IoT based Air Quality Monitoring System with Power Consumption Optimization and Air Quality Parameters Prediction using Deep Learning

A SYNOPSIS

Submitted in partial fulfillment of the
requirements for the award of the degree
of

DOCTOR OF PHILOSOPHY

in

COMPUTER SCIENCE & ENGINEERING

By

BAROT VIRENDRAKUMAR KANTILAL

Under Guidance of

DR. VIRAL V. KAPADIA

सत्यं शिवं सुन्दरम्

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
FACULTY OF TECHNOLOGY & ENGINEERING
THE MAHARAJA SAYAJIRAO UNIVERSITY OF BARODA
VADODARA-390002 (INDIA)

MARCH 2021

ABSTRACT

With the fast developing economy, industrial park construction and production processes, there is increase in the probable issues related to environmental pollution, especially air pollution accidents. Air pollution may be explained as the contamination of the atmosphere by gaseous, liquid, or solid wastes or by-products that can danger human health and welfare of plants and animals or produce undesirable odors. The air pollution monitoring has been emerged as a critical issue in developing countries such as India. Recently, in Delhi, AQI has crossed the bar of 1000, which may cause enormous breathing problems, cancer-like diseases and chronic respiratory conditions to the citizens of the country. Therefore, availing air pollutant concentration information in real time (monitoring) to citizens with handy tools like web and mobile interface help citizens to avoid any health hazards. Along with monitoring air pollutant concentration forecasting is an effective method of protecting public health by providing an early warning against harmful air pollutants. IoT has the potential to monitor air pollution by providing real-time updates related to sudden changes in the air quality. The main motive of this work is to develop complete air quality monitoring system for real time reporting of air pollutants using IoT, along with efficient method for prediction of such air pollutants from the historical data.

Many efforts have been made to monitor air quality using IoT based technologies, still, the air quality monitoring system using IoT is an open research area because of the challenges of IoT discipline such as complex architecture, no standardization, less memory, power consumption, interfacing of sensors, reliable delivery and security. The proposed research work makes the use of light-weight protocol and compatible devices to transmit air quality parameters to the remote server without building or setting up complex network. There are various kinds of "Things" used in the proposed monitoring system that includes NodeMCU(controller), HiveMQ(cloud broker), Sensors, Python Script(Paho MQTT subscriber) and Android. These things coordinate with each other for the purpose of air quality parameter (temperature, humidity, Carbon Monoxide, PM 2.5 and PM 10) collection, transmission, storage and retrieval individually towards the implementation of the complete air quality monitoring system. The proposed research work represents the implementation of power consumption reduction scheme during sensing (reading) phase. Also we have proposed event based transmission method to reduce number of transmissions (messages) which further reduce power consumption. Proposed system is implemented and tested with a variety of Quality of Service levels to confirm the reliability of the system under employed architecture. A customized web interface and mobile application are designed to represent updates of pollutants at different indoor and outdoor sites. The system also avails data logging in the data base for further analytics and prediction purpose. In this work, deep learning based framework is also proposed to predict air quality parameters such as particulate matters (PM 2.5 and PM 10) and carbon monoxide(CO). Long short term memory(LSTM) neural network based model that processes sequences in both forward and backward direction to consider influence of time steps(observations) in both directions is employed. For further learning and to improve the prediction performance, the stacking of unidirectional layers is implemented. The performance of the model is optimized by fine-tuning of various hyperparameters like epochs, regularization techniques for overfitting resolution, and various merging options for the bidirectional input layer. The proposed model achieves good optimization and performs better than a simple LSTM and RNN based model. Moreover, attention-based mechanism is adopted to focus on timesteps that are more significant for prediction purpose. The addition of self-attention mechanism improves the performance further and works well for longer sequences and extended time horizons also. Experiments are conducted using the recently collected real-world data and results are evaluated using mean square error(MSE) loss function metric.

TABLE OF CONTENTS

INTRODUCTION

This chapter briefly introduces about the air pollutions and hazardous effect of it. It also tells the motive behind this work, the problem statement and what are the research contributions of this work.

1.1 Air Pollution and Hazardous Effects

In the era of a new industrial revolution widely known as Industry 4.0, the internet of things, big data, and artificial intelligence have empowered a variety of industrial sectors and started focusing heavily on areas such as Smart Agriculture, Smart Transportation, Smart Environmental Monitoring, Smart Manufacturing and many more. Such industrial sectors have revamped the practices of tradition industries by facilitating 24 * 7 interconnectivity, automation, real-time data analytics, and IoT based predictive maintenance. However, along with the latest advancements and innovations in the new phase of the industrial revolution, air pollution has emerged as one of the major concerns of the 21st century. With the fast-developing economy, an increase in industrial park construction, and production processes, air quality monitoring has become an alarming issue for all the industrial sectors and society. For this purpose, a rigorous analysis of existing systems was undertaken to identify and mitigate research gaps. Existing research works did not focus on power consumption optimization and also neglects to assess the performance of the system at a variety of quality of service levels. In this study, a complete air quality framework has been proposed to monitor and reduce the impacts of air pollution including power consumption reduction, and have also done reliability assessment at the variety of service levels in indoor and outdoor environments. In general, air pollution can be described as the contamination of the surrounding atmosphere by various gaseous, liquid and solid wastages which can cause enormous effects on human health, the welfare of plants and animals across the globe [1]. These air pollutants have the capability to create innumerable health risks in the form of various health diseases such as sinusitis, asthma, allergic, organic dust toxic syndrome, nasal irritation, central nervous system symptoms, deice and many more. According to the reports of [2], household air pollution and outdoor air pollution had resulted in 4.3 million and 3.7 million pollution-related deaths. It can be observed that the majority of indoor and outdoor pollution-related deaths occurred due to heart, lung and respiratory diseases [9,11]. The WHO report [2] also suggests that near about 91% of the world’s population lives in places where air quality levels have exceeded WHO prescribed limits. The air pollution index(AQI) monitoring has emerged as a critical issue in developing countries such as India. Recently, in Delhi, AQI has crossed the bar of 1000, which may cause enormous breathing problems, cancer-like diseases and chronic respiratory conditions to the citizens of the country. WHO report [2,5] also indicates that more deaths have happened due to pollution-related issues rather than road accidents in the last decade.

