Pragmatic Method Based on Intelligent Big Data Analytics to Prediction Air Pollution

Samaher Al_Janabi(✉), Ali Yaqoob, and Mustafa Mohammad

Department of Computer Science, Faculty of Science for Women (SCIW), University of Babylon, Babylon, Iraq
samaher@itnet.uobabylon.edu.iq

Abstract. Deep learning, as one of the most popular techniques, is able to efficiently train a model on big data by using large-scale optimization algorithms. Although there exist some works applying machine learning to air quality prediction, most of the prior studies are restricted to several-year data and simply train standard regression models (linear or nonlinear) to predict the hourly air pollution concentration. The main purpose of this proposal is design predictor to accurately forecast air quality indices (AQIs) of the future 48 h. Accurate predictions of AQIs can bring enormous value to governments, enterprises, and the general public -and help them make informed decisions. We Will Build Model Consist of four Steps: (A) Determine the Main Rules (contractions) of avoiding emission (B) Obtaining and pre-processing reliable database from (KDD CUP 2018) (C) Building Predator have multi-level based on Long Short-term Memory network corporative with one of optimization algorithm called (Partial Swarm) to predict the PM2.5, PM10, and O3 concentration levels over the coming 48 h for every measurement station. (D) To evaluate the predictions, on each day, SMAPE scores will be calculated for each station, each hour of the day (48 h overall), and each pollutant (PM2.5, PM10, SO_x, CO, O3 and NO_x). The daily SMAPE score will then be the average of all the individual SMAPE scores.

Keywords: Air pollutant prediction · Big data · Prediction · LSTM · PSO

1 Introduction

The proposed title deals with the study of gases that cause air pollution. The proposed title deals with the study of gases that cause air pollution. Air pollution has become a serious threat to large parts of the Earth's population [1]. Air pollution control has attracted the attention of governments, industrial companies and scientific communities. There are two sources of air pollution: the first source is natural sources such as volcanoes, forest fires, radioactive materials, etc. The second source is industrial sources produced by human activities such as factories, vehicles, remnants of war and others. These pollutants produce different types of gases, including sulfur oxides (SOx), carbon monoxide (CO), nitrogen oxides (NO_x) and ozone (O3). Another type of contaminant is $PM_{2.5}$ (particulate matter), a mixture of compounds (solids and liquid droplets), the most dangerous types of contaminants cause cardiovascular disease plus

Y. Farhaoui (Ed.): BDNT 2019, LNNS 81, pp. 84–109, 2020.
https://doi.org/10.1007/978-3-030-23672-4_8

pollutant PM_{10}. This work calculates concentrations of air pollutants (6 types of air pollution above) using a predictive model capable of handling large data efficiently and producing very accurate results. Therefore, a long short-term memory model using the swarm algorithm was used and developed. So that the network is trained to concentrate this polluted air on a number of stations and every hour and the results are evaluated using Root Mean error the scale of assessment so that this model can predict within the next 48 h. These data are analyses, conversion, testing, modelling, accurate information mining, restructuring and storage. This information helps us make decisions about these data and takes them across several stages before they are entered into the network. It is the data collection phase of the different version. This is the data problem to be expected. Are the determinants and stage of understanding the nature of these data in themselves? This data is a very important stage where the data we deal with contains missing data processed by each column and then these data are entered after analysis in a model developed and evaluated by SMAPE.

The remind of this paper organization as follow. Section 2 show the related work. Section 3 explain the main concept used to handle this problem. Section 4 show the prototype of pragmatic system while Sect. 5 show the results. Finally, Sect. 6 description the discussion and conclusion of the problem statement.

2 Related Works

The issue of air quality prediction is one of the vital topics that are directly related to the lives of people and the continuation of a healthy life in general. Since the subject of this thesis is to find a modern predictive way to deal with this type of data that is huge and operates within the field of data series. Therefore, in this part of the thesis, we will try to review the works of the former researchers in the same area of our problem and compare these works in terms of five basic points, namely the database used. The methods to assess the results, the advantages of the method, and their limitations.

Bun et al. [1] used "deep recurrent neural network (DRNN)" that is reinforced with the novel pre-training system using auto-encoder principally design for time series prediction. Moreover, sensors chosen is performed within DRNN without harming the accuracy of the predictions by taking of the sparsity found in the system. The method used to solve particulate matter ($PM_{2.5}$) that is one of the air populations. this method reveals results in more accuracy than the poor performance of the "Noise reduction (AE)". Evaluation the results based on four measures: "Root mean square error (RMSE), precision (P), Recall (R) and F-measure (F)". Our work is similar that work by using the same technique RNN while we predicate based on "long short-term memory".

Samaher et al. [2] the researcher found in this work the hybrid system uses "genetic neural computing (GNC)" to analyze and understand the resulting data from the concentration of dissolved gases. Where it is used in four subgroups for the purpose of analysis and assembly based on the C57.104 Specified by "IEEE" using "GA". Is inserted the clustering data into the neural network for the purpose of predicting different types of errors. This hybrid system generates decision rules which identify the error accurately. There are two measures used in this work are: "The Davies–Bouldin

(DB) index and MSE". This work has proven to provide less cost to solve the problem. And through this method facilitates the process of prediction and identify more accurate ideas through the analysis of errors and ways to address them. This work is similar to our work in terms of using neural networks but the difference is using a" Swarm" algorithm with "LSTM".

Xiang et al. [3] used model "spatiotemporal deep learning (STDL)"-based on air quality prediction method. It used a "stacked autoencoder (SAE)" method to extract inherent air quality characteristics, in addition, it is trained in a greedy layer-wise method. it compared your' model with traditional time series prediction models, the model can predict the air quality of all stations at the same time and shows the temporal stability in all seasons. In addition, a comparison with the "spatiotemporal artificial neural network (STANN)", "autoregression moving average (ARMA)", and "support vector regression (SVR)" models demonstrates. The results of that model evaluation by three measures are "RMSE", "MAE" "MAPE". Our work is similar that work by using the same technique RNN to prediction the air quality index but we deal with huge data, "long-short-term memory" ("LSTM") to enhance network work.

Xiang et al. [4] used "long short-term memory neural network" extended ("LSTME") model which is the association of spatial-temporal links to predict the concentration of air pollutants. Long-term memory layers (LSTM) have been used to automatically extract potential intrinsic properties from historical air pollutants data, and assistive data, Contains meteorological data and timestamp data, has been incorporated into the proposed model to improve performance. This technique evaluation by three measures "RMSE", "MAE" and "MAPE". A comparison with the "spatiotemporal artificial neural network (STANN)", "autoregression moving average (ARMA)", and "support vector regression (SVR)" models demonstrates. Our work is similar that work by using long-short-term memory ("LSTM") as part of the repeated neural network structure. While differ by using another evaluation measure.

Osama et al. [5] "shown a new deep learning-based ozone level prediction model, which considers the pollution and weather correlations integrally. This deep learning model is used to learn ozone level features, and it is trained using a grid search technique. A deep architecture model is utilized to represent ozone level features for prediction. Moreover, experiments demonstrate that the proposed method for ozone level prediction has superior performance. The outcome of this study can be helpful in predicting the ozone level pollution in Aarhus city as a model of smart cities for improving the accuracy of ozone forecasting tools". Results of that model evaluation based on "RMSE", "MAE", "MAPE", squared "R^2" and correlation coefficient. Our work will also use memory in ("LSTM") for the processing large data, but differ by finding the optimal structure of that neural network by partial swarm algorithm.

