTM models. Menares et al. (2021) proposed a LSTM and DFFNN model to predict maximum $PM_{2.5}$ concentrations. They concluded that LSTM models respond better to $PM_{2.5}$ event prediction when appropriate pollutant and meteorological variables are selected. LSTM models can recall important synoptic patterns useful for $PM_{2.5}$ prediction over time through memory units. The LSTM algorithm is reliable and accurate in predicting $PM_{2.5}$ utilizing historical air quality and meteorological data.

Developing an air quality prediction model using the deep learning method can be considered in several dimensions. One of those dimensions is determining the most appropriate imputation method for detecting the missing values and outliers and removing and filling them with new data. Missing and erroneous data in any time series can be caused by human error, device failure, or downtime due to routine maintenance. For the forecasting model, missing and erroneous data are not handled by an appropriate method, leading to unreliable forecasting results (Mir et al., 2022). For this purpose, several imputation methods, such as linear interpolation, mean imputation, mode-median imputation, K-Nearest Neighbor imputation, and deep learning imputation methods, have been frequently used (Quinteros et al., 2019). Samal et al. (2021) estimated the $PM_{2.5}$ data, which consisted of missing data, with deep learning methods. In addition to traditional machine learning methods, deep learning methods such as CNN, RNN, and LSTM were used in the study comparison. Another dimension is the selection of a deep learning algorithm that will solve the relationship in the multidimensional and complex relationship data set. No single algorithm is suitable for every data set used in prediction models. Many algorithms need to work with training datasets of different types, volumes, and accuracy. The primary goal of the prediction model development studies is to determine the appropriate algorithm that will give reliable and acceptable performance for the selected data set. The recent increase in the development of deep learning models for prediction purposes has led to many new algorithms. Therefore, investigating the effect of combining various deep learning algorithms on model prediction performances is considered

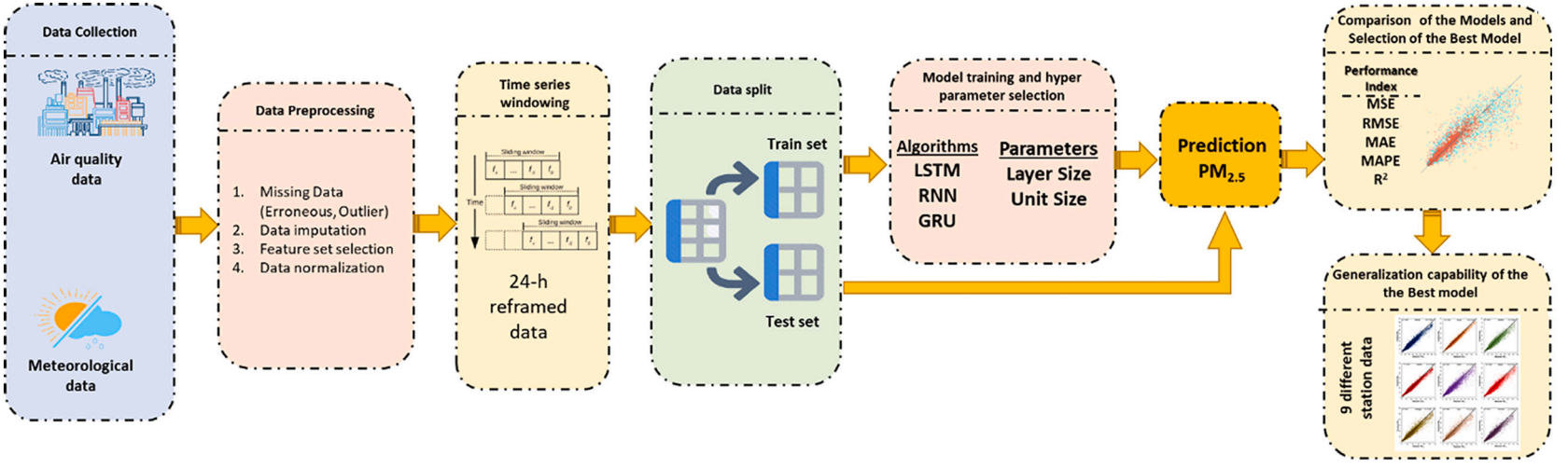


Fig. 1. Workflow for prediction of hourly $PM_{2.5}$ concentrations.

another dimension. Li et al. (2020) predicted $PM_{2.5}$ data in the Beijing region using CNN-LSTM deep learning algorithms in a hybrid way. In the study, a successful model with an RMSE of 18.99 was developed by combining the CNN method's feature extraction with the advantages of the LSTM method in time series analysis. The last dimension evaluates the developed deep learning model's generalization ability. The generalization ability of the developed model (how good the model is in estimating new samples that it has not seen before) is essential in terms of applicability and dissemination of the model to other regions and data sets with similar characteristics. Du et al. (2022) experimentally performed the models they developed for three Chinese cities (Beijing, Tianjin, and Shijiazhuang). That study emphasizes the importance of predicting pollutants that threaten human health, such as $PM_{2.5}$, and states that the developed model can be used for cities with similar characteristics. Considering all these dimensions, predicting the air pollutant concentrations in a particular region is crucial. Overall, the main objectives of this study are as follows:

(i). To develop a deep learning model that can predict hourly $PM_{2.5}$ concentrations using a specific time series dataset comprising air quality and meteorological data with a complex and potentially non-linear relationship.
(ii). To determine the effectiveness of the time windowing strategy in estimating 25th-hour $PM_{2.5}$ concentrations using historical data from the previous 24 h in a developed deep learning model for estimating hourly $PM_{2.5}$ concentrations.
(iii). To present a comparative analysis combining different deep learning algorithms like LSTM, RNN, and GRU based on performance evaluation criteria, i.e., MSE, RMSE, MAE, MAPE, and R^2.
(iv). To evaluate the proposed model's generalization ability and applicability in hourly $PM_{2.5}$ concentration prediction using data from different regions with potentially similar characteristics.

The remainder of this study is organized as follows: Section 2 describes the study area and its characteristics, as well as the datasets, data preprocessing stage, and time windowing strategy. Also, it defines the basic concepts of the deep learning models, namely LSTM, RNN, and GRU. Section 3 compares the model performances and discusses the experimental results with the studies published in the literature and evaluates the generalization ability of the proposed deep learning model. Finally, Section 4 presents the concluding remarks and possible extensions of this work with future research suggestions.

2. Material and methods

In this study, the workflow of the proposed approach is summarized in Fig. 1. Accordingly, meteorological and air quality data were combined to create the data set in the first step. Data preprocessing steps were carried out in the second stage of preprocessing data, such as processing missing data and detecting outlier data. Then, the data set was taken into a 24-h time frame and adapted to the deep learning model. The data set was divided into training and testing, and the model parameters were determined with a two-stage approach. The obtained estimation results were evaluated with different performance indices. Finally, the generalization ability of the proposed model was assessed.

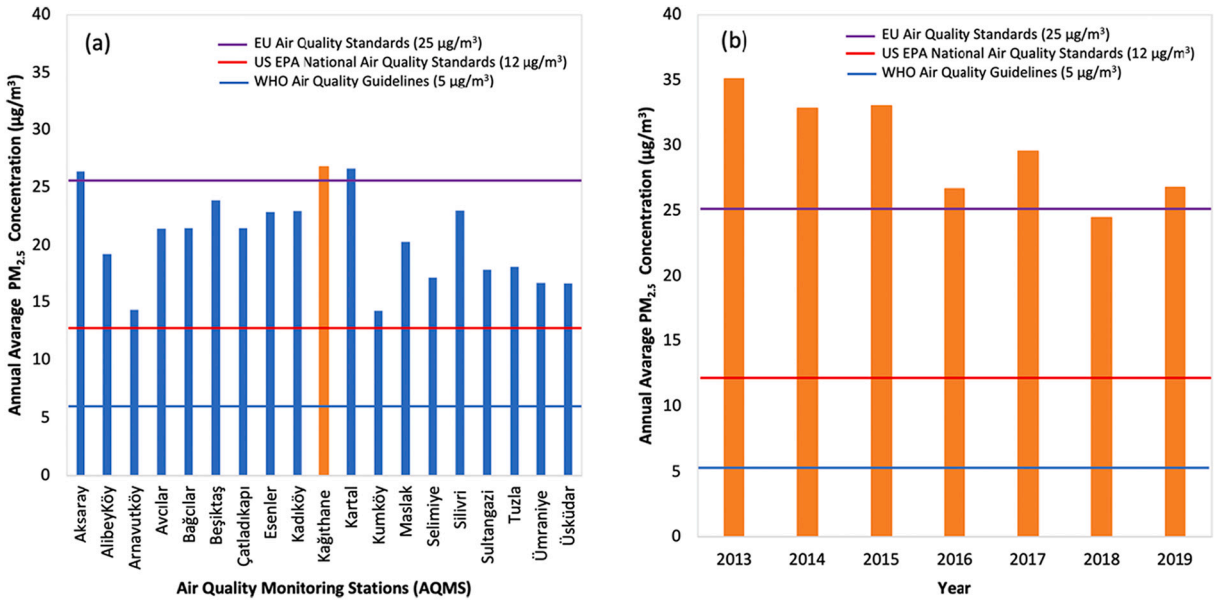


Fig. 2. Evaluation of $PM_{2.5}$ concentrations according to international standards for Istanbul metropolitan city **a)** Annual average $PM_{2.5}$ concentrations monitored AQMS in Istanbul for 2019 **b)** Annual average $PM_{2.5}$ concentrations at Kağıthane AQMS between 2013 and 2019 (SIM, 2022).

2.1. Study area

Istanbul province is situated at the Black Sea's entrance, lying on Europe and Asia separated by the Bosphorus Strait, covering nearly 5342 km^2. Istanbul has a diverse topography with altitudes ranging from 0 to 537 m. This has an impact on local climatic conditions as well as air pollution. At the end of 2021, Istanbul was Turkey's most populous urban city, with a population of 15.6 million (about one-fifth of Turkey's population), which is increasing daily. The city has the highest population density, with 3049 people per square kilometer (110 people/km^2 in the country) (TUIK, 2022). Depending on the population growth, the number of motor vehicles and residences also increases; vehicle emissions and urban heating also increase. Istanbul is also significantly affected by particulate matter pollution carried from Europe by westerly winds and dust transport from Africa over the Sahara Desert, especially by southerly winds in autumn and early spring (Ağaç, 2016). Overall, all these factors play an essential role in negatively changing the air quality of Istanbul.