Such air pollution-related issues have arisen due to rapid urbanization, enormous population growth and the shortfall in the availability of the required infrastructure, and a lack of pre-defined health framework for air quality monitoring [3,4]. According to [5], a significant concern for air pollution is the presence of a high concentration of particulate matter in the surrounding atmosphere. Furthermore, incomplete coal combustion and vehicle emission can cause detrimental effects on human health because they release fine particulate matter (PM2.5) directly to the atmosphere. PM2.5 can cause harmful biological effects to various human organs such as lungs, a nervous system [6,10], and the outermost layer of skin [7]. Carbon monoxide (CO) present in the atmosphere is responsible for an enormous proportion of the poisonings and life losses, reported all around the globe [12]. In various cases, Carbon monoxide level increases to a level that causes coma and even deaths sometimes [13].

1.2 Motivation for This Work

Air pollution has been a critical concern of the 21st century. IoT technologies have the potential to monitor air pollution by providing real-time updates related to sudden changes in air quality [8]. Many efforts have been made to monitor air quality using IoT based technologies, still, the air quality monitoring system using IoT is an open research area because of the challenges of IoT discipline such as complex architecture, no standardization, less memory, power consumption, interfacing of sensors, reliable delivery and security.

- Confirming to the smart city context air quality monitoring and real time reporting with handy tools like mobile application can help in avoiding health hazards and providing ease of life in urban area.
- The historically data(logged) can assist in identifying characteristics, seasonal patterns, sources of pollution, distribution of pollutants at various region (intra and intercity) of air pollutants.
- Forecasting helps user in providing early warning about the pollutants. The development of accurate air quality (air pollution) parameter forecasting model or system in urban areas can help people in avoiding health hazards by the decision making of cancelling to be outdoor or to visit certain critical places.
- Also, the forecasting results inform government agencies to execute traffic control or any such policy implementation looking at the critical levels of air pollution in the future.

1.3 Problem Statement, Objectives, Research Contributions

1.3.1 Problem Statement

- To build IOT based system that collects air quality parameters from any remote resource constrained site, process and transmits the observation in real time to the server by addressing various issues of IOT architecture, sensor interfacing, reliable delivery and power consumption.
- To process, store and avail visualization of the observed parameters and do the prediction using deep learning from the available historical observation data.

1.3.2 Objectives

- To identify critical air quality parameters and prepare IoT based smart node that comprises of parameter specific sensors, interfaced to the processing board by keeping in consideration of the economical aspect.
- To design or develop some strategy for power consumption optimization for increasing the battery lasting period of smart node.
- To standardize the IoT layered architecture and propose the complete framework with for monitoring (processing, transmission, logging and providing GUI) of identified air quality parameters from the resource constrained remote site.
- To test the system performance over parameters like throughput and reliability.
- To design methodology or model based on deep learning for forecasting/predicting air quality parameters from the logged historical parameters.
- To transform raw data collected in to the form suitable for such prediction model and test the system performance for prediction accuracy. Also design strategy to improve performance further.

1.3.3 Research Contributions

1. IoT based systems suffers from challenges like no standardization and complex architecture design. In proposed work we standardize the layered architecture provide the framework starting from collection (at deployment sites) to logging and representation of the air quality parameters on different platforms such as web and mobile interface at the same time by providing semantic interoperability and data harmonization between various objects. The underlying architecture is benefited with following unique features.

- No relaying nodes (i.e in case of zigbee based implementation) or network setup required.
- System is implemented with broker based architecture and cloud broker HiveMQ is utilized for MQTT communication protocol implementation rather any local broker. So, system can be scaled easily.
- System implements authentication of publisher or subscriber node at broker during registration phase by use of unique ClientID.
- System implements integrity check while reading parameters from sensors to avoid any physical level sensing error.

2. Power consumption is one of the major challenges for the IoT based monitoring system because of the battery operated nodes deployment at the resource constrained sites. In proposed research work we provide power consumption optimization scheme. The sensing operation of the smart node is divided in five phases. The scheme utilizes sleep mode of the controller and particulate matter sensor with the help of customized hardware design for power saving during sampling period. The proposed scheme is able to reduce the power consumption almost 54% per sampling period and battery lasting period is increased 1.8 times as exhibited in experiment.
3. Average power usage of smart node relies on number of measurements taken periodically and the number of transmission of measured data. Proposed system also experimented with novel event based transmission to reduce overall transmission and ultimately reduction in power consumption. The aperiodic transmission scheme, updates receiver (transmits data) only if significant change in current (recent) measurement is observed compared to the last N measurement. Under the employed scheme with the set threshold criterion the transmissions are reduced from 240 to 64 (on an average) over 6 hour's transmission. Also the scheme takes care of the no reporting for longer period to balance the tradeoff between number of transmission and tolerance of parameter changes for reporting.
4. IoT based ecosystem such as air quality monitoring systems are transporting real-time updates to the remote server. In such real time system reliable delivery of the messages to the subscribers is very important. Reliable delivery or accuracy is one of the metrics representing Quality of Service provided by system. Thus implementation of QoS adds value to such diverse system by providing performance, visibility and usability of the services offered. Many few efforts have been attempted to implement and assess performance of implemented system under complex architecture design. We have implemented and tested the system with two QoS levels scheme. QoS level 0 is fire and forgets kind of communication where the publisher sends the message and

never checks for the delivery of the message at the destination. QoS level 1 guarantees MQTT packet delivery at least once. Even though subscriber temporary unavailable due to low bandwidth or other issues, it will get the published messages from broker once the subscriber is UP(available) again. Under employed architecture with QoS level 0 also we are able to achieve 98% accuracy. QoS level 1 increase the accuracy performance at the cost of increased end to end delay.

5. RNN is widely used approach for time series prediction due to the backpropogation with time algorithm usage for learning. We perform the timestep based moving window training and predicted the future timestep value three parameters CO, PM 2.5 and PM 10 using tensorflow backend and keras. RNN suffers from vanishing gradient problem as the sequence sample size increases which also can be observed in the performance evaluation.

