

Autonomous Online Predictive Monitoring System for Hazardous Air Pollution Prediction: A Case Study

Ayed A. Salman and Fawaz S. Al-Anzi

Department of Computer Engineering, Kuwait University, Kuwait

Article history

Received: 30-12-2015

Revised: 05-04-2016

Accepted: 31-05-2016

Corresponding Author:

Fawaz S. Al-Anzi

Department of Computer

Engineering, Kuwait

University, Kuwait

Tel: (965)24987412

Email: Fawaz.Alanzi@ku.edu.kw

Abstract: Environment is increasingly becoming an important issue in the world of politics and global economy as well as people's life. Environmental deterioration resulted from human activity; misuse of natural resources. Pollution is now a global issue that ecologically, economically and politically requires imminent, intelligent and global solutions. This paper present findings and results of a real project, supported by the Environmental Public Authority of Kuwait, intended to design, implement and test an autonomous intelligent system responsible for online monitoring and assessing hazardous air pollution in industrial areas in the state of Kuwait. The system is capable of acquiring data from sensors planted into the industrial area and communicates its captured data through wireless network to a server. The server analyzes and learns from data. The system goes further by introducing the idea of "self-controlled decision-making" process. The findings presented here show great promises for such approach and discuss several design issues in such environment.

Keywords: Air Pollution, Air Monitoring Systems, Data Mining, Intelligent Systems, WEKA, Environmental Prediction System

Introduction

Environment is increasingly becoming an important issue in the world of politics and global economy as well as people's life. Environmental deterioration resulted from human activity; misuse of natural resources and pollution is now a global issue that ecologically, economically and politically requires global solution. Environmental problems occur due to the interaction between two complex systems, human-society system and ecological system. Today, the most notable environmental problems in the world include global warming leading to climate change, land degradation, water pollution contributing to human health problems, deforestation resulting desertification, destruction of species, ozone depletion, increasing urban and industrial wastes, etc. Globally, pollution of freshwater resources, deforestation and improper waste dumping, among other abuses, are human conducts bankrupting natural resources of future generations (Dubey, 2014). Human activity and life is changing the environment in ways, on scale, quite unlike in any other era, making our common future in jeopardy.

Large percentage of people's illness in third world countries is directly linked to the pollution of their

natural environment (Stephens, 1995; Adeola, 1996; Sebastian, 1990; Azid *et al.*, 2014). Improved environment resulting improved public health is therefore a clear element in the struggle and the strategy of enhancing human quality health. In general terms, population growth, economic development and growing inequality in income all put greater pressure on the ecosystems. Moreover, poverty and political conflict, which are the features of most developing countries, also cause environmental damage. Environmental degradation increases the poverty of those who are already poor especially in those parts of the world where livelihoods and lives are closely dependent on natural environment.

Kuwait is by no means an exception in the above situation (Al-Awadhi, 2014). There are substantial challenges of environmental concerns in the country due to human life activities. The second gulf war and the crimes committed by previous Iraqi regime against Kuwait and its people were of all kinds that included environmental crimes. In fact one of the largest environmental catastrophes made by humans was done here in Kuwait when the Iraqi regime plowed up oil wells and spelled others over land and sea (Literathy, 1993; Lange *et al.*, 2002). This act affected not only

Kuwait but also the global ecosystem. Actually consequences of this act are still unfolding. Added to this is the pollution contributed by urban and industrial activities such as vehicle trafficking and industrial plants among many (Al-Awadhi, 2014), which are expanding as Kuwait population and its demands increases. As mentioned before the impacts inflicted on the ecosystem by human's activities are hard to track let alone identify their consequences short and long. This is due to the complexity of both human and ecological system as well as their interaction.