Currently, 38 air quality monitoring stations (AQMS) throughout Istanbul monitor air pollution at different locations. Although it varies by station, seven parameters are generally monitored at these stations: PM_{10}, $PM_{2.5}$, SO_2, CO, NO_2, NO_x, and O_3. Only two stations, $PM_{2.5}$ air pollution, were monitored between 2013 and 2018 (Ümraniye AQMS on the Asian and Kağıthane AQMS on the European side of the city). As of 2019, the number of stations measuring $PM_{2.5}$ air pollution has reached 19 stations (SIM, 2022). Turkey's national standard for $PM_{2.5}$ air pollution has not yet been regulated, but the draft legislation aims to reduce the $PM_{2.5}$ limit value from 30 μg/m^3 annually in 2021 to 25 μg/m^3 by 2029 (Gümüşel, 2022). Therefore, the comparison of annual average $PM_{2.5}$ concentrations based on the international standard values for the Istanbul metropolitan area is illustrated in Fig. 2. The annual average $PM_{2.5}$ concentrations monitored in AQMS of Istanbul in 2019 are demonstrated in Fig. 2a. The annual average $PM_{2.5}$ concentrations of the Kağıthane station between 2013 and 2019 were selected as the study area illustrated in Fig. 2b. The annual average $PM_{2.5}$ concentrations at all stations exceeds the standards set by the US EPA, WHO, and EU air quality. The observations are slightly below the EU air quality limit value except for Aksaray, Kağıthane, and Kartal stations, as in Fig. 2a. Furthermore, Fig. 2b shows that the $PM_{2.5}$ concentrations in the Kağıthane District are consistently higher than the international limit values, and the annual average $PM_{2.5}$ concentrations in this region are currently about five times the WHO revised annual air quality guideline value of 5 μg/m^3 (WHO, 2022b). Considering this, predicting the $PM_{2.5}$ air pollution in Istanbul will enable the detection of possible adverse effects in advance and take necessary measures to protect public health by preparing various air pollution reduction scenarios, warnings, and health practices.

In this study, the District of Kağıthane was selected as the study area for model development because the number of observed air pollutant parameters and the measurement period is high, the number of missing data is relatively low, and air pollution is considerably higher in this region than in other locations. The Kağıthane District is in a narrow valley on Istanbul's European side. Kağıthane valley's elevation ranges from sea level to 130 m, with an average of 90 m. The district has a population density of 29,000 persons per

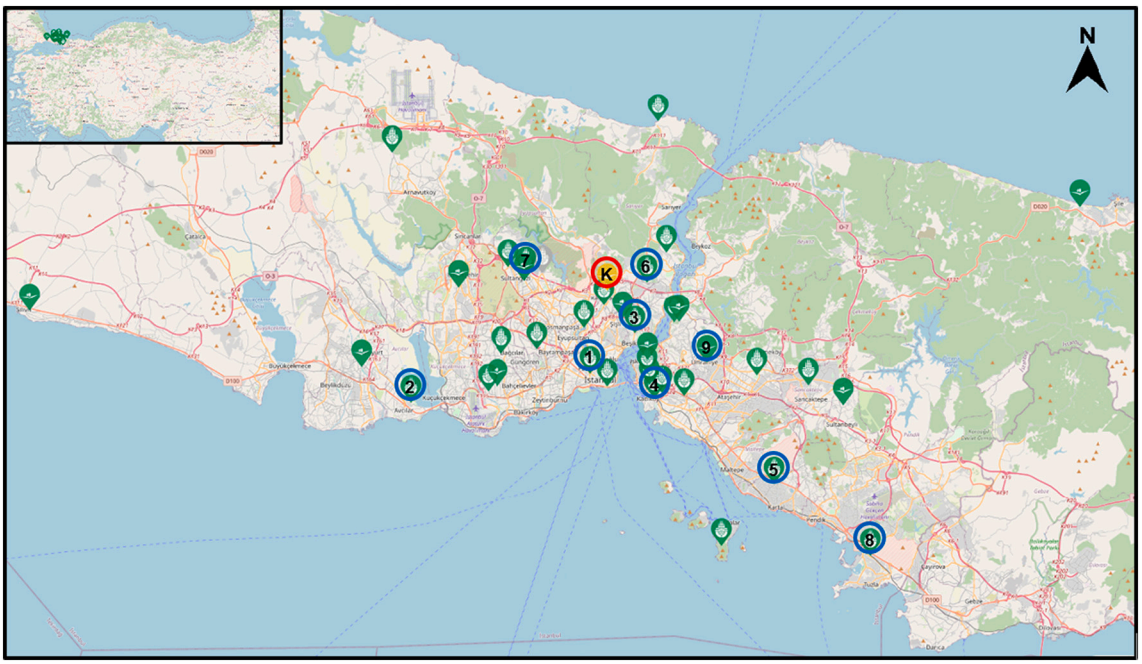


Fig. 3. Distribution of Monitoring Stations in Istanbul metropolitan city and the location of the Kağıthane (K) monitoring station (in red circle) used for model development and other nine monitoring stations (in blue circle), including Aksaray (1), Avcılar (2), Beşiktaş (3), Kadıköy (4), Kartal (5), Maslak (6), Sultangazi (7), Tuzla (8) and Ümraniye (9) district data used to test model generalization ability (AQMS, 2022). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

km^2 and a total size of 16 km^2. It is also surrounded by various industrial zones and highways (Efe et al., 2022).

2.2. Data acquisition and description

In this study, hourly time series data of air quality and meteorological values for the Kağıthane District (Istanbul-Turkey) were obtained from the Kağıthane Air Quality Monitoring Station (KAQMS) by the Ministry of Environment, Urbanization and Climate Change of Turkey (SIM, 2022) and the Turkish State Meteorological Service, respectively (TSMS, 2022). Additionally, the data sets from other stations in the city were utilized to evaluate the generalization ability of the best $PM_{2.5}$ prediction model constructed using Kağıthane district data. Fig. 3 illustrates the distribution of all monitoring stations across the city and the locations of the monitoring stations.

The data set used in this study to develop deep learning models for $PM_{2.5}$ prediction was obtained hourly from Kağıthane AQMS for around five years, from January 1st, 2015, to November 30th, 2019. The KAQMS data were utilized for model construction (training and testing) because it has data set for longer than other stations. The KAQMS has urban, traffic, and industrial source characteristics established on March 1st, 2013. The station is located 6 m from the roadway and 91 m above sea level (Latitude: 41.0923, Longitude: 28.9747). The station's measurement height is 4 m (KAQMS, 2022).

Table 1 demonstrates the descriptive statistics such as average (avg), standard deviation (SD), minimum (min) and maximum (max) of the parameters in the data set used for model development. The dataset contains some missing values commonly seen in practice because of sensor malfunction, sensor sensitivity, power outages, computer system failure, routine maintenance, human error, and other reasons. In this study, the missing values are between 3.4% and 3.7% for air quality parameters and 2.0% to 2.2% for meteorological parameters (excluding temperature). The availability of missing values in a data set might lead to bias, affecting the model's prediction performance. As a result, in this study, outliers in the data set were eliminated first, and any missing data was filled using the linear interpolation method. The total data dimension was (43,057, 10), which means that the data table contained 43,057 rows and ten columns. Then the data was divided into two portions, with 80% of the data utilized to train the models and the remaining 20% used to test the models.

A time series is a sequence of values measured in discrete or continuous-time units over a time step. Time series prediction has been an early warning and control tool in various studies. Time series analysis aims to predict future changes at observation points over time. As illustrated in Fig. 4, the data set used in this study is a typical multivariate time series that typically includes real-valued air pollutants and meteorological parameters. When the graph is examined in more detail, it is seen that there are irregularities in both air quality and meteorological data. However, temperature data only show seasonal and annual variation patterns. Air quality data is significantly associated with meteorological observation data. High wind speeds reduce $PM_{2.5}$ concentrations, whereas high humidity increases air pollution. Moreover, high atmospheric pressure is associated with good air quality (Du et al., 2021). Therefore, multivariate time series data sets, including air quality and meteorological features, are critical for air quality forecasting. Since time series consists of sequential data, sequential data should be taken to determine training and test sets. The data used to train and test the models were obtained at KAQMS from 01/01/2015 to 30/11/2019 (43,057 h). Similarly, the data sets provided by another nine monitoring stations covering the dates 01/01/2019 through 30/11/2019 (a total of 7960 h for each station) were used to evaluate the model generalization ability. The air quality dataset contains hourly averaged concentrations of air pollutants, including $PM_{2.5}$ ($\mu g/m^3$), NO ($\mu g/m^3$), NO_2 ($\mu g/m^3$), NO_X ($\mu g/m^3$), O_3 ($\mu g/m^3$) and SO_2 ($\mu g/m^3$). Among them, $PM_{2.5}$ is selected as the target feature. Similarly, the meteorological dataset includes atmospheric temperature (°C), relative humidity (%), air pressure (mbar), and wind speed (m/s) (SIM, 2019).

Due to their typically large size and multiple and heterogeneous sources, today's real-world datasets can contain erroneous data and outliers. Such data sets may negatively affect the prediction accuracy of the developed models. Therefore, data quality should be improved using the data preprocessing step before training the model to avoid bias. During the data preprocessing stage, erroneous data and outliers were initially recorded as missing data, then filled with the linear interpolation method (Zhang and Thorburn, 2022). The linear interpolation method is the simplest imputation method that assumes a linear relationship between the missing and non-missing values. This method connects two data points with a straight line. The equation of the linear interpolation function is as follows (Huang, 2021):

Table 1
Descriptive statistics of air quality and meteorological parameters from January 1st, 2015 to November 30th, 2019 (43,057 h) at Kağıthane AQMS.