Lifeng et al. [6] the author found that obtaining the best air quality prediction uses the GM model (1.1) the fractional order accumulation (FGM (1.1)) to find the Expected the average annual concentrations of "PM2.5, PM10, SO2, NO2, and 8-h O3 And O-24 h". The measure used in this search is "MAPE". Using the method of "FGM (1.1)" obtained much higher than the traditional GM model (1.1), the average annual concentrations of "PM2.5, PM10, SO2, NO2, O8-O3, and O3 24-h" will decrease from 2017 to 2020. This work is similar to our work in terms of predicting the concentration of air pollutants and finding ways to address them, but the difference in terms of the method of prediction using "LSTM".

Olalekan et al. [7] in this work the method was used Sensor measurement: SNAQ boxes and network deployment, Sensor measurement validation, Source apportionment to create a predictive model for modeling (ADMS-Airport), the concentration of pollutants to determine the air quality model. The results showed in this study can be applied in many environments that suffer from air pollution. Which will reduce the potential health effects of air quality and the lowest cost, as well as monitor greenhouse gas emissions? This work is similar to our work to determine the concentration of air pollutants but the method used for our work is "LSTM –RNN".

Congcong et al. [8] in order to extract high temporal-spatial features have been used by the Merge of the "convolutional neural network (CNN)" and "long short-term memory neural network (LSTM-NN), and meteorological data and aerosol data were also Merge, in order to refinement model prediction performance. The data collected from 1233 air quality monitoring stations in Beijing and the whole of China were used to verify the effectiveness of the proposed C-LSTME model. The results show that the present model has achieved better performance than current state-of-the-art Technologies for different time predictions at various regional and environmental scales. This technique evaluation by three measures "RMSE", "MAE" and "MAPE". Our work used "LSTM" through RNN but after fined the best structure of that network. It differs by using another evaluation measures.

Zhigen et al. [9] the prediction method was based on "classification and regression tree ("CART")" and was combined with the "ensemble extreme learning machine (EELM)" method. Subgroups were created by dividing datasets by creating a shallow hierarchy tree by dividing the data set through "CART". Where at each node of the tree "EEL Models" are done using the training samples of the node, to minimize the verification errors sequentially to all the tree sub-trees by identifying the numbers of hidden intestines where each node is considered as root. Finally, EEL models for each path of the leaf is compared to the root on each leaf and only one path is selected with the smallest error checking the leaf. The measures that used are RMSE and MAPE. This method proved that the experimental results of the measurements used: can address the global-local duplication of the method of prediction in each leaf, "CART-EELM" work better than the models "RF, v-SVR, and EELM", "CART-EELM" also shows superior performance compared "EELM and K-means-EELM seasonal". Our work is similar to this work using the same set of six data for air pollution "(PM2.5, O3, PM10, SO2, NO2, CO)", but we differ in terms of the mechanism of reducing air pollutants where we use the RNN method.

Hongmin et al. [10] using a new air quality forecasting method, and proposing a new positive analysis mechanism that includes complex analysis, improved prediction units, pre-treatment data, and air quality control problems. This system analyzes the original series using the entropy model and data processing process. The "MOMVO" algorithm was used to achieve the required standards and "LSSVM" to achieve the best accuracy in addition to stable prediction. There are three ratings used in this work are: "RMSE", "MAE" and "MAPE". The result of the application of the proposed method on the data set showed a good performance in the analysis and control of air quality, in addition to the approximation of values with high precision. Our work used the same

Table 1. Compare among the previous works

Name	Dataset/database	Method	Evaluation	Advantage
Bun et al. [1]	Air quality index "(AQI)" http://uk-air.defra.gov.uk	"(DRNN)" that is enhanced with a novel pre-training method using auto-encoder	• "(RMSE)" • "(P)" • "(R)" • "(F)"	1- The numerical experiments show that DRNN with our proposed pre-training method is superior to when using a canonical and a state-of-the-art auto-encoder training method when applied to time series prediction. The experiments confirm that when compared against the PM2:5 prediction system VENUS 2- NN was known as "(RNN)", in contrast with "(FNN)", has been shown to exhibit very good performance in modeling temporal structures and has been successfully applied to many real-world problems
Samaher et al. [2]		• "(GNC)" • "BPNN"	• "MSE" • "(DB)"	This work has proven to provide less cost to solve the problem. And through this method facilitates the process of prediction and identify more accurate ideas through the analysis of errors and ways to address them
Xiang et al. [3]	Air quality using PM2.5 (http://datacenter.mep.gov.cn/).	1- "(STDL)" based air quality prediction method 2- "(SAE)"	• "(RMSE)" • "(MAE)" • "(MAPE)"	3- Compared with traditional time series air quality prediction models, our model was able to predict the air quality of all monitoring stations simultaneously, and it showed satisfactory seasonal stability. We evaluated the performance of the proposed method and compared it with the performance of the "STANN", "ARMA", and "SVR" models and the results showed that the proposed method was effective and outperformed the competitors
Xiang et al. [4]	Air quality using PM2.5 (http://datacenter.mep.gov.cn/)	• "(LSTM)"	• "(RMSE)" • "(MAE)" • "(MAPE)"	The "LSTME" model is capable of modeling time series with longtime dependencies and can automatically determine the optimum time lags. To evaluate the performance of our proposed model, Compared with Six different models, including our "LSTME", the traditional "LSTM" "NN", the "STDL", the "TDNN", the "ARMA" and the "SVR" models
Osama et al. [5]	Citypulse dataset. (ftp://http://iot.ee.surrey.ac.uk:)	1-Deep learning approach 2-In-Memory computing	• "(RMSE)" • "(MAE)" • "(MAPE)" • "(R^2)" • "(r)"	The proposed method is evaluated on citypluse dataset and compared with SVM, NN, and GLM models. the comparison results show that the proposed model is efficient and superior as compared to already existing models

(continued)

Table 1. (*continued*)

Name	Dataset/database	Method	Evaluation	Advantage
Lifeng et al. [6]	BBC, 2013. Beijing Smog: when Growth Trumps Life in China. www.bbc.com/news/magazine-21198265	"FGM(1,1)"	• "RMSE" • "MAPE"	Using the method of "FGM (1.1)" obtained much higher than the traditional GM model (1.1), the average annual concentrations of "PM2.5, PM10, SO2, NO2, O8-O3, and O3 24-h" will decrease
Olalekan et al. [7]		• "SNAQ boxes and network deployment" • Sensor measurement validation. Source apportionment		The results showed in this study can be applied in many environments that suffer from air pollution
Congcong et al. [8]	Hourly PM2.5 concentration data from 1233 air quality monitoring stations in China collected from January 1, 2016, to December 31, 2017, were acquired from the Ministry of Environmental Protection of China (http://datacenter.mep.gov.cn/).	combination of the "(CNN)" and "(LSTM-NN)"	• "(RMSE)" • "(MAE)" • "(MAPE)"	(1) The addition of PM2.5 information from neighboring stations, which contributes to the spatiality of the data, can considerably improve the prediction accuracy of the model (2) The supplement of auxiliary data can help predict sudden changes in air quality, thereby improving the prediction performance of the model. Moreover, compared to meteorological data, the "AOD" data contributes more to the accuracy of the model (3) The present model can efficiently extract more essential spatiotemporal correlation features through the combination of "3D-CNN" and stateful "LSTM", thereby yielding higher accuracy and stability for air quality prediction of different spatiotemporal scales

(*continued*)

Table 1. (*continued*)

Name	Dataset/database	Method	Evaluation	Advantage
Zhigen et al. [9]	Yancheng city, which is one of the 13 cities under the direct administration of Jiangsu Province, China. Yancheng city spans between northern latitude 32°34′–34°28′, eastern longitude 119°27′–120°54′	"CART" and "EELM"	• "(RMSE)" • "(MAPE")	The experimental results of the measurements used: can address the global-local duplication of the method of prediction in each leaf, "CART-EELM" work better than the models "RF, v-SVR, and EELM", "CART-EELM" also shows superior performance compared "EELM and K-means-EELM seasonal
Hongmin et al. [10]	The datasets from eight sites in China. (https://data.epmap.org)	• "LSSVM" • "MOMVO"	• "(RMSE)" • "(MAE)" • "(MAPE)"	The application of the proposed method on the data set showed a good performance in the analysis and control of air quality, in addition to the approximation of values with high precision

evaluation measures but it differs by using "LSTM" through RNN but after fined the best structure of that network.