6. To overcome the problem and further minimize the loss function (increase performance) for longer sequences, we use simple LSTM and stacking with forward LSTM +backward LSTM based model. The model is checked with four merge modes and the prediction performance is further improved with the applied model. The proposed approach outperforms both RNN and Simple LSTM based prediction.

7. LSTM based cascaded model improves the performance but found to be suffering from over fitting. To overcome the issue, we experimented the model with two regularization techniques and found good performance with dropout based technique. The model is also fine-tuned by proper hyperparameter setting to improve the performance. We applied dropout layers in the model between hidden layers. Adding dropout layers make the model dynamic by changing the number of LSTM cells while backpropogation is applied for particular batch. We tested the model with varying dropout values and found good convergence for validation data at 0.3 dropout value. The final loss for validation also found to be further minimized and stabilized at around 300 epochs.

8. Attention-based mechanism is adopted to focus on timesteps that are more significant for prediction purpose. The addition of self-attention mechanism improves the performance further and works well for longer sequences and extended time horizons also. The applied self-attention mechanism is also one of the first attempts to the best of our knowledge, in the field of sequence to sequence air quality prediction.

2. LITERATURE STUDY

To address the issue of air pollution, the internet of things has emerged as one of the critical technologies in the development of smart cities. In developing countries such as India, the government has also utilized the Internet of Things framework to provide various public sector services to the citizens. IoT has the capability to facilitate interrelated machines/devices embedded with a variety of sensors to send or receive data with each other using device identification without active human interaction. Advancements in the area of IoT have changed the lives of human beings by bringing smart concepts [14-17] such as smart healthcare, smart environment, smart city and many more.

2.1. Air quality monitoring system reviewed

Many researchers are actively involved in the environmental monitoring field due to their concern to the critical problem. Li et al. have investigated a method of analysing hourly monitoring data produced by 71 fixed monitoring stations distributed over Taiwan using neural networks [18]. The conducted study lacked an air quality framework for air pollution data collection and monitoring. Reisinger et al. have described a differential optical absorption spectroscopy (DOAS)-based instrument for measuring air pollutants [19]. The sensors utilized in this approach have a very long optical path around 100 m to 20 km at an interval of 20 minutes. Richards et al. have designed a high-throughput sensor network to address the issue of real-time urban air pollution monitoring using GUSTO sensor technology [20]. Their system had measured a variety of pollutants accurately in the form of ppb (part per billion Levels) at short intervals of 2s. The proposed approach was an enhancement of the DOAS based project of Reisinger et al. team. DOAS and GUSTO based projects have used a wired network of optical sensors. The complete setup of the proposed systems was based on wired infrastructure due to the unavailability of technologies such as wireless sensor networks and the internet of things.

With the advancements in the field of wireless sensor networks, sensing technology and internet of things, fellow researchers have attempted many researches to implement air quality monitoring systems to measure various air quality parameters. Al-Haija et al. and Kularatna et al. have presented the microcontroller-based system using general-purpose gas sensors [21, 22]. The experiments were conducted at the home premises in Vadodara India. However, the system did not facilitate to report the monitoring results in near real-time. Al-ali et al. and Devarakonda et al. have implemented and tested the distributed pollution monitoring system using the GPRS public network [23, 24]. The pollution data from various mobile sensor arrays were transmitted to a central local server. SocialCops is a data intelligence company located in India which has developed a project to measure air pollution in Delhi. This project module comprises of sensors, GPS shield and GPRS shield. In this approach, data is transmitted with GPRS network [25]. As per the recent trends, GPRS technology is considered outdated technology due to the latest advancements. Abraham et al., 2014 have developed a system that measures indoor levels of CO, Ozone and CO2 using the ZigBee mesh network [26]. However, they did not conduct any experiments in outdoor environments. Kumar et al., Ferdoush et al., Baccol et al. and Morawska et al. have reported a monitoring system of air pollution parameters [27-30]. In all the proposed approaches, Zigbee modules have been implemented to establish communication. ZigBee modules possess a limitation of distance in the case of relaying packets. The implementation of ZigBee technology requires a higher number of ZigBee nodes to relay the data over a longer distance. The proposed solution is also a costly alternative. Sharma et al. have developed a gas sensor based embedded system. In this approach, the CNT gas sensor was developed and the MSP430 controller was used to detect ammonia but they did not deploy any data transmission methods [31]. Tiwari et al. have developed a system for monitoring methane, temperature and humidity using Raspberry Pi, which communicates the received data to the local web server [32]. This approach was implemented at Bits Pilani research lab in India. Marques et al. have developed an air quality monitoring system for ambient assisted living. The system was designed to monitor LPG propane gas using MQ6 sensor connected to the local system (laptop) for data transmission [33]. Hong-di et al. have presented a methodology to predict PM2.5 with the available data from the surrounding monitoring stations [34]. The proposed approach did not consider direct measurements of air quality parameters. A wireless mobile air pollution monitoring application was designed by Dhingra et al. and Huang et al. [35-37]. They used cloud-based services in acquiring the data in a cost-effective manner using low-cost sensors. However, the system was not tested in indoor and outdoor environments to check its reliability and accuracy. In this approach, generic air quality sensors were used instead of specialized sensors, which can directly measure pollutant parameters such as PM2.5 and PM10 and provide more insights about AQI. In developing countries such as India, government is running the National Air Quality Monitoring Program [38]. Program is handling a network of 779 fixed stations. The government of India is funding projects related to air quality pollution looking at the alarming situations in metro cities. However, the program has failed to reach to urban and suburban areas. So there is an immediate need to establishing a complete air quality framework to monitor air pollution

2.2. Air quality parameters prediction system reviewed

Numerous air quality prediction methodologies have been presented by researchers, which can be classified into categories like statistical methods, machine learning based approaches and recently, the deep learning based approaches. Statistics based approach comprises principal component analysis(PCA), Coefficient analysis, linear as well non-linear regression based model [39,44,41] and interpolation based [40] implementations. These methods suffer due to lack of ability to model non-linear as well as multivariate types of data. Machine learning based approach comprises fuzzy methods [45, 46], genetic algorithm [42] and support vector [43, 46] based implementations.