Dataset	Identifier	Unit	Missing (%)	Avg	SD	Min	25%	50%	75%	Max
Air quality parameters	$PM_{2.5}$	$\mu g/m^3$	1570 (3.6%)	26.20	17.02	0.01	14.51	20.75	32.31	86.69
	SO_2	$\mu g/m^3$	1444 (3.4%)	5.91	5.13	0.03	2.34	3.92	7.83	24.30
	NO	$\mu g/m^3$	1593 (3.7%)	33.92	33.27	0.01	11.17	20.53	44.78	142.29
	NO_2	$\mu g/m^3$	1516 (3.5%)	36.53	26.45	0.01	17.22	29.42	49.18	144.89
	NO_X	$\mu g/m^3$	1545 (3.6%)	93.18	80.48	0.04	37.81	64.32	121.14	370.65
	O_3	$\mu g/m^3$	1473 (3.4%)	44.81	28.97	0.02	19.10	43.07	68.17	189.79
Meteorological factors	Temperature	°C	–	15.67	7.63	0.01	9.36	15.85	22.09	37.68
	Wind speed	m/s	884 (2.1%)	1.87	1.17	0.02	0.92	1.68	2.69	7.32
	Relative humidity	%	966 (2.2%)	77.48	16.75	12.52	66.73	80.08	91.14	100.00
	Air pressure	mbar	851 (2.0%)	1006.11	6.51	979.76	1001.70	1005.44	1010.18	1033.30

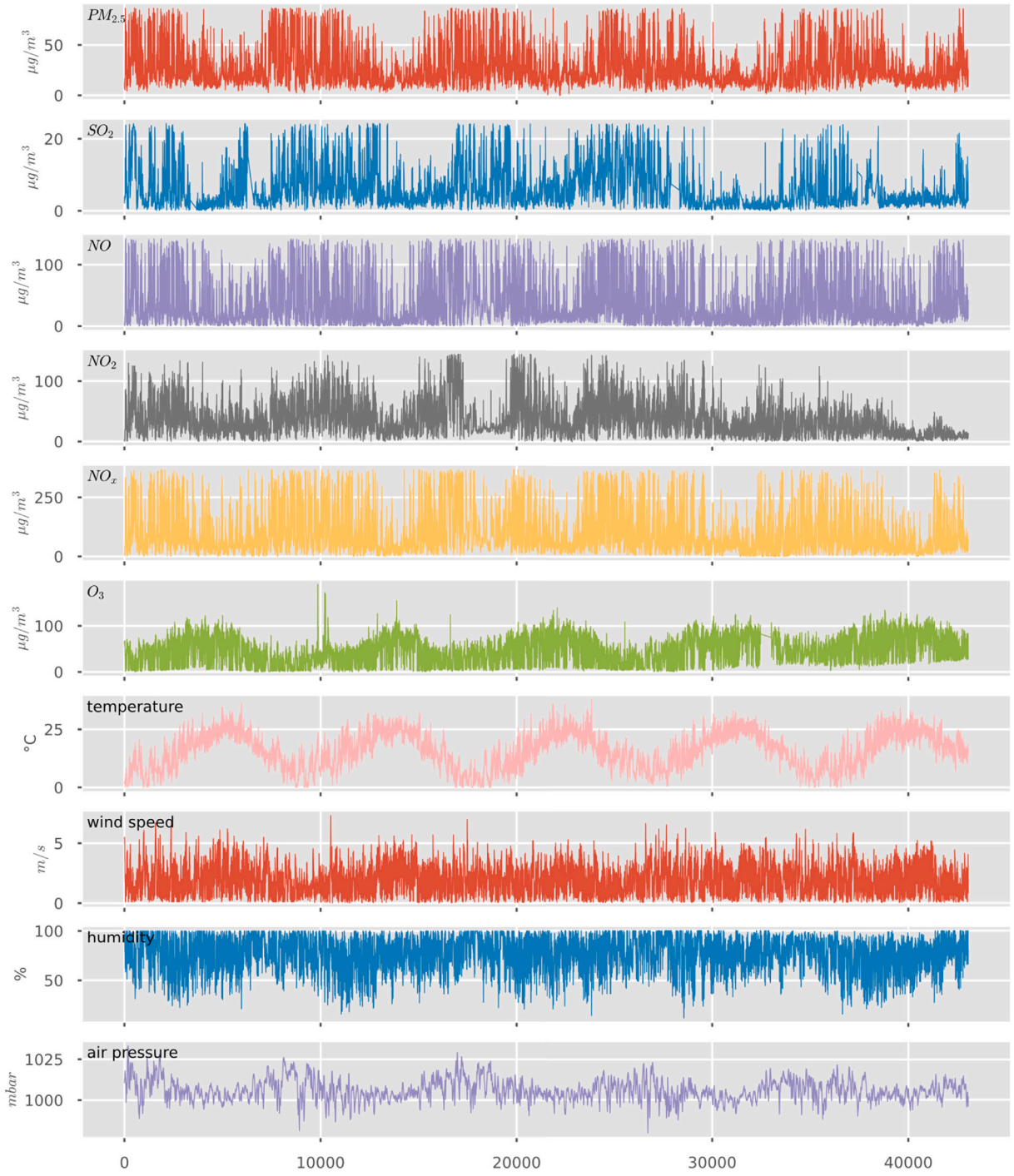


Fig. 4. Time series plots of hourly air quality and meteorological data for Kağıthane district from 01/01/2019 through 30/11/2019.

$$f_1(x) = b_0 + b_1(x - x_0) \tag{1}$$

Where x is the independent variable. x_0 is a known value of the independent variable, and $f_1(x)$ is the value of the dependent variable for a value x of the independent variable. The following formulas are used to determine b_0 and b_1.

$$b_0 = f(x_0) \tag{2}$$

$$b_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} \tag{3}$$

In order to improve the prediction accuracy, the data is then scaled to the range [0–1] using the Min-Max normalization method. The method is given in Eq. (4) (Guo et al., 2020).

$$Z_i = \frac{x_i - min(x)}{max(x) - min(x)} \tag{4}$$

Where,
$Z_i = i^{th}$ normalized value ranging from 0 to 1.
$x_i = n^{th}$ observed value for the variable x.
min(x) = minimum value in the data set.
max(x) = maximum value in the data set.

2.3. Time windowing strategy

Time-series prediction is estimating future information using past and present samples. As a result, this study's time windowing strategy (also referred to as "next hour prediction") is a supervised learning problem. The main goal of this strategy is to generate forecasting datasets from current data by windowing the original dataset to produce historical datasets ($t = 1$, $t = 2$, …,$t = 24$) and the next-hour dataset ($t = 25$). Then, the model combines all the datasets to create an entry time-series dataset that can be combined with the target dataset to predict future values (t = 25, $t = 26$, …,t = n). This strategy is summarized in Fig. 5.

2.4. Deep learning models

2.4.1. Recurrent neural network (RNN)

RNNs are a variant of ANNs that mimic human brain neural connections and aim to have a memory to store information. RNNs were developed to learn sequenced data such as voice recognition (Graves et al., 2013), in which data points are dependent on each other, and thereby, the sequence allows for the estimation of the following word (Kim and Lee, 2016; Zoph et al., 2016; Park et al., 2018), machine translation, and time series (Walid and Alamsyah, 2017; Tokgöz and Ünal, 2018; Canizo et al., 2019). According to the RNN concept, information from the previous states is also used to produce the output value. However, ANNs only output from current inputs. In other words, memory cell, a key term, has been developed so that the data points can also be used to predict the following data. This reflects the properties of the sequential data in the output variable by storing or ignoring some information.

RNNs have input, hidden, and output layers like a classical ANN (Assaad et al., 2008). As shown in Fig. 6, RNNs can be represented as a chain that affects each other and is interconnected. Each node indicates the neural network cell at a single timestep. *X* is the

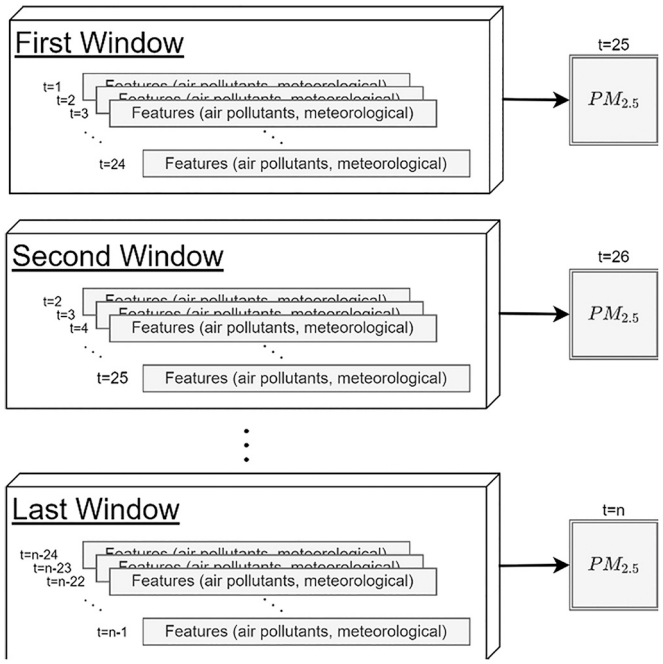


Fig. 5. Illustration of the time windowing strategy.

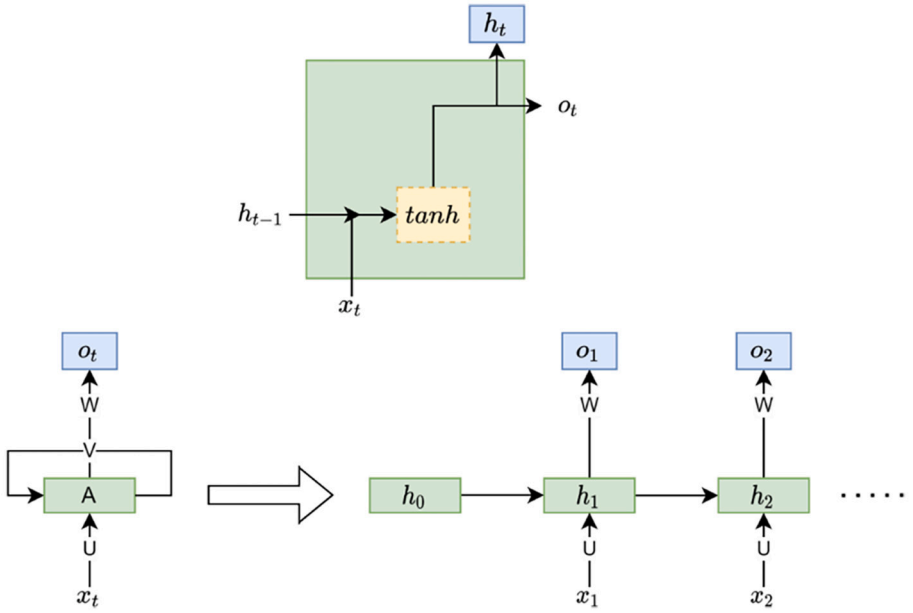


Fig. 6. A sample RNN cell and its chain structure.

features set and X_t indicates the feature's state at t time. O_t indicates the output value at t time and h_t is the hidden state. There is a connection within the RNN from the input layer to the hidden layer, from the hidden layer to the output layer. These links are illustrated by the weight matrices U, W, and V. Equations are given by:

$$h_t = tanh(Wh_{t-1} + UX_t) \tag{5}$$

$$y_t = tanh(Vh_t) \tag{6}$$

The presence of LSTM, GRU, and RNN structures in the popular DL libraries, especially TensorFlow(Abadi et al., 2016) and Keras (Chollet, 2015), has allowed us to see frequent comparison studies in the literature. RNNs can maintain memory for up to 10-time steps and pay more attention to recent information (Graves, 2012). Therefore, information earlier than 10-time steps can have little effect on predictions. Variations such as LSTM and GRU have been developed within the RNN to address drawbacks (vanishing gradient problem etc.) in RNNs (Pascanu et al., 2013). In this study, multivariate time series forecasting will be carried out on an hourly $PM_{2,5}$ dataset. The target variable, $PM_{2.5}$, is affected not only by the current state but also by previous states. For this reason, RNN and its variants LSTM and GRU DL models were preferred as methods in this study, and their performance was compared.