Table 1 Shown the compare among the previous works based on five points are types of dataset, methodology used, evaluation measures, and advantages.

3 Main Concept

3.1 Big Data

There are three types of data. The first type is small data: these data are not subject to normal distribution and therefore cannot be determined and difficult to predict due to their small size and less than 30 samples, the second type is the normal data is the most common data that is organized in the table consists of columns and rows, For large data that we will deal with in our work. These big data cannot be handled in the traditional way due to their large size. From TB to ZB, these big data are either structural, semi-structured or unorganized. In large data, there are three conditions that must be met: First, the existence of available data called the data domain has three characteristics of volume, Velocity and Variety. The second condition: There is a statistic for this data called (statistical domain), which has three characteristics are Veracity, Variability and Validity. The third condition is the existence of the beneficiary of this data called (intelligence analysis domain) which has three characteristics of Visibility, value and Verdict [11].

3.2 Big Data Analysis Stages

The attention to large data has become of great importance to many organizations and companies over the last few years so many technologies have been developed to meet the challenges of dealing with huge data The process of analysis of the statement through several stages: the data deployment stage is the stage of data collection from different sources, the stage of understanding the problem: the phase of understanding the problem of data and the problem of this proposal are the gases and vehicles that result from air pollution, data exploration stage: After understanding the data problem, Itself. Does this data contain missing data? Is this data of an integer type, type of characters, or mixture between them?, data pre-processing stage This is a very important stage because the data we deal with contains missing data to be processed by each column in addition to the filtering of data. After processing the data, we need to build a predictive model for this data, that called the data modelling stage and then evaluate the data using one of the different measurements called the data assessment stage [12].

3.3 Deep Learning

It is a new research field that examines the creation of theories and algorithms that allow the machine to learn by itself by simulating neurons in the human body. And one of the branches of science that deals with artificial intelligence science. From an

Automated Learning Science branch, most in-depth learning research focuses on finding methods to derive a high degree of strippers by analyzing a large data set using linear and nonlinear mutants. “Discoveries in this area have proven significant, rapid and effective progress in many areas, including face recognition, speech recognition, computer vision, and natural language processing.” Deep learning is a branch of learning, which is a branch of artificial intelligence. Where deep learning is divided into three basic categories: Deep networks under supervision or deep discrimination Models, deep unsupervised networks or giant models, deep hybrid networks” [13].

3.4 Prediction

Prediction is the process of studying and making guesses about future time events using specific mechanisms across time periods and different spatial intervals. It is a way for policymakers to develop plans to address future risks. The process of prediction goes through several stages, the most important of which are: determining the purpose of forecasting, gathering the data needed for a predictable phenomenon, analyzing and extracting data, and selecting the appropriate model or mechanism to predict the phenomenon under consideration and to take the appropriate decision. The forecasting process is capable of predicting the state of data based on advance data, a necessary step to give very low error rates [14] (Fig. 1).

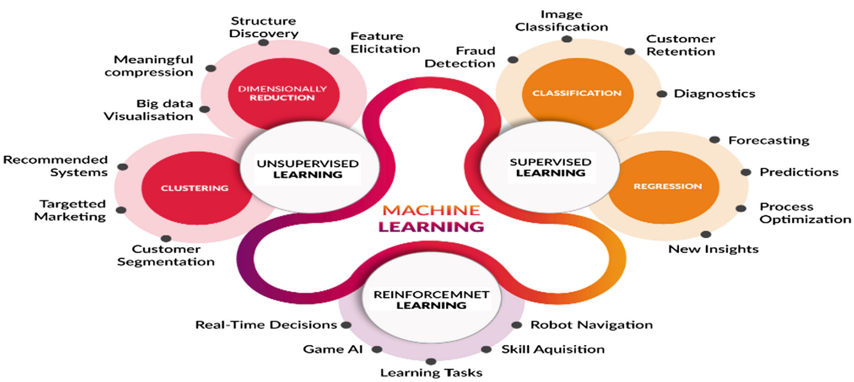

Fig. 1. Main Types of Machine Learning Techniques [19]

3.5 Air Pollution

Pollution is a serious concern that has attracted the attention of industrial companies, the international community and scientific and health communities because it contains substances harmful to the lives of humans and other living organisms. These pollutants produce different types of concentrations that cause air pollution, including sulfur dioxide SO_2, Nitrogen Dioxide gas NO_2, Carbon monoxide gas CO, Ozone gas O_3, Vehicles with mixed composition $PM_{2.5}$ and PM_{10}. Predicting concentrations that cause air pollution is very important for health community organizations and to launch an early warning and decide on these concentrations [15].

4 A Prototype of Smart Air Quality Prediction Model (SAQPM)

4.1 Problem Statement

The problem of air pollution has serious health aspects to humans and other living organisms because of the presence of harmful concentrations and very dangerous. So the prediction of air pollution attracted many companies, governments, and the scientific community to study these concentrations that cause air pollution. Where many techniques were used but did not produce results at the required level for several reasons, including Most of the techniques used cannot deal with very large data in addition to some techniques lose data because they do not contain a memory able to save this data until deep learning techniques emerged.

Deep learning, as one of the most popular techniques, is able to efficiently train a model on big data by using large-scale optimization algorithms. Although there exist some works applying machine learning to air quality prediction, most of the prior studies are restricted to several-year data and simply train standard regression models (linear or nonlinear) to predict the hourly air pollution concentration. The main purpose of this LSTM-RNN model is to predict future concentrations of air pollution 48. The challenge in this work is how to choose an algorithm that can predict the large data set at high resolution, taking into account that previous readings are not ignored, that is, they maintain all readings.

Objectives:

We will review the main objectives of this research (Fig. 2)

- Identify key constraints to avoid emissions.
- To give a specific definition of air pollution.
- Build LSTM-RNN prediction model in conjunction with a swarm algorithm to predict. The air-polluting concentration of each station and each hour to predict the concentrations of air pollution during future 48.
- Validating the suggested LSTM-RNN force using SMAPE to measure results.

In this paper, we will attempt to find answer for the following questions

- How particle swarm can be useful in building a recurrent neural network (RNN)?
- How to build a multi-layer model with a combination of two technologies) LSTM-RNN with particle swarm) so that this model can predict the concentrations of air pollution accurately and efficiently over the next 48 h?
- Is SMAPE measure enough to evaluate the results of suggesting predictor?
- How to install sensors and get data related to concentrations of air pollution?
- How can you combine two deep learning techniques that lead to reduced time and the complexity of the training phase?