Air pollution is one of the major environmental problems existing in Kuwait. Environmental Public Authority (EPA) in Kuwait is a Governmental body responsible monitoring and assessing environmental situations within the state of Kuwait. EPA constantly support project to enhance such monitoring process. In this study, supported by EPA, we showed clear evidences that we were able to monitor the quality of air and identify when it may become hazardous to human life by building an autonomous system that will constantly monitor air quality intelligently. The system is a combination of technological instrumentation with algorithmic solutions from the Artificial Intelligence community. The basic idea behind such system is that whenever a gas concentration rises above a certain threshold (acceptable norms) the system will issue an alert in different means to the decision makers. For the system to work correctly it adapt to the constantly changing conditions, due to human or natural causes. This adaptation is achieved using an algorithmic technique called "data mining" that reach that with minimum human intervention. Data mining techniques and algorithm are widely used in many application related to predicting air pollution concentrations (Bell *et al.*, 2015; Ming *et al.*, 2009; Pandey *et al.*, 2013; Hájek and Olej, 2013).

This process is based on measurements generated by a mobile meteorological station located nearby an environmentally hazardous region with. The system acquires data from sensors attached to the station and communicates its captured data through wireless network to a server. The server analyzes and learns from data. The system goes further by introducing the idea of "self-controlled decision-making" process. The idea relies on utilizing collected historical data to construct prediction models. The strength of such system then comes from using these models to forecast the expected values of the monitored gases for the next 1, 12 and 24 h period. This is done using data mining algorithms such as Decision trees (Quinlan, 1986) or Neural Networks (Hagan *et al.*, 1996). As the collected data increases, the online system, through the data mining process, learns how to enhance its prediction models so that forecasting

becomes more accurate. Moreover, this system is able of extracting rules; which can be used at any other site to process data offline. The system goes better and better as time passes in predicting such problems and taking the right decisions correctly.

The system is able of producing prediction models as well as values for the monitored gases (CO, CO₂, NO₂, SO₂ and H₂S) and is able to adapt to different minutes-by-minutes dynamically changing environmental factors. The following sections describe all phases the system went through and concludes with important observations one should care of when deciding to adopt such system for large scale environment.

Intelligent Monitoring System: The Process

The Developed system has two major stages, as shown in Fig. 1, to accomplish its task:

- Data acquisition
- Data processing and mining

In first stage, the system acquires the data (raw data) from a number of sensors. Each of these sensors is responsible for reading the levels of a certain gas. These gases are CH₄, CO, CO₂, NO, NO₂, SO₂ and H₂S. The system reads raw data from a text file generated by the monitoring system. This file consists of lines of captured data that register gases levels along with other factors (pressure, wind speed...etc) every 5 min. Raw data is then stored in the database DB (Fig. 1). At the same time the average readings for 1, 12 and 24 h periods are calculated using raw data then stored in DB.

This processed data (i.e., averaged) is then divided, in the second stage by the data mining engine, into training and testing data to enhance over the prediction models that predict the values for the gases mentioned above over the next 1, 12 and 24 h periods. These predicted values are then again stored in DB and reported to the user along with the real data (Fig. 1).

For this effort, we adopted an objected oriented programming strategy (Snyder, 1986) with layered approach for the sake of high portability and adaptively. In another word, the system is divided into objects that encapsulate certain functionality/s and in turn these objects are distributed between layers. From programming point of view, this approach will ensure maximum robustness, scalability and system maintenance. The following section will describe the system and its major functionalities.

System Layers

The system uses three logically well-separated layers: Presentation, Business and data layers. A separate component was added to handle offline prediction by anywhere, anytime.