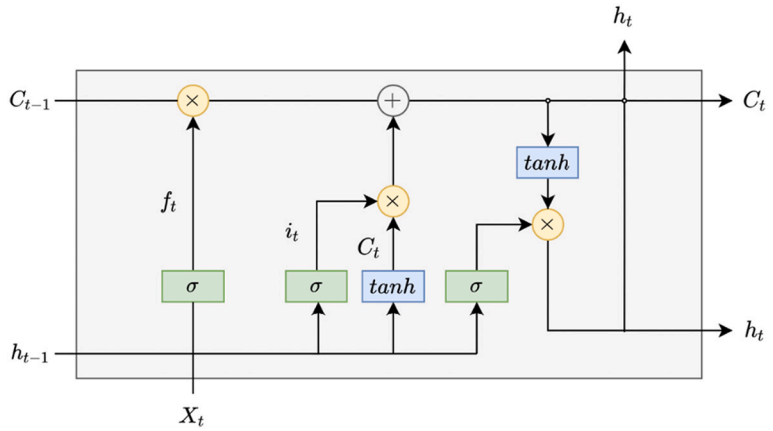


Fig. 7. The structure of LSTM ($\boldsymbol{X_t}$ indicates the values that the feature receives at $\boldsymbol{t}$ time. $\boldsymbol{H_t}$ is the output cell of the previous cell. Inside the LSTM cell, memory is indicated by $\boldsymbol{c_{t-1}}$. $\boldsymbol{W}$ is the weight matrix, and $\boldsymbol{B}$ is the term bias. The sigmoid function ($\boldsymbol{\sigma}$), hyperbolic tangent function (tanh), processes the X(t) variable, the $\boldsymbol{h}$ variable from the previous learning.

2.4.2. Long short-term memory (LSTM)

LSTM, the variant of RNNs customized for long-term series use, was first introduced to solve the gradient vanishing problem. LSTM has a proven successful application in sequential data, i.e., time series with long timesteps (Cao et al., 2019; Chimmula and Zhang, 2020). Because in the structure of the LSTM, the address is shown to the memory unit as a solution to the vanishing gradient problem (Greff et al., 2016; Abdullah et al., 2019). The first learnings are forgotten a lot towards the vanishing gradient problem. Therefore, while the initial inputs have little effect on the output, the most recent inputs have a lot of impacts (Graves, 2012).

LSTMs carry memory cells like RNNs (Abidogun, 2005). In addition, memory has been improved with three gate cells: input, forget, and output gates (Freeman et al., 2018). The input gate determines which information is transferred from the previous state to the present state. LSTM uses a series of gates and feedback loops that are self-trained on the input data. A single structure of an LSTM cell is given in Fig. 7.

$$\sigma(t) = \frac{1}{1+e^{-t}}\ tanh(t) = \left(\frac{e^{t}-e^{-t}}{e^{t}+e^{-t}}\right) \tag{7}$$

A separate multiplicative operation is performed at each gate (Eq. (8)). The f_t equation is multiplied by the forgotten variable of the previous state. Then it is determined at the input gate whether to update the variable (Eq. (9)). Finally, states go to the output gate and determine the output variable (Eq. (10)).

$$f_t = \sigma\left(W_f(\ h_{t-1}, X_t) + B_f\ \right) \tag{8}$$

$$i_t = \sigma(W_i(\ h_{t-1}, X_t) + B_i\ ) \tag{9}$$

$$o_t = \sigma(W_o(\ h_{t-1}, X_t) + B_o\ ) \tag{10}$$

2.4.3. Gated recurrent units (GRU)

The GRU unit has been developed as a variant of RNN, like LSTMs. It has a less complex structure than LSTMs. The input, forget, and exit cells in the LSTMs have been updated here as exit, update (z), and reset (r) gates, as shown in Fig. 8. Through these gates, it is determined whether the current situation will be affected by the condition in the past. In the reset gate, it is decided how to combine the input in the memory with the new input. How long before the information remains in the memory is determined by the update gate. The following equations are given for three gates of GRU:

$$r = \sigma(W_r(\ h_{t-1}, X_t) + U_r X_t\ ) \tag{11}$$

$$z = \sigma(W_z(\ h_{t-1}, X_t) + U_z X_t\ ) \tag{12}$$

$$c = tanh(W_c(\ h_{t-1}{}^{*}r) + U_c X_t\ ) \tag{13}$$

$$h_t = (z^{*}c) + ((1-z)^{*}h_{t-1}\ ) \tag{14}$$

2.5. Performance evaluation metrics

The study employs five metrics to assess the models' prediction performance and determine the efficacy of the proposed method. The evaluation metrics include Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Square Error (MSE), Mean Absolute Percentage Error (MAPE), and Coefficient of Determination (R^2). The formulation of these metrics is presented in Eqs. (15–19).

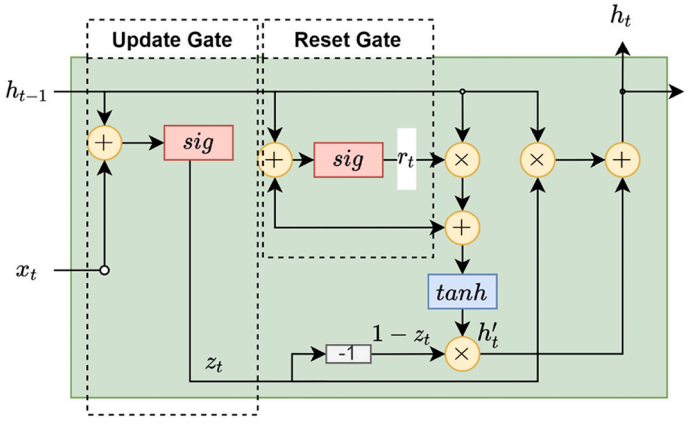


Fig. 8. The structure of Gated Recurrent Units (GRU).

Table 2
The $PM_{2.5}$ forecasting performance results of the developed models.

Stage	Models	Hidden Layer Size	Number of Neurons	Algorithms	Train MSE	Train RMSE	Train MAE	Train MAPE	Train R^2	Test MSE	Test RMSE	Test MAE	Test MAPE	Test R^2
Stage 1. LSTM model development and parameter optimization	Model#1	1	5	LSTM	27.45	5.002	3.228	0.138	0.956	28.19	5.137	3.315	0.142	0.930
	Model#2	1	10	LSTM	27.04	4.966	3.221	0.138	0.956	27.89	5.122	3.323	0.142	0.926
	Model#3	1	20	LSTM	30.79	5.262	3.405	0.154	0.935	31.56	5.394	3.490	0.158	0.912
	Model#4	1	30	LSTM	45.93	8.920	4.656	0.201	0.904	33.16	5.160	3.581	0.201	0.880
	Model#5	1	40	LSTM	30.13	5.196	3.395	0.147	0.944	31.19	5.379	3.514	0.153	0.911
	Model#6	2	5	LSTM	46.94	9.008	4.693	0.202	0.896	48.15	9.240	4.815	0.207	0.873
	Model#7	2	10	LSTM	26.93	4.941	3.219	0.135	0.963	27.27	5.003	3.260	0.136	0.951
	Model#8	2	20	LSTM	25.92	4.902	3.136	0.133	0.973	26.11	4.937	3.159	0.134	0.965
	Model#9	***2***	***30***	***LSTM***	***24.95***	***4.771***	***3.099***	***0.130***	***0.976***	***25.20***	***4.818***	***3.129***	***0.132***	***0.966***
	Model#10	2	40	LSTM	66.44	13.02	6.156	0.270	0.838	67.40	13.21	6.245	0.274	0.826
	Model#11	4	5	LSTM	168.4	33.49	13.77	0.617	0.515	169.1	33.64	13.84	0.620	0.512
	Model#12	4	10	LSTM	127.0	25.25	10.70	0.476	0.647	127.0	25.24	10.70	0.476	0.647
	Model#13	4	20	LSTM	187.3	37.45	15.21	0.682	0.461	184.8	36.94	15.00	0.673	0.468
	Model#14	4	30	LSTM	167.4	33.40	13.72	0.614	0.522	166.9	33.29	13.67	0.612	0.523
	Model#15	4	40	LSTM	206.0	41.14	16.59	0.745	0.397	204.2	40.78	16.45	0.739	0.401
Stage 2. Evaluating of different deep learning algorithms	Model#16	2	30	GRU+GRU	25.81	4.853	3.132	0.133	0.975	26.05	4.900	3.162	0.134	0.965
	Model#17	***2***	***30***	***LSTM+LSTM***	***24.60***	***4.739***	***3.056***	***0.129***	***0.980***	***24.84***	***4.785***	***3.086***	***0.130***	***0.970***
	Model#18	2	30	RNN+RNN	31.99	5.281	3.407	0.154	0.934	31.54	5.371	3.465	0.157	0.917
	Model#19	2	30	GRU+LSTM	67.86	13.15	6.239	0.278	0.835	68.35	13.241	6.283	0.280	0.828
	Model#20	2	30	GRU+RNN	26.76	4.940	3.214	0.134	0.964	27.31	5.042	3.281	0.137	0.944
	Model#21	2	30	LSTM+RNN	46.47	8.967	4.686	0.202	0.897	27.53	5.093	3.291	0.139	0.936
	Model#22	2	30	LSTM+GRU	29.19	5.099	3.305	0.141	0.949	30.10	5.258	3.409	0.145	0.920
	Model#23	2	30	RNN+GRU	32.79	5.453	3.521	0.194	0.911	33.93	5.641	3.642	0.201	0.880
	Model#24	2	30	RNN+LSTM	48.92	9.184	4.796	0.205	0.879	31.15	5.352	3.522	0.153	0.911

Hyperparameters for Model#1 - #24
Learning rate:0.001, Activation function: Relu, Optimizer: Adam, Batch size: 32, and Loss function: Mean Absolute Error.
All models were run ten times, and the average score of performance values is shown in the table.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2 \tag{15}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2}{n}} \tag{16}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\, y_i - \widehat{y}_i \,| \tag{17}$$

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \widehat{y}_i}{y_i}\right| \tag{18}$$

$$R^2 = \left[\frac{n\sum_{i=1}^{n} y_i\widehat{y}_{ii} - \left(\sum_{i=1}^{n} y_i\right)\left(\sum_{i=1}^{n}\widehat{y}_i\right)}{\sqrt{\left[n\sum_{i=1}^{n} y_i{}^2 - \left(\sum_{i=1}^{n} y_i\right)^2\right] \times \left[n\sum_{i=1}^{n}\widehat{y}_i^2 - \left(\sum_{i=1}^{n}\widehat{y}_i\right)^2\right]}}\right]^2 \tag{19}$$

Where, n is the number of observations, y_i is the predicted value and $\widehat{y}_i$ is the actual value. MSE, RMSE, MAE, and MAPE are used to evaluate the amount of error in the forecast results. The smaller values mean, the better the prediction performance of the model. R^2 is used to assess the fit of the prediction results to the actual data. The closer the value is to 1, the higher the degree of data fit and the better the prediction effect.

