Araştırma Makalesi https://doi.org/10.46810/tdfd.1201415 Research Article

Türk Doğa ve Fen Dergisi
Turkish Journal of Nature and Science
www.dergipark.gov.tr/tdfd

Deep Learning Based Air Quality Prediction: A Case Study for London

Anıl UTKU[1], Umit CAN[1*]

[1] Munzur University, Engineering Faculty, Computer Engineering Department, Tunceli, Türkiye
Anıl UTKU ORCID No: 0000-0002-7240-8713
Umit CAN ORCID No: 0000-0002-8832-6317

**Corresponding author: ucan@munzur.edu.tr*

(Received: 8.11.2022, Accepted: 13.12.2022, Online Publication: 28.12.2022)

Keywords
Deep learning,
London,
$PM_{2.5}$ prediction,
Machine learning,
Air quality prediction

Abstract: Although states take various measures to prevent air pollution, air pollutants continue to exist as an important problem in the world. One air pollutant that seriously affects human health is called $PM_{2.5}$ (particles smaller than 2.5 micrometers in diameter). These particles pose a serious threat to human health. For example, it can penetrate deep into the lung, irritate and erode the alveolar wall and consequently impair lung function. From this, the event $PM_{2.5}$ prediction is very important. In this study, $PM_{2.5}$ the prediction was made using 12 models, namely, Decision Tree (DT), Extra Tree (ET), k-Nearest Neighbourhood (k-NN), Linear Regression (LR), Random Forest (RF), Support Vector Machine (SVM), Extreme Gradient Boosting (XGBoost), Multi-Layer Perceptron (MLP), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Gated Recurrent Unit (GRU), and Long Short-Term Memory (LSTM) models. The LSTM model developed according to the results obtained achieved the best result in terms of MSE, RMSE, MAE, and R^2 metrics.

Derin Öğrenme Tabanlı Hava Kalitesi Tahmini: Londra İçin Bir Vaka Çalışması

Anahtar Kelimeler
Derin öğrenme,
Londra,
$PM_{2.5}$ tahmini,
Makine öğrenmesi,
Hava kalitesi

Öz: Hava kirliliğini önlemek için her ne kadar devletler çeşitli önlemler alsada dünyada hava kirleticileri önemli bir problem olarak varlığını sürdürmektedir. İnsan sağlığına ciddi etkileri bulunan hava kirleticilerinden biri ise $PM_{2.5}$ (çapı 2,5 mikrometreden küçük partiküller) olarak adlandırılır. Bu patiküller insan sağlığını ciddi tehdit etmektedir. Örneğin akciğere derinlemesine nüfuz edebilir, alveol duvarını tahriş edebilir ve aşındırabilir ve sonuç olarak akciğer fonksiyonunu bozabilir. Bundan olayı $PM_{2.5}$ tahmini çok önemlidir. Bu çalışmada Decision Tree (DT), Extra Tree (ET), k-Nearest Neighbourhood (k-NN), Linear Regression (LR), Random Forest (RF), Support Vector Machine (SVM), Extreme Gradient Boosting (XGBoost), Multi-Layer Perceptron (MLP), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Gated Recurrent Unit (GRU) ve Long Short-Term Memory (LSTM) modelleri olmak üzere toplam 12 model kullanılarak $PM_{2.5}$ tahmini yapılmıştır. Elde edilen sonuçlara göre geliştirilen LSTM modeli MSE, RMSE, MAE ve R^2 metrikleri cinsinden en iyi sonucu elde etmiştir.

1. INTRODUCTION

As a result of the high population density and increasing rapid industrialization throughout the world, an increase in the emission of air pollutants has inevitably occurred. Gaseous pollutants and particulate matter, which are very small particles, form air pollutants. The different properties of these tiny particles (PM), including their size, determine the impact power of PM. Defined as a PM group, $PM_{2.5}$, despite its small diameter length, has a large surface area and therefore can transport various toxic substances, pass through the filtration of the nasal hairs, reach the end of the respiratory tract with the airflow, and accumulate there. Thus, it can damage other parts of the body through air exchange in the lungs [1]. It has also been shown that long-term exposure to $PM_{2.5}$, is associated with fatal outcomes by increasing the incidence of diseases such as heart disease (16% increase) and stroke (14% increase) [2].

Increasing urbanization plays a vital role in public health exposure to deadly problems. As a result of industrial activities, intense air pollution occurs. Since people are exposed to $PM_{2.5}$ for a long time, lung cancer, heart disease, and mortality rates increase. Air pollution prediction become a popular area of research. Besides $PM_{2.5}$, the primary air pollutants in urban areas are carbon dioxide (CO2), carbon monoxide (CO), nitrogen dioxide (NO2) and nitrogen monoxide (NO), and PM10 particulate matter. Numerous control measures can be taken to reduce $PM_{2.5}$ emissions. For example, industrial production vehicles equipped with new technologies that reduce the emission of air pollutants can be used. In cities, individual heating technologies can be updated instantly. By reducing the use of lump coal, the use of coal with low gas emission rates can be encouraged or incentives can be applied to switch to cleaner fuels such as natural gas. Steps are also needed to reduce dust from construction sites, including promoting more green spaces [3].

Accurate prediction of $PM_{2.5}$ concentration has an important place in determining air quality and taking necessary precautions in this regard. Accurate and effective predictions can guide policymakers in this regard. In the literature, statistical models, deterministic, and machine learning-based methods are used as the basis for $PM_{2.5}$ prediction. Machine learning methods achieve very successful results because they can successfully use complex linear and non-linear relationships. For example, Ma et al. [4] achieved better results on daily $PM_{2.5}$ prediction with the XGBoost method than the WRF-Chem model, which is a deterministic model.

