2018 IEEE 16th Int. Conf. on Dependable, Autonomic & Secure Comp., 16th Int. Conf. on Pervasive Intelligence & Comp., 4th Int. Conf. on Big Data Intelligence & Comp., and 3rd Cyber Sci. & Tech. Cong.

Air pollution forecasting using RNN with LSTM

Yi-Ting Tsai
Dept. of Computer Science and Information Engineering
National Taipei University
New Taipei, Taiwan, R.O.C
s710683131@webmail.ntpu.edu.tw

Yu-Ren,Zeng
Dept. of Computer Science and Information Engineering
National Taipei University
New Taipei, Taiwan, R.O.C
s7106831@webmail.ntpu.edu.tw

Yue-Shan Chang
Dept. of Computer Science and Information Engineering
National Taipei University
New Taipei, Taiwan, R.O.C
ysc@mail.ntpu.edu.tw

***Abstract*—With the advance of technology, it is increasingly exhaust emissions have caused air pollution. In particular, PM2.5 (Particulate Matter) has been proven that it has a great correlation with human health. Therefore, the detection and prediction of PM 2.5 air pollution is an important issue. There are countries around the world have built a variety of sensing devices for monitoring PM2.5 concentrations. There were also many studies have been constructed to predict and forecast various air pollution. Therefore, how to accurately forecast PM2.5 becomes an important issue in recent year. In this paper we propose an approach to forecast PM2.5 concentration using RNN (Recurrent Neural Network) with LSTM (Long Short-Term Memory). We exploit Keras, which is a high-level neural networks API written in Python and capable of running on top of Tensorflow, to build a neural network and run RNN with LSTM through Tensorflow. The training data used in the network is retrieved from the EPA (Environmental Protection Administration) of Taiwan from year 2012 to 2016 and is combined into 20-dimensions data; and the forecasting test data is the year 2017. We have conducted experiments to evaluate the forecasting value of PM2.5 concentration for next four hours at 66 stations around the Taiwan. The result shows that the proposed approach can effectively forecast the value of PM2.5.**

Keywords—Air pollution, Forecasting, LSTM, RNN, Air quality

I. INTRODUCTION

Air pollution is increasingly serious with industrial development. In particular the PM (Particulate Matter) has been shown that it has a great correlation with human health [10] that concluded short exposures to PM10 and PM2.5 are associated with increases in mortality. Therefore, to effectively monitor and forecast PM2.5 concentration is an important issue. Many countries have deployed a large number of sensors to monitor daily air quality [11]. In Taiwan, the Environmental Protection Administration (EPA) has deployed over 70 air quality sensors to monitor daily air quality and to offer the information to people. How to accurately forecast the air pollution is increasingly becoming an important issue.

In the recent year, there were many studies have been conducted to forecast air quality, in order to have a warning of air pollution early [3, 4, 5, 6, 7, 8, 9, 14]. Many researches have tried to use neural network to predict the possible air quality in the near future. It can be believed that the current PM2.5 concentration is strongly correlation with various data in the past period of time, these data consists of the pollutant emissions from factory chimneys, the emissions from steam locomotives, and even the impact of foreign pollution sources. It is the problem to exploit historical ground-level air quality and weather related data to accurately forecast the PM2.5.

In addition, the Long Short-Term Memory networks (LSTM) [1,2] embedded in the Recurrent Neural Network (RNN), in which the connections between units form a directed graph along a sequence. This allows it to exhibit dynamic temporal behavior for a time sequence. It is a popular model and has shown great potential in many prediction models that rely on previous data. Therefore, we hope to exploit LSTM to solve this type of problem. It takes advantage of the characteristic of making loops in the data network of pollution source, so that the information can exist for a long time without losing the past information.

