Heliyon 6 (2020) e05618

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

A machine learning-based model to estimate PM2.5 concentration levels in Delhi's atmosphere

Saurabh Kumar, Shweta Mishra, Sunil Kumar Singh *

Department of Computer Science & Information Technology, Mahatma Gandhi Central University, Bihar, India

ARTICLE INFO

Keywords:
Computer science
PM2.5 prediction
Regression
Atmospheric pollution
Time series analysis
Machine learning

ABSTRACT

During the last many years, the air quality of the capital city of India, Delhi had been hazardous. A large number of people have been diagnosed with Asthma and other breathing-related problems. The basic reason behind this has been the high concentration of life-threatening PM2.5 particles dissolved in its atmosphere. A good model, to forecast the concentration level of these dissolved particles, may help to prepare the residents with better prevention and safety strategies in order to save them from many health-related diseases. This work aims to forecast the PM2.5 concentration levels in various regions of Delhi on an hourly basis, by applying time series analysis and regression, based on various atmospheric and surface factors such as wind speed, atmospheric temperature, pressure, etc. The data for the analysis is obtained from various weather monitoring sites, set-up in the city, by the Indian Meteorological Department (IMD). A regression model is proposed, which uses Extra-Trees regression and AdaBoost, for further boosting. Experimentation for comparative study with the recent works is done and results indicate the efficacy of the proposed model.

1. Introduction

Air is a mixture of several organic gases needed for life support. However, several factors [1] such as deforestation, modernization, industrialization, vehicle emissions, and population super explosion contribute in polluting the air by dissolving several harmful gases such as Nitrogen Dioxide (NO_2), Sulphur Dioxide (SO_2), lead (Pb), Carbon Monoxide (CO), Ozone (O_3). Several factors contribute to pollution including stubble burning [2] along with hazardous particulate matters such as PM2.5 and PM10 [1]. These particulate matters are mainly composed of tiny solid and liquid particles suspended in the air [3], with diverse chemical composition involving some organic compounds as well as compounds like SO_4^{2-}, NO_3^- etc [4].

The main and the most hazardous constituent among these pollutant particles are PM2.5 particles, which as evident by its name, refers to fine atmospheric particulate matter (PM) with a diameter of fewer than 2.5 μm'', about 3% the diameter of a human hair. The concentration of PM2.5 is measured in $\mu g/m^3$. These particles are extremely dangerous for health and can easily penetrate deep into the lungs, irritate, and corrode the alveolar wall, and consequently impair the lung functions [5]. The adverse effect of PM2.5 is not only limited to asthma [6], respiratory inflammation, jeopardization of lung functions, various cardiovascular diseases [7, 8, 9] but even may cause cancer [10]. These fine particles, if enter into the lungs, might complement the severity of COVID-19 infection as the novel coronavirus also attacks the respiratory system [11]. If the concentration of these pollutant particles is very high in the atmosphere, it severely affects our health and may cause life-threatening problems in a short span of time [12]. Studies have established that particulate matters affect the human health even at the genetic level [13].

The work proposed in this paper is considering the air pollution of Delhi, the most affected place during winter. Data, for the experimentation, is collected from the Central Pollution Control Board [14].

1.1. Air quality monitoring in Delhi

In Delhi, air pollution monitoring is done through continuous and manual ambient air quality monitoring (CAAQM) stations. As per the National Air Quality Monitoring Program (NAMP) [15] of the Central Pollution Control Board (CPCB), manual air pollution monitoring is carried out at Sarojini Nagar, Chandni Chowk, Mayapuri Industrial Area, Pitampura, Shahadra, Shahzada Bagh, Nizamuddin, Janakpuri, Siri Fort, and ITO across the Delhi. Apart from the manual air monitoring stations, continuous air quality monitoring is also carried out at 11 locations viz. Anand Vihar, Civil Lines, DCE, Dilshad Garden, Dwarka, IGI Airport, ITO,

* Corresponding author.
E-mail addresses: sksingh@mgcub.ac.in, sunilsingh.jnu@gmail.com (S.K. Singh).

https://doi.org/10.1016/j.heliyon.2020.e05618
Received 12 March 2020; Received in revised form 5 June 2020; Accepted 24 November 2020

Mandir Marg, Punjabi Bagh, R.K. Puram, & Shadipur. The map with all the monitoring stations in Delhi is given in Figure 1 in which the dark circled station (R. K. Puram) is used for the study in the model.

1.2. National ambient air quality standards (NAAQS)

The Environmental Protection Agency is amenable for time to time review and suggesting updates on all national ambient air quality standards (NAAQS) [16]. To reduce the severe effects of air pollution and to efficiently manage ambient air quality, it is a must for a nation to define national ambient air quality standards (NAAQS). In India, the Central Pollution Control Board firstly adopted ambient air quality standards on November 11, 1982, as per Section 16 (2) of the Air (Prevention and Control of Pollution) Act, 1981, which were further revised in 1994 and then in 2009. According to the latest guidelines, the acceptable concentrations of PM2.5 in India are 40–60 $\mu g/m^3$.

The proposed work is aimed to predict the PM2.5 pollutant particle concentration in Delhi Air efficiently and well in advance so that preventative steps can be taken to save the human lives from its hazardous impact.

The outline of the paper is as follows. Section 2 discusses the related work in the area of PM2.5 particulate matter measuring techniques and a few machine learning techniques. Acquisition, pre-processing, and analysis of the data is mentioned in section 3. Section 4 discusses the problem definition, while section 5 covers the definition and algorithmic part of the proposed model. The performance of the proposed model is analyzed by carrying out some experiments in section 6. A comparative study has been done in section 6 to show the effectiveness of the model. Section 7 concludes the work.

2. Related works

Since the last few years, most of the metropolitan cities across the globe do experience pollution levels breaching all international standards [17, 18] which causes many life-threatening issues. Although there are many responsible factors behind health hazards issues, PM2.5 is one of the crucial particulate matter responsible for it. The life-threatening impact of PM2.5 particulate matter brings researchers' attention towards this to propose some accurate model for forecasting PM2.5 levels in the polluted air.

Few models have explored this area to measure the pollutant particle levels in the air. Time series analysis of historical atmospheric data and further performing regression over it had been at the core of these models.

The primary models, for measuring the pollution level, have been based on statistical methods including Kalman Filtering [19] and single variable linear regression [20]. However, these have failed considerably in producing a good accuracy level. This initiated a trend to use Machine Learning and Neural Network-based approaches [21] for predicting PM2.5 as these can easily consider multiple attributes simultaneously. The models like Non-Linear Regression [22] and Neural Network Regression improved the accuracy significantly. Still, in these models, giving importance to the dependency on previous PM2.5 values is completely missed. So, when the Time Series component was combined with existing machine learning (ML) based models, the accuracy level of the measurement is quite improved.