Effective planning is needed to control the emission levels of air pollutants to keep air quality at a high level. From this point of view, it is essential for the success of these plans to make an accurate prediction of $PM_{2.5}$ concentration for the future. For example, a study conducted in Delhi by Masood and Ahmad [5] tried to predict $PM_{2.5}$ by using air quality values covering two years. Two different models, SVM and Artificial Neural Network (ANN) were used for this prediction. According to the results obtained, the prediction performance of ANN is higher than that of SVM. Danesh Yazdi et al. [6] used an ensemble machine learning method to predict $PM_{2.5}$ intensity in the Greater London area. The ensemble method included RF, k-NN, and Gradient Boosting Machine (GBM) methods, and successful results were obtained. The sparseness of meteorological vertical observations has resulted in limitations in the forward prediction of air and air pollution. $PM_{2.5}$ estimation was made using photographic data from the Beijing region of China. Feng et al. [7] used a common camera to automatically photograph day and night lights in an urban area in Beijing between 2019 and 2020. The photographs they obtained can show the processes of cloud, fog, and precipitation by characterizing the scattering effect of $PM_{2.5}$ on visible sunlight or lamps. By using these features, estimation was done with machine learning methods. Successful results were obtained with DT and MLP among these methods. Lv et al. [8] applied three machine learning algorithms to correct deviations in their study to improve numerical prediction accuracy. Their results showed that the RF and SVM models offer much better prediction performance. Enebish et al. [9] compared the performance of six different machine learning algorithms to predict $PM_{2.5}$ concentrations using data between 2010 and 2018 in Ulaanbaatar, Mongolia. Results on the advantage and applicability of machine learning approach in predicting $PM_{2.5}$ levels in an environment with limited resources and extreme levels of air pollution are discussed. In another study, Multiple Additive Regression Trees, Deep Feedforward Neural Network methods, and LSTM based models were used by Karimian et al. [10] to predict $PM_{2.5}$ concentrations effectively at different time intervals. According to the results obtained, the LSTM-based model gave better results. Pak et al. [11] presented a study on the $PM_{2.5}$ prediction for Beijing, China. The hybrid model, developed using CNN and LSTM models, was used to predict the next day's average $PM_{2.5}$. Xiao et al. [12] proposed an ensemble machine learning-based approach. In this study, in which satellite data is used, first of all, missing satellite data is filled with multiple assignments. Then, for the modelling area, China, seven regions were obtained using a spatial clustering method to control for unobserved spatial heterogeneity. Machine learning models such as RF, Generalized Additive Model (GAM), and XGBoost are trained separately for each region. A generalized ensemble model is proposed to incorporate predictions from the models. Kleine Deters et al. [13] developed a machine-learning model that makes effective $PM_{2.5}$ predictions for Quito, Ecuador, a medium-sized city at high altitudes. In this study, weather data obtained from the city of Quito was used. This method aims to categorize different levels of $PM_{2.5}$, has yielded very effective results.

This study, it is aimed to predict the short-term $PM_{2.5}$ values using the measurement data obtained from the Eltham measurement station in London between January 1, 2019 and May 1, 2019. DT, ET, k-NN, LR, RF, SVM, XGBoost, MLP, CNN, RNN, GRU, and LSTM models compared for $PM_{2.5}$ prediction. According to the results obtained, RF gave better results than other machine learning methods. This study emerges as an important study both in terms of the results it has achieved and by comparing a total of twelve successful machine learning and deep learning methods in this field.

2. DEEP LEARNING BASED AIR QUALITY PREDICTION

In this section, the general developmental stages of machine learning models developed for the effective prediction of $PM_{2.5}$, a hazardous air pollutant, were explained. These stages were shown in Figure 1.

2.1. Dataset

In this study, a dataset consisting of hourly $PM_{2.5}$ measurement values taken from 7 different stations in London was used. The dataset consists of hourly measurement values for a total of 120 days between January 1, 2019, and May 1, 2019. The dataset consists of utc, location, parameter, unit, and value attributes. Utc represents time and date information. Location refers to the station name. Parameter refers to $PM_{2.5}$ measurements. The unit stands for µg/m3 measurement unit. The value represents the measured $PM_{2.5}$ value. In this study, the data obtained at the London Eltham measuring station were used. The dataset is available on Kaggle [14]. Figure 2 shows $PM_{2.5}$ concentrations over time for the London dataset.