2.6. Model development

In this study, the proposed deep learning and baseline models are implemented using Python packages, including Keras (version 2.6.0), Tensorflow backend, and Scikit-learn (Pedregosa et al., 2011; Chollet, 2015; Abadi et al., 2016), matplotlib, and seaborn for visualizations of the results (Hunter, 2007; Waskom, 2021), Papermill for the experimental designs. Heavier workloads were run on Google Colab equipped with NVIDIA's Tesla T4 GPU. The rest of the models were executed on a computer with Intel(R) Core (TM) i7-5600U CPU @ 2.60GHz 2.59 GHz 12,0 GB RAM. The model development process is divided into two stages described below.

Stage 1 (selection of hidden layer size and numbers of neurons).

At this stage, hidden layer size and number of neuron optimizations were made to create the LSTM model used in the $PM_{2.5}$ prediction. The high number of neurons increases the computation time and memory requirement. On the other hand, a low number of neurons causes underfitting. As the number of layers increases in deep learning algorithms, the model learns better, but the back-propagation reaches the first layers less (Szandała, 2021). Other parameters for the developed model are learning rate, activation function, optimizer, epoch, batch size, and loss function.

Stage 2 (prediction performance evaluation of different deep learning algorithms).

In Stage 2, the $PM_{2.5}$ prediction performances of GRU + GRU, LSTM+LSTM, RNN + RNN, GRU + LSTM, GRU + RNN, LSTM+RNN, LSTM+GRU, RNN + GRU, RNN + LSTM algorithms are evaluated. The Models were run ten times to achieve robust evaluation results.

Table 3
The hyperparameters of the best deep learning model (Model#17).

Hyperparameters	Value
Batch Size	32
Optimizer	Adam
Loss function	Mean Absolute Error
Number of hidden layers	2
Number of neurons in each layer	30
Learning rate	0.01
Models used in the layers	LSTM + LSTM
Dropout rate	0
Epoch	50
Batch size	32
Activation function	'relu'
Train size	0.80
n hours	24
SEED	10 different seed
Input Shape	[null, 24, 10]
Duration	16:31 min

3. Experimental results and discussion

We applied a two-stage modeling approach to predict the hourly $PM_{2.5}$ concentration using deep learning algorithms. The performance results of all models developed in this study are presented in Table 2. In the first part of Table 2, it is seen that the LSTM model (model#9) with two hidden layers and 30 neurons in each layer has the best performance values for training and testing with MSE = 24.95 and 25.20, RMSE = 4.771 and 4.818, MAE = 3.099 and 3.129, MAPE = 0.13 and 0.132, and R^2 = 0.976 and 0.966, respectively ($p < 0.05$). The second part of Table 2 presents the results of the models developed using different learning algorithms. By comparing the performance results of those models, it was found that model#16 (GRU + GRU), model#17 (LSTM+LSTM), and model#20 (GRU + RNN) had performance values that were quite similar to each other. However, model#17 had the best performance values for training and testing with MSE = 24.60 and 24.84, RMSE = 4.739 and 4.785, MAE = 3.056 and 3.086, MAPE = 0.129 and 0.13, and R^2 = 0.98 and 0.97 ($p < 0.05$). Consequently, model#17 (LSTM+LSTM) outperforms other developed models in terms of prediction accuracy. On the other hand, model #19 (GRU + LSTM) had the lowest prediction performance results in this study. Table 3 summarizes the hyperparameters selected for model#17.

Violin plots of all models with R^2 value have been provided in Fig. 9. The violin plot represents the distribution and average of the experiments, in which the white colour indicates medians and interquartile ranges are indicated by boxplots with black colour. Blue and orange areas show the distribution of the performance. Accordingly, model#9 has the best result in the first stage. In the second stage, model#17 is observed to work well.

In Fig. 10, $PM_{2.5}$ predicted values obtained from the LSTM+LSTM model and actual values are compared. A comparison of the first one hundred hours of data is shown for better readability of the graph in the figures. Similarly, the scatter plot graphs for predicted and actual values are in Fig. 11. Overall, the performance of the LSTM+LSTM model is the best among the other models, as shown in Table 2 and Fig. 10a and b. In predicting accuracy, model#9 and model#17 (LSTM+LSTM) have the lowest MSE, RMSE, MAE, and MAPE and the highest R^2 values among the forecasting models and better prediction accuracy.

As shown in Table 4, the results obtained from this study were compared with the model results developed for predicting $PM_{2.5}$ concentrations by deep learning those published in the literature based on model prediction performances, and the results were evaluated. The results obtained from this study were compared with the model results developed for predicting $PM_{2.5}$ by deep learning in the literature. Huang and Kuo (2018) used APNet (CNN_LSTM) algorithm in $PM_{2.5}$ concentration prediction in Beijing, China. They compared the performance of APNet with the support vector machine, random forest, decision tree, MLP, CNN, and LSTM models. As a result, they determined the highest prediction accuracy in the CNN-LSTM model (APNet). In another study, Li et al. (2020) used the hourly values of $PM_{2.5}$ concentration and meteorological data in Beijing to predict $PM_{2.5}$ concentration for the next day. That study

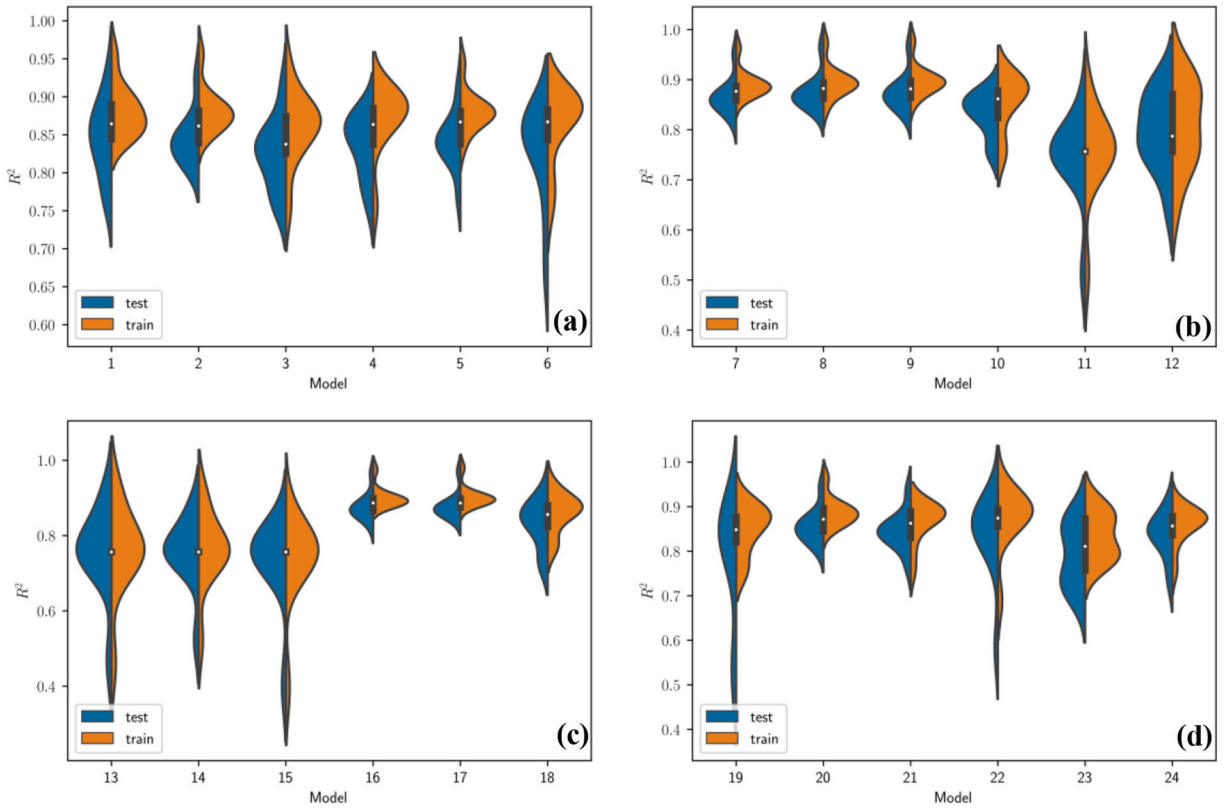


Fig. 9. Violin plots of all developed models, (a) Model#1-Model#6, (b) Model#7-Model#12, (c) Model#13-Model#18, (d) Model#19-Model#24.