4.2 Particle Swarm Optimization (PSO)

Is a scientific technique developed to improve solutions problems and solutions to solve the best solution to these problems and is one of the latest areas of evolution in

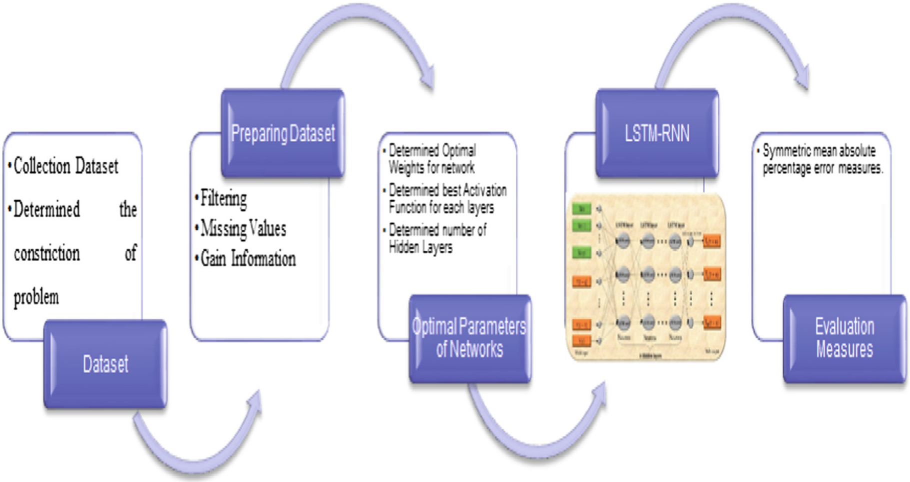

Fig. 2. Block Diagram of Proposed DLSTM-RNN

the field of artificial intelligence, developed by the world (Kennedy and Eberhart) in 1990, and the idea of PSO of the behavioral and social behavior of bird droppings through the idea of research About the food, where the bird squadron are looking for food from one place to another and that some birds in the squadron have the ability to distinguish the smell of food in a strong and effective with information for these birds about the best place to eat because some birds send information among themselves during the search process and Inspection of the best place for food, when the bird flock is explored for a good place for food quality, these birds use this place to get better food. Thus, the work of the swarm algorithm is the search process and replication process of the best solutions within the specific research area. The PSO algorithm can be used to solve optimization problems and problems that change by the time [16, 17].

4.2.1 Basic Components of the Bird Swarm Algorithm (PSO)

The swarm algorithm consists of the numbers of the population of the swarm called particles. Symbolizes them (n) which consists of n = (n1, n2, …, ni) which moves within the swarm of the search space determined by the type of problem which is multi-dimensional and the search for good initial solutions. Particles depend on their own expert and also rely on the experts and experiences of neighboring particles within the swarm, The PSO algorithm is randomly assigned to the number of particles of the squadron in the search area. The use of the squadron particles when creating the PSO algorithm depends on the velocity of the particles that comprise $V_i^t = (V_1^t, V_2^t, \ldots, V_i^t)$ the location of the particles that are composed of $X_i^t = \left(X_1^t, X_2^t, \ldots, X_i^t\right)$, where it is determined based on the previous cases of the best location of the particle itself and symbolizes it $P_{best,i}^t$, The best location of the particles in the entire swarm symbolizes it

$G^t_{best,i}$, Depending on the dimensions of problem d, consisting of (d1, d2,…dj), The speed and location of each particle are adjusted according to the following equations:

$$V_i^{t+1} = (V_1^t + c_1 r_1^t \left(P^t_{best,i} - X_i^t\right) + c_2 r_2^t \left(G^t_{best,i} - X_i^t\right) \tag{1}$$

$$X_i^{t+1} = X_i^t + V_i^t \tag{2}$$

Where:

V_i^t: Particle velocity i in swarm in dimension j and frequency t.
X_i^t: The location of the particle i in a swarm in dimension j and frequency t.
c_1: acceleration factor related to Pbest.
c_2: Acceleration factor related to gbest.
r_1^t, r_2^t: random number between 0 and 1.
t: Number of occurrences specified by type of problem.
$G^t_{best,i}$: gbest position of swarm
$P^t_{best,i}$: pbest position of particle.

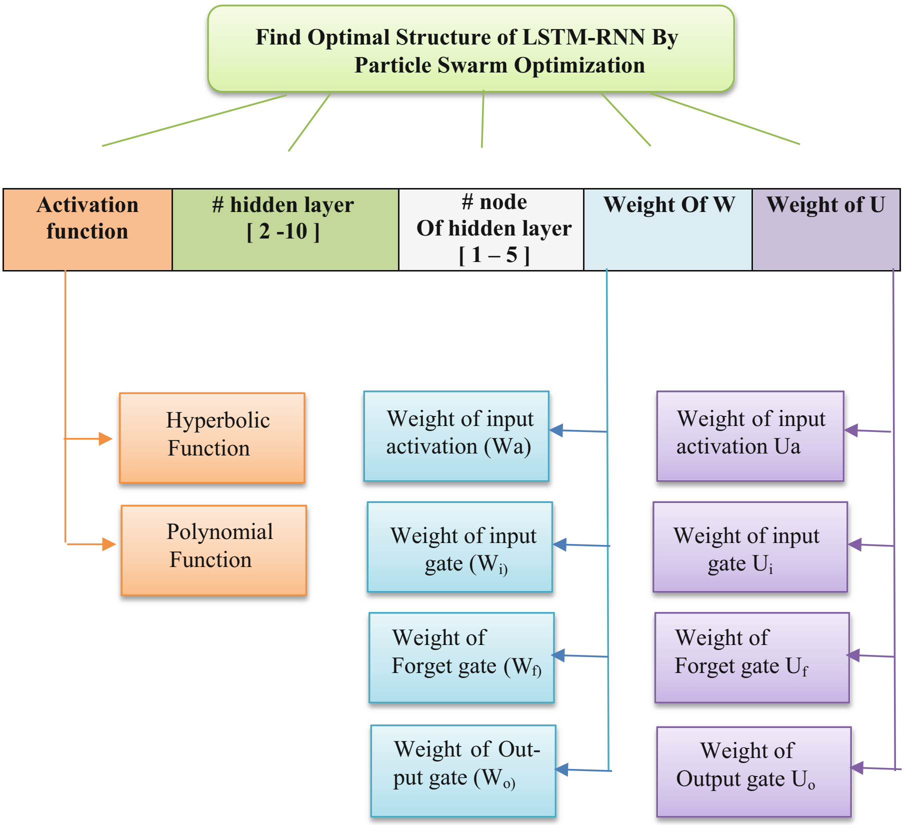

Fig. 3. PSO algorithm is to optimize the LSTM-RNN

The aim of the PSO algorithm is to optimize the structure of LSTM-RNN from determined the best activation function for each layer, determined number of hidden layers in network, number of neurons in each hidden layer, and the optimal weights between each layer and the layer next it as described in the diagram (Fig. 3 and Tables 2 and 3):

Table 2. Hyperbolic Functions with discretion [20]

Hyperbolic function		
Name of hyperbolic function	#variable	Function
Sinh	One	F(x) = $\frac{e^x - e^{-x}}{2}$
Cosh	One	F(x) = $\frac{e^x + e^{-x}}{2}$
Tanh	One	F(x) = $\frac{e^x - e^{-x}}{e^x + e^{-x}}$
Sinh^{-1}	One	F(x) = $\frac{2}{e^x - e^{-x}}$
Cosh^{-1}	One	F(x) = $\frac{2}{e^x + e^{-x}}$
Tanh^{-1}	One	F(x) = $\frac{e^x + e^{-x}}{e^x - e^{-x}}$

Where: x is input and F(x) is output.