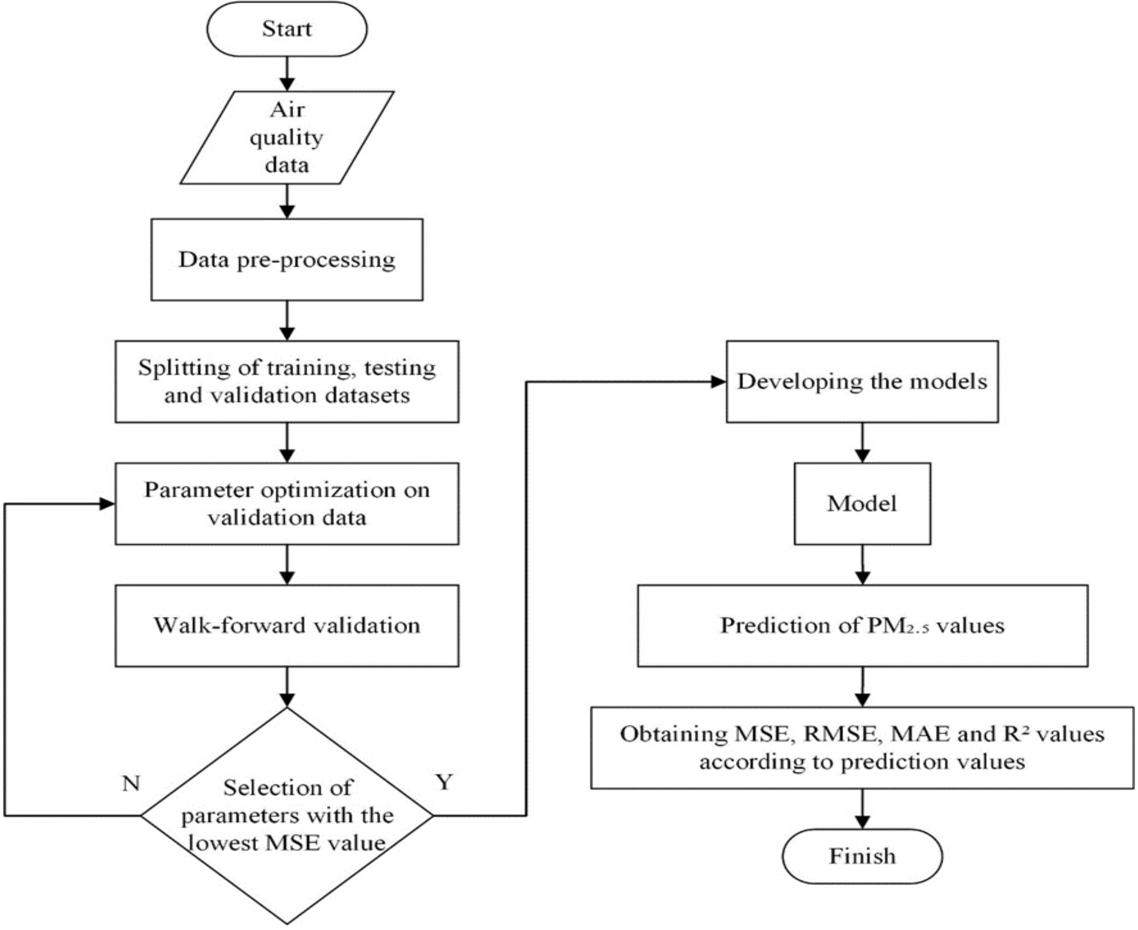
Start
Air quality data
Data pre-processing
Splitting of training, testing and validation datasets
Parameter optimization on validation data
Walk-forward validation
Selection of parameters with the lowest MSE value
N
Y
Developing the models
Model
Prediction of PM2.5 values
Obtaining MSE, RMSE, MAE and R² values according to prediction values
Finish

Figure 1. Flow chart of developed machine learning based prediction models

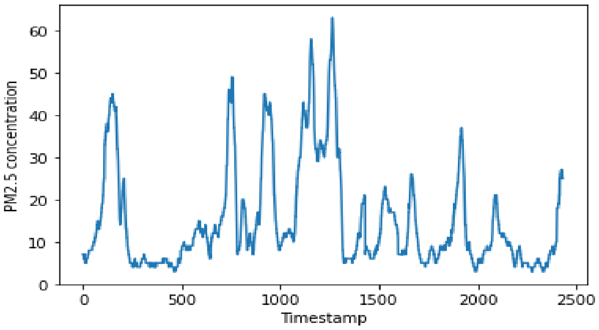

Figure 2. Variation of $PM_{2.5}$ concentrations over time

2.2. Data Pre-processing

In the data pre-processing stage, empty and incorrect fields in the dataset were checked. The dataset used in this study is a time series dataset. Time series data refers to data ordered according to a certain time index. In order to apply machine learning methods to time series data, it is necessary to transform the dataset into a supervised learning problem structure. The sliding window method is used to transform the time series data into a supervised learning problem structure. In the sliding window method, observation data as much as the specified window size is taken as input. The value to be predicted at the next time step is the output. For example, considering the sliding window in 3 dimensions, it is aimed to predict a value at time t_4 using the observation values at time steps t_1, t_2, and t_3. The sliding window shifts one unit to the right after each prediction is made. It is aimed to predict the value in the next time step by using the inputs in the window size determined in this way.

As seen in Figure 3, the dataset is structured as a supervised learning problem in which the pollution value at a certain time can be predicted using the $PM_{2.5}$ values from the past time steps, using the sliding window method.