In this paper we would like to propose an approach to forecast PM2.5 concentration using RNN with LSTM. We exploit Keras[1], which is a high-level neural networks API written in Python and capable of running on top of Tensorflow[2], to build a neural network and run RNN with LSTM through Tensorflow. The training data used in the network is retrieved from the EPA (Environmental Protection Administration)[3] of Taiwan from year 2012 to 2016 and is combined into 20-dimensions data; and the forecasting test data is the year 2017. We have conducted experiments to evaluate the forecasting value of PM2.5 concentration for next four hours at 66 stations around the Taiwan. The result shows that the proposed approach can effectively forecast the value of PM2.5.

The rest of the paper is organized as follows. Section II survey some related work relevant to this study. Section III presents the RNN with LSTM architecture and its internal components of LSTM, including input gate, forget gate, and output gate. Section IV presents the data preprocessing and experiments design. Section V is the experimental results of forecasting evaluation. Finally, we give a concluding remarks and future work in Section VI.

II. RELATED WORK

As mentioned above, many studies have shown that PM2.5 can cause respiratory, cardiovascular and nervous system damage [13]. PM 2.5 causes asthma, lung cancer,

[1] https://keras.io
[2] https://www.tensorflow.org
[3] https://www.epa.gov.tw/mp.asp?mp=epaen

978-1-5386-7518-2/18/$31.00 ©2018 IEEE
DOI 10.1109/DASC/PiCom/DataCom/CyberSciTec.2018.00178

cardiovascular disease, stroke, birth defects and premature death. Therefore, government of many countries has deployed a lot of air quality sensors to monitor regional pollution and to perform early warning of air quality. Many researches have tried to use neural network to predict the possible air quality in the near future. In [3], authors exploited three neural network models: Multilayer Perceptron (MLP), Radial Basis Function (RBF), Square Multilayer Perceptron (SMLP) respectively, to forecast the PM2.5. The result showed that RBF network has good prediction results for PM2.5. In the [4], authors surveyed some air quality prediction approaches and make a comparison between Recurrent Network Model (RNM), Change Point modeling Model with RNM (CPDM), Sequential Network Construction Model (SNCM), and Self Organizing Feature Maps (SOFM). It showed that the SOFM has better forecasting performance and neural networks have sufficient capabilities for air quality forecasting to handle noise and error data.

In [5], a variety of traditional machine learning and neural networks are combined to do the forecasting of air quality and showed that the use of meteorological and land characteristics can enhance the accuracy of regional PM2.5 forecasting. In the [6], authors used the ANN (artificial neural networks) and k-means clustering to identify the main sources of contamination at a single site without the need for costly chemical analysis. It has shown that the improvement of ANN input parameters through k-means clustering significant improvement in PM10 and PM2.5 prediction accuracy at high concentrations of air pollutants also has better prediction performance.

In [7], it shows that Ozone and PM10 have a great correlation with PM2.5. In some cases, training the model through seasonal data can improve the accuracy of the prediction. In [8], authors used random variables as input data for training proposed ANN models, which is a kind of fairly high prediction accuracy with less input data. In the [9], authors exploited satellite-retrieved aerosol optical depth (AOD) products to improve the forecasting accuracy for a large range area of PM2.5 forecasting. From this, we can know that the use the flow of air pollutants at high altitude in prediction accuracy has greatly improved. In this paper, we only exploit the historical ground-level air quality and weather related data to forecast the PM2.5.

III. RNN AND LSTM ARCHITECTURE

A. RNN

As well known, RNN [1] mainly deals with the processing of sequence data, such as text, speech, and climate. This type of data exists in an orderly relationship with each other; each piece of data is associated with the previous piece. For example, in a text, a word in a sentence is related to its preceding word. In the climate data, the temperature of a day is related to the temperature of previous day. Therefore, we can form many sets of sequences from the data using time from a set of continuous data, and the correlation between sequences can be observed from multiple sets of sequences. The most easily observed associations in PM2.5 data are rainy days and PM10. After the rain, pollutants in the air were washed down to the ground by rain, and the PM2.5 concentration was low at that time. When the PM10 concentration is too high, the PM2.5 concentration will also increase. We hope to use RNN to help us find that we cannot observe the correlation between time series data.