Table 3. Polynomial Functions with discretion [20]

Polynomial functions		
Name of polynomial function	#variable	Function
Linear	One	F(x) = p1+p2*x1
Linear	Tow	F(x) = p1+p2*x1+p3*x2
Linear	Three	F(x) = p1+p2*x1+p3*x2+p4*x3
Quadratic	One	F(x) = p1+p2*x1+p3*x1^2
Quadratic	Tow	F(x) = 1+p2*x1+p3*x1^2+p4*x2 +p5*x2^2+p6*x1*x2
Cubic	One	F(x) = p1+p2*x1+p3*x1^2 +p4*x1^3
Product	Tow	F(x) = p1+p2*x1*x2
Ratio	Tow	F(x) = p1+p2*(x1/x2)
Logistic	One	F(x) = p1+p2/(1+ exp(p3*(x1-p4)))
Log	One	F(x) = p1+p2*Log(x1+p3)
Exponential	One	F(x) = p1+p2*exp((p3*(x1+p4))
Asymptotic	One	F(x) = p1+p2/(x1+p3)

Where:
x is input
f(x) is output
p1, p2, p3, p4: is constants

Algorithm #1: PSO [21]

```
Set of parameters
A: Population of agents, p_i: Position of agent a_i in the solution space, f: Objective function
v_i : Velocity of agent's a_i , V(a_i): Neighborhood of agent a_i (fixed)
[x*] = PSO()
P = Particle_Initialization()
For i=1 to it_max
  | For each particle p in P do
  |   | fp = f(p)
  |   | If fp is better than f(pBest)
  |   |   | pBest = p
  |   | End
  | End
  | gBest = best p in P
  | For each particle p in P do
  |   | # v = v + c1*rand*(pBest – p) + c2*rand*(gBest – p)
  |   | # p = p + v
  | End
End
```

4.3 Long Short-Term Memory (LSTM)

LSTM was proposed in 1997 by Sepp Hochreiter and Jürgen Schmidhuber. By introducing Crossover Fixed Error (CEC) modules, LSTM deals with gradient and burst problems. The initial version of the block included LSTM cells, input and output gateways. LSTM achieved record results in natural language text compression, unauthorized handwriting recognition and won the Handwriting Competition (2009). LSTM networks were a key component of the network, which achieved a standard audio error rate of 17.7% over the traditional natural speech data set (2013). (LSTM) are units of the Recurrent Neural Network (RNN). The RNN is often called LSTM (or LSTM only). The common LSTM module consists of a cell, an input port, an output port, and a forgotten gateway. The cell remembers values at random intervals and the three gates regulate the flow of information inside and outside the cell. LSTM networks are well suited for classifying, processing and predicting predictions based on time series data, where there may be an unknown delay for important events in a time series. LSTMs have been developed to deal with fading and fading problems that can be encountered when training traditional RNNs. The relative lack of sense of gap length is the LSTM feature on RNNs, Hidden Markov models and other sequential learning methods in many applications [18].

4.3.1 LSTM Architecture and Algorithm

There are many structures for LSTM modules. The common structure of a cell (the memory part of the LSTM module) and three "organizations", usually called portals, consist of the flow of information within the LSTM module: an input gateway, an output gateway, and a forgotten gateway. Some differences in the LSTM module do not contain one or more of these portals or may have other portals. For example, duplicate units do not contain portals (GRUs) on the output portal.

Intuitively, the cell is responsible for tracking dependencies between elements in the input sequence. The input gateway controls the extent to which a new value is flowing into the cell. The ny gate controls how long a value in the cell and the output gateway controls how much the value in the cell is used to calculate the output activate the LSTM module. LSTM port activation function is often the logistics function.

There are connections to and from LSTM portals, a few of which are frequent. The weights of these links, which must be learned during training, determine how the gates work.

4.3.2 The Variables in LSTM – RNN

This algorithm required setup multi variables at the beginning then through it work will update these variables by apply computation operations. as shown below (Fig. 4):

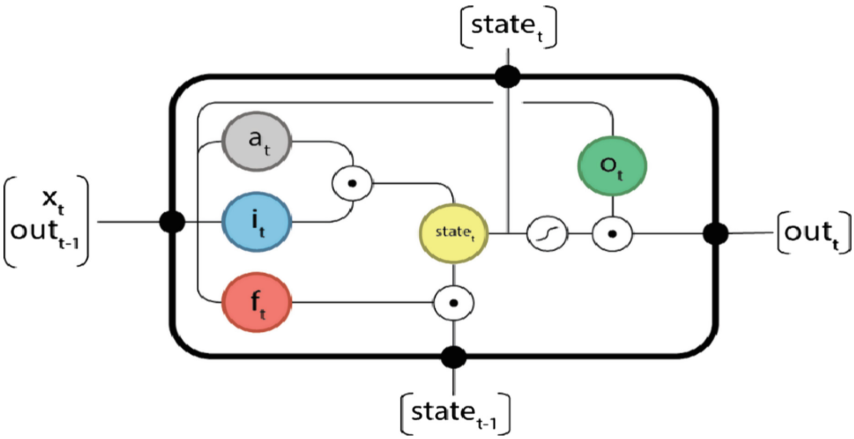

Fig. 4. LSTM cell

Step 1: The forward components
Step 1.1: *Compute the gates:*
Input activation:

$$\mathrm{a_t = tanh(W_a.\ X_t +\ U_a.\ out_{t-1} + b_a)} \tag{3}$$

Input gate:

$$\mathrm{i_t\ =\ \sigma(WI.\ X_t +\ U_i.\ out_{t-1} + b_i)} \tag{4}$$

Forget gate:

$$\mathrm{f_t \sigma\ (Wf_. X_t +\ U_f.\ out_{t-1} + b_f)} \tag{5}$$

Output gate:

$$\mathrm{o_t \sigma(Wo.\ X_t +\ U_o.\ out_{t-1} + b_o)} \tag{6}$$

Then fined:
Internal state:

$$\mathrm{State =\ a_t \odot i_t +\ f_t \odot\ state_{t-1}} \tag{7}$$

Output:

$$\mathrm{outt = tanh(state)\ \odot\ ot} \tag{8}$$

Where

$$\text{Gate } \mathrm{S_t} = \begin{bmatrix} a_t \\ i_t \\ f_t \\ o_t \end{bmatrix}, \mathrm{W} = \begin{bmatrix} W_a \\ W_i \\ W_f \\ W_o \end{bmatrix}, \mathrm{U} = \begin{bmatrix} U_a \\ U_i \\ U_f \\ U_o \end{bmatrix}, \mathrm{b} = \begin{bmatrix} b_a \\ b_i \\ b_f \\ b_o \end{bmatrix}$$

Step. 2: The backward components:
Step 2.1. Find
Δt the output difference as computed by any subsequent.
ΔOUT the output difference as computed by the next time-step

$$\mathrm{\delta out_t = \Delta_t + \Delta out_t} \tag{9}$$

$$\delta state_t = \delta out_t \odot o_t \odot \left(1 - tanh^2(state_t)\right) + \delta state_{t+1} \odot f_{t+1} \tag{10}$$

Step 2.2: Gives:

$$\delta a_t = \delta state_t \odot i_t \odot (1 - a_t^2) \tag{11}$$

$$\delta i_t = \delta state_t \odot a_t \odot i_t \odot (1 - i_t) \tag{12}$$

$$\delta f_t = \delta state_t \odot state_{t-1} \odot f_t \odot (1 - f_t) \tag{13}$$

$$\delta o_t = \delta out_t \odot \tanh(state_t) \odot o_t \odot (1 - o_t) \tag{14}$$

$$\delta x_t = W^t.\delta state_t \tag{15}$$

$$\delta out_{t-1} = U^t.\delta state_t \tag{16}$$

Step 3: update to the internal parameter

$$\delta W = \sum\nolimits_{t=0}^{T} \delta gates_t \otimes x_t \tag{17}$$

$$\delta U = \sum\nolimits_{t=0}^{T} \delta gates_{t+1} \otimes out_t \tag{18}$$

$$\delta b = \sum\nolimits_{t=0}^{T} \delta gates_{t+1} \tag{19}$$

Algorithm #2: DLSTM-RNN

Input: X: Dataset of Air Pollution
Output: prediction value of $PM_{2.5}$, PM_{10}, O_3, SO_2, CO, NO_2