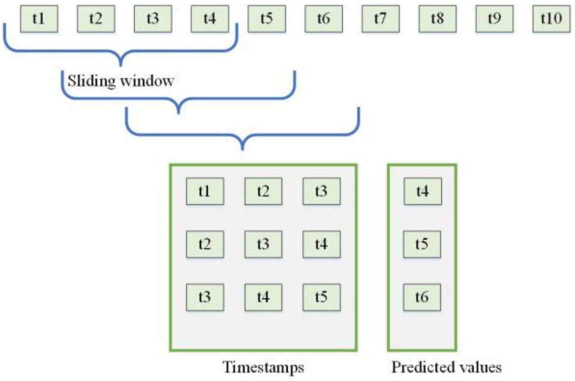

Figure 3. Sliding window method

After transforming the data into a supervised learning problem structure, the measurement values were normalized in the range of 0-1 using MinMaxScaler. The purpose of normalization is to change the values of the numeric columns in the dataset to a common scale without breaking the differences in the range of values. Normalization affects the performance of the developed model and the stability of the training. After the normalization step, the data was split into 80% training and 20% testing. 10% of the training data was split for validation. The dataset was split into training/test sets with 1945 rows of training data and 487 rows of test data. Validation data was used for the optimization of model parameters. In order for the compared machine learning algorithms to give the best results, parameter optimization was made using the GridSearchCV library. In the GridSearch method, a model is built separately with all combinations for the hyperparameters to be tested in the model and their values, and the most successful hyperparameter set is determined. With parameter optimization, the parameters with the lowest MSE values were determined and models were created.

In time series modelling, predictions over time become less and less accurate. Therefore, it is important to retrain the model with real observation data in order to obtain more accurate predictions. Walk forward validation method, as seen in Figure 4, refers to the inclusion of test data in the training process with actual observation values after prediction.

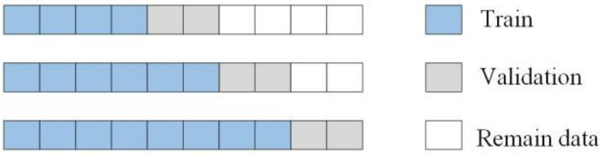

Figure 4. Walk forward validation method

2.2. Prediction Models

In this study, popular machine learning and deep learning methods are used for air quality prediction. These methods are briefly described in this section.

DT: It is a sequential model that efficiently and harmoniously combines a set of core tests in which a numerical feature is compared with a threshold value in each test [15]. DT is an important method in solving classification and regression problems. DT is one of the powerful methods widely used in application areas such as machine learning, image processing, and pattern recognition. For example, it is widely used in marketing, fraud detection, and scientific discovery tasks. The ID3, C4.5, and C5.0 algorithms, which are the classical algorithms of the decision tree, have advantages such as high classification speed, strong learning ability, and simple use [16].

ET: This method is a new tree-based ensemble method for supervised classification and regression problems [17]. It is an effective method used to overcome the disadvantages of traditional DTs and basically consists of strongly randomizing both the feature and the choice of breakpoint when splitting a tree node. Because of the randomization feature for numeric inputs, the ET method is useful for problems involving a large number of numeric features. In such cases, it often contributes to increased accuracy [18].

k-NN: In this method, determining the class of an element in the sample space is determined according to the class of k elements close to it. This method was proposed by T. M. Cover and P. E. Hart [19]. kNN is a classification algorithm based on measuring the distance between sample data. In kNN, the proximity between test samples and training samples is calculated by various distance-measuring methods [20]. k-NN is used for classification, clustering, and regression in various research areas such as financial modelling, image interpolation, and visual category recognition [21].

LR: This model proposed by Vapnik [22] is a statistical model and plays a dominant role in the field of statistical modelling. The LR model aims to develop a regression function to make accurate predictions. Since LR, which is the basis of current methods, has a linear structure, its parameters are easier to interpret. In addition, LR offers successful solutions when the sample space size is small [23].

RF: This method is an ensemble method that makes predictions based on its results from a collection of DTs and was introduced by Ho [24]. In this method, the predictions obtained by combining the DTs are collected. If the number of variables is more than the number of observations in a problem, RF shows high performance in these problems. It has easy adaptability [25].

SVM: It was proposed by Vapnik for solving classification and regression problems [26]. The decision function of the SVM is an optimal "hyperplane" for classifying observations of one class from another based on patterns of knowledge about those observations. This hyperplane can then be used to identify the most likely label for the invisible data [27]. The strength of an SVM comes from its ability to learn data classification models with balanced accuracy and repeatability. Also, SVM is one of the classic machine learning techniques that can help solve big data classification problems. It is especially successful in applications in big data environments [28].

XGBoost: The XGBoost algorithm based on the gradient boosting algorithm was proposed by Chen and Guestrin [29]. XGBoost is one of the best-performing algorithms in supervised learning tasks. It can be used for both regression and classification problems. Apart from basic computing, Xgboost is preferred by data scientists due to its high execution speed [30]. The key innovation of XGBoost is that it adds an edit component to the loss function. Thus, the complexity of the resulting community is taken into account along with predictability in each compartment. In addition, XGBoost allows its users to reduce model overfitting by adjusting multiple hyper-parameters such as forest complexity, learning rate, regularization terms, and column subdomains. XGBoost offers additional innovations such as processing missing data with nodes [31].

MLP: This model, which consists of input, output, and hidden layer, is a feed-forward neural network. The input layer transmits the incoming signal to other layers for processing, and the output layer gives the predictions or classification results. It uses a back propagation technique for learning [32]. A random number of hidden layers placed between the input and output layers is the true computation engine of MLP [33].

CNN: It was first proposed by Fukushima [34] in 1988. It is one of the most widely used and popular deep learning networks. The main advantage of CNN over its predecessors which makes it the most used feature is that it automatically detects important features without any human supervision [35]. CNN gives very good results in pattern recognition applications. It has been successfully used in different application areas by extracting automatic features in areas such as speech recognition and computer vision [36].