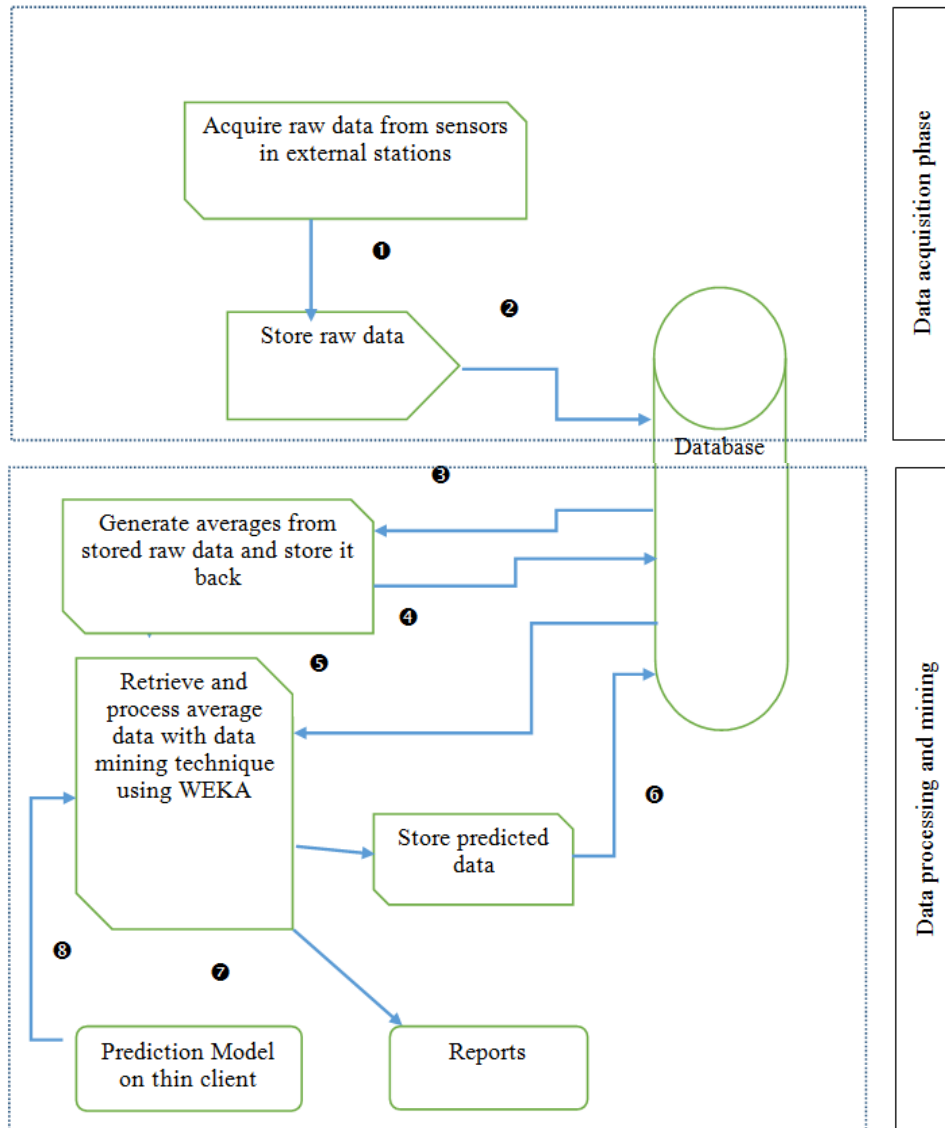


Fig. 1. System main stages

The major function of the presentation layer is to present a communication platform with user/s, in terms of providing a human interface through which the user can:

- Supply input to the system
- Monitor system's status and performance
- View system's output and reports

In presentation layer the system administrator can specify the text file that contains raw data (captured by monitoring system). The system then will try to read that file, either partially or as new file.

On the other hand, the main operations of the business layer include:

- Calculate averages of raw data (real data) for 1, 12 and 24 h periods
- Process real data (as training/testing data) using WEKA tool (Witten and Frank, 1999) to generate and enhance prediction models

When the system starts for the first time it expects two things from administrator:

Raw data file. If this is missing the system will go to sleep for 5 min and recheck again; it keeps doing so until it is forcibly stopped.

A date-time value that marks the Capturing Starting Date-Time (CSDT) of raw data file. If the user did not supply this value the system will consider the operating system's current date-time as CSDT.

This is necessary for the system to produce accurate prediction with accordance to the live situation of the monitoring station. (e.g., prediction for Saturday of the week is based on data collected with monitoring station on and up to Friday of the previous week). Knowing CSDT makes predicted values to be generated over the correct time periods. Since the monitoring system (sensors) produces one line (new reading for gases levels and weather conditions) of raw data every 5 min, knowing the CSDT will enable the system to store each raw data with its corresponding capturing timestamp as follows:

- For first data line in raw data file (Engine Time = CSDT)
- For the rest of data lines (Engine Time = Engine Time +5 Min)

Raw data then averaged over three time periods 1, 12 and 24 h. This is done as follows:

- For 1 h period the system calculates the average of raw data for the past 1 h from current engine time. In another words:
 - Retrieve all raw data from DB whose timestamp fall within the period (= Engine Time -1 h)
 - Calculate the average for each gas using retrieved raw data
- Do the same thing for 12 and 24 h periods
- The averaged data are then stored in DB again for later use by data mining engine to calculate prediction

Prediction Model

Data mining process uses an algorithm to do prediction that adopts a multi predictor voting system. These predictors are estimated from the artificial intelligence community research areas, namely:

- Linear regression (Weisberg, 2005)
- Pace regression (Wang and Witten, 1999)
- Multi-Layer perceptron (Ruck, 1990)

Both linear regression and Pace regression tries to estimate mathematical model parameters that mimic the behavior of the complex environmental system monitored. Multi-layer perceptron adopt a technology from machine learning community called neural network and feed forward learning algorithm. All algorithms utilize historical data to formulate a model (linear or non-linear equation) about such environmental system. The prediction capability of such systems enhances as more data are feed in. Building on such capabilities, designing the process as

a multi-predictor voting system, enhances the system multi-fold. Voting system do enhance over multiple decisions by different subsystems.

Offline Client

Thin client is a piece of software that can work independently of the server. To be able to do prediction onsite the team developed a thin client software that can take the model/s file generated by the system and uses it directly on the produced raw data. This will eventually help in decentralizing prediction process and leaving the system with a primary target of enhancing produced prediction models through continuous training and testing.

The system can be reused by the offline client, after producing a prediction model, to process collected data onsite and without being online. All what is required is the model and data file to run. This represents agents which are self-contained with respect to their processing capability. It is worth mentioning that the system will produce a prediction for each gas separately.

Design Issues

During development stage many issues were faced that were not accounted for during the design stage. These include:

- Handling missing data and dealing with negative values at the data preparation and filtering stage
- Preserving the system's state when intentional shutdown occurs and recovering from a crash when unintentional shutdown occurs
- Solving concurrency problem when trying to read raw data file while uploading new data from monitoring station
- Multi norm handling

The following subsection presents our approach to solve such issues.

Missing Data Handling

The first encountered issue was handling missing values in raw data file. Namely speaking there are two kinds of missing data:

- Missing fields within a raw data line
- Complete missed line/s

While reading raw data file, if the system finds that some of values are missed (incomplete raw data line) it will predict the missed value/s. If the whole line is missed the system will replace the missed line with the last accepted line. This action is done as long as the number of missed records do not exceed max Number of

Missed Record variable. Currently if the system finds that missed records exceeding 3 consecutive records the system will ignore (it will not be stored in the database) the current missed record, increment its engine time by 5 min and logs this incident as well as issuing an alert message to the user. We took this decision to account for two type of incidents: If the monitoring station was taken a sudden disturbance for any reasons, we wait for it to adjust back to normal not more than 15 min, during such we try to approximate missed readings as previously read value before disruption without much loss of generality. In case, disruption (occurring for any reasons), go beyond the 15 min period, then we account for it as missed value and keep reading position vacant until a manual interference by the monitoring station administrator is taken place.

On the other hand, some times and due an error in sensors negative values appear in raw data. The system treats negative value/s like missing ones. In addition the system will log this line in text file.

Shutdown Problem

In real-time situation, a system shutdown (either intentional or unintentional) is a reality. The project team treated the shutdown problem very efficiently. As mentioned previously elsewhere in the study, the system keeps reading from raw data file specified by the user until the end of the file. When last record from the current file is read, the system will:

- Try to acquire an updated version the file or if the user supplied a new file name the system will try to acquire it
- If the system fails to acquire an updated/new raw data file the system will go into sleep mode for 5 min and will issue an alert for contacts list about the unavailability of new raw data. The system will repeat step 2 until new raw data is available

If an intentional shutdown occurs then it is only a matter of restating the starting timestamp and the system will continue working as normal, while in case of unintentional shutdown, then the system will recover even easier since it will continue reading from the last record it stop short reading.