1. Set of Parameters: X_t , OUT_t , W_a , W_i , W_f , W_o ,U_a , U_i , U_f , U_o , b_a , b_i , f_t , b_o , $STEAT_t$.
2. The forward components
3. Call PSO
4. For each Time (t) in X and OUT, apply Dataset by:
5. Compute: a_t, i_t, f_t, o_t.
6. # Input activation: $a_t = \tanh (W_a . X_t + U_a. out_{t-1} +b_a)$
7. # Input gate: $i_t = \sigma(WI. X_t + U_i. out_{t-1} +b_i)$
8. # forgte gate: $f_t = \sigma (Wf. X_t + U_f . out_{t-1} +b_f)$
9. # Output gate: $O_t = \sigma (Wo. X_t + U_o . out_{t-1} +b_o)$
10. Compute $STATE_t$, OUT_t
11. # internal state : $state_t = a_t \odot i_t + f_t \odot state_{t-1}$
12. # output : $out_t = \tanh(state_t) \odot o_t$
13. The Backward components.
14. Given Δt the output difference as computed by any subsequent.
15. Given ΔOUT the output difference as computed by the next time-step
16. For each time (t) to update OUTt,$STATEt$
17. # $\delta out_t = \Delta_t + \Delta out_t$
18. # $\delta state_t = \delta out_t \odot o_t \odot \left(1 - \tanh^2(state_t)\right) + \delta state_{t+1} \odot f_{t+1}$
19. For each time (t) to update of input Xt and ΔOUTt.
20. Update of Input activation, Input gate, Forget gate and Output gate.
21. # $\delta a_t = \delta state_t \odot i_t \odot (1 - a_t^2)$
22. # $\delta i_t = \delta state_t \odot a_t \odot i_t \odot (1 - i_t)$
23. # $\delta f_t = \delta state_t \odot state_{t-1} \odot f_t \odot (1 - f_t)$
24. #$\delta o_t = \delta out_t \odot \tanh (state_t) \odot o_t \odot (1 - o_t)$
25. # $\delta x_t = W^t . \delta state_t$
26. # $\delta out_{t-1} = U^t . \delta state_t$
27. END
28. END
29. The final updates to the internal parameters is compute:
30. # $\delta W = \sum_{t=0}^{T} \delta gates_t \otimes x_t$
31. # $\delta U = \sum_{t=0}^{T} \delta gates_{t+1} \otimes out_t$
32. # $\delta b = \sum_{t=0}^{T} \delta gates_{t+1}$
33. Using SMAPE to evaluate the resulted.
34. # SAMPE = $\frac{1}{n} \sum_{t=1}^{n} \frac{|F_t - A_t|}{|A_t + F_t|/2}$.

END.

---

5 Experiment

Using (DLSTM) networks in Python and how you can use them to make appropriate with concentrations of air pollution predictions! In this network, we will see how you can use a time-series model known as Long Short-term Memory. LSTM models are powerful, especially for retaining a long-term memory, by design, as you will see later.

5.1 Data and Method

We will be using data from KDD cup 2018, where contain the name of the station and Pollution time for each of the following concentrations per hour (Table 4).

Table 4. Before handle the missing values

No	Utc_Time	PM2.5	PM10	NO2	O3	CO	SO2
6	1/1/2017 19:00	429.0	141.0	6.5	3.0	9.0	NaN
7	1/1/2017 20:00	211.0	110.0	3.3	11.0	NaN	NaN
… …							
309327	11/24/2017 1:00	19.0	20.0	19.0	0.3	28.0	2.0
309329	11/24/2017 2:00	9.0	21.0	27.0	28.0	0.4	2.0

The concentrations are: $PM_{2.5,}$ $PM_{10,}$ Sox, CO, NOx, O3.

The table above shows that there are incomplete values that will be preprocessed by a MEAN equation.

Then we will preprocess the missing values through each column (Table 5):

Table 5. After handle the missing values

No	Utc_Time	PM2.5	PM10	NO2	O3	CO	SO2
6	1/1/2017 19:00	429.0	141.0	6.5	3.0	9.0	11.212
7	1/1/2017 20:00	211.0	110.0	3.3	11.0	15.78	11.212
… …							
309327	11/24/2017 1:00	19.0	20.0	19.0	0.3	28.0	2.0
309329	11/24/2017 2:00	9.0	21.0	27.0	28.0	0.4	2.0

After processing for each column using MEAN the following table shows the description of the results in the previous table (Table 6).

Table 6. The description of the data after the preprocessing

	PM2.5	PM10	NO2	CO	O3	SO2
Count	200.000	200.000	200.000	200.000	200.000	200.000
Mean	179.949	134.376	27.180	17.205	15.788	11.212
Std	131.835	123.790	56.373	20.671	11.056	2.788
Min	5.000	4.600	0.200	0.200	2.000	2.000
Max	500.000	561.000	208.000	79.000	61.000	37.000

5.2 Data Visualization

Now let's see what sort of data you have. You want data with various patterns occurring over time (Fig. 5).

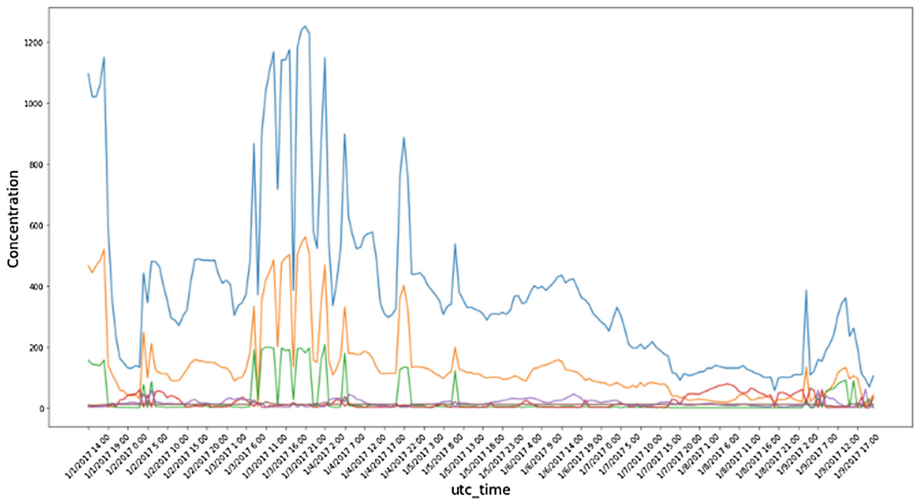

Fig. 5. Data Visualization

This graph already says a lot of things. The specific reason I picked this data that this graph is bursting with different behaviors of for concentrations of air pollution over time. This will make the learning more robust as well as give you a chance to test how good the predictions are for a variety of situations.

Another thing to notice is that the values at the beginning 2017 are much higher and fluctuate more than the values close to the last days. Therefore, you need to make sure that the data behaves in similar value ranges throughout the time frame. You will take care of this during the data normalization phase.

5.3 Normalizing the Data

Before the normalizing, we will the Data process Splitting Data into a Training set and a Test set, where we use the 70% for training and 30 for testing.

Now we need to define a scaler to normalize the data. Min Max Scalar scales all the data to be in the region of 0 and 1. You can also reshape the training and test data to be in the shape [data_size, num_features].

Due to the observation, we made earlier, that is, different time periods of data have different value ranges, and we normalize the data by splitting the full series into windows. If we don't do this, the earlier data will be close to 0 and will not add much value to the learning process. Here you choose a window size of 2500.