Fig. 1 shows a simple RNN architecture. Basic RNN has multiple neuron-node to form a network. Each node (neuron) has a time-varying real-valued activation. Each connection has a real-valued weight, which can be modifible in each case. In the architecture, the output of the neuron at time t-1 will be passed to the input at time t and add the data of itself at time t to generate the output at time t. Recurrently exploiting the neuron node to cascade multiple neuron elements to form a RNN.

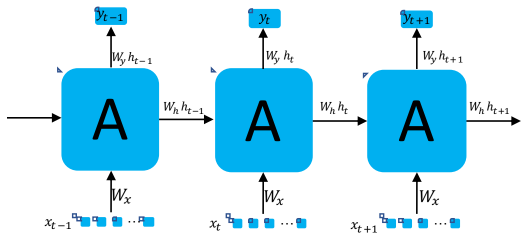


Fig. 1. RNN architecture

The recursive formulas of RNN are shown in equations (1)(2).

$$h_t = tanh(W_h h_{t-1} + W_x x_t) \quad (1)$$

$$y_t = W_y h_t \quad (2)$$

where y_t is output vector, h_t is hidden layer vector, x_t is input vector, and W_h is weighting matrix.

Although theoretical RNN can handle such long-term dependencies (Long Dependencies) problem, the longer the interval time step (the data to be referenced is at a longer time point), the W_h and W_x (weighting matrix) will continue to multiply recurrently with previous output. This will cause the vanishing gradient and exploding gradient problem. To solve this problem, we use Long Short-Term Memory (LSTM) networks to improve this problem.

B. LSTM

According to the Wikipedia's[4] definition, long short-term memory (LSTM) units, as shown in Fig. 2, are a building unit for layers of a (RNN). A RNN composed of LSTM units is often called an LSTM network. The difference between LSTM and traditional RNN neural networks is that each neuron in LSTM is a memory cell. The LSTM links the previous data information to the current neurons. Each neuron contains three gates: input gate, forget gate, and output gate. Using the internal gate, the LSTM can solve the problem of long-term dependence of the data. Next we present the internal gates of LSTM and describe how the LSTM architecture solves long-term dependency problems.

[4] https://en.wikipedia.org/wiki/Long_short-term_memory

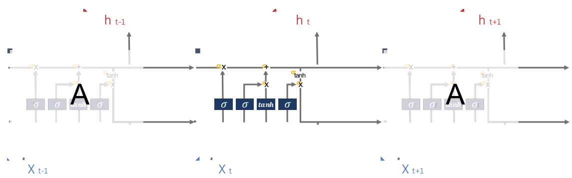


Fig. 2. LSTM architecture

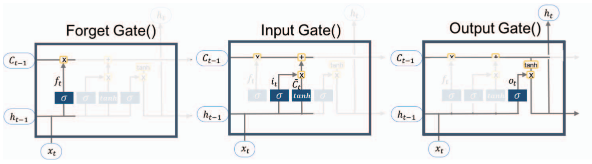


Fig. 3. LSTM's Gates

1) Forget gate

In Fig. 3, the forget gate (Equation (3)) determines which information will be discarded from the cell. By entering the output h_{t-1} at the previous unit (t-1) and adding the current time t input X_t into a $Sigmoid$ function S(t) (Equation (4)), as shown in Fig. 4, a value between 0 and 1 is generated. This vaule will multiply with cell state C_{t-1} to determine how much information will be forgotten or remembered. The 0 means completely forgotten, while 1 is completely remembered. W and b are the weight matrices and bias vector parameters respectively, which need to be learned during training.