Concurrency Problem

Read/write problem to the same file is a well-known concurrency problem (Lampert, 1977) that should be taken extreme care about. In system such as ours, server can try to read from file that is written in the same time by monitoring station sensory machines. When monitoring system produces raw data it registers the data in a text file. The developing team is designing uploading module to upload the file into the system's

server (into upload folder). The system will then move the raw data file to its working directory for reading operations which leaves the upload folder free to take a new raw data file at any time from the uploading software. In this way we alleviate the problem of concurrent read/write operations to the same raw data file.

Dealing with Multi-Norms

Finally, the system may be deployed in different areas that have different gas level thresholds (e.g., residential area and industrial area). Because of this the system is designed to deal with different sets of norms and standards. Right now the system can work with residential or industrial norms agreed standards by EPA. Changing norms will only configure the system to issue alerts at different levels for monitored gases.

Testing

After finishing the development and to verify the correctness of the development we tested the system. The deployment of the system was fairly simple. The first thing we needed to do was to insure the periodical update for the server. For this, we designed a program that copies the data file generated by the sensors system to a directory from which we periodically send any changes to the server using ftp protocol. When we put this program into production, we find that our system suffers from some problems that could affect its prediction accuracy.

The main and major obstacle had to do with how the sensors system constructs its data file and how captured data are written to it. When the mobile station sensory system starts it produces an initial file for the current month and it fills it with many lines of consecutive -0.999E04 repeatedly. Each line of those eventually will be replaced by captured data. The number of lines for a monitored month is given as follows:

$$\frac{60 \text{ min} \times 24 \text{ hr} \times \text{number days in a month}}{5 \text{ min}}$$

Each 5 min the system replaces an existing line with captured data. If for any reason the sensors system did not capture any data the corresponding line of initial consecutive -0.999E04 stays as is and is not replaced by actual data. On top of that, the server reads each line in the received data file and increments its internal clock by 5 min. The server keeps reading the file received till completion, each line containing the initial false data of consecutive -0.999E04 is normally considered a missed record during reading. Reading data files by the server and capturing data by sensors are normally two different processes in terms of timing. Server at certain timing can read data file while sensors are capturing new data file. Some of the files read by the server many not yet

completely filled by the sensors' reading in the mobile station. So the data file maybe sent including the remaining $-0.9999E+04$ lines at the end which are not yet replaced. The problem is that these lines at the end of the file are not missed records, instead, they are not-yet-replaced lines and should be logically replaced in the future when their exact time arrived. This will cause the server to increment its internal clock to the end of the month and this will cause the server to stop prediction until the end of the month. To alleviate this problem we devised a simple program that works as follows: Before sending the data file the program will truncate all consecutive lines that contains $-0.9999E+04$ starting at the end of the file and until it faces the first non $-0.9999E+04$ line from the end of the file. This will leads to eliminating false missed-data look-alike lines.

After solving this first obstacle we performed life testing with different server's file-update periods of 5, 15 and 30 min. We found out in practice, 5 min was too short, at such period, approximately we were sending the same data file about 3 times before any new data is written to the data file. A 30 min update period meant that we only have one chance to send the data needed to predict the next hour. A more balanced period should be chosen in between of those two periods, taking into consideration that we are relying on EDGE technology for data communication between monitoring site and server location and from our observation its performance fluctuates due to reasons that ranges from traffic over network to weather conditions. For all the above, 15 min was the suitable updating period for two reasons:

- It allows 3 possible sending attempts to send to the server the necessary data within the current hour to make prediction for the next hour (minimal prediction period)
- Within 15 min the file updating is likely to happen so the server will always be supplied with new data

After tackling these two obstacles the server was ready to accept and process data in a smooth timely manner. The server behaved as expected, the data received was used in enhancing the prediction quality for different gases for 1, 12 and 24 h periods. Figure 2 shows the prediction for NO_2 . from the graph you can see that as long as there is any gradual change in gas concentration level the prediction is fairly reliable, actually you can see in Fig. 2a (before 6000) that the predicted and actual (average) graphs coincides. Figure 2b shows that error percentage in the range 0-6000 is below 10% (per in mind that the server was able to achieve such a prediction accuracy within a 15 days of operation). After 6000 the spikes becomes more frequent and from this point onward the error percentage increases dramatically.