We can now smooth the data using the exponential moving average. This helps us to get rid of the inherent raggedness of the data concentrations and produce a smoother curve. Note that we should only smooth training data.

5.4 Data Generator

We are first going to implement a data generator to train our model. This data generator will have a method called. unroll batches (…) which will output a set of num_unrollings batches of input data obtained sequentially, where a batch of data is of size [batch_size, 1]. Then each batch of input data will have a corresponding output batch of data.

For example, if num_unrollings=3 and batch_size=4 a set of unrolled batches It might look like

- **input data**: [x_0, $x_1$0, $x_2$0, $x_3$0, $x_4$0, $x_5$0], [x_1, $x_1$1, $x_2$1, $x_3$1, $x_4$1, $x_5$1], [x_2, $x_1$2, $x_2$2, $x_3$2, $x_4$2, $x_3$5]
- **output data:** [x_1, $x_1$1, $x_2$1, $x_3$1, $x_4$1, $x_5$1], [x_2, $x_1$2, $x_2$2, $x_3$2, $x_4$2, $x_5$2], [x_3, $x_1$3, $x_2$3, $x_3$3, $x_4$3, $x_5$3]

As shown in the results below (Table 7):

Table 7. Data generator to train our model

Unrolled index 0:	
Inputs:	[0.86032474 0.79311657 0.79409707 0.8310883 0.90970576 0.79311657]
Output:	[0.86032474 0.90970576 0.90970576 0.09216607 0.90970576 0.90970576]
Unrolled index 1:	
Inputs:	[0.79311657 0.79409707 0.8310883 0.90970576 0.90970576 0.79409707]
Output:	[0.8310883 0.41866928 0.8310883 0.09216607 0.41866928 0.90970576]
… …	

(*continued*)

Table 7. (*continued*)

Unrolled index 132:	
Inputs:	[0.79409707 0.86032474 0.8310883 0.90970576 0.79409707 0.90970576]
Output:	[0.2070513 0.79409707 0.90970576 0.2070513 0.2070513 0.02807767]
Unrolled index 133:	
Inputs:	[0.8310883 0.79311657 0.90970576 0.8310883 0.8310883 0.79311657]
Output:	[0.2070513 0.41866928 0.02807767 0.90970576 0.2070513 0.79409707]

6 Discussions and Conclusions

In this paper, we used PSO algorithm to find the parameter of best structure related to RNN, these parameters include activation function (i.e., one from six types of Hyperbolic, or one from twelve types of Polynomial Functions), and then from the Experiment we find the best Hyperbolic function is Tanh and the best Polynomial Functions is Linear (Three parameter).

Through the PSO we find the best number of hidden layers is three and the best number of nodes is four. Also, find the best weights of the input between the input layer and hidden layer are (Table 8):

Table 8. The weights of the input between input and first hidden layers

Node 1				Node 2				Node 3			
W_a	W_i	W_f	W_o	W_a	W_i	W_f	W_o	W_a	W_i	W_f	W_o
0.378	0.199	0.305	0.506	0.163	0.263	0.979	0.992	0.545	0.576	0.376	0.889
0.059	0.716	0.352	0.521	0.411	0.986	0.836	0.793	0.928	0.174	0.177	0.901
0.399	0.228	0.628	0.243	0.728	0.440	0.519	0.826	0.204	0.838	0.954	0.411
0.663	0.708	0.069	0.910	0.725	0.698	0.181	0.902	0.522	0.695	0.121	0.136
Node 4				Node5				Node6			
W_a	W_i	W_f	W_o	W_a	W_i	W_f	W_o	W_a	W_i	W_f	W_o
0.486	0.219	0.191	0.990	0.091	0.648	0.422	0.156	0.515	0.797	0.680	0.883
0.018	0.968	0.928	0.564	0.352	0.405	0.386	0.865	0.940	0.938	0.253	0.580
0.062	0.893	0.098	0.950	0.159	0.136	0.356	0.475	0.951	0.447	0.096	0.827
0.725	0.823	0.400	0.291	0.862	0.312	0.074	0.475	0.303	0.544	0.842	0.279

Then find the best weights of input between hidden layers as explained in Tables 9, 10 and 11. While, we explained the weights of recurrent connections in Tables 12 and 13:

After building and developing the LSTM model by the PSO algorithm, this model consists of several layers capable of predicting concentrations of air pollutants. We have (32 stations) products six types of concentrations that cause air pollution

Table 9. The optimal weights of the input **between** first and second layer

Node1				Node2				Node3				Node4			
W_a	W_i	W_f	W_o	W_a	W_i	W_f	W_o	W_a	W_i	W_f	W_o	W_a	W_i	W_f	W_o
0.882	0.256	0.604	0.403	0.373	0.999	0.252	0.875	0.350	0.841	0.158	0.912	0.113	0.814	0.287	0.782
0.447	0.412	0.067	0.900	0.223	0.860	0.594	0.913	0.087	0.273	0.174	0.537	0.300	0.895	0.082	0.636
0.338	0.467	0.832	0.209	0.039	0.091	0.547	0.717	0.286	0.918	0.394	0.215	0.603	0.608	0.930	0.460
0.661	0.362	0.657	0.592	0.900	0.763	0.934	0.029	0.542	0.767	0.998	0.794	0.619	0.623	0.899	0.483

Table 10. The optimal weights of input between second and third layer

Node1				**Node2**				**Node3**				**Node4**			
W_a	W_i	W_f	W_o	W_a	W_i	W_f	W_o	W_a	W_i	W_f	W_o	W_a	W_i	W_f	W_o
0.067	0.724	0.363	0.178	0.709	0.755	0.743	0.843	0.757	0.989	0.547	0.671	0.311	0.930	0.457	0.718
0.422	0.321	0.590	0.597	0.707	0.120	0.699	0.269	0.991	0.030	0.576	0.247	0.238	0.367	0.716	0.740
0.161	0.665	0.109	0.467	0.305	0.992	0.343	0.327	0.053	0.084	0.889	0.092	0.049	0.719	0.816	0.687
0.783	0.388	0.733	0.263	0.130	0.879	0.799	0.906	0.731	0.674	0.420	0.765	0.154	0.443	0.198	0.323

Table 11. The weights of input between third and output layer

Node1				**Node2**				**Node3**				**Node4**			
W_a	W_i	W_f	W_o	W_a	W_i	W_f	W_o	W_a	W_i	W_f	W_o	W_a	W_i	W_f	W_o
0.410	0.492	0.598	0.842	0.651	0.777	0.599	0.033	0.456	0.983	0.647	0.569	0.608	0.006	0.192	0.476
0.137	0.681	0.510	0.839	0.741	0.934	0.680	0.817	0.346	0.848	0.984	0.007	0.041	0.557	0.350	0.329
0.801	0.171	0.317	0.079	0.099	0.613	0.227	0.671	0.385	0.218	0.903	0.890	0.053	0.579	0.625	0.936
0.605	0.418	0.610	0.850	0.351	0.303	0.095	0.008	0.152	0.601	0.087	0.990	0.549	0.913	0.731	0.373
0.132	0.830	0.711	0.139	0.092	0.545	0.181	0.717	0.291	0.376	0.722	0.297	0.036	0.084	0.776	0.853
0.456	0.033	0.562	0.703	0.981	0.003	0.254	0.185	0.839	0.431	0.363	0.470	0.792	0.674	0.376	0.420

Table 12. The weight of recurrent connections

U_a	U_i	U_f	U_o
0.390	0.279	0.435	0.622

Table 13. The result of DLSTM-RNN and SMAPE

	PM2.5	PM10	NO2	CO	O3	SO2
DLSTM-RNN	79.29	78.07	73.48	11.60	12.25	14.00
	..	…	…	…	…	…
	67.88	71.18	66.94	7.01	9.37	10.44
SMAPE	13.61	12.25	11.83	4.70	4.52	4.46
	…	…	…	…	…	…
	12.25	11.93	11.61	4.38	4.41	4.22

(PM2.5, PM10, Sox, CO, NOx, O3) so within one hour we have (192) reading, within one day (4608) and within 30 days after the training process the network has become our (138240) Read. After the training, DLSTM-RNN can predict air pollution concentrations over the next 48 h based on previous training.