Methods like Multi-Layered Perceptron Regression [23] and regression trees [24] based methods such as Decision Tree Regression [25], Random Forest Regression [26], Lasso, etc. came into the forefront for this analysis. Later on, for further improvement in accuracy level, boosting techniques were also combined along-with the existing models. A good example being XGBoost [27].

A study on air pollution prediction, through machine learning approaches, was done by Guan & Sinnott [28]. In this, they have applied LSTM (Long Short-Term Memory) Networks on the air pollution data based on Melbourne, Australia. It is observed that the LSTM network is able to detect PM2.5 concentration in the air quite significantly.

A few machine learning-based models for PM2.5 prediction is given by Joharestani et al. [29]. In this, they have applied XGBoost, Random Forest, and deep learning on multisource remote sensing data to predict the PM2.5 pollutant particles in Tehran's urban area, Iran. It is observed that XGBoost is the best performing model in comparison to the other two in terms of R2-Score, MAE, and RMSE values [28].

A few boosting techniques e.g. AdaBoost is used frequently to improve the quality of results produced by diverse machine learning models. There are a lot of use-cases available for time series forecasting assisted by the boosting techniques. An ensemble approach-based model [30] utilized boosting in agribusiness time series forecasting to achieve quality results. Xiao et al. [31] combined AdaBoost with LSTM for sea surface temperature forecasting. A gradient-boosted decision tree algorithm, based on Kalman Filter, was introduced by Li et al [32]. Boosted LSTMs were used for Internet Traffic Forecasting by Bian et al. [33]. Gradient boosting was also utilized to improve the performance of the Delay-Based Reservoir Computing System by Tao et al. [34]. AdaBoost

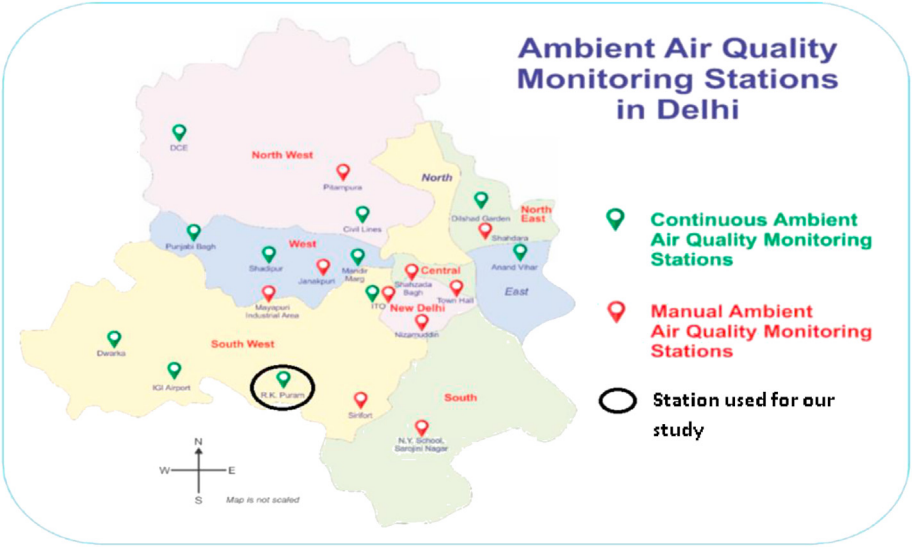

Figure 1. Air quality monitoring stations in Delhi.

was combined with SVM for time-series signal classification in epileptic seizure diagnosis by Hadeethi et al. [35].

The Extra Trees Classifier as well as Regressor have also found a wide range of applications in varied domains. Li et al. [36] stacked Extra Trees with LSTM for Dam Displacement Time Series Prediction. John et al. utilized Extra Trees Regression for Real-Time Lane Estimation [37]. Extra Trees produced commendable results in daily streamflow forecasting also, as suggested by Tyrallis et al. [38].

The proposed work is an attempt to accurately predict the PM2.5 levels and to increase the prediction accuracy, especially in Delhi's atmosphere. A model, for this, is proposed based on Extra-Trees-Regressor [39] boosted with AdaBoost [40].

Extra-Trees is a tree-based ensemble technique, which strongly randomizes both the cut point choice and the attributes involved in it while splitting a tree node. It is used for supervised classification but can also be extended for regression problems [39]. AdaBoost, stands for Adaptive Boosting, is a boosting algorithm used in conjunction with the learning algorithms to complement on its performances [40, 41].

3. Data acquisition and analysis

Machine learning uses historical data as an input to predict the future output. The data acquisition, pre-processing, and its analysis are discussed as follows.

3.1. Data acquisition

The data, used in the model, is of Delhi, India, and is collected from the monitoring station set up in the R K Puram area, by Delhi Pollution Control Committee (DPCC). DPCC is an autonomous regulatory body under the Central Pollution Control Board CB) of India. The dataset for the model is downloaded from the official CPCB website [14]. The downloaded data was in .csv (comma separated values) format, consisting of multiple features, from which irrelevant features were removed. The data collected is on an hourly basis from the 1st of January, 2018 to the 30th of November, 2019.

The feature sets utilized, along with their respective units of measurement, are mentioned in Table 1.

3.2. Data pre-processing

Many machine learning techniques often lead to better results when applied to pre-processed data. Therefore, in the proposed model, we have applied Z-Square and Boxplot methods to observe the outlier data and also considering the relevancy of the data, decided that the values of PM2.5 concentration above 600 and below 10 will be considered to be the outlier and are removed [42]. These extreme values generally occurred due to some faulty detection or reading, which happens due to human error while taking records manually.

In the model, the samples with missing values, are also excluded from the data. Although, these kinds of attributes were negligible in number.

Table 1. Feature, Details, and it's Unit.

Features	Details	Unit
SR	Solar Radiance	W/m^2
BP	Buoyancy Pressure by air	mmHg
AT	Atmosphere Temperature	^{0}C
WS	Wind Speed	m/s
VWS	Vertical Wind Speed	Degrees
WD	Wind Direction	Degrees
PM2.5	Particulate Matter 2.5,m	µg/m^3

3.3. Data analysis

After cleaning and pre-processing the data, it is subjected to further analysis including time-series analysis and analyzing the overall impact of every feature on the PM2.5 value [43].

Figure 2 shows the dependency of PM2.5 values on the timings of a day. The format is xx-yy-zz with xx denoting month, yy the date, and zz denotes the hour of the particular day. It is clearly evident that approaching the mid-night, values are fairly higher while in the evening, it is relatively low.