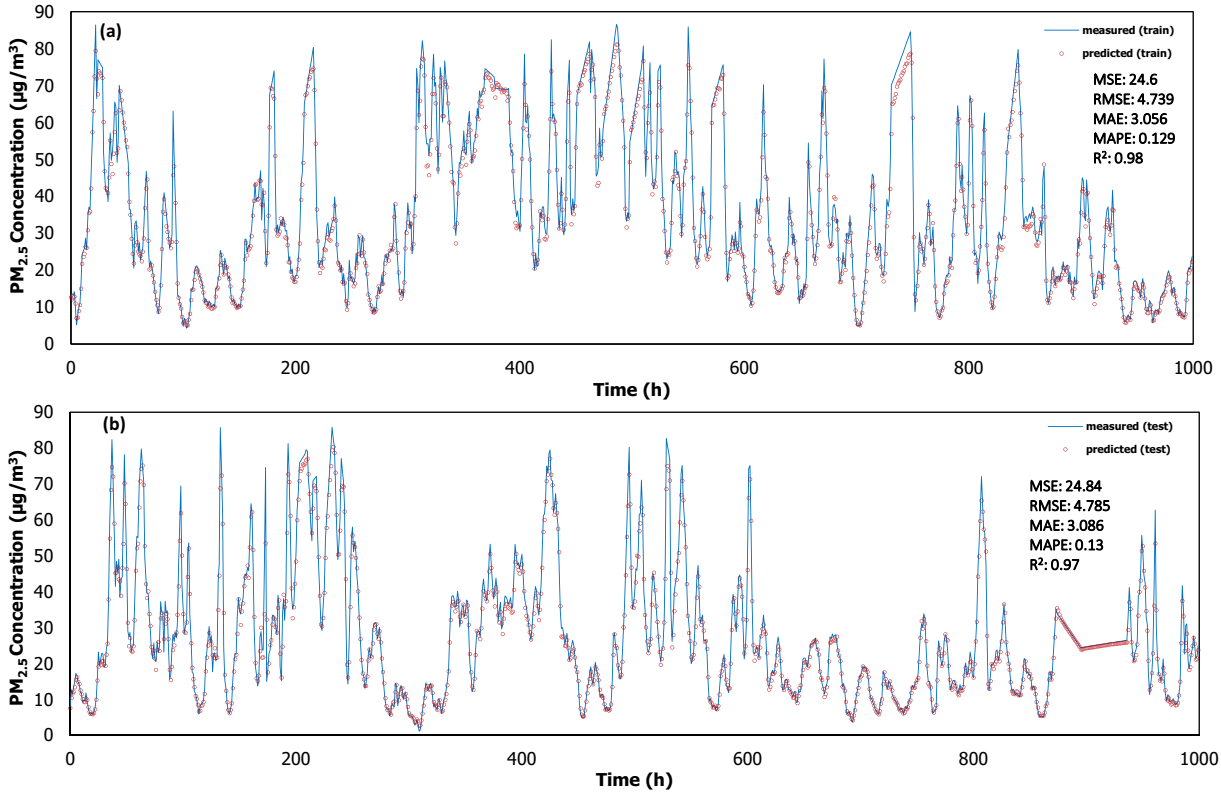


Fig. 10. Comparison of predicted vs. measured $PM_{2.5}$ values for Kağıthane District using the LSTM+LSTM model a) train data set and b) test data set from Kağıthane AQMS ($p < 0.05$).

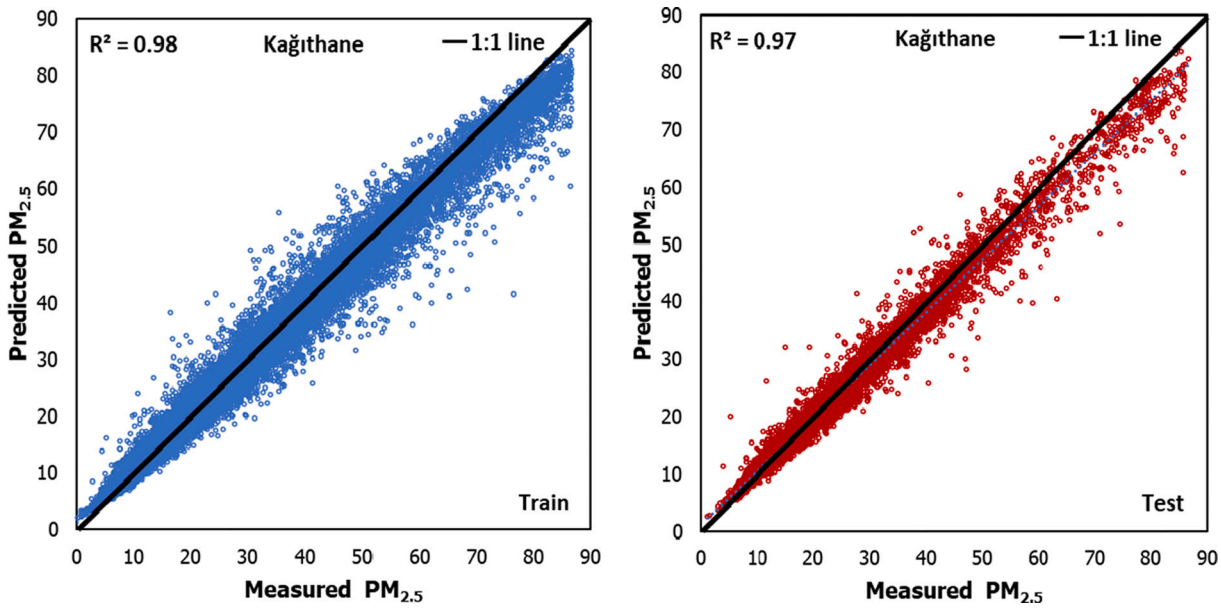


Fig. 11. Scatter plots for predicted vs. measured $PM_{2.5}$ values for Kağıthane District using the LSTM+LSTM model ($p < 0.05$).

finds that hybrid model (CNN-LSTM) performance is higher than single LSTM. Also, multivariate models' performances are higher than univariate models. Finally, Ma et al. (2019) investigated the prediction accuracy of Neural Networks (bi-directional Long Short-Term Memory- BLSTM and the transfer learning technique) in $PM_{2.5}$ prediction. They concluded that the proposed TL-BLSTM model

Table 4
Compared to existing studies published in the literature.

Reference	Study area	Method	Performance		
			MAE	RMSE	R^2
(Huang and Kuo, 2018)	$PM_{2.5}$ and meteorological data (Beijing, China).	ApNet (CNN - LSTM)	14.634	24.228	-
(Li et al., 2020)	$PM_{2.5}$ and meteorological data. (Beijing, China).	CNN-LSTM (Multivariate)	13.969	17.930	-
(Ma et al., 2019)	$PM_{2.5}$ (Guangdong, China)	TL- BLSTM	6.184	8.653	–
The proposed model	$PM_{2.5}$ and meteorological data (Kağıthane, İstanbul, Turkey)	LSTM-LSTM	3.086	4.785	0.97

performed better at the larger temporal resolutions. When all these studies are evaluated together for $PM_{2.5}$ prediction performances based on the performance index, it is understood that deep learning models are suitable for the future prediction of air pollutants. As a result, when all these studies were compared in terms of $PM_{2.5}$ prediction based on the performance index, as seen in Table 4, it was seen that the results of the proposed deep learning model were compatible with the other literature studies and had better performance results.

3.1. Evaluating the generalization ability of the developed model

In this section of the study, we assess the generalization ability of the proposed LSTM+LSTM model using data sets from nine different stations (Aksaray, Avcılar, Beşiktaş, Kadıköy, Kartal, Maslak, Sultangazi, Tuzla and Ümraniye districts of Istanbul metropolitan city) that have never been utilized for the models training and testing. The generalization performance results of the model for the stations are shown in Table 5. It is seen that the proposed LSTM+LSTM model has excellent performance values for all stations, and the best performances are obtained for the Kadıköy and Avcılar stations considering the generalization performances. When the *p*-values of the predictions are examined, all algorithms have p-value$<$0.05 which means all prediction results are acceptable. Besides, in order to test the validity of the RMSE, it is necessary to evaluate the SD values along with the RMSE performance as suggested by Ritter and Muñoz-Carpena (2013). Although the optimal value of RMSE is considered as zero, it is more likely that RMSE will be high in data sets with high SD values. In this context, the SD/RMSE ratio is presented in Table 5 to evaluate the algorithm's performance. Ritter and Muñoz-Carpena (2013) proposed four performance ratings based on the SD/RMSE ratio. If SD/RMSE value is higher than 3.2, the performance rating is “Very good”; if it is between 2.2 and 3.2, then the performance rating is “Good”; between 1.2 and 2.2 is “Acceptable”, and lower than 1.7 values are classified as “Unsatisfactory”. In this study, it is seen that the performance of Kartal district has a “Very good” performance rating, and the performance of the others has a “Good” performance rating.

In Fig. 12, scatter plot graphs for predicted versus actual values are shown for nine districts. According to those plots, there is an excellent match between the measured and predicted $PM_{2.5}$ concentrations. Consequently, the proposed LSTM+LSTM model has outstanding generalization ability for hourly $PM_{2.5}$ prediction of Istanbul metropolitan city.

4. Conclusions

One of the most effective ways to reduce health risks, environmental effects, and economic losses caused by $PM_{2.5}$ air pollution is to create a model that can predict future $PM_{2.5}$ concentrations. This study developed a deep-learning model using LSTM, RNN, and GRU deep-learning algorithms to predict hourly $PM_{2.5}$ concentrations. Among the developed models, it has been observed that the LSTM+LSTM model has the best prediction performance compared to other models and can predict hourly $PM_{2.5}$ concentrations with high accuracy based on the performance index. In addition, the developed model was applied to data from 9 different regions and discovered that it had outstanding generalization ability. Moreover, the proposed model's performance results were compared to the published studies in the literature, and it was seen that the proposed model outperformed the others. Overall, by estimating $PM_{2.5}$ air pollution using the proposed model, possible adverse effects will be determined, and necessary measures will be taken to protect public health by preparing various air pollution reduction scenarios and warnings.

On the other hand, this study has some limitations. Firstly, the effectiveness of the proposed model for other air pollutants, except $PM_{2.5,}$ has not been studied. Secondly, the feature importance ranking of the input parameters has not been evaluated. All these

Table 5
The generalization performances of the LSTM+LSTM model for nine different stations ($p < 0.05$).