The spikes shown in Fig. 2 are also observed for other monitored gases (CO_2 , H_2S , etc) and this reflects itself in error increase. The spikes result from a sudden change (increase or decrease) in gas concentration for a short period of time. This kind of sudden change prevents the server's prediction engine from devising an accurate prediction model, in other words, the server in this case is receiving erroneous training examples. Despite such obstacles, the server can recover from such a situation elegantly: The server is designed to enhance itself by recurrent training; as soon as the spikes disappear the server starts to devise more and more accurate prediction models. All what the server needs are spike free data and time to correct its prediction results.

Observations

The testing stage exposed the server to real environment and this gave us a good idea about what may be faced and what sorts of modification is needed when lunching such system live. There are two crucial points needed to be address in order for the server to work correctly in live environment:

- Receiving data in a timely manner by the server
- Ensuring the quality of received data

To insure a timely data receiving process we used EDGE technology which is based on GSM wireless network. Our testing phase shows that such way of communication is not as reliable as expected; its performance is highly susceptible to weather condition and its upload rate decreases a lot as traffic over network increases. In a production environment this could lead to delivery delays which could render prediction useless as the server will start to produce results for past hours. In some occasions the network fails even to establish a connection which would prevent delivery. For future real-time deployment, we recommend the usage of more reliable but costly communication links like microwave connection for example.

As shown in Fig. 2 the server's prediction was effected by data spikes. We investigated the reason behind their occurrence and found out that in all the cases of spike occurrence the sensor system was behaving erroneously. This was due either to a system calibration or due to failure in one of the sensors. The problem is that at all the time the sensor system was writing to the data file and under these conditions this data was erroneous or at best not reliable. The sensor system should have a mechanism to alert and preferably stops data producing as long as it is in an unreliable state. The sensor we used in our testing stage lacks such mechanisms. The project team recommends relying on much reliable sensory systems (which was beyond the scope of such study) in future studies.

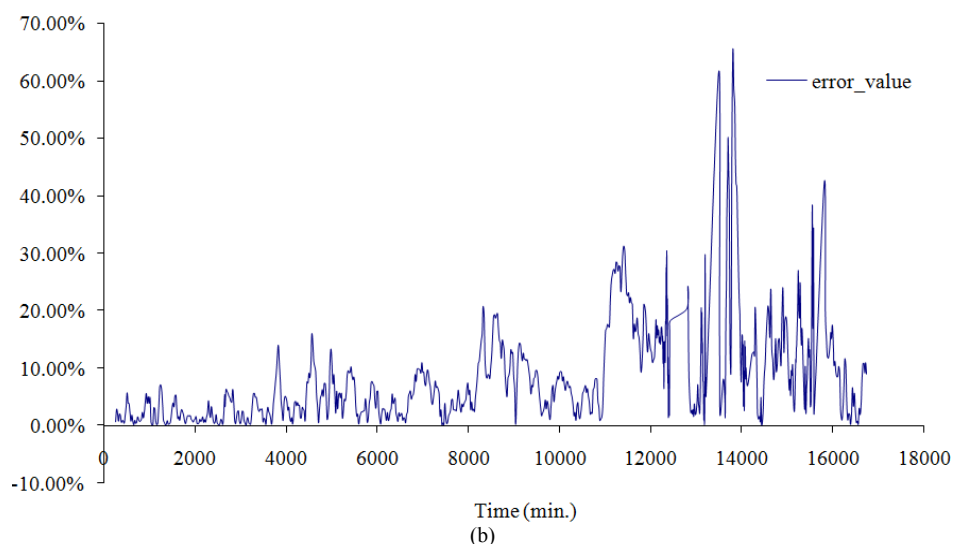
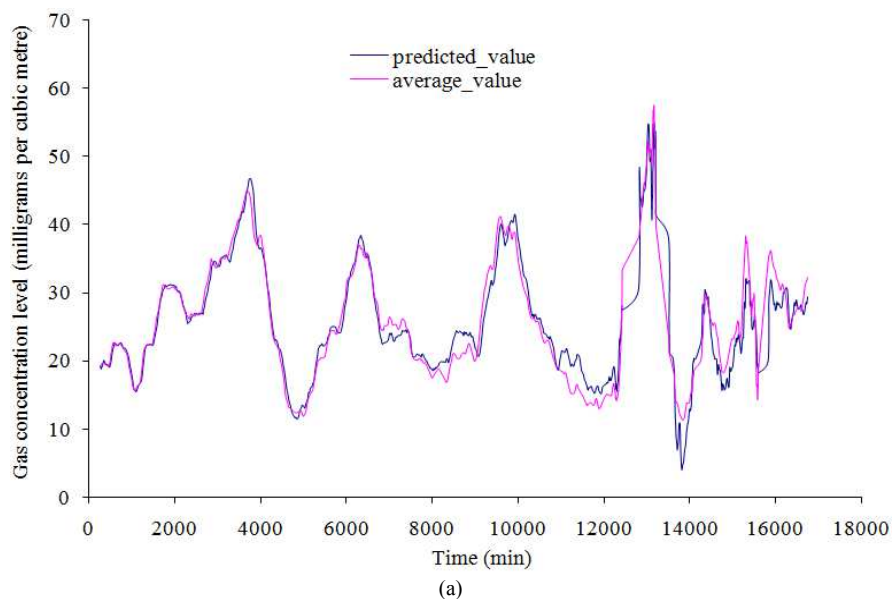