RNN: This model is a simple adaptation of a standard feedforward neural network to model sequential data. Text, video, and audio data are sequential data and this model has been used frequently in the processing of this data [37]. At each time step, the RNN receives an input, updates its latent state, and makes a prediction. In the traditional RNN architecture, the RNN can refresh the current state based on past state connections and input states. This is done in a circular structure. The high-dimensional latent state and non-linear nature of RNN are great advantages [38].

GRU: It is a version of RNNs. GRU is similar to LSTM in terms of its internal structure and RNN method with its organization of input and output structures [39]. GRUs are popular methods; the main reason for this is the computational cost and simplicity of the model. GRUs are simpler RNN approaches than standard LSTM in terms of topology, computational cost, and complexity. This technique combines forgetting and entry gates into a single update gate and can combine cell state and hidden state with some other modifications [40].

LSTM: It was suggested by Hochreiter and Schmidhuber [41]. LSTM has an RNN structure and also has a multilayer cell structure. Further LSTM includes state memory. LSTM neural networks gave successful results in pattern recognition and classification tasks, and categorizing audio and images. It is used in many fields, from medicine to statistics, because its sequential data processing features are strong [42].

2.2. Evaluation Metrics

MSE, RMSE, MAE, and R^2 metrics are metrics used to measure prediction accuracy in regression problems. The MSE is calculated using Equation 1 by averaging the squares of the differences between the actual observation values and the predicted values.

$$MSE = \frac{1}{n}\sum\nolimits_{i=1}^{n}(y-\hat{y})^2 \qquad (1)$$

y is the actual values, $\hat{y}$ predicted values and n is the number of samples.

The RMSE is calculated by taking the square root of the MSE and measuring the standard deviation of errors. The RMSE is calculated using Equation 2.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(|y-\hat{y}|)^2} \qquad (2)$$

MAE expresses the mean of the absolute values of the differences between the actual observation values and the predicted values. Calculates the mean of errors. MAE is calculated using Equation 3.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y-\hat{y}\,| \qquad (3)$$

R^2 is a measure of the fit of the applied models to the dataset. R^2 evaluates the distribution of data points around the regression line. Higher R^2 values for the same dataset indicate lower errors between actual and predicted values. R^2 is calculated using Equation 4.

$$R^2 = \frac{\sum(y-\hat{y})^2}{(y-\bar{y})^2} \qquad (4)$$

$\hat{y}$ is the predicted y values, and $\bar{y}$ is the average of the y values.

3. EXPERIMENTAL RESULTS

In this study, a comparative analysis of DT, ET, k-NN, LR, RF, SVM, XGBoost, MLP, CNN, RNN, GRU, and LSTM models for $PM_{2.5}$ predictions is presented. The results obtained according to MSE, RMSE, MAE, and R^2 metrics for each applied algorithm were analysed comparatively.

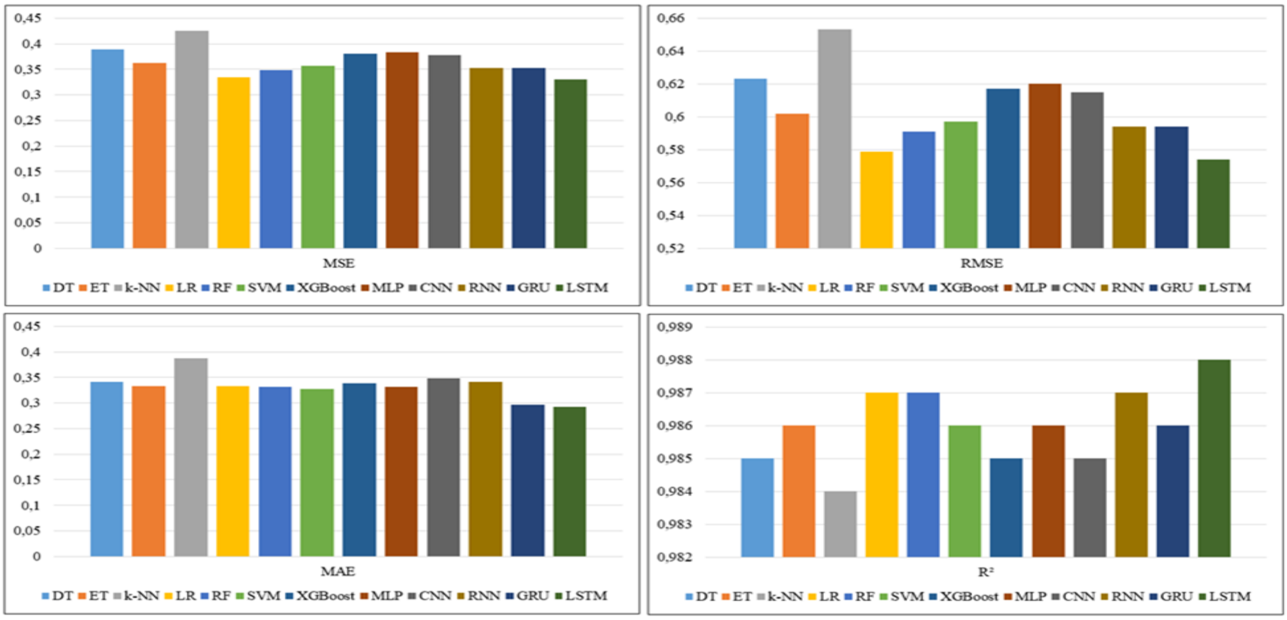

Figure 5. Comparative experimental results

In this study, the sliding window size was tested using values between 2 and 20 to determine the sliding window size. Table 1 shows an example of the test results according to the MAE metric of the models applied to express w different sliding window sizes.