Figure 3 shows the effect of the month on PM2.5 value. Observations from Figure 3 is the associated seasonality factor on PM2.5. In the winter months (November–January), the values of PM2.5 are relatively higher as can be easily visualized from Figure 3. For the afore-mentioned months, it can also be observed that PM2.5 values are quite high sometimes more than 500. While in summer seasons, the PM2.5 values are largely concentrated in the low-value zones which are in the range of 0–100 as shown with the thick parts.

Further, the impact of wind is also analyzed on PM2.5. When the wind speed was high, PM2.5 value was found to lower and vice versa for still wind. Wind speed vs PM2.5 value scatter plot is shown in Figure 4, which also affirms the same as we have higher concentrations for wind speed close to 0. It significantly decreases with an increase in the wind speed.

It can also be observed that when the wind speed is 6–10 m/s, PM2.5 value is almost negligible.

4. The problem definition

In an attempt to tackle the menace of air pollution, efficient measurement of the level of fine particles present in the air is very essential. Many factors such as wind speed, direction, solar radiance, etc. contribute to determining its level. The specific particles, we are concerned about, is PM 2.5. Although many techniques have been applied to measure the PM2.5 concentration level, it is still a challenging task to measure it accurately because of its time-dependent behavior and varied dependency on many other factors such as vehicle CO2 emissions, stubble burning in the outskirts of Delhi, etc.

The task on a broader level is of time-based regression where we have to predict the continuous values of PM2.5, which are dependent on various meteorological features as well as the previous values of PM2.5 recorded in a time-series format.

Precisely, the problem dealt in the proposed model is the accurate determination of PM2.5 concentrations in the atmosphere of Delhi-NCR well in advance for precautionary measures.

5. The proposed model

This section presents the proposed model along with the required fundamental preliminaries. A detailed discussion of its structure has also been given.

5.1. Preliminaries

For the correct prediction of PM2.5 concentration levels, an ensemble-based regression method has been used. Such as an Ensemble-based method is Extra-Trees Regressor. It is a short form of Extremely Randomized Trees which tries to build a purely randomized tree that will be structurally independent on the output of the learning sample.

Thereafter, for boosting the performance, the AdaBoost (Adaptive Boosting) machine learning meta-algorithm is applied. It is used to enhance the performance of the weak learning regression trees by giving some more weights to misclassified samples.

For the Extra Trees, let's assume that we have a learning sample (LS) of size N as given in Eq. (1).

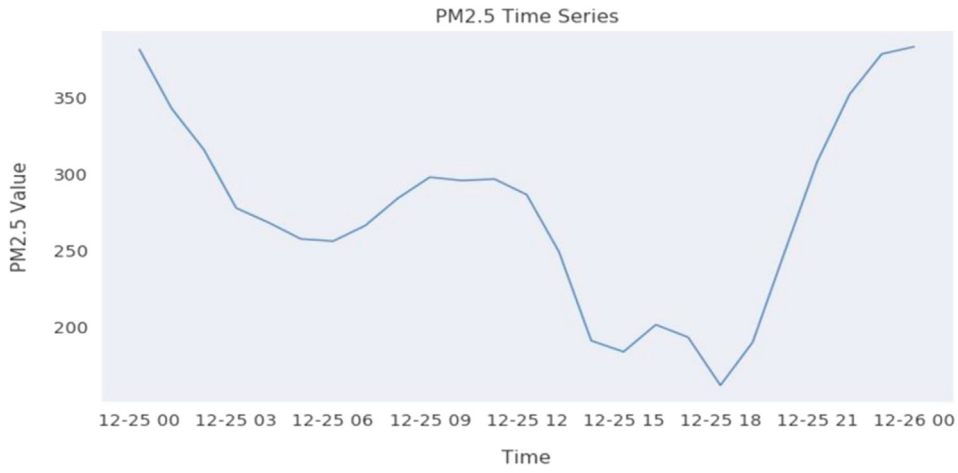

Figure 2. Time vs PM2.5 Value Plot.

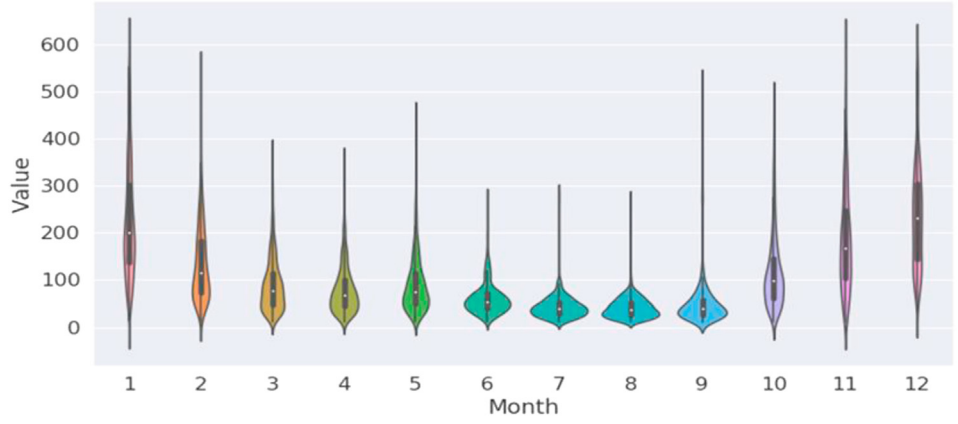

Figure 3. Month vs PM2.5 Value Violin Plot.

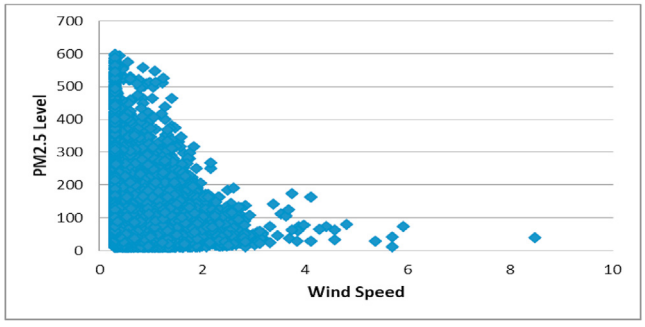

Figure 4. Wind Speed vs PM2.5 Value Plot.

$$LS_N = \{X^i, y^i : i = 1, 2.....N\} \tag{1}$$

Where X^i is the attribute vector for an n-dimensional dataset represented as given in Eq. (2).

$$X^i = (X_1^i .....X_M^1) \tag{2}$$

And y^i is the corresponding output value to each X_i.
$\forall(i_1, ..., i_m) \in \{0, ..., N\}$ by $I_{(i_1,....i_n)}(x)$

Using all these, the prediction by an Extra Trees Regressor is given by Eq. (3).

$$\hat{y}(x) = \sum_{i_1=0}^{N} ..... \sum_{i_2}^{N} I_{(i_1,.....i_n)}(x) \sum_{X \subset \{x_1,.....x_n\}} \lambda^X_{(i_1,.....i_n)} \prod_{x_j \varepsilon X} x_j \tag{3}$$

Where λ (lambda) denotes the real-valued trainable parameter.