Station	MSE	RMSE	MAE	MAPE	R^2	SD	SD/RMSE
Aksaray	29.41	5.531	3.570	0.152	0.932	16.220	2.932
Avcılar	28.52	5.364	3.462	0.147	0.961	12.512	2.262
Beşiktaş	30.02	5.646	3.644	0.155	0.913	13.023	2.355
Kadikoy	28.46	5.353	3.455	0.147	0.963	14.118	2.552
Kartal	30.09	5.659	3.652	0.155	0.911	21.257	3.843
Maslak	29.70	5.585	3.605	0.153	0.923	13.209	2.388
Sultangazi	29.99	5.640	3.640	0.155	0.914	12.762	2.307
Tuzla	30.32	5.702	3.681	0.156	0.904	14.476	2.617
Ümraniye	30.06	5.652	3.648	0.155	0.912	14.025	2.536

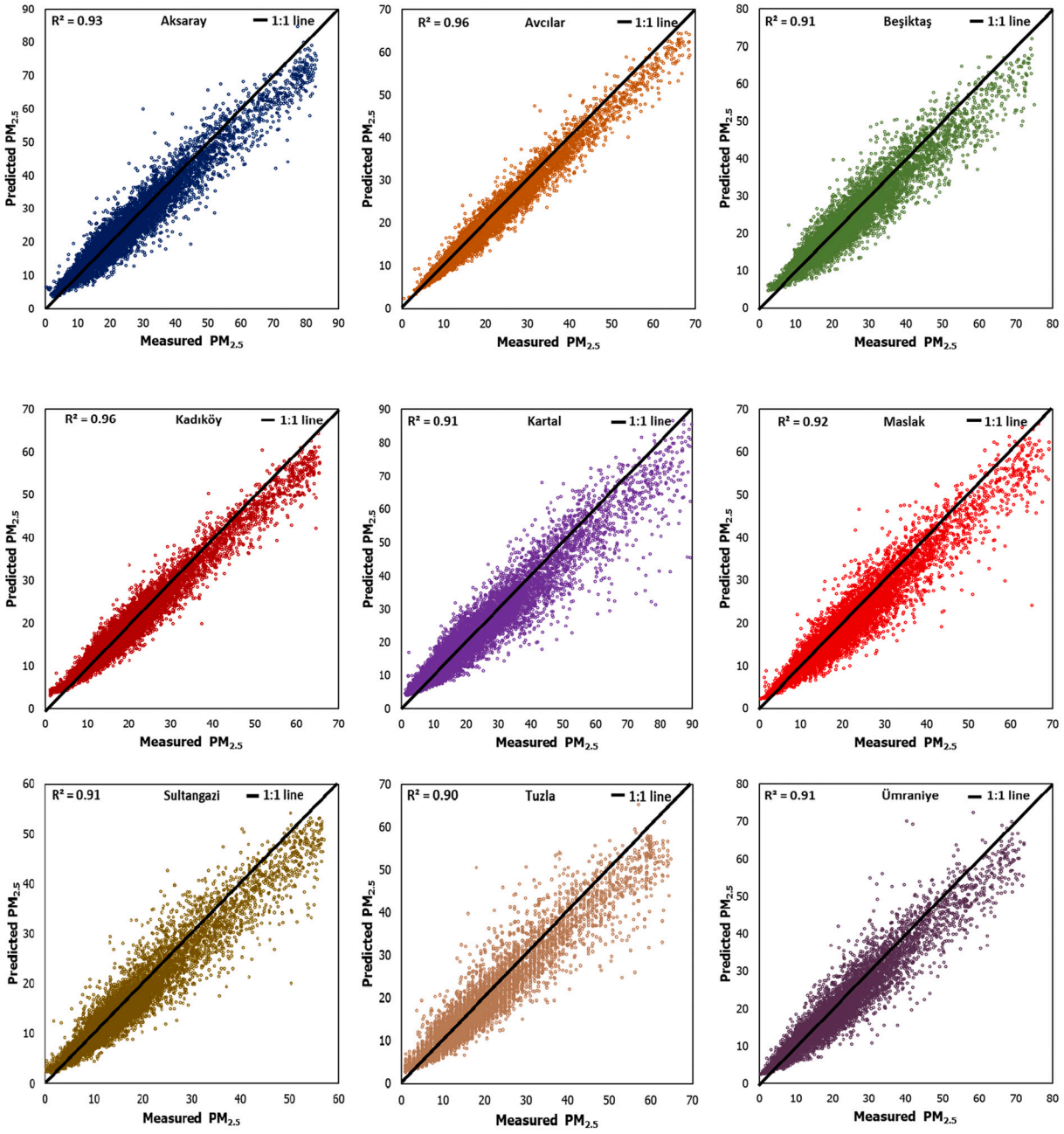


Fig. 12. Scatter plots for predicted vs. measured PM_{10}values for validated stations using the LSTM models ($p < 0.05$).

limitations are planned for future studies.

CRediT authorship contribution statement

Beytullah Eren: Conceptualization, Methodology, Resources, Software, Writing – review & editing, Visualization, Validation, Supervision. **İpek Aksangür:** Data curation, Writing – original draft, Formal analysis. **Caner Erden:** Methodology, Writing – review & editing, Software, Data curation, Visualization.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to

influence the work reported in this paper.

Data availability

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Acknowledgments

The authors thank the Ministry of Environment and Urban Planning and the Turkish State Meteorological Service for providing the air quality and meteorological dataset for this study. The authors are thankful to Professor Fatih Karadağlı for his reviews and suggestions to improve the organization of this study.

References

Abadi, M., Barham, P., Chen, J., et al., 2016. Tensorflow: a system for large-scale machine learning. In: 12th USENIX Symposium on Operating Systems Design and Implementation, pp. 265–283.
Abdullah, S., Ismail, M., Ahmed, A.N., Abdullah, A.M., 2019. Forecasting particulate matter concentration using linear and non-linear approaches for air quality decision support. Atmosphere 10, 667.
Abidogun, O.A., 2005. Data Mining, Fraud Detection and Mobile Telecommunications: Call Pattern Analysis with Unsupervised Neural Networks. PhD Thesis,. University of the Western Cape.
Ağaç, K., 2016. İstanbul Kağıthane Bölgesinde Yüksek PM10 Konsantrasyonlarının Meteorolojik Olarak Incelenmesi. Doctoral Dissertation. Istanbul Teknik Üniversitesi.
AQMS, 2022. Hava Kalitesi İzleme Merkezi. Hava Kalitesi İzleme Projesi - İstanbul Büyükşehir Belediyesi. https://havakalitesi.ibb.gov.tr/Pages/AirQuality. Accessed 9 May 2022.
Assaad, M., Boné, R., Cardot, H., 2008. A new boosting algorithm for improved time-series forecasting with recurrent neural networks. Inform. Fusion 9, 41–55.
Bai, L., Wang, J., Ma, X., Lu, H., 2018. Air pollution forecasts: an overview. IJERPH 15, 780. https://doi.org/10.3390/ijerph15040780.
Baklanov, A., Smith Korsholm, U., Nuterman, R., et al., 2017. Enviro-HIRLAM online integrated meteorology–chemistry modelling system: strategy, methodology, developments and applications (v7.2). Geosci. Model Dev. 10, 2971–2999. https://doi.org/10.5194/gmd-10-2971-2017.
Bekkar, A., Hssina, B., Douzi, S., Douzi, K., 2021. Air-pollution prediction in smart city, deep learning approach. J. Big Data 8, 161. https://doi.org/10.1186/s40537-021-00548-1.
Bui, T.-C., Le, V.-D., Cha, S.-K., 2018. A Deep Learning Approach for Forecasting Air Pollution in South Korea Using LSTM arXiv:180407891 [cs, stat].
Cabaneros, S.M., Calautit, J.K., Hughes, B., 2019. A review of artificial neural network models for ambient air pollution prediction. Environ. Model. Softw. 119, 285–304. https://doi.org/10.1016/j.envsoft.2019.06.014.
Canizo, M., Triguero, I., Conde, A., Onieva, E., 2019. Multi-head CNN–RNN for multi-time series anomaly detection: an industrial case study. Neurocomputing 363, 246–260.
Cao, J., Li, Z., Li, J., 2019. Financial time series forecasting model based on CEEMDAN and LSTM. Phys. A: Statist. Mech. Appl. 519, 127–139.
Chimmula, V.K.R., Zhang, L., 2020. Time series forecasting of COVID-19 transmission in Canada using LSTM networks. Chaos, Solitons Fractals 135, 109864.
Chollet, F., 2015. Keras. https://github.com/fchollet/keras.
Czernecki, B., Marosz, M., Jędruszkiewicz, J., 2021. Assessment of machine learning algorithms in short-term forecasting of PM10 and PM2.5 concentrations in selected polish agglomerations. Aerosol Air Qual. Res. 21, 200586 https://doi.org/10.4209/aaqr.200586.
Du, S., Li, T., Yang, Y., Horng, S.-J., 2021. Deep air quality forecasting using hybrid deep learning framework. IEEE Trans. Knowl. Data Eng. 33, 2412–2424. https://doi.org/10.1109/TKDE.2019.2954510.
Du, P., Wang, J., Yang, W., Niu, T., 2022. A novel hybrid fine particulate matter (PM2.5) forecasting and its further application system: case studies in China. J. Forecast. 41, 64–85. https://doi.org/10.1002/for.2785.
Efe, B., Öztaner, Y.B., Deniz, A., Unal, A., 2022. Analysis of air pollutants in Kagithane valley and Istanbul metropolitan area. Air Qual. Atmos. Health 15, 1027–1041. https://doi.org/10.1007/s11869-022-01184-1.
Freeman, B.S., Taylor, G., Gharabaghi, B., Thé, J., 2018. Forecasting air quality time series using deep learning. J. Air Waste Manage. Assoc. 68, 866–886. https://doi.org/10.1080/10962247.2018.1459956.
Graves, A., 2012. Supervised sequence labelling. In: Supervised Sequence Labelling with Recurrent Neural Networks. Springer, pp. 5–13.
Graves, A., Jaitly, N., Mohamed, A., 2013. Hybrid speech recognition with deep bidirectional LSTM. In: 2013 IEEE Workshop on Automatic Speech Recognition and Understanding. IEEE, pp. 273–278.
Greff, K., Srivastava, R.K., Koutník, J., et al., 2016. LSTM: a search space odyssey. IEEE Trans. Neural Networks Learn. Sys. 28, 2222–2232.
Guan, Y., Xiao, Y., Rong, B., et al., 2021. Long-term health impacts attributable to PM2.5 and ozone pollution in China's most polluted region during 2015–2020. J. Clean. Prod. 321, 128970 https://doi.org/10.1016/j.jclepro.2021.128970.
Gümüşel, Y.D., 2022. ÇİSİP: Hava Kirliliği Azalmıyor, Türkiye'nin 2029 Hedefi DSÖ Tavsiyesinin 5 Katı - Temiz Hava Hakkı. In: Temiz Hava Hakkı. https://www.temizhavahakki.com/cisip-turkiyenin-2029-hedefi-dso-tavsiyesinin-5-kati/. Accessed 11 May 2022.
Guo, C., Liu, G., Chen, C.-H., 2020. Air pollution concentration forecast method based on the deep ensemble neural network. Wirel. Commun. Mob. Comput. 2020, 1–13. https://doi.org/10.1155/2020/8854649.
Huang, G., 2021. Missing data filling method based on linear interpolation and lightgbm. J. Phys. Conf. Ser. 1754, 012187 https://doi.org/10.1088/1742-6596/1754/1/012187.
Huang, C.-J., Kuo, P.-H., 2018. A deep CNN-LSTM model for particulate matter (PM2.5) forecasting in smart cities. Sensors 18, 2220. https://doi.org/10.3390/s18072220.
Hunter, J.D., 2007. Matplotlib: a 2D graphics environment. Comput. Sci. Eng. 9, 90–95. https://doi.org/10.1109/MCSE.2007.55.
Isaev, E., Ajikeev, B., Shamyrkanov, U., et al., 2022. Impact of climate change and air pollution forecasting using machine learning techniques in Bishkek. Aerosol Air Qual. Res. 22, 210336 https://doi.org/10.4209/aaqr.210336.
KAQMS, 2022. Kagithane (Urban). Air Quality Monitoring Project - Istanbul Metropolitan Municipality. https://havakalitesi.ibb.gov.tr/Pages/AirQualityDetails/d9f15eae-128a-4a71-ad30-141b4728a520. Accessed 11 May 2022.
Karimian, H., Li, Q., Wu, C., et al., 2019a. Evaluation of different machine learning approaches to forecasting PM2.5 mass concentrations. Aerosol Air Qual. Res. 19, 1400–1410. https://doi.org/10.4209/aaqr.2018.12.0450.
Karimian, H., Li, Q., Wu, C., et al., 2019b. Evaluation of different machine learning approaches to forecasting PM2.5 mass concentrations. Aerosol Air Qual. Res. 19, 1400–1410. https://doi.org/10.4209/aaqr.2018.12.0450.
Kim, H., Lee, J.-H., 2016. A recurrent neural networks approach for estimating the quality of machine translation output. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 494–498.
Leong, W.C., Kelani, R.O., Ahmad, Z., 2020. Prediction of air pollution index (API) using support vector machine (SVM). J. Environment. Chem. Eng. 8, 103208 https://doi.org/10.1016/j.jece.2019.103208.