Recent development in deep learning based forecasting approaches have shown very good prediction accuracy, outperforming the statistics and machine learning based methods in various domains. Deep learning methods include recurrent neural network (RNN) and Long Short Term Memory (LSTM) based neural network model [47,48]. LSTM network performed better compared to RNN due to the gated cell based mechanisms in LSTM unit [48,49]. LSTM based fully connected network was utilized to predict particulate matter concentration at targeted stations [50,51] by the researchers. Authors [52] predicted air pollution parameters with a combination of CNN and LSTM based networks to further improve the efficiency of a simple LSTM based network model. Thanongsak xayasouk and team [53] predicted air quality using deep auto encoder and LSTM based model. Authors fine-tuned the performance by hyperparameter setting and compared the performance using two approaches.PM2.5 predictions by combining convolution networks and bidirectional GRU (gated recurrent unit) network over particulate matter data of Beijing city carried out to enhance the performance of simple GRU [54]. Authors [55] used bi-directional LSTM network for air pollution severity category prediction. The bi-directional approach learns features in both forward and backward directions to improve the LSTM performance further. Weitian Tong and his team [56] used bidirectional LSTM for spatiotemporal interpolation of particulate matter PM2.5 concentrations. The model focused both spatial and temporal factors. So LSTM based approach is one of the popular choice in air quality parameter prediction. Some attempts also made to further improve the performance of LSTM based model as discussed.

LSTM based bi-directional deep learning network model that takes the advantage of both forward and backward direction time step observation in learning during training is applied in the proposed work. The observed data are modeled into sequences and a sliding window based approach for transformation of training data in to supervised data is used. The model can predict the next time step value for air quality parameters from the given test sequence. In the field of air quality prediction, one such approach of using bidirectional LSTM [55] is available but the work is done for label classification where label is various severity category rather than actual value prediction. The performance of such bi-directional LSTM is heavily influenced by the way forward and backward layer output are merged. In proposed work, the model is tested with various merging functions(options) to optimize the performance. Moreover, the model is using further stacking above the input bidirectional learning to improve the performance of air quality parameter prediction. To our best knowledge, such approach is novel in the field of sequence to sequence (moving window) based prediction for air quality or air pollution timeseries (data) parameters.

3. PROPOSED WORK

3.1 Implementation of IoT based air quality monitoring system

3.1.1 Details of a controller and sensing units

ESP8266 12E/NodeMCU controller

ESP8266 12E/NodeMCU is an open-source cost-effective IoT development platform. This module is equipped with a 32-bit microcontroller with 4MB memory for storing sensing programs. The "NodeMCU" requires 3.3V power supply. There are plenty of IoT development boards/processing units are available, selection of processing unit affected by inherent constraint of application domain and budget. Cost is very important aspect considering vast deployment of sensor monitors needed in air quality monitoring system. Arduino and ESP8266 12E/NodeMCU are most economic options. However, both processing units contain microcontroller and comparatively less processing memory than other alternatives. In our application sensing nodes do not involve any edge computing (analytics at data origin) task which demand for more processing power. The task of sensor node is to read sensor data at employed frequency and transmit over IEEE 802.11.

SDS021 particulate matter sensor

The module gives output using the UART protocol with a bit rate of 9600 bits/second. Detection range is 0.0-999.9 µg /m^3. The sensor works at 5V operating voltage.

ZE07-CO carbon monoxide(CO) sensor

ZE07-CO module depends on an electrochemical principle to measure the Carbon Monoxide. The sensor works at 5~12 V with the 9600-bit rate for UART output which measures in the range 0-500 ppm.

DHT22 temperature and humidity sensor

DHT22 is a low-cost digital temperature and humidity sensor. The sensor gives measured value as an output (in the form of a calibrated digital signal on the data pin).

3.1.2 Layered architecture of IoT based monitoring system

The layered architecture of system has been classified into five layers. Figure 1 shows the topology design of the proposed system, and figure 2 represents the detailed circuit design of the interfacing of sensors with a controller ESP8266 12E.

- **Physical sensing layer** - The physical sensing layer consists of a variety of modules such as SDS021 Particulate Matter Sensor, ZE07-CO carbon monoxide sensor and DTH22. These sensor modules are capable of measuring air quality parameters such as PM10, PM2.5 CO, temperature, and humidity. These sensor modules are connected to a microcontroller ESP8266 12E. The microcontroller is equipped with IEEE 802.11 and TCP/IP stack. The data available from sensor in UART data frame is converted to text value.

- **Communication and Networking layer -** The Wi-Fi access point is essential for the transmission of sensor data using Message Queue Telemetry Transport (application layer protocol) over internet (with underlying network protocol-TCP). The measured parameters' values are converted to MQTT messages.

- **Cloud service layer -** Controller, along with the equipped sensors, forms a smart node that can be identified as mqtt publisher. Publisher (smart node) publishes the air quality data from an individual sensing site to MQTT broker with topics specific to air pollutant parameters. This layer is also responsible device authentication based on unique clientID provided by node during initialization process utilizing the ClientID field of MQTT message field. Data available from mqtt messages are stored in topic value hierarchy at broker.

- **Data processing layer -** This layer is responsible for fetching data from cloud broker, process raw data of messages and store data in format compatible with application layer requirement and suited for further analysis. MQTT subscriber is deployed on a server that is responsible for fetching observed data of a remote site available at broker. Data received from broker is handed over for rendering to graphical user interface and stored in a database for analytics.

- **Application layer -** The application layer provides real-time air quality monitoring updates via web and mobile GUI interfaces.