For AdaBoost regression, we define a cutoff value (Ɛ). If an output value deviates from its actual value by Ɛ, it will be considered as a misclassified sample. While training our regressors which are considered weak, we increase the weight of the misclassified sample in each iteration to improve the performance.

The equation to represent this proceeding is given in Eq. (4),

$$F_T(x) = \sum_{t=1}^{T} f_t(x) \tag{4}$$

where each $f(t)$ is a weak learner that takes as input an object x and returns a value indicating the predicted value of the object. If it fails to the cutoff, it is considered as an input for the next weak learner.

5.2. The algorithm

Combining both ensemble-based learning and the Extra-Trees Regressor algorithm, we propose the algorithm for the model, given as Algorithm 1.

Algorithm 1. ET with Ada Boost

Input: D (A Data set with M features and N samples), n (Number of Decision stumps)

▷ *where decision stump are decision trees of unit depth*

Output: Prediction

1. ***For each*** $X_i \in D$ *where* $i \in (1, n)$
2. $W_i = {}^{1}/_{N}$ *where* ▷ W_i *indicates the weight assigned to each sample*
3. ***End***
4. ***For*** $j \leftarrow 1$ *to* q ▷ q *is number of decision stumps*
5. *Select* $\lceil \sqrt{M} \rceil$ *features from M as* F_j
6. *Construct traning set* $T_j \subset$ D *with* F_j *features where every* $X_i (\in T_j)$ *will be selected*with probability $\#P_i = W_i \Big/ \sum_{i=1}^{N} W_i$
7. Feed T_j to decision stumps regressor R_j

▷ *An API call to Decision Tree Regressor of sklearn library*

8. Compute loss L_i for all $X_i \in T_j$ by Mean Square Error *Calculation*
9. Compute $\bar{L} = \sum_{i=1}^{N} L_i P_i$ ▷ $\bar{L}$ *is the calculated mean loss*
10. Compute $\beta = \bar{L} / (1 - \bar{L})$ ▷ β indicates regressor confidence
11. Update $W_i = W_i * \beta^{(1-L_i)}$ ▷ $i \in (1, n)$
12. **End**
13. ▷ for a test sample S, Y_j being the prediction by the J^{th} stump
14. $Y_{pred} = \sum_{j=1}^{q} Y_j \Big/ q$

In the algorithm, decision stump is defined as a machine learning model consisting of a one-level decision tree. For example, the root of the tree is immediately connected with the terminal nodes.

Step 1 to Step 3 of the algorithm is the initialization of weights for all the samples in the dataset D.

Step 4 to Step 12 is for the training of the proposed model where firstly for a predefined number of decision stumps, we select some features from the training set randomly and also select some samples based on a probability value dependent on weights. Then iteratively train the decision stumps and compute loss. This loss is used to compute the confidence to update the weights. This process continues until we train the last decision stump. The total number of decision stumps is also called the number of estimators in machine learning terminology.

Step 13 and Step 14 are used for predicting the values based on the trained decision stumps by taking the mean value of all predictions.

6. The performance analysis

The model is simulated by writing the program in Python 3.6 programming environment. To perform a few comparative studies Keras library, which is an open-source neural network library in Python, is used.

This section shows the experimental evaluation of the model. The first section of the experiment shows the importance of the predicted values while the second set of experiments is based on the comparative study. The last section includes the comparative analysis with some recent work which shows the importance of the proposed model.

6.1. Actual vs predicted values

The data set is divided into two subsets, a training set and a holdout set (a set not used for fitting the model) of 80:20 of entire data respectively i.e. 80% of the data is used for learning (training) while remaining 20% of the data is used to test the results.

The test data results, shown in Figure 5, exhibits that the predicted values of PM2.5 are very close to the actual measured values. The plot of Actual vs Predicted forms an almost straight line with a slope of 45° from the origin. This validates that the actual and predicted values are almost equal which indicates the effectiveness of the model.

6.2. The comparative study

The proposed model is compared with several existing models, listed as follows.

i. **Linear Regression:** A Multiple Linear Regression-based approach was utilized in a raw manner to predict the values.
ii. **MLP Regression:** This is basically an Artificial Neural Network based approach tuned to use for regression problems. A 4-layer model is used for the comparison with this work.
iii. **Lasso:** It is a type of linear regression that uses shrinkage i.e. data values are shrunk towards a central point (mean for our case). This model is well suited for multi-collinear datasets such as our use-case.
iv. **Elastic Net:** A variation of linear regression that aims to minimize the penalties of ridge regression and lasso regression.
v. **Random Forest:** A collection of multiple decision trees [16].
vi. **Extra Trees:** An ensemble-based technique which is a variation of random forest. It is actually a more randomized version of random forest.
vii. **Decision Tree:** An adaptation of normal decision trees for regression problems where prediction is made using entropy and information gain of data attributes.
viii. **DT+AdaBoost:** Decision tree boosted using Adaptive Boosting.
ix. **RF+AdaBoost:** Random forest boosted using Adaptive Boosting.
x. **XGBoost:** XGBoost or Xtreme Gradient Boosting is a boosting technique used to boost the performance of decision trees.

A comparative study has been done to show the efficacy of the proposed model (ET+AdaBoost) with respect to the other ten models selected for the performance study.

To show the comparative performance, these well-recognized models have been implemented with the metrics which are used to evaluate the performance of the proposed model. The metrics used for the performance evaluation are as follows.

Mean Absolute Error (MAE) [44], which is basically the horizontal separation between two continuous values, which is the arrays of actual and predicted values of PM2.5 concentrations, is computed with the help of Eq. (5).

$$MAE = \frac{1}{n} \sum_{i=1}^{n} \left| \hat{y}_i - y_i \right| \tag{5}$$

Where n is the number of samples, y_i indicates the actual value and $\hat{y}_i$ represents the predicted values.

The second metric, used for the comparison, is Root Mean Squared Error (RMSE). It is computed by taking the standard deviation of the predicted values from actual values on a dataset and is represented in Eq. (6).