Fig. 2. (a) NO₂ predicted vs. real (average) value over 24 h. Notice how the value increase between the two graphs as spikes appears
 (b) The error percentage for NO₂ prediction for 24 h period

Conclusion and Recommendation

The system presented in this study is able to:

- Tackles the problem of real-time monitoring for environmental hazardous gasses
- Present a very accurate prediction model for different periods in the future such as 1, 12 and 24 h timeline starting from any moment in the present time
- Presented its intelligence in an offline agent format that is able to work and predict accurately without the need to referring back to the server. This component is essential for manual and standalone

sites that does not, or prevented from communication for any reason with the server

- Overcomes many practical issues discusses in previous section
- Have an impressive and highly accurate performance during testing stage; as long as the system receives correct data (from the sensors) in a timely manner the system is able of performing an accurate prediction

Reliable communication should be established in order for the system to produce results in proper time. More over the sensor's system should contain some self-diagnosing mechanisms which can be interfaced

programmatically. In this way our system can get an early warning about the quality of captured data being received. This would prevent the system from processing faulty data which would affect its prediction accuracy. Of course the effect of faulty data will be offset over time. In other words, over the course of time the system will receive new data which is used to retrain the system and due to this the system will produce new prediction models where faulty data effect is marginalized. Another area for enhancements is incorporating methods that enhance data filtering where bogus data can be removed.

In summary the project presents the following recommendations:

- The project's served as proof of concept. It shows that incorporating artificial intelligence in examining complex systems is helpful in understanding them. Specifically speaking, artificial intelligence ability in discovering patterns and models coupled with the great computational power of computers presents a great tool (which if used correctly) can help human experts in understanding complex systems and uncover existing subtle patterns that can explain some unusual behavior
- The system is an effective tool with practical usefulness. We recommend adopting the system on large scale for all industrial or residential areas
- Project results' usefulness can be extended to different sectors such as oil industry
- System bottlenecks represented by monitoring sensory readings and communication equipments. These two problems can be solved easily with enough budgets to purchase newly accurate sensory machines and high-end communication equipments that assure reliability and availability

Acknowledgment

The author's thanks Sameera Al-Mathkoori, Shaker Madi from the EPA for their valuable support and help in facilitating this project. The authors would like also to thank Ali Alameen and Tariq Hamadah for their help in coding the system.

Funding Information

The authors would like to thank the Environmental Public Authority (EPA) of Kuwait for the financial support for such project.

Author's Contributions

Ayed A. Salman: Supervised, designed, checked and verified the code. He supervised the data acquisition from environmental sensors. He analyzed the data and chose the data mining algorithms and tuned them. He

helped in drafting the results and conclusions. He wrote the first draft of the paper.

Fawaz Al-Anzi: Reviewed the results, suggested part of the reliability mitigation solutions for different design issues occurred while operating the proposed system. He participate in drafting the conclusion. He exercised English proofreading onto the manuscript.