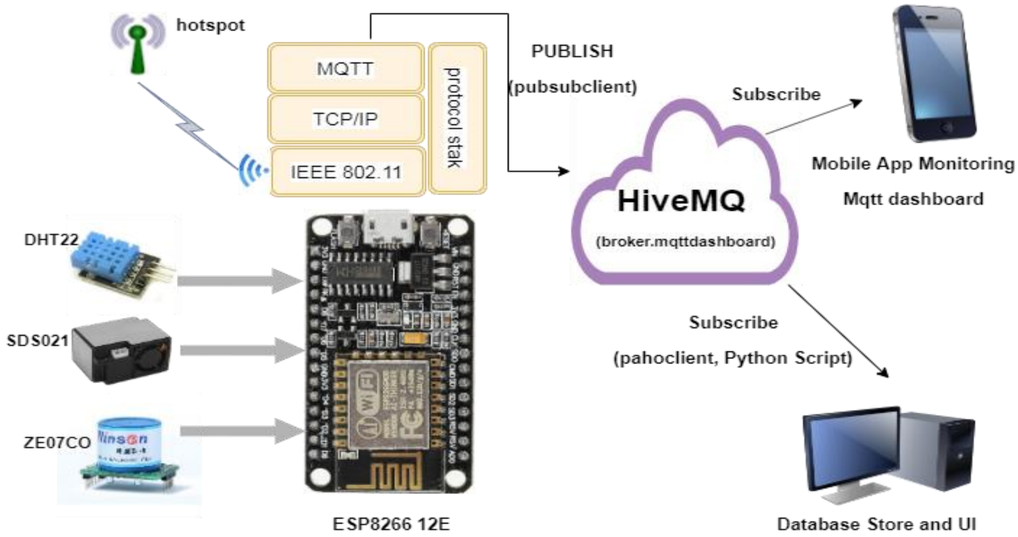


Fig. 1. Topology design

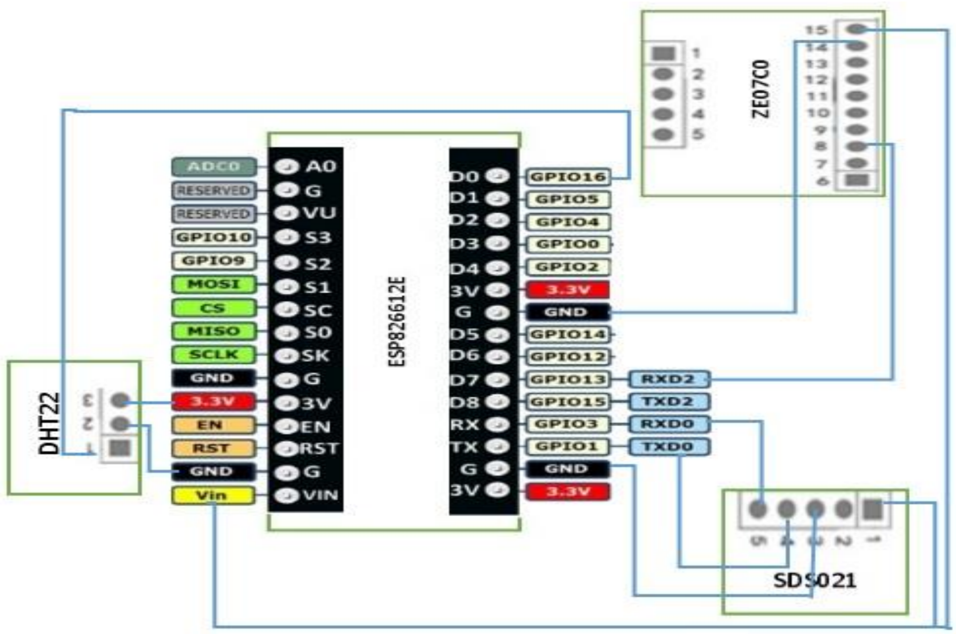


Fig. 2. Detailed circuit design of sensor interfacing

3.1.3. Detailed working flow of IoT based air quality monitoring system

Figure 3 shows detailed working flow of the system. Publish-subscribe pattern requires a message broker in between a publisher and a subscriber in MQTT protocol. HIVEMQ cloud broker is used as a public cloud broker. MQTT publisher is implemented on NodeMCU, which sends sensor data in the form of MQTT messages to an HIVEMQ cloud broker. Data acquisition at a server (MQTT subscriber) has been implemented using python language packages “paho-mqtt” and “paho.mqtt.client”. Moreover, the sensing layer is programmed using the Arduino IDE, which is responsible for reading data from the sensors using API calls designed for air quality monitoring system.