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}} \tag{6}$$

The last used performance metric is R2-Score which is sometimes also known as R-squared. This metric is also called a coefficient of regression. R2-Score [44] is used to measure the closeness of the predicted values to

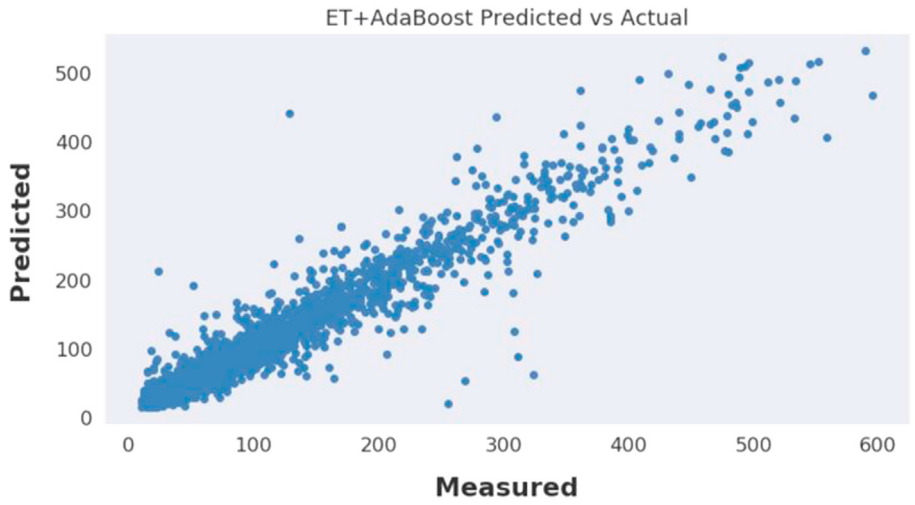

Figure 5. Actual Measured vs Predicted PM2.5 Value Plot.

actual values on a regression line. It can be calculated with the help of the relation given in Eq. (7).

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \mu)^2} \tag{7}$$

Where μ indicates the mean of actual values.

The performance of the proposed model (ET + Adaboost) for MAE (Mean Absolute Error) is shown in the form of bar plots to indicate its importance. The lower is the value of *MAE*; the closer is the forecasted values to the actual ones.

The outcome of the proposed model is pointed with red color as shown in Figure 6.

Observations from Figure 6 show that the mean absolute error of the proposed model (ET+ AdaBoost) is 14.79, which is very close to the DT+AdaBoost model and lower than the remaining other models.

Figure 7 indicates the comparative study of the RMSE value of the proposed model with other existing models. From Figure 7, one can observe that the *RMSE* of the proposed model is the minimum amongst all other models, which signifies the fact that actual values, when compared with the forecasted ones by the proposed model, are less erroneous than the one predicted by the other models. It shows the significance of the proposed model with respect to other models.

The next set of experiments are carried out to compare the R2-Score of the proposed model with other models mentioned above.

Observation from Figure 8, shows that the proposed model attains the highest *R2-Score*, which indicates that the prediction accuracy of the proposed model is quite high in comparison to other existing models on the used data set.

Overall observations from Figures 6 and 7 and, 8 show that the proposed model is comparatively much effective than other existing models for predicting the PM2.5 concentration level in the polluted air.

6.3. Comparison with recent work

For the comparative performance of the proposed model with some recently published work two recent proposed models such as LSTM by Guan & Sinnott [17] and XGBoost by Joharestani et al. [18] are considered. These two models have also been used for the prediction of PM2.5 pollutant particles.

The same three metrics, MAE, RMSE, and R2-Score, have been used for the comparative study.

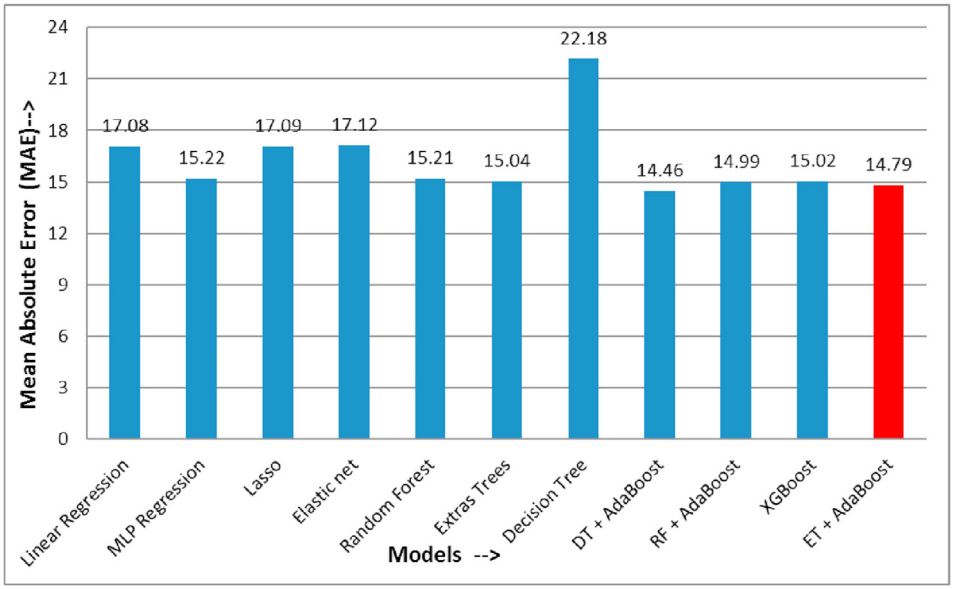

Figure 6. Mean absolute error.

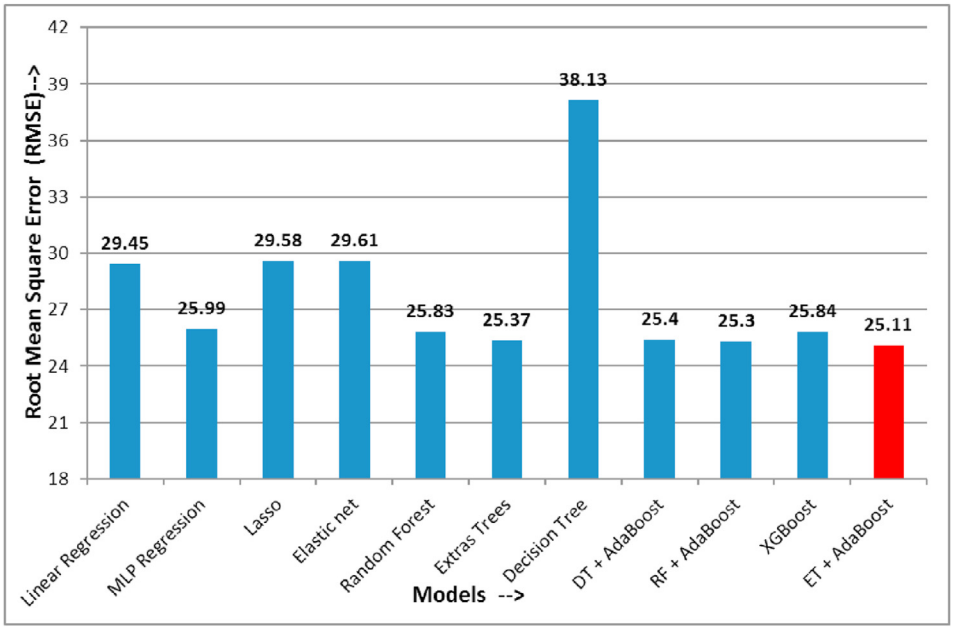

Figure 7. Root mean square error.