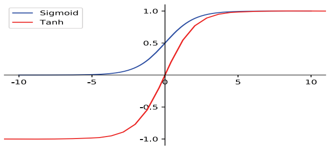


Fig. 4. $Tanh$ and $Sigmoid$ Function

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t\,] + b_f) \tag{3}$$

$$S(t) = \frac{1}{1+e^{-t}} \tag{4}$$

2) Input gate

In Fig. 4, the input gate determines which new information to remember in cell state. By entering the output h_{t-1} at the previous time t-1 and adding the current time t input X_t into a $Sigmoid$ function (Equation 4), a value i_t (Equation 5) between 0 and 1 is generated to decide how much new information cell state need to be remembered. At the same time, a $tanh$ in Fig. 3 layer obtains an election message $\tilde{C}_t$ (Equation 6) to be added to the cell state by inputting the output h_{t-1} at the previous time t-1 and adding the current time t input information X_t. Multiply the values i_t and $\tilde{C}_t$ just obtained by the Sigmoid function to get the updated information that we really want to add to the cell state C_t (Equation 7).

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t\,] + b_i) \tag{5}$$

$$\tilde{C}_t = tanh(W_C \cdot [h_{t-1}, x_t\,] + b_C) \tag{6}$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{7}$$

3) Output gate

In Fig. 4, the output gate determines which information will be output in cell state. By entering the output h_{t-1} at the previous point in time t-1 and adding the current time t input X_t into a $Sigmoid$ function, a value o_t (equation 8) between 0 and 1 is generated to determine how many cells state information that need to output. The cell state is first activated in the $tanh$ layer before being multiplied by o_t (between 1 and -1). The result of multiplication is the output information h_t (equation 9) of the LSTM block at time t.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t\,] + b_o) \tag{8}$$

$$h_t = o_t * tanh(C_t) \tag{9}$$

IV. Data Preprocessing and Experimemts Design

For the data sets used in the work are 77 air quality stations constructed by Environmental Protection Administration Executive Yuan, R.O.C (Taiwan). Some stations do not include in the work because lack of enough data. Finally, there are 66 stations that were selected as experimental data from 2012 to 2017. This paper uses 20 dimensional data retrieved from EPA of Taiwan. When there are missing values in the data, we use the average value to fill in the data. And uses Min-Max Normalization to limit values in each dimension between 0 and 1. Use the 20-dimensional data from the previous 72 hours to process the data to train the LSTM neural networks. Finally, we put the previous 72 hours of 20-dimensional data into our trained model to forecast the next hour's PM2.5 data. These data are collected, preprocessed, and stored using a big data analysis platform [14]. The data preprocessing steps are as shown in Fig. 5. We will describe the steps in following subsections.

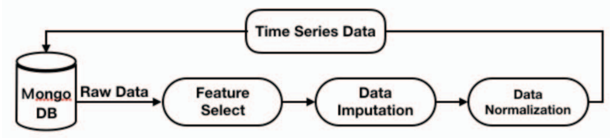


Fig. 5. Data Preprocessing

A. *Data Preprocessing*

1) Feature Selection

In this dataset, some station-level dimensions have up to 21 different types of sensing data (SO2, CO, O3, PM10, PM2.5, NOX, NO, NO2, THC, NMHC, CH4, UVB, Temperature, RAINFALL, Humidity, WIND_SPEED, WIND_DIREC, WS_HR, WD_HR, PH_RAIN, RAIN), But not all kinds of station data are related to PM2.5, so in this experiment we used 15 kinds of data. Data sensing interval is one hour. In addition, we add the two-dimensional variance data to include the hourly variation of the PM2.5 and PM10. We also add the three-dimensional data of month, weekday and hour. In the month, the monsoon causes the impact of foreign pollution sources. On weekday we believe there will be different sources of emissions from weekdays and holidays. In the hour, there may be different traffic or industrial emissions between day and night. Finally, we used a total of 20 different data types, such as Table I, as the data source for the forecasting. There will be a total of 52,608 data points in each station, and each data field will have 20 data fields.