Table 1. Test results for determining the sliding window size

Model	***w*=3**	***w*=4**	***w*=5**	***w*=10**	***w*=15**	***w*=20**
DT	0.341	0.360	0.362	0.521	0.522	0.525
ET	0.333	0.356	0.357	0.437	0.441	0.459
k-NN	0.388	0.414	0.443	0.546	0.600	0.701
LR	0.333	0.334	0.334	0.335	0.335	0.338
RF	0.331	0.348	0.354	0.422	0.429	0.432
SVM	0.328	0.326	0.335	0.343	0.351	0.352
XGBoost	0.339	0.357	0.363	0.445	0.449	0.452
MLP	0.342	0.325	0.331	0.351	0.360	0.367
CNN	0.349	0.327	0.332	0.414	0.424	0.435
RNN	0.331	0.325	0.328	0.346	0.360	0.365
GRU	0.297	0.323	0.327	0.343	0.347	0.354
LSTM	0.292	0.315	0.319	0.319	0.320	0.322

As seen in Table 1, as a result of the experimental studies, when the sliding window size is 3, the lowest error rate was obtained for all models applied. Comparative experimental results are shown in Table 2 and Figure 5.

Table 2. Comparative experimental results

Model	**MSE**	**RMSE**	**MAE**	**R^2**
DT	0.389	0.623	0.345	0.985
ET	0.363	0.602	0.333	0.986
k-NN	0.426	0.653	0.388	0.984
LR	0.335	0.579	0.333	0.987
RF	0.349	0.591	0.331	0.987
SVM	0.357	0.597	0.328	0.986
XGBoost	0.380	0.617	0.345	0.985
MLP	0.384	0.620	0.331	0.986
CNN	0.378	0.615	0.349	0.985
RNN	0.353	0.594	0.342	0.987
GRU	0.353	0.594	0.297	0.986
LSTM	**0.330**	**0.574**	**0.292**	**0.988**

In Table 1 and Figure 5, it is seen that all the models applied can be used successfully in $PM_{2.5}$ predictions. All compared algorithms have very low MSE, RMSE, and MAE values. Also, R^2 values are high for all models.

Experimental results showed that LSTM had better results than other models compared. After LSTM, GRU, SVM, MLP, RF, LR, ET, RNN, XGBoost, DT, CNN, and k-NN algorithms were successful, respectively. Figure 5 shows the prediction results of LSTM on the test data.

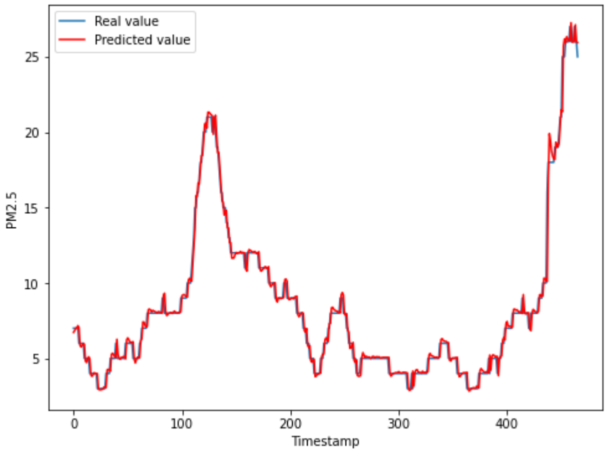

Figure 4. Prediction results of LSTM on test data

As can be seen in Figure 4, the LSTM successfully predicted the fluctuations in the dataset and outperformed the other models compared.

4. CONCLUSION

This study, it is aimed to estimate the short-term $PM_{2.5}$ values using the data obtained from the Eltham measuring station in London between January 1, 2019, and May 1, 2019. DT, ET, k-NN, LR, RF, SVM, XGBoost, MLP, CNN, RNN, GRU, and LSTM models were compared practically according to MSE, RMSE, MAE, and R^2 metrics. Experimental results have shown that LSTM gives more successful results than other models. After LSTM, GRU, SVM, MLP, RF, LR, ET, RNN, XGBoost, DT, CNN, and k-NN algorithms were successful, respectively.

The reason why RF gives better results than k-NN is the values of the features in the dataset. RF basically assumes local similarities and very similar samples are classified in the same way. k-NN can select only the most similar samples based on distance.

The fact that RF has better experimental results than LR can be interpreted as generally that LR performs better when the number of noise variables is less than or equal to the number of explanatory variables. SVM supports both linear and non-linear solutions using kernel trickery. SVM handles outliers better than LR when training data is scarce. In this study, SVM and LR had a close performance.

The fact that LR has better experimental results than k-NN can be interpreted as k-NN is a non-parametric model and LR is a parametric model. The fact that SVM has better performance than k-NN can be interpreted as SVM being more sensitive to outliers. If the training data is much larger than the number of features, k-NN may be more successful than SVM. However, SVM outperforms k-NN when there are lots of features and less training data.

The fact that RF outperforms DT and ET can be interpreted as DT and ET placing high emphasis on a certain set of features. RF randomly selects features during the training process. As such, it is not heavily dependent on any particular feature set. RF can generalize better data. This random selection of features makes RF much more accurate than DTs.

The better performance of RF than XGBoost can be explained by the concept of bias in tree structures. XGBoost relies on weak learners (high bias, low variance) i.e. shallow trees. But RF uses fully grown DTs (low bias, high variance). It performs the error reduction task by reducing the variance.