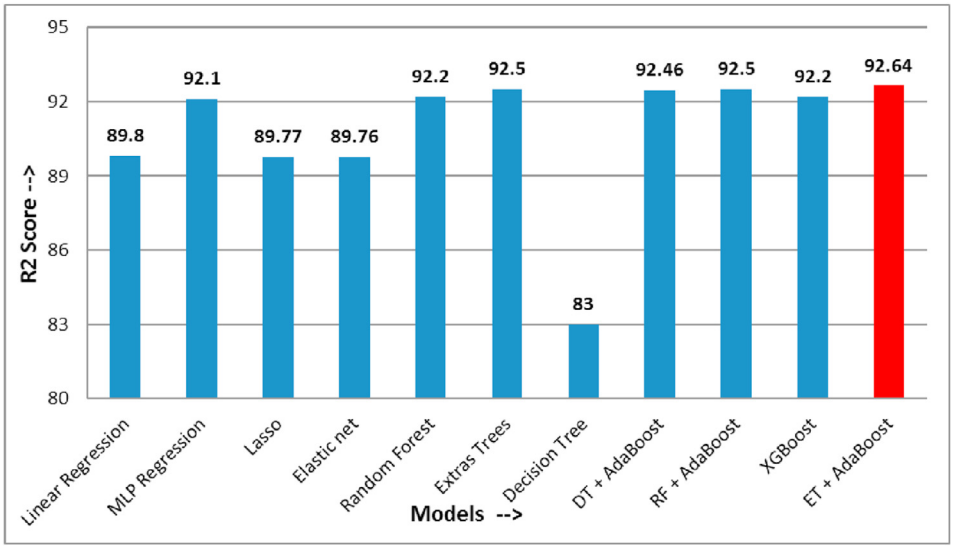

Figure 8. R2-score.

A comparative study of MAE and RMSE, shown in Figure 9, indicates that the proposed model (ET+AdaBoost) incurs not only the lowest mean absolute error but the lowest root mean square error also.

While comparing the R2-Score of the proposed model with LSTM and XGBoost models, it is observed that the performance of the proposed model (ET + AdaBoost) is significantly better. The proposed model is contributing to the highest R2-score, as shown in Figure 10.

Overall, observations from Figures 6 and 7, 8, 9, and 10 show that the performance of the proposed model is better as it contributes to comparatively lower MAE and RMSE values. On R2-Score, it exhibits the highest among all the models. Thus, on the basis of lower MAE and RMSE values and higher R2-Score, we can conclude that the proposed model is the most appropriate model for PM2.5 prediction.

7. Discussion

We have studied the estimation of PM2.5 pollutant particles in Delhi Air, India. The data was collected from the R K Puram area, by DPCC since 1st January 2018 to 30th November 2019.

In the data analysis phase, it has been observed from the violin plot that the PM2.5 concentration is quite high. It is more than 500 in the months of November to January, while in other months especially in the summer season, its value is quite low lying in the range of 0–100. Wind impact on the data shows that in case of low wind speed, PM2.5 concentration is quite high. When the speed increases and is in the range of 4–6 m/second, its value is low. Above 6 m/second, PM2.5 value is almost negligible.

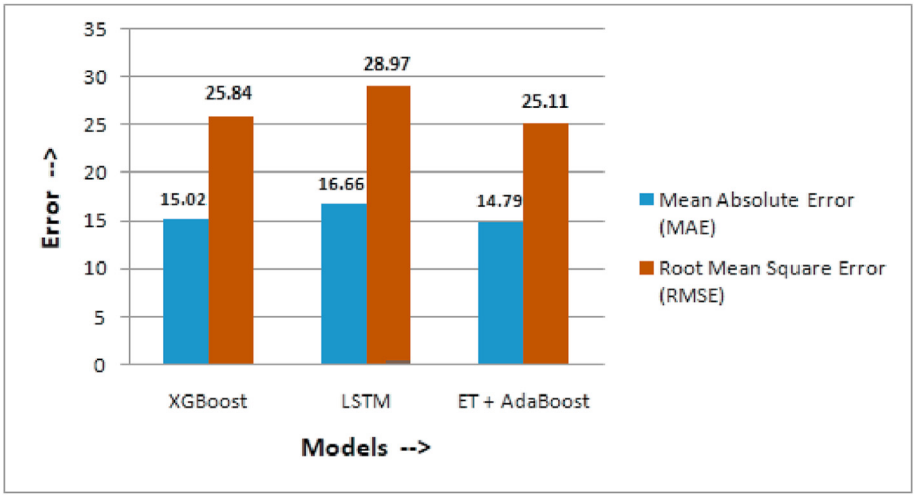

Figure 9. Error comparison.

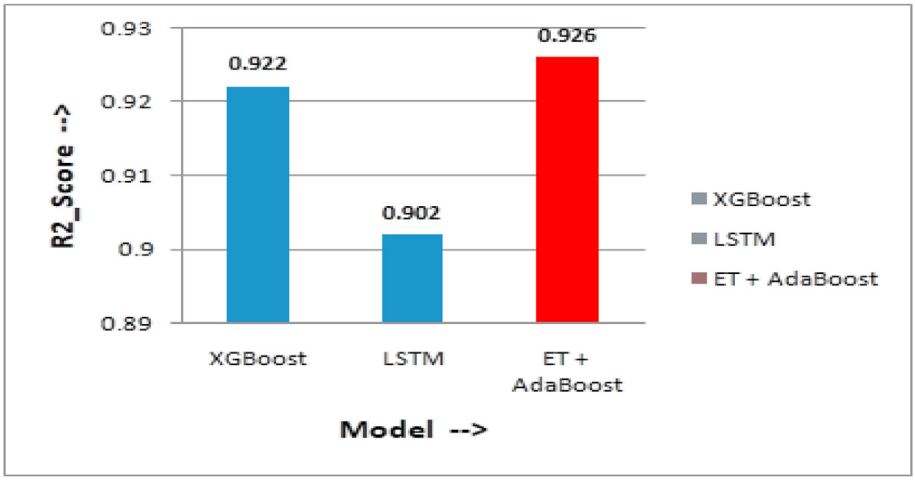

Figure 10. R2-Score comparison.