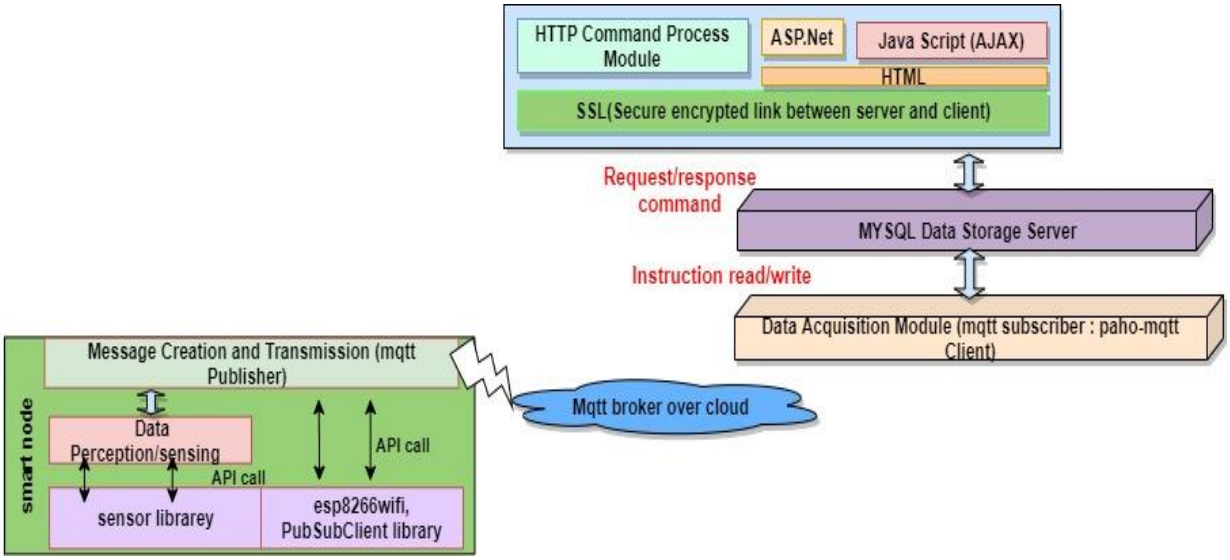


Fig. 3. Detailed workflow of IoT based air quality monitoring system

3.1.4. Power consumption optimization scheme implementation

Power usage has been optimized by switching to a sleep mode at appropriate intervals instead of hardware optimization. In sleep mode, device draws very less power compared to an active mode. ESP8266 12E controller facilitates three types of sleep modes: modem, light and deep-sleep. In a light-sleep mode, the controller draws around 0.9 mA compared to 120 mA in an active mode. So, to achieve power consumption optimization, controller has been switched to a light sleep mode as and when it is not acquiring data from the sensors. While the controller is reading data from the sensors, the RF transceiver is kept on an idle mode to reduce the power consumption. As shown in Figure 2, particulate matter sensor module SDS021 and CO sensor module ZE07C0 will be connected to [vin] of a controller. [Vin] is the direct output from the supply source, which provides power back up to an Esp826612E controller. Whenever a controller is in a light sleep mode, [vin] supply will not remain idle and it will supply the necessary power to connected sensors. In general, it is essential to keep the CO sensor in the active mode most of the time because it requires a 3-minute warm-up time whenever it will be switched on and off. So it is not advisable to switch-off the CO sensor during the communication phase. On the contrary, it is possible to power down particulate matter sensor module SDS021 by switching it to a sleep mode.

The sensing operation of the system can be divided in five phases as shown in Table 1. Initially, system will remain in the P1 phase where controller will switch into a light sleep mode for the duration of

60 seconds. At this stage, SDS021 sensor module will remain in a hibernation mode and DHT22 sensor module will remain in a sleep mode. After the expiration of a timer, a controller will switch into an active mode from a light sleep mode. Whenever a proposed system will enter into the p2 phase, DHT22 sensor module will wait for 10 seconds for stabilization and recording of temperature and humidity data. In the case of p3 phase, a controller will activate the SDS021 sensor module. It will record particulate matter data for the duration of 10 seconds and switch into a sleep mode. After recording CO data, the proposed system will enter into the P5 phase from the p4 phase. Eventually, a publisher turns on transceiver and transmits the gathered data using MQTT messages.

Phases	**controller**	**DHT22**	**SDS021**	**ZE07CO**	**Wi-Fi**
P1	light sleep	Power down	sleep	active	OFF
P2	active	active	sleep	Active	OFF
P3	active	active	active	Active	OFF
P4	active	active	sleep	Active	OFF
P5	active	active	sleep	Active	ON

Table 1. Sensing cycle phases for various components of sensing unit during parameter reading

3.1.5. Event based transmission towards power usage reduction

Average power usage of smart node relies on number of measurements taken periodically and the number of transmission of measured data. As shown in table 1 the transceiver is kept off and only activated when transmission occurs. Power consumption can still be reduced if number of transmissions are reduced, in addition to the power optimization scheme implemented. The aperiodic transmission scheme, updates receiver only if significant change in curren measurement is observed compared to the last or previous measurement. The well-known techniques include send on delta (SoD)[57], send on area(SoA)[58] and Send on prediction(SoP)[59]. SoD technique updates the measurement when difference between current measurement and previous one reaches a set threshold. In SoA the condition is to transmit when integral of the difference between recent and previous measurement over the interval reaches the set threshold. SoP calculates prediction value based on few previous terms (measured values) and difference between prediction and current value is utilized for transmission decision.

$$|\,\overline{X_t} - \overline{X_{t-1}}\,| > \delta\,\overline{X_{t-1}} \qquad (1)$$

Where, $\overline{X_t} = \frac{\sum_{k=t-(N-1)}^{k=t} X_k}{N}$ and $\overline{X_{t-1}} = \frac{\sum_{k=t-(N-1)}^{k=t-1} X_k}{N-1}$

Proposed system also experimented with event based transmission to reduce overall transmission and ultimately reduction in power consumption. Event based transmission scheme uses condition given in equation 1 for transmission decision. If the condition is satisfied transmission takes place otherwise not. Here $\overline{X_t}$ is the average of last N measurement including the latest measurement at time t and $\overline{X_{t-1}}$ is the average of previous N-1 measurement till time step t-1(excluding the recent measurement). If the change in the average due to recent measurement is more than δ percent than change is considered significant and transmission takes place. If any of the CO, PM 2.5 and PM 10 follows the given condition than the MQTT message is transmitted otherwise message transmission is skipped. It is possible that gradual and steady increase in measurement, never (or for longer period of time) makes the condition in equation 1 to be true. In that case the measurement not reported to receiver for very longer period of time. To avoid such scenario, we added one more condition, that is continuous nine transmission skipping will trigger one transmission on tenth measurement.

3.1.6. Quality of Service

IoT based ecosystem such as air quality monitoring systems are transporting real-time parameters and updates to the remote server. Such system can also provide threshold based notifications based on received data. In such real time system reliable delivery of the messages to the subscribers is very important parameter. Reliable delivery or accuracy is one of the metrics representing Quality of Service provided by system. Thus implementation of QoS adds value to such diverse system by providing performance, visibility and usability of the services offered. Many few efforts have been attempted to implement and assess performance of implemented system under complex architecture design. There are three levels of QoS in MQTT. Level 0 is least reliable communication out of the three levels. It is fire and forgets kind of communication where the publisher sends the message and never checks for the delivery of the message at the destination. In QoS level 1 publisher keeps the message until the acknowledgment using the PUBACK message from the broker. The publisher publishes the message again if PUBACK is not received from a subscriber. QoS level 2 ensure exactly one-time message delivery at the subscriber. At this point, a sender and a receiver will use various message identifiers for the synchronization of delivery. A publisher sends the message again with a duplicate flag if PUBREC is not acknowledged.