The fact that RNN is more successful than CNN can be interpreted with the architectures of these models. CNN is a feed-forward neural network. RNN is a feedback neural network. In CNN the size of the input is fixed whereas in RNN the size of the input is variable. The feedback structure in the RNN architecture enabled the past observations to be remembered and presented to the network as input again.

The fact that GRU is more successful than RNN can be interpreted with the structure of GRU that allows long-term dependencies to be remembered. RNN is not successful enough when long-term dependencies need to be learned because of the disappearing gradient problem. The fact that LSTM is more successful than other models compared is that LSTM's architecture includes special units in addition to other iterative neural network architectures. LSTM contains cells that can hold information in memory for long periods. In addition, there are doors used for remembering and forgetting information. This makes it easier to learn about long-term dependencies.

Air pollution forecasting has individual, national, and global implications. Accurate estimation of air pollution is important in terms of people who are sensitive to polluted air, determining public policies, and taking measures to reduce air pollution. The experimental results obtained in this study have shown that air pollution values can be successfully predicted by artificial intelligence methods. By using artificial intelligence-supported air pollution prediction models, it is possible to predict future pollution values and thus take measures to reduce pollution at the national and global levels. In future studies, more successful predictions can be made by using hybrid versions of traditional machine learning models and deep learning models. In addition, these studies can be expanded by using more data sets.

REFERENCES

[1] Xing YF, Xu YH, Shi MH, The impact of PM2. 5 on the human respiratory system. J. Thorac. Dis. 2016;8(1), E69. https://doi.org/ 10.3978/j.issn.2072-1439.2016.01.19Lian YX.

[2] Hayes RB, Lim C, Zhang Y, Cromar K, Shao Y, Reynolds HR, et al. PM2. 5 air pollution and cause-specific cardiovascular disease mortality. Int. J. Epidemiol. 2020;49(1), 25-35.

[3] He K, Yang F, Ma Y, Zhang Q, Yao X, Chan CK, et al. The characteristics of PM2. 5 in Beijing, China. Atmos. Environ. 2001; 35(29), 4959-4970. https://doi.org/10.1016/S1352-2310(01)00301-6

[4] Ma J, Yu Z, Qu Y, Xu J, Cao Y. Application of the XGBoost machine learning method in PM2. 5 prediction: A case study of Shanghai. Aerosol Air Qual. Res. 2020; 20(1), 128-138. https://doi.org/10.4209/aaqr.2019.08.0408

[5] Masood A, Ahmad K. A model for particulate matter (PM2. 5) prediction for Delhi based on machine learning approaches. Procedia Comput. Sci. 2020; 167, 2101-2110. https://doi.org/10.1016/j.procs.2020.03.258

[6] Danesh Yazdi M, Kuang Z, Dimakopoulou K, Barratt B, Suel E, Amini H, et al. Predicting fine particulate matter (PM2. 5) in the greater London area: an ensemble approach using machine learning methods. Remote Sens. 2020; 12(6), 914. https://doi.org/10.3390/rs12060914

[7] Feng L, Yang T, Wang Z. Performance evaluation of photographic measurement in the machine-learning prediction of ground PM2. 5 concentrations. Atmos. Environ. 2021;262, 118623. https://doi.org/10.1016/j.atmosenv.2021.118623

[8] Lv L, Wei P, Li J, Hu J. Application of machine learning algorithms to improve numerical simulation prediction of PM2. 5 and chemical components. Atmos. Pollut. Res. 2021; 12(11), 101211. https://doi.org/10.1016/j.apr.2021.101211

[9] Enebish T, Chau K, Jadamba B, Franklin M. Predicting ambient PM2. 5 concentrations in Ulaanbaatar, Mongolia with machine learning approaches. J. Exposure Sci. Environ. Epidemiol. 2021; 31(4), 699-708. https://doi.org/10.1038/s41370-020-0257-8

[10] Karimian H, Li Q, Wu C, Qi Y, Mo Y, Chen G, et al. Evaluation of different machine learning approaches to forecasting PM2. 5 mass concentrations. Aerosol Air Qual. Res. 2019; 19(6), 1400-1410. https://doi.org/10.4209/aaqr.2018.12.0450

[11] Pak U, Ma J, Ryu U, Ryom K, Juhyok U, Pak K, et al. Deep learning-based PM2. 5 prediction considering the spatiotemporal correlations: A case study of Beijing, China. Sci. Total Environ. 2020;699, 133561. https://doi.org/10.1016/j.scitotenv.2019.07.367

[12] Xiao Q, Chang HH, Geng G, Liu Y. An ensemble machine-learning model to predict historical PM2. 5 concentrations in China from satellite data. Environ. Sci. Technol. 2018;52(22), 13260-13269. https://doi.org/10.1021/acs.est.8b0291

[13] Kleine Deters J, Zalakeviciute R, Gonzalez M, Rybarczyk Y. Modeling PM2. 5 urban pollution using machine learning and selected meteorological parameters. J. Electr. Comput. Eng. 2017: 5106045. https://doi.org/10.1155/2017/5106045

[14] Pollution PM2.5 data London 2019 Jan to Apr. Access time: 10 September 2022. https://www.kaggle.com/siddharthnobell/pollution-pm25-data-london-2019-jan-to-apr

[15] Charbuty B, Abdulazeez A. Classification based on decision tree algorithm for machine learning. J. Appl. Sci. Technol. Trends. 2021; 2(01), 20-28. https://doi.org/10.38094/jastt20165

[16] Brijain M, Patel R, Kushik MR, Rana K. A survey on decision tree algorithm for classification. Int. J. Eng. Dev. Res. 2014;2(1).