The performance analysis section of the model shows that actual and predicted values are very close to each other and forms a 45° degree slope as shown in Figure 5. In this work, mainly three metrics have been used to measure the performance of the proposed model, which ae as follows; the first one is Mean Absolute Error (MAE) is used to measure the horizontal separation between actual and measured values.

Root Mean Squared Error is used as a second metric and R2-score is used as third metric to measure the closeness of the predicted values to actual values on a regression line. Figures 6, 7, and 8 show that ET+AdaBoost proposed model is quite effective to measure the PM2.5 pollutant particles. Because its R2-Score is highest and RMSE lowest among all the models. Even MAE is very close to the other optimal models.

In the comparative analysis section, the proposed model is compared with the other recent works and it has been observed that ET+AdaBoost is one of the best performing models.

8. Conclusion

Machine learning-based models are often found to be useful for accurately analyzing the time series based data. In this work, we have proposed an effective ML-based model for the prediction of PM2.5 as the air quality measure in the atmosphere of Delhi. The proposed machine learning-based model is an ensemble of Extra Tree Regressor and Adaptive Boosting. In this, adaptive boosting helps to enhance the learning rate. Performance study indicates that the proposed model i.e. ET+AdaBoost is fairly good in terms of actual and predicted values because these values are almost equal.

To perform the comparative study with other existing models, we have used MAE, RMSE, and R2-score as performance metrics. The comparative analysis reflects that the proposed model is significantly better not only with the traditional contemporary models but even with the new-age models like random forest and boosted decision trees.

A comparison with few recent works, like LSTM and XGBoost, also indicates that the proposed model is not only contributing to the lowest MAE and RMSE even exhibits the highest R2-Score also which concludes the significance of the model. Because PM2.5 particles are susceptible to human lives, therefore, even a slight improvement in the result may create a major difference.

Though the model is generalized well, it still has some limitations associated with it. Being a time-series forecasting task, it continuously requires data for performance analysis. AdaBoost does have one specific problem; it is sensitive towards outliers as the output on any sample is also a function of its predecessor. So one outlier sample affects the prediction on multiple continuous samples. Also, boosting techniques do require extra time for training. So with a large amount of data, it takes more time and is difficult to scale up. Extra Trees improves the overfitting problem of generic random forest algorithm to a large extent, but still somehow prone to overfitting.

Future application of this model can easily be extended to other geographical areas and problems such as predicting water contamination levels, total air quality index, other fine particle concentrations, etc. We can also use this model for other time series-based regression problems, especially in weather forecasting. We can display the continuous forecast made by this model on hourly data, at different places of the city, for the benefit of the common people as air pollution remains a severe health hazard. Thus, the proposed model can be mapped with many real-life genuine problems related to air pollution.

Declarations

Author contribution statement

S. K. Singh: Conceived and designed the experiments; Wrote the paper.

S. Kumar: Performed the experiments; Contributed reagents, materials, analysis tools or data.

S. Mishra: Analyzed and interpreted the data.

Funding statement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data availability statement

Data associated with this study has been deposited at https://app.cpcbccr.com/ccr/#/caaqm-dashboard/caaqm-landing/caaqm-comparison-data.

Declaration of interests statement

The authors declare no conflict of interest.

Additional information

No additional information is available for this paper.

Acknowledgements

Authors would like to acknowledge the editors and the anonymous reviewers for their valuable suggestions in order to improve the manuscript. We also acknowledge Mahatma Gandhi Central University, Bihar, India for the support and cooperation.