We use the different station to cut this data set because the air pollution sources around each station are different. The station near the industrial area and the station near the commercial residential area are different in relation to time data and different gas dimensions. Therefore, different stations will have different weights for the influence of PM2.5 on each data field. We have trained the different stations separately to make it easier for the neural network to learn local pollution source emission characteristics.

Table I. Data Set

Variables (Lag=1 Hour)	**Unit**
Real time concentratio	
Pm2.5, Pm10	μg/m³
SO_2, O_3, NO_X, NO_2, NO	ppb
CO	ppm
Temperature	℃
Humidity	%
Hour mean concentration	
Wind Speed Hour	m/sec
Wind Direct Hour	degrees
Hour Accumulated concentration	
Rainfall	mm
last 10 minutes per hour	
Wind Speed	m/sec
Wind Direct	Degrees
Date Time	
Month	1 to 12
Weekday	0 to 6
Hour	0 to 23
Hours rate of change	
Pm2.5, Pm10	%

2) Data Smoothing

In the original data, due to instrument failure, the instrument, program, and manual check are invalid, resulting in the presence of missing values in the data. In this paper, we use the arithmetic average of the data fields for each dimension to replace the missing values for this dimension field.

3) Data Normalization

When the scale of each field of the data set is different, it will cause the problem that the training time is too long or the best solution cannot be found when the neural network uses the gradient descent method to find the best solution. This problem causes the neural network to iterate many times before it converges, or even fails to converge, resulting in a decrease in accuracy. So we used Min-Max Normalization (10) to limit the values of different dimensions between 0 and 1.

$$z = \frac{x - min(x)}{max(x) - min(x)} \tag{10}$$

4) Time Series Data

Since we use the LSTM neural network, we must sort the data according to the time. We arranged the 72-hour data into a 72*20 matrix and used the 73-hour PM2.5 data as our forecast target.

B. *Experimental Software platform and parameter of LSTM*

We use the first 72 hours to predict the next hour's data by using the sorted time series data. In the activation Function we used linear function, because single-point station data isn't hugely enough to use nonlinear activation functions. In the batch size we used three days (24*3 Hours) data. Optimizer function we used Adam method [12]. Loss function we used mean squared error. By monitoring the value of test data loss function, stop the training model when it is not decreasing, and save the current best model.

V. FORECASTING EVALUATION

This paper collates the information of 77 air quality monitor stations in Taiwan from 2012 to 2017, combining multiple gases and PM25 concentrations with local climate data. Through LSTM neural network training model, 2012 to 2016 as training data, and 2017 data as test data. In the Fig. 6 through experimental comparison to prove that use the LSTM neural network than the ANN neural network can have a better accuracy . In assessing the accuracy of model predictions, we used root-mean-square error (RMSE) and Mean Absolute Error (MAE). RMSE (11) and MAE (12) are commonly used as a measure of the difference between predicted and observed values. RMSE can evaluate the degree of change and accuracy of data. The smaller the RMSE value, the better the accuracy of the prediction model to describe the experimental data. MAE can better reflect the actual situation of the prediction error. The experimental results show that using the LSTM neural network has good accuracy in forecasting PM2.5.

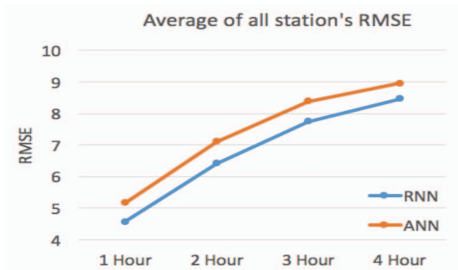


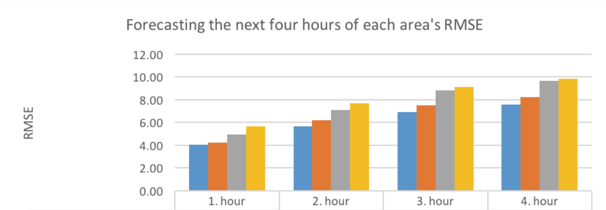
Fig. 6. Compared forecasting of PM2.5 from ANN with LSTM