Ethics

This article is original and never published before. Part of the results shown here was presented in a panel discussion in a GET-16 conference, Kuwait, without being published in a proceedings.

References

- Adeola, F.O., 1996. Environmental contamination, public hygiene and human health concerns in the third world: The case of Nigerian environmentalism. *Environ. Behav.*, 28: 614-646.
DOI: 10.1177/001391659602800503
- Al-Awadhi, J.M., 2014. Measurement of air pollution in Kuwait City using passive samplers. *Atmospheric Climate Sci.*, 4: 253-271.
DOI: 10.4236/acs.2014.42028
- Azid, A., H. Juahir, M.E. Toriman, M.K.A. Kamarudin and A.S.M. Saudi *et al.*, 2014. 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: 2063-2063.
DOI: 10.1007/s11270-014-2063-1
- Bell, M., A.S. Bergantino, M. Catalano and F. Galatioto, 2015. Prediction of air pollution peaks generated by urban transport networks. Working papers SIET.
- Ming C., Yafeng Y., Min X., 2009. Prediction of hourly air pollutant concentrations near urban arterials using artificial neural network approach. *Transport. Res.*, 14: 32-41. DOI: 10.1016/j.trd.2008.10.004
- Dubey, A., 2014. Air pollution modeling for human exposure predictions: A review. *Res. J. Recent Sci.*, 3: 24-27.
- Hagan, M., H. Demuth and M. Beale, 1996. *Neural Network Design*. 1st Edn., PWS Publishing Company, Boston.
- Hájek, P. and V. Olej, 2013. Prediction of air quality indices by neural networks and fuzzy inference systems-the case of Pardubice microregion. *Eng. Applic. Neural Netw.*, 383: 302-312.
DOI: 10.1007/978-3-642-41013-0_31
- Lampert, L., 1977. Concurrent reading and writing. *Commun. ACM*, 20: 806-811.
DOI: 10.1145/359863.359878

- Lange, J.L., D.A. Schwartz, B.N. Doebbeling, J.M. Heller and P.S. Thorne, 2002. Exposures to the Kuwait oil fires and their association with asthma and bronchitis among gulf war veterans. *Environ. Health Perspectives*, 110: 1141-1146. PMID: 12417486
- Literathy, P., 1993. Considerations for the assesment of environmental consequences of the 1991 Gulf War. *Marine Pollut. Bull.*, 27: 349-356. DOI: 10.1016/0025-326X(93)90042-I
- Pandey, G., B. Zhang and L. Jian, 2013. Predicting submicron air pollution indicators: A machine learning approach. *Environ. Sci. Process Impacts*, 15: 996-1005. DOI: 10.1039/c3em30890a
- Quinlan, J.R., 1986. Induction of decision trees. *Mach. Learn.*, 1: 81-106. DOI: 10.1007/BF00116251
- Ruck, D.W., S.K. Rogers, M. Kabrisky, M.E. Oxley and B.W. Suter, 1990. The multilayer perceptron as an approximation to a Bayes optimal discriminant function. *IEEE Trans. Neural Netw.*, 1: 296-298. DOI: 10.1109/72.80266
- Sebastian, I., 1990. Issues in urban air pollution: Ankara diagnostic report. World Bank, Environment Working Paper, Washington.
- Snyder, A., 1986. Encapsulation and inheritance in object-oriented programming languages. *Proceedings of the Conference Proceedings on Object-Oriented Programming Systems, Languages and Applications*, (SLA' 86), ACM., New York, pp: 38-45. DOI: 10.1145/28697.28702
- Stephens, C., 1995. The urban environment, poverty and health in developing countries. *Health Policy Plan*, 10: 109-121. DOI: 10.1093/heapol/10.2.109
- Wang, Y. and I.H. Witten, 1999. Pace regression. (Working paper 99/12). University of Waikato, Department of Computer Science, Hamilton, New Zealand.
- Weisberg, S., 2005. *Applied linear regression*. Wiley Interscience.
- Witten, I.H. and E. Frank, 1999. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. 1st Edn., Morgan Kaufmann, San Francisco, ISBN-10: 1558605525, pp: 371.