Then we used the SMAPE error rate scale to evaluate the results from the DLSTM-RNN network for the least or the nearest error.

The combination of LSTM and SWARM has reduced training time on the network because the SWARM algorithm has provided the best function for activation and has identified the number of hidden layers and the number of nodes in each hidden layer, adding that they provide better weights, but at the same time complicate the network for the above reason.

Compliance with Ethical Standards.

Conflict of Interest: The authors declare that they have no conflict of interest.

Ethical Approval: This article does not contain any studies with human participants or animals performed by any of the author.

Appendix

See Table 14.

Table 14. Terms and meaning

DLSTM	Developed long short – term memory
LSTM	Long short-term memory
PSO	Particle Swarm Optimization
SMAPE	Symmetric mean absolute percentage error
PM2.5	Particulate matter that have a diameter of less than 2.5 μm
PM10	particulate matter 10 μm or less in diameter
O3	Ozone is the unstable triatomic form of oxygen
Sox	sulfur oxides
CO	carbon monoxide
NO_x	nitrogen oxides
Θ	is the element-wise product or Hadamard product
$\otimes$	Outer products will be represented
σ	represents the sigmoid function
a_t	Input activation
i_t	Input gate
f_t	Forget gate
o_t	Output gate
$State_t$	Internal state
Out_t	Output
W	the weights of the input
U	the weights of recurrent connections

References

1. Ong, B.T., Sugiura, K., Zettsu, K.: Dynamically Pre-Trained Deep Recurrent Neural Networks using environmental monitoring Data for Predicting PM2.5. https://doi.org/10.1007/s00521-015-1955-3
2. Al-Janabi, S., Rawat, S., Patel, A., Al-Shourbaji, I.: Design and evaluation of a hybrid system for detection and prediction of faults in electrical transformers. Int. J. Electri. Power Energy Syst. **67**, 324–335 (2015). https://doi.org/10.1016/j.ijepes.2014.12.005
3. Li, X., Peng, L., Hu, Y., Shao, J., Chi, T.: Deep learning architecture for Air quality predictions. Environ. Pollut. **231**, 997–1004 (2017). https://doi.org/10.1007/s11356-016-7812-9
4. Li, X., Peng, L., Yao, X., Cui, S., Hu, Y., You, C., Chi, T.: Long short-term memory neural network for air pollutant concentration predictions: Method development and evaluation. Environ. Pollut. 997–1004 (2017). https://doi.org/10.1016/j.envpol.2017.08.114
5. Ghoneim, O.A., Manjunatha, B.R.: Forecasting of Ozone Concentration in Smart City using Deep Learning, pp. 1320–1326 (2017). https://doi.org/10.1109/ICACCI.2017.8126024
6. Lifeng, W., Li, N., Yang, Y.: Prediction of air quality indicators for the Beijing-Tianjin-Hebei region. Clean. Prod. **196**(2018), 682–687 (2018). https://doi.org/10.1016/j.jclepro.2018.06.068
7. Popoola, O.A.M., Carruthers, D., Lad, C., Bright, V.B., Mead, M.I., Stettler, M.E.J., Saffell, J.R., Jones, R.L.: Use of networks of low-cost air quality sensors to quantify air quality in urban settings. Atmos. Environ. **194**, 58–70 (2018)
8. Wen, C., Liu, S., Yao, X., Peng, L., Li, X., Hu, Y., Chi, T.: A novel spatiotemporal convolutional long short-term neural network for air pollution prediction. Sci. Total Environ. **654**, 1091–1099 (2019). https://doi.org/10.1016/j.scitotenv.2018.11.086
9. Shang, Z., Deng, T., He, J., Duan, X.: A novel model for hourly PM2.5 concentration prediction based on CART and EELM. Sci. Total Environ. **651**, 3043–3052 (2019). https://doi.org/10.1016/j.scitotenv.2018.10.193
10. Li, H., Wang, J., Li, R., Haiyan, L.: Novel analysis forecast system based on multi-objective optimization for air quality index. Clean. Prod. **208**, 1365–1383 (2019). https://doi.org/10.1016/j.jclepro.2018.10.129
11. Buyya, R., Calheiros, R.N., Astjerdi, A.V.: Big data: principles and paradigms. Big Data: Principles and Paradigms, pp. 1–468 (2016). https://doi.org/10.1016/c2015-0-04136-3
12. Oussous, A., Benjelloun, F.-Z., Lahcen, A.A., Belfkih, S.: Big data technologies. Adv. Parallel Comput. 28–55 (2019). https://doi.org/10.3233/978-1-61499-814-3-28
13. Liu, S., Wang, Y., Yang, X., Lei, B., Liu, L., Li, S.X., Ni, D., Wang, T.: Deep learning in medical ultrasound analysis: a review. Engineering (2019). https://doi.org/10.1016/j.eng.2018.11.020
14. Al-Janabi, S., Alkaim, A.F.: A nifty collaborative analysis to predicting a novel tool (DRFLLS) for missing values estimation. J. Soft Comput. (2019). https://doi.org/10.1007/s00500-019-03972-x. Springer
15. Aunan, K., Hansen, M., Liu, Z., Wang, S.: The hidden hazard of household air pollution in Rural China. Environ. Sci. Policy **93**, 27–33 (2019). https://doi.org/10.1016/j.envsci.2018.12.004
16. Inácio, F., Macharet, D., Chaimowicz, L.: PSO-based strategy for the segregation of heterogeneous ro Botic swarms. J. Comput. Sci. 86–94 (2019). https://doi.org/10.1016/j.jocs.2018.12.008

17. Matos, J., Faria, R., Nogueira, I., Loureiro, J., Ribeiro, A.: Optimization strategies for chiral separation by true moving bed chromatography using Particles Swarm Optimization (PSO) and new Parallel PSO variant. Comput. Chem. Eng. 344–356 (2019). https://doi.org/10.1016/jcompchemeng.2019.01.020
18. Hu, M., Wang, H., Wang, X., Yang, J., Wang, R.: Video facial emotion recognition based on local en Hanced motion history image and CNN-CTSLSTM networks. J. Vis. Commun. Image Represent. 176–185 (2019). https://doi.org/10.1016/j.jvcir.2018.12.039
19. Al_Janabi, S., Mahdi, M.A.: Evaluation prediction techniques to achievement an optimal biomedical analysis. Int. J. Grid Util. Comput. (2019)
20. Al-Janabi, S., Alwan, E.: Soft mathematical system to solve black box problem through development the FARB based on hyperbolic and polynomial functions. In: IEEE, 2017 10th International Conference on Developments in eSystems Engineering (DeSE), Paris, pp. 37–42 (2017). https://doi.org/10.1109/dese.2017.23
21. Al_Janabi, S., Al_Shourbaji, I., Salman, M.A.: Assessing the suitability of soft computing approaches for forest fires prediction. Appl. Comput. Inf. **14**(2), 214–224 (2018). ISSN 2210-8327, https://doi.org/10.1016/j.aci.2017.09.006