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(f_i - y_i)^2} \tag{11}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|(f_i - y_i)| \tag{12}$$

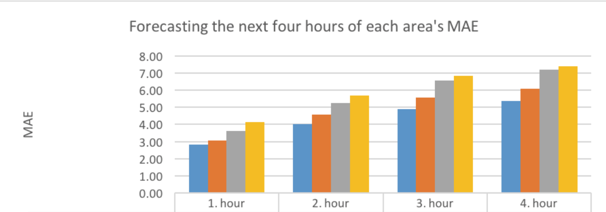
In the parameter adjustment of the neural network, we found that deepening the LSTM neural network can't improve the accuracy of the predicted PM2.5. As we increase the breadth of LSTM neural networks, there is a slight improvement in the accuracy. Using $Softmax$ Activation function will cause the Gradient descent fail to find the best solution. In this experiment, we use linear as the Activation function. In reference to the length of the past, the past 3 hours, 8 hours, 24 hours, and 72 hours were used as the data for predicting the future PM2.5 values, and the data used in the past 72 hours for prediction has the best prediction accuracy.

Fig. 7 shows the average RMSE and Fig. 8 shows the MAE of the forecasted PM2.5 values for each area. RMSE will have a larger value due to the larger change in PM2.5. Because MAE does not consider the degree of change, RMSE is relatively large compared to MAE. The training data set used by each station is the same in the data field, data acquisition time, and data preprocessing. However, the experimental results show that the average RMSE values in each region are not significantly different, but the RMSE between the individual stations is quite different. We believe that because of the lack of air pollution characteristics that affect individual areas, the average annual values of PM2.5 and PM2.5 at most RMSE stations are large. In industrial areas and traffic-intensive stations, the dramatic changes in PM2.5 will also make RMSE have larger values. We have determined that the data field we are using does not allow the LSTM neural network to accurately learn the pollution values from industrial and traffic emissions.



	1. hour	2. hour	3. hour	4. hour
Northern Taiwan	4.03	5.66	6.89	7.56
Middle Taiwan	4.23	6.20	7.51	8.23
Southern central Taiwan	4.96	7.08	8.82	9.67
Southern Taiwan	5.67	7.67	9.11	9.85

Fig. 7. Each area's RMSE



	1. hour	2. hour	3. hour	4. hour
Northern Taiwan	2.80	4.01	4.89	5.36
Middle Taiwan	3.07	4.57	5.55	6.07
Southern central Taiwan	3.62	5.25	6.57	7.19
Southern Taiwan	4.13	5.70	6.82	7.39

Fig. 8. Each area's MAE

LSTM neural network can use the data of the past period of time to accurately forecast the value of PM2.5 concentration in a short time. It can learn the PM2.5 concentration trends accurately over a long period of time. When a station is subjected to unpredictable external sources of pollution or due to the short-term changes in climate and landforms, it cannot predict accurately and often underestimated before the longer-term time. There is an accurate forecasting of the value of pollution at lower concentrations.

VI. CONCLUTION AND FUTURE WORK

Using LSTM neural network to forecast PM2.5 results is not bad when PM2.5 is not affected by climate or air pollution from abroad. The MAE can reach a minimum of 1.644 at a maximum of 5.543 within one hour of forecasting. Even in the long-term forecast, the PM2.5 value can be accurately predicted to rise. In the future, we will also increase the data on air pollutants emitted by transportation and industry so that LSTM neural network can better learn the characteristics of local air pollutants. In the prediction of PM2.5 values in the more distant future through different atmospheric altitude airflow data, the fugitive or cumulative area of air pollutants can be learned more accurately. In the prediction of more distant PM2.5 values through different atmospheric height airflow data, the flow or cumulative characteristics of air pollutants can be learned more accurately. And it can provide accurate PM2.5 prediction data for longer time.