References

[1] A. Zanobetti, M. Franklin, P. Koutrakis, J. Schwartz, Fine particulate air pollution and its components in association with cause-specific emergency admissions, Environ Health 8 (1) (2009) 58.
[2] P. Chawala, H. Sandhu, Stubble burn area estimation and its impact on ambient air quality of Patiala & Ludhiana district, Punjab, India, Heliyon 6 (1) (2020), e03095.
[3] K. Adams, D.S. Greenbaum, R. Shaikh, A.M. van Erp, A.G. Russell, Particulate matter components, sources, and health: systematic approaches to testing effects, J. Air Waste Manag. Assoc. 65 (5) (2015) 544–558.
[4] C.I. Davidson, R.F. Phalen, P.A. Solomon, Airborne particulate matter and human health: a review, Aerosol. Sci. Technol. 39 (8) (2005) 737–749.
[5] Y.-F. Xing, Y.-H. Xu, M.-H. Shi, Y.-X. Lian, The impact of PM2. 5 on the human respiratory system, J. Thorac. Dis. 8 (1) (2016) E69.
[6] T.C. Lewis, et al., Air pollution–associated changes in lung function among asthmatic children in Detroit, Environ. Health Perspect. 113 (8) (2005) 1068–1075.
[7] I. Bos, et al., No exercise-induced increase in serum BDNF after cycling near a major traffic road, Neurosci. Lett. 500 (2) (2011) 129–132.
[8] L. Jacobs, et al., Subclinical responses in healthy cyclists briefly exposed to traffic-related air pollution: an intervention study, Environ Health 9 (1) (2010) 64.
[9] A. Bhatnagar, Environmental cardiology: studying mechanistic links between pollution and heart disease, Circ. Res. 99 (7) (2006) 692–705.
[10] A. Valavanidis, K. Fiotakis, T. Vlachogianni, Airborne particulate matter and human health: toxicological assessment and importance of size and composition of particles for oxidative damage and carcinogenic mechanisms, J. Environ. Sci. Health, Part C 26 (4) (2008) 339–362.
[11] S. Kumar, S. Mishra, S.K. Singh, Deep Transfer Learning-Based COVID-19 Prediction Using Chest X-Rays, medRxiv, 2020, p. 2020.
[12] J. Schwartz, D.W. Dockery, L.M. Neas, Is daily mortality associated specifically with fine particles? J. Air Waste Manag. Assoc. 46 (10) (1996) 927–939.
[13] D.W. Graff, M.T. Schmitt, L.A. Dailey, R.M. Duvall, E.D. Karoly, R.B. Devlin, Assessing the role of particulate matter size and composition on gene expression in pulmonary cells, Inhal. Toxicol. 19 (sup1) (2007) 23–28.
[14] Central Control Room for Air Quality Management - Delhi NCR, 2019. https://app.cpcbccr.com/ccr/#/caaqm-dashboard/caaqm-landing/caaqm-comparison-data.
[15] J. Huang, X. Pan, X. Guo, G. Li, Health impact of China's Air Pollution Prevention and Control Action Plan: an analysis of national air quality monitoring and mortality data, The Lancet Planetary Health 2 (7) (2018) e313–e323.
[16] J. Padgett, H. Richmond, The process of establishing and revising national ambient air quality standards, J. Air Pollut. Contr. Assoc. 33 (1) (1983) 13–16.
[17] P. Kumar, et al., New directions: air pollution challenges for developing megacities like Delhi, Atmos. Environ. 122 (2015) 657–661.
[18] C.B. Guerreiro, V. Foltescu, F. De Leeuw, Air quality status and trends in Europe, Atmos. Environ. 98 (2014) 376–384.
[19] I. Djalalova, L. Delle Monache, J. Wilczak, PM2. 5 analog forecast and Kalman filter post-processing for the Community Multiscale Air Quality (CMAQ) model, Atmos. Environ. 108 (2015) 76–87.
[20] Y. Guo, Q. Tang, D.-Y. Gong, Z. Zhang, Estimating ground-level PM2. 5 concentrations in Beijing using a satellite-based geographically and temporally weighted regression model, Remote Sens. Environ. 198 (2017) 140–149.
[21] A. Azid, et al., Prediction of the level of air pollution using principal component analysis and artificial neural network techniques: a case study in Malaysia, Water, Air, Soil Pollut. 225 (8) (2014) 2063.
[22] D.R. Michanowicz, et al., A hybrid land use regression/AERMOD model for predicting intra-urban variation in PM2. 5, Atmos. Environ. 131 (2016) 307–315.
[23] Q. Zhou, H. Jiang, J. Wang, J. Zhou, A hybrid model for PM2. 5 forecasting based on ensemble empirical mode decomposition and a general regression neural network, Sci. Total Environ. 496 (2014) 264–274.
[24] L. Breiman, J. Friedman, R. Olshen, C. Stone, Classification and Regression Trees, Wadsworth International Group, Belmont, USA, 1984. Chapter 9. Bibliography.
[25] L. Breiman, Bagging predictors, Mach. Learn. 24 (2) (1996) 123–140.
[26] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[27] T. Chen, C. Guestrin, Xgboost: a scalable tree boosting system, in: Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785–794.
[28] R.O. Sinnott, Z. Guan, Prediction of air pollution through machine learning approaches on the cloud, in: 2018 IEEE/ACM 5th International Conference on Big Data Computing Applications and Technologies (BDCAT), IEEE, 2018, pp. 51–60.
[29] M. Zamani Joharestani, C. Cao, X. Ni, B. Bashir, S. Talebiesfandarani, PM2. 5 prediction based on random forest, XGBoost, and deep learning using multisource remote sensing data, Atmosphere 10 (7) (2019) 373.
[30] M.H.D.M. Ribeiro, L. dos Santos Coelho, Ensemble approach based on bagging, boosting and stacking for short-term prediction in agribusiness time series, Appl. Soft Comput. 86 (2020) 105837.
[31] C. Xiao, N. Chen, C. Hu, K. Wang, J. Gong, Z. Chen, Short and mid-term sea surface temperature prediction using time-series satellite data and LSTM-AdaBoost combination approach, Remote Sens. Environ. 233 (2019) 111358.
[32] L. Li, S. Dai, Z. Cao, J. Hong, S. Jiang, K. Yang, Using improved gradient-boosted decision tree algorithm based on Kalman filter (GBDT-KF) in time series prediction, J. Supercomput. (2020) 1–14.
[33] G. Bian, J. Liu, W. Lin, Internet Traffic Forecasting Using Boosting LSTM Method," DEStech Transactions on Computer Science and Engineering, 2017.
[34] J.-Y. Tao, Z.-M. Wu, D.-Z. Yue, X.-S. Tan, Q.-Q. Zeng, G.-Q. Xia, Performance enhancement of a delay-based Reservoir computing system by using gradient boosting technology, IEEE Access 8 (2020) 151990–151996.
[35] H. Al-Hadeethi, S. Abdulla, M. Diykh, R.C. Deo, J.H. Green, Adaptive boost LS-SVM classification approach for time-series signal classification in epileptic seizure diagnosis applications, Expert Syst. Appl. 161 (2020) 113676.
[36] Y. Li, T. Bao, J. Gong, X. Shu, K. Zhang, The prediction of Dam displacement time series using STL, extra-trees, and stacked LSTM neural network, IEEE Access 8 (2020) 94440–94452.
[37] V. John, Z. Liu, C. Guo, S. Mita, K. Kidono, Real-time lane estimation using deep features and extra trees regression, in: Image and Video Technology, Springer, 2015, pp. 721–733.
[38] H. Tyralis, G. Papacharalampous, A. Langousis, Super Learning for Daily Streamflow Forecasting: Large-Scale Demonstration and Comparison with Multiple Machine Learning Algorithms, 2019 arXiv preprint arXiv:1909.04131.
[39] P. Geurts, D. Ernst, L. Wehenkel, Extremely randomized trees, Mach. Learn. 63 (1) (2006) 3–42.
[40] Y. Freund, R.E. Schapire, A desicion-theoretic generalization of on-line learning and an application to boosting, in: European Conference on Computational Learning Theory, Springer, 1995, pp. 23–37.

[41] J.C.-W. Chan, D. Paelinckx, Evaluation of Random Forest and Adaboost tree-based ensemble classification and spectral band selection for ecotope mapping using airborne hyperspectral imagery, Remote Sens. Environ. 112 (6) (2008) 2999–3011.
[42] S. Chowdhury, S. Dey, L. Di Girolamo, K.R. Smith, A. Pillarisetti, A. Lyapustin, Tracking ambient PM2. 5 build-up in Delhi national capital region during the dry season over 15 years using a high-resolution (1 km) satellite aerosol dataset, Atmos. Environ. 204 (2019) 142–150.
[43] H. Karimian, et al., Evaluation of different machine learning approaches to forecasting PM2. 5 mass concentrations, Aerosol Air Quality Res. 19 (6) (2019) 1400–1410.
[44] A. Botchkarev, Evaluating Performance of Regression Machine Learning Models Using Multiple Error Metrics in Azure Machine Learning Studio, 2018. Available at SSRN 3177507.