[17] Geurts P, Ernst D, Wehenkel L. Extremely randomized trees. Mach. Learn. 2006;63(1), 3-42. https://doi.org/10.1007/s10994-006-6226-1

[18] Sharaff A, Gupta H. Extra-tree classifier with metaheuristics approach for email classification. In Advances in computer communication and computational sciences. 2019. https://doi.org/189-197. 10.1007/978-981-13-6861-5_17

[19] Cover T, Hart P. Nearest neighbor pattern classification. IEEE Trans. Inf. Theory, 1967;13(1), 21-27. https://doi.org/10.1109/TIT.1967.1053964

[20] Ali N, Neagu D, Trundle P. Evaluation of k-nearest neighbour classifier performance for heterogeneous data sets. SN Appl. Sci. 2019; 1(12), 1-15. https://doi.org/10.1007/s42452-019-1356-9

[21] Ertuğrul ÖF, Tağluk ME. A novel version of k nearest neighbor: Dependent nearest neighbor. Appl. Soft Comput, 2017;55,480-490. https://doi.org/10.1016/j.asoc.2017.02.020

[22] Vapnik VN. Statistical learning theory. Wiley;1998.

[23] Su X, Yan X, Tsai CL. Linear regression. Wiley Interdiscip. Rev. Comput Stat. 2012;4(3), 275-294. https://doi.org/10.1002/wics.1198

[24] Ho TK. Random decision forests. Proceedings of the International Conference on Document Analysis and Recognition, ICDAR, Montreal, Canada, 1995. pp. 278–282.

[25] Biau G, Scornet E. A random forest guided tour. Test. 2016;25(2), 197-227. https://doi.org/10.1007/s11749-016-0481-7

[26] Drucker H, Burges CJ, Kaufman L, Smola A, Vapnik V. Support vector regression machines. Adv. Neural Inf. Process. Syst. 1997; 9, 155-161.

[27] Pisner DA, Schnyer DM. Support vector machine. Mach. Learn. 2020. https://doi.org/10.1016/b978-0-12-815739-8.00006-7

[28] Suthaharan S. Support vector machine. Machine learning models and algorithms for big data classification, Springer, Boston, MA, 2016. pp. 207-235.

[29] Chen T, Guestrin C. XGBoost. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 2016. pp. 785–794.

[30] Osman AIA, Ahmed AN, Chow MF, Huang YF, El-Shafie A. Extreme gradient boosting (Xgboost) model to predict the groundwater levels in Selangor Malaysia. Ain Shams Eng. J. 2021; 12(2), 1545-1556. https://doi.org/10.1016/j.asej.2020.11.011

[31] Sagi O, Rokach L. Approximating XGBoost with an interpretable decision tree. Inf. Sci. 2021;572, 522-542. https://doi.org/10.1016/j.ins.2021.05.055

[32] Desai M, Shah M. An anatomization on breast cancer detection and diagnosis employing multi-layer perceptron neural network (MLP) and Convolutional neural network (CNN). Clin. eHealth. 2021; 4, 1-11. https://doi.org/10.1016/j.ceh.2020.11.002

[33] Abirami S, Chitra P. Energy-efficient edge based real-time healthcare support system. In Advances in computers. Elsevier; 2020, Vol. 117, No. 1, pp. 339-368. https://doi.org/10.1016/bs.adcom.2019.09.007

[34] Fukushima K. Neocognitron: A hierarchical neural network capable of visual pattern recognition. Neural Netw. 1988; 1, 119–130.

[35] Alzubaidi L, Zhang J, Humaidi AJ, Al-Dujaili A, Duan Y, Al-Shamma O, et al. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. J. Big Data. 2021;8(1), 1-74. https://doi.org/10.1186/s40537-021-00444-8

[36] Botalb A, Moinuddin M, Al-Saggaf UM, Ali SS. Contrasting convolutional neural network (CNN) with multi-layer perceptron (MLP) for big data analysis. In 2018 International conference on intelligent and advanced system (ICIAS), Kuala Lumpur, Malaysia: IEEE; 2018. pp. 1-5. https://doi.org/10.1109/ICIAS.2018.8540626

[37] Sutskever I, Martens J, Hinton GE. Generating text with recurrent neural networks. In ICML. 2011.

[38] Yu Y, Si X, Hu C, Zhang J. A review of recurrent neural networks: LSTM cells and network architectures. Neural Comput. 2019;31(7), 1235-1270. https://doi.org/10.1162/neco_a_01199

[39] Yang S, Yu X, Zhou Y. Lstm and gru neural network performance comparison study: Taking yelp review dataset as an example. In 2020 International workshop on electronic communication and artificial intelligence (IWECAI). Shanghai, China: IEEE; 2020. pp. 98-101. https://doi.org/ 10.3978/10.1109/IWECAI50956.2020.00027

[40] Alom MZ, Taha TM, Yakopcic C, Westberg S, Sidike P, Nasrin MS, et al. A state-of-the-art survey on deep learning theory and architectures. Electron. 2019;8(3), 292. https://doi.org/10.3390/electronics8030292

[41] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997;9(8), 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

[42] Smagulova K, James AP. A survey on LSTM memristive neural network architectures and applications. Eur. Phys. J. Spec. Top. 2019;228(10), 2313-2324.