ACKNOWLEDGEMENT

This work was partially supported by Ministry of Science and Technology of Taiwan, Republic of China under Grant No. MOST 106-3114-M-305 -001 –A and by National Taipei University under Grant No. 106-NTPU_A-H&E-143-001 and 107-NTPU_A-H&E-143-001.

REFERENCES

[1] Lin, T., Horne, B., Tino, P., & Giles, "Learning long-term dependencies in NARX recurrent neural networks," IEEE Transactions on Neural Networks, 1996,7(6), 1329–1338.

[2] Gers, F. A., Schmidhuber, J., & Cummins, "Learning to forget: continual prediction with LSTM," Neural Computation, October 2000, p.2451-2471

[3] J.B. Ordieresa, E.P. Vergara, R.S. Capuz, "Neural network prediction model for fine particulate matter (PM2.5) on the USeMexico border in El Paso (Texas) and Ciudad Juarez (Chihuahua)," Environmental Modelling & Software, 2005, p.547-559

[4] S. V. BaraiA, K. DikshitSameer, Sharma,"Neural Network Models for Air Quality Prediction: A Comparative Study," Soft Computing in Industrial Applications, 2007, p.290-305

[5] Qingping Zhou, Haiyan Jiang, Jianzhou Wang, Jianling Zhou, "A hybrid model for PM2.5 forecasting based on ensemble empirical mode decomposition and a general regression neural network," Science of the Total Environment, 2014, p.264–274

[6] M.A. Elangasinghe, N. Singhal, K.N. Dirks, "Complex time series analysis of PM10 and PM2.5 for a coastal site using artificial neural network modelling and k-means clustering," Atmospheric Environment, 2014, p.106-116

[7] Giorgio Corani, "Air quality prediction in Milan: feed-forward neural networks, pruned neural networks and lazy learning," Ecological Modelling, 2005, p.513–529

[8] Ana Russoa, Frank Raischel, Pedro G. Lind, "Air quality prediction using optimal neural networks with stochastic variables," Atmospheric Environment, 2013, p.822-830

[9] Xuefei Hu, Lance A. Waller, Alexei Lyapustin, "Estimating ground-level PM2.5 concentrations in the Southeastern United States using MAIAC AOD retrievals and a two-stage model," Remote Sensing of Environment, January 2014, P.220-232.

[10] Feng Lu, Dongqun Xu, Yibin Cheng, Shaoxia Dong, Chao Guo, Xue Jiang, Xiaoying Zheng, "Systematic review and meta-analysis of the adverse health effects of ambient PM2.5 and PM10 pollution in the

Chinese population," Environmental Research, Vol. 136, January 2015, pp. 196-204.

[11] Yung-Sheng Lin, Yu-Hsiang Chang, YUE-SHAN CHANG , "Constructing PM2.5 Map Based on Mobile PM2.5 Sensor and Cloud Platform,"2016 IEEE International Conference on Computer and Information Technology (CIT), Nadi, Fiji, 8-10 Dec. 2016, pp. 702-707.

[12] Diederik P. Kingma, Jimmy Ba, "Adam: A Method for Stochastic Optimization," 2015 International Conference for Learning Representations, May 7 - 9, 2015, San Diego, https://arxiv.org/abs/1412.6980v9.

[13] F. Dominici, R. D. Peng, M. L. Bell, L. Pham, A. McDermott, S. L. Zeger, et al. (2006). "Fine particulate air pollution and hospital admission for cardiovascular and respiratory diseases." JAMA: The Journal of the American Medical Association, 295, 1127–1134.

[14] Yue Shan Chang, Kuan-Ming Lin, Yi-Ting Tsai, Yu-Ren Zeng, Cheng-Xiang Hung, "Big data platform for air quality analysis and prediction," 2018 27th Wireless and Optical Communication Conference (WOCC), 30 April - 1 May 2018, Hualien, Taiwan, pp.111-113.