Li, T., Hua, M., Wu, X., 2020. A hybrid CNN-LSTM model for forecasting particulate matter (PM2.5). IEEE Access 8, 26933–26940. https://doi.org/10.1109/ACCESS.2020.2971348.
Lu, X., Sha, Y.H., Li, Z., et al., 2021. Development and application of a hybrid long-short term memory – three dimensional variational technique for the improvement of PM2.5 forecasting. Sci. Total Environ. 770, 144221 https://doi.org/10.1016/j.scitotenv.2020.144221.
Ma, J., Cheng, J.C.P., Lin, C., et al., 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., Yu, Z., Qu, Y., et al., 2020. Application of the XGBoost machine learning method in PM2.5 prediction: a case study of Shanghai. Aerosol Air Qual. Res. 20, 128–138. https://doi.org/10.4209/aaqr.2019.08.0408.
Menares, C., Perez, P., Parraguez, S., Fleming, Z.L., 2021. Forecasting PM2.5 levels in Santiago de Chile using deep learning neural networks. Urban Clim. 38, 100906 https://doi.org/10.1016/j.uclim.2021.100906.
Miller, L., Xu, X., 2018. Ambient PM2.5 human health effects—findings in China and research directions. Atmosphere 9, 424. https://doi.org/10.3390/atmos9110424.
Mir, A.A., Kearfott, K.J., Çelebi, F.V., Rafique, M., 2022. Imputation by feature importance (IBFI): a methodology to envelop machine learning method for imputing missing patterns in time series data. PLoS One 17, e0262131. https://doi.org/10.1371/journal.pone.0262131.
Mukherjee, A., Agrawal, M., 2017. World air particulate matter: sources, distribution and health effects. Environ. Chem. Lett. 15, 283–309. https://doi.org/10.1007/s10311-017-0611-9.
Nerobelov, G., Timofeyev, Y., Smyshlyaev, S., et al., 2021. Validation of WRF-Chem model and CAMS performance in estimating near-surface atmospheric CO2 mixing ratio in the area of Saint Petersburg (Russia). Atmosphere 12, 387. https://doi.org/10.3390/atmos12030387.
Pak, U., Ma, J., Ryu, U., et al., 2020. Deep learning-based PM2.5 prediction considering the spatiotemporal correlations: a case study of Beijing, China. Sci. Total Environ. 699, 133561 https://doi.org/10.1016/j.scitotenv.2019.07.367.
Park, H., Cho, S., Park, J., 2018. Word RNN as a baseline for sentence completion. In: 2018 IEEE 5th International Congress on Information Science and Technology (CiSt). IEEE, pp. 183–187.
Pascanu, R., Mikolov, T., Bengio, Y., 2013. On the difficulty of training recurrent neural networks. In: International Conference on Machine Learning. PMLR, pp. 1310–1318.
Pedregosa, F., Varoquaux, G., Gramfort, A., et al., 2011. Scikit-learn: machine learning in Python. J. Mach. Learn. Res. 12, 2825–2830.
Perez, P., Menares, C., 2018. Forecasting of hourly PM2.5 in south-west zone in Santiago de Chile. Aerosol Air Qual. Res. 18, 2666–2679. https://doi.org/10.4209/aaqr.2018.01.0029.
Quinteros, M.E., Lu, S., Blazquez, C., et al., 2019. Use of data imputation tools to reconstruct incomplete air quality datasets: a case-study in Temuco, Chile. Atmos. Environ. 200, 40–49. https://doi.org/10.1016/j.atmosenv.2018.11.053.
Ritter, A., Muñoz-Carpena, R., 2013. Performance evaluation of hydrological models: statistical significance for reducing subjectivity in goodness-of-fit assessments. J. Hydrol. 480, 33–45. https://doi.org/10.1016/j.jhydrol.2012.12.004.
Rubal, Kumar D., 2018. Evolving differential evolution method with random forest for prediction of air pollution. Proced. Comput. Sci. 132, 824–833. https://doi.org/10.1016/j.procs.2018.05.094.
Samal, K.K.R., Babu, K.S., Das, S.K., 2021. Multi-directional temporal convolutional artificial neural network for PM2.5 forecasting with missing values: a deep learning approach. Urban Clim. 36, 100800 https://doi.org/10.1016/j.uclim.2021.100800.
SIM, 2019. Kağıthane 2 (Kentsel). Hava Kalitesi İzleme Projesi - İstanbul Büyükşehir Belediyesi. https://havakalitesi.ibb.istanbul/Pages/AirQualityDetails/d9f15eae-128a-4a71-ad30-141b4728a520. Accessed 21 Feb 2022.
SIM, 2022. SIM (Continuous Monitoring Center). T.C. Ministry of Environment, Urbanisation and Climate Change. https://www.havaizleme.gov.tr/Services/AirQuality/Default.ltr.aspx. Accessed 27 Dec 2021.
Srivastava, N., Blond, N., 2022. Impact of meteorological parameterization schemes on CTM model simulations. Atmos. Environ. 268, 118832 https://doi.org/10.1016/j.atmosenv.2021.118832.
Szandała, T., 2021. Review and comparison of commonly used activation functions for deep neural networks. In: Bhoi, A.K., Mallick, P.K., Liu, C.-M., Balas, V.E. (Eds.), Bio-Inspired Neurocomputing. Springer, Singapore, pp. 203–224.
Tokgöz, A., Ünal, G., 2018. A RNN based time series approach for forecasting turkish electricity load. In: 26th Signal Processing And Communications Applications Conference (SIU). IEEE, pp. 1–4.
TSMS, 2022. Turkish State Meteorological Service. https://mgm.gov.tr/eng/about.aspx. Accessed 11 May 2022.
Tucker, W.G., 2000. An overview of PM2.5 sources and control strategies. Fuel Process. Technol. 65–66, 379–392. https://doi.org/10.1016/S0378-3820(99)00105-8.
TUIK, 2022. Address Based Population Registration System Results for 2021. https://data.tuik.gov.tr/Bulten/Index?p=Adrese-Dayali-Nufus-Kayit-Sistemi-Sonuclari-2021-45500. Accessed 10 May 2022.
US EPA O, 2014. Criteria Air Pollutants. https://www.epa.gov/criteria-air-pollutants. Accessed 27 Apr 2022.
Walid, A., Alamsyah, I.U., 2017. Recurrent neural network for forecasting time series with long memory pattern. In: J. Phys. Conf. Ser, 012038.
Wang, Z., Maeda, T., Hayashi, M., et al., 2001. A nested air quality prediction modeling system for urban and regional scales: application for high-ozone episode in Taiwan. Water Air Soil Pollut. 130, 391–396. https://doi.org/10.1023/A:1013833217916.
Wang, L., Zhang, Y., Wang, K., et al., 2016. Application of weather research and forecasting model with chemistry (WRF/Chem) over northern China: sensitivity study, comparative evaluation, and policy implications. Atmos. Environ. 124, 337–350. https://doi.org/10.1016/j.atmosenv.2014.12.052.
Waskom, M.L., 2021. seaborn: statistical data visualization. J. Open Source Software 6, 3021. https://doi.org/10.21105/joss.03021.
WHO, 2022a. World Health Organization. "Burden of Disease from the Joint Effects of Household and Ambient Air Pollution for 2016". v2 May 2018.
WHO, 2022b. WHO Air Quality Guideline Values. https://www.who.int/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health. Accessed 9 May 2022.
Yan, R., Liao, J., Yang, J., et al., 2021. Multi-hour and multi-site air quality index forecasting in Beijing using CNN, LSTM, CNN-LSTM, and spatiotemporal clustering. Expert Syst. Appl. 169, 114513 https://doi.org/10.1016/j.eswa.2020.114513.
Yu, Y., Zou, W.W., Jerrett, M., Meng, Y.-Y., 2022. Acute health impact of convectional and wildfire-related PM2.5: a narrative review. Environment. Adv. 100179 https://doi.org/10.1016/j.envadv.2022.100179.
Zakarin, E.A., Baklanov, A.A., Balakay, L.A., et al., 2021. Simulation of air pollution in Almaty City under adverse weather conditions. Russ. Meteorol. Hydrol. 46, 121–128. https://doi.org/10.3103/S1068373921020072.
Zhang, Y., Thorburn, P.J., 2022. Handling missing data in near real-time environmental monitoring: a system and a review of selected methods. Futur. Gener. Comput. Syst. 128, 63–72. https://doi.org/10.1016/j.future.2021.09.033.
Zhou, L., Schwede, D.B., Wyat Appel, K., et al., 2019. The impact of air pollutant deposition on solar energy system efficiency: an approach to estimate PV soiling effects with the community multiscale air quality (CMAQ) model. Sci. Total Environ. 651, 456–465. https://doi.org/10.1016/j.scitotenv.2018.09.194.
Zoph, B., Vaswani, A., May, J., Knight, K., 2016. Simple, fast noise-contrastive estimation for large rnn vocabularies. In: Proceedings of the 2016 conference of the north American chapter of the Association for Computational Linguistics: human language technologies, pp. 1217–1222.