3.2 Implementation of air quality parameter prediction using deep learning

3.2.1 Proposed LSTM based Neural Network (bidirectional and stacking) Model

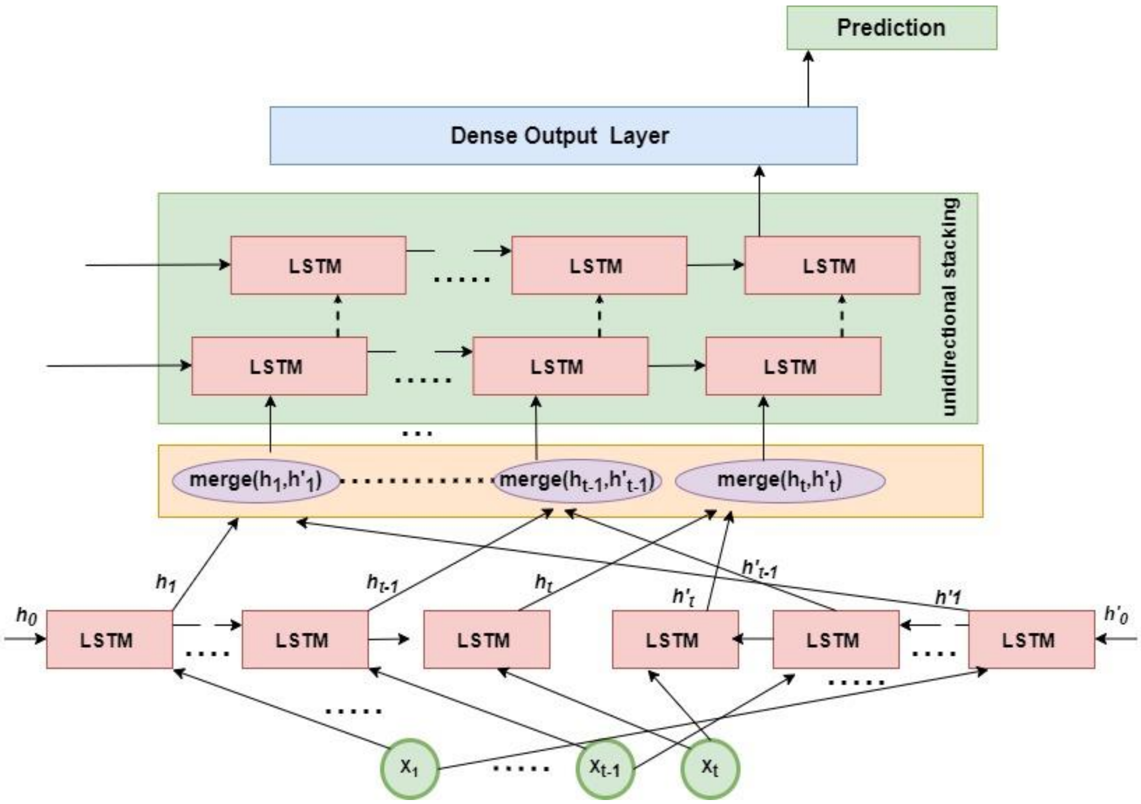


Fig. 4. Proposed LSTM based model

Authors[60] proposed a bidirectional recurrent neural network first time, which works on sequences in two directions forward as well backward. Forward pass and backward pass handled by separate RNN layer. Bidirectional Long Short Term Memory network was proposed in the year 2005[61] in the field of signal processing. It connects two separate layers (forward and backward processing) to the same output(merge) layer. The approach proved to perform well than unidirectional [61,62]. In our proposed model for training, two sequences are processed as shown in figure 4. The forward pass layer outputs sequence ***h*** iteratively which is calculated using inputs in forwarding direction *T-1* to *T+1*. The backward pass layer outputs vector ***h'*** iteratively which is calculated using inputs in reverse direction*T+1* to *T-1*. Both the forward and backward layer outputs are calculated by equations used for standard LSTM unit. The input vector to next layer is generated using merge function. The Merge function uses output of respective time step from LSTM cell in forward and backward layer as shown in figure 4.

Various researches [63-65] have presented that deep neural network building by progressively stacking recurrent layers on top of each other works more effectively. The approach lets the hidden state at every level to work at different timescale and every layer provides its own abstraction level. The proposed model implements deep LSTM architectures where several stacked unidirectional LSTM layers are added on top of each other. Output of one such layer is given as input to the next layer. The bottom layer in the proposed training model utilizes forward and backward pass for learning the features in given sequences. The top layers of the model utilize the learned features from the bottom layer for further learning. The number of layers to be stacked which gives optimum result is one of the hyperparameter and decided through experiments. The proposed training model with forward and backward layer as first layer with further unidirectional stacking as shown in figure 4 will be referred as the FBLSTM the model now onwards in the paper.

3.2.2 Data Preparation

For the purpose of developing a model for prediction of time series data of pollutants collected, we need to transform the time series data into suitable data structure for supervised learning. The approach [66] used is relying on a prediction methodology that models many to many mapping of inputs to outputs by keeping the stochastic interdependencies of time series events. It consists of predicting the next k values as per the below equation 2.

$$P(x_t, \ldots\ldots x_{t-l+1}) = (x_{t+k}, \ldots., x_{t+1}) \tag{2}$$

Where l indicates the number of past observations utilized for prediction of k next events and $t \in \{l, \ldots. n - k\}$. The right-hand side of the equation is indicating observation included in the target or output window and k is representing the size of the output window. The left-hand side of the equation is representing observations included in input window with size l. Moving window or sliding window method employed by partitioning the time series observations of pollutants of length n in to sample sequences of length *[input window (l) + output window (k)]*. So overall there are total *[n-(input window size +output window size) +1]* such samples in sample space Output window is kept of size one, so only the next time step is predicted in experiments and input window size l is kept to be 60.

3.2.3 Data preprocessing and metrics

During data pre-processing normalization of time series data is performed using linear scaling given with equation 3. The linear scaling method transforms the observations into a new interval defined by lower bound *(lb)* and upper bound *(ub)*. The new transformed pollutant time series data lies in the range [0,1].

$$X_{i\_new} = lb + \frac{X_i - MIN(X)}{MAX(X) - MIN(X)} * (ub - lb) \tag{3}$$

Transformed training data is modelled into supervised learning data as discussed in the data preparation subsection. There are various evaluation metrics MAPE, MSE and RMSE available that can be used in performance evaluation of the prediction model. Mean Square Error (MSE) is used as an evaluation metric (MSE loss function in Keras) in the experiments conducted that can be defined as per equation 4, where X_{pred_i} is the predicted value and X_{actual_i} is the actual value of the i^{th} air quality parameter observation.

$$MSE = \frac{1}{n}\sum_{n=1}^{n}(\mathrm{X}_{\mathrm{pred\_i}} - \mathrm{X}_{\mathrm{actual\_i}})^2 \tag{4}$$

3.2.4 Attention mechanism

Attention mechanism in neural network lets the training model to give more importance to the features that are having more influence on output. Self attention is applied on the input timestep vector of the RNN layer to focus more on important timestep values [67,68]. Self attention mechanism assigns some weight to the each input sample according to its importance or influence on output. Self attention is proven to improve the performance of the neural network [67-69]. Self attention is applied in the model shown in figure 2 and experiments are carried out to check the effect on learning with different dimensions, discussed later in the result section. Output vector V ={V1, V2, V3, …, Vt } of the last unidirectional hidden layer in LSTM stacking is given as input to the attention mechanism or attention layer. Self attention mechanism assigns the weight αi to each of the Vi based on the importance to output as per equations 5 and 6.

$$\alpha_i = \frac{\exp(e_i)}{\sum_{i=1}^{t}\exp(e_i)} \tag{5}$$

$$e_i = fun(W_{i,}V_i\,) \tag{6}$$

Here W_i is the weight applied during training for each timestep V_i in LSTM based learning through back propagation with time and *fun* is the function applied for calculation of e_i which is tanh function in experiments. Softmax in Keras is used for self attention which calculates the normalized weight α_i as per the equation 5. Sofmax function takes care that addition of all the weights(α_i) is one. The final Context vector output from the self attention layer is $V\,‘=\{V’_1,\ V’_2,\ V’_3,\ \ldots,\ V’_t\,\}$ *where* $V’_i$ can be obtained as per equation 7.

$$V'_i = \alpha_i * V_i \tag{7}$$

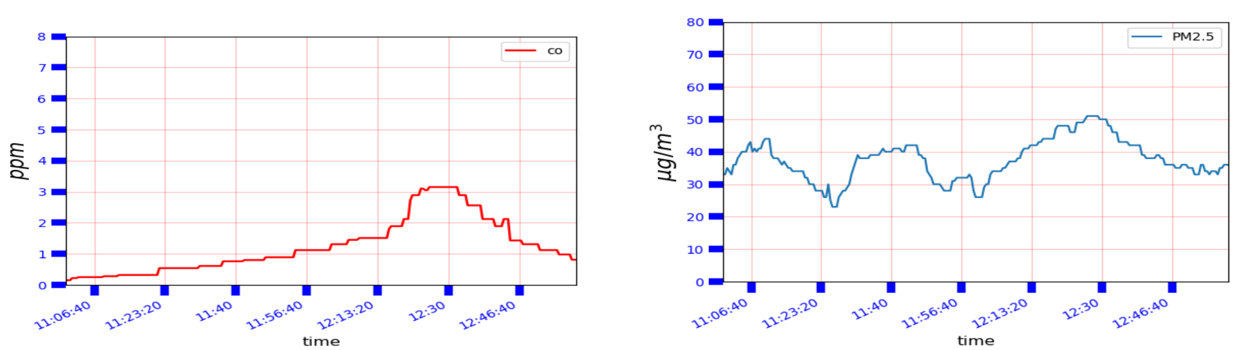
4. Testing and Result Analysis

In order to make sure that we achieved our goal, we performed extensive testing of our all the proposed approaches. This chapter shows result analysis carried out to see the fruitfulness of our research work.

4.1 Results and discussion of IoT based air quality monitoring system

4.1.1 Visualization and data analytics

The installation and testing of the proposed system done at indoor and outdoor sites. The indoor site experiments were conducted at the home (site 2), and outdoor experiments were done at Bhavnagar, India (site 1). In the conducted experiments, HiveMQ MQTT broker was used. MQTT messages were published with QoS levels 0 and 1. Figure 5 depicts the real-time graphical user interface of the observed data received by MQTT subscriber from smart node deployed at outdoor site. Real time graphs of sensor data implemented using “matplotlib” library inside MQTT subscriber script using python. As shown in Figure 5, obtained results display observed data of CO, PM2.5 and PM10. Observation has been received at interval of 60 seconds and displayed by updating every one-hour sliding window. Along with real time graph rendering of observation parameters MQTT subscriber fetches readings from the sensing units and also stores these data in MYSQL for analysis, however it can be stored in any database of choice.



(a) (b)

Fig. 5. Some snaps of GUI cum graphs generated for live monitoring of air quality parameters at home rooftop (site 1: outdoor) per one-hour sliding window (a) CO (b) PM 2.5

Figure 6(a) shows distribution and comparison of observed pollutants based on the data collection of day time for indoor and outdoor deployments. PM2.5 values vary in the range of 23 to 51 µg /m^3 with the median of 35, PM10 varies in range 98 to a maximum value of 126 µg /m^3 with observed median of 110. The maximum observed value of CO at site 1(outdoor) is 3.15 ppm. The obtained readings are recorded and analyzed for 6 hours for site 1(outdoor) and site 2(indoor). From the obtained results of site 1, it is observed that the temperature is varying in the range of 32.5◦C to 34.8◦C, and humidity is varying in the range of 24.1 % to 31.1 %. In the case of site 2(indoor), median values for CO, PM10, and PM 2.5 are recorded around 0.38 ppm, 80 µg /m^3 and 24µg /m^3 respectively. During day-time, the maximum value of PM10 has been recorded is 102 µg /m^3 for site 2(indoor), which is 23% lesser compared to maximum value recorded at site1. Figure 6(b) and Figure 6(c) show the observed data of Particulate matter 10, 2.5 and CO during day time for outdoor and indoor deployments. They also depict the differences in the air quality level between both the sites. It can be observed that the indoor environment site (site 2) is less polluted compared to the outdoor site (site 1). Moreover, the observation of site 1(outdoor) also indicates that it has recorded a higher concentration of PM2.5, PM10 and CO air pollutants during mid-day. The observations of site 1 also pinpoints that the recorded value of parameters PM10 and PM2.5 have fluctuated around 110 and 35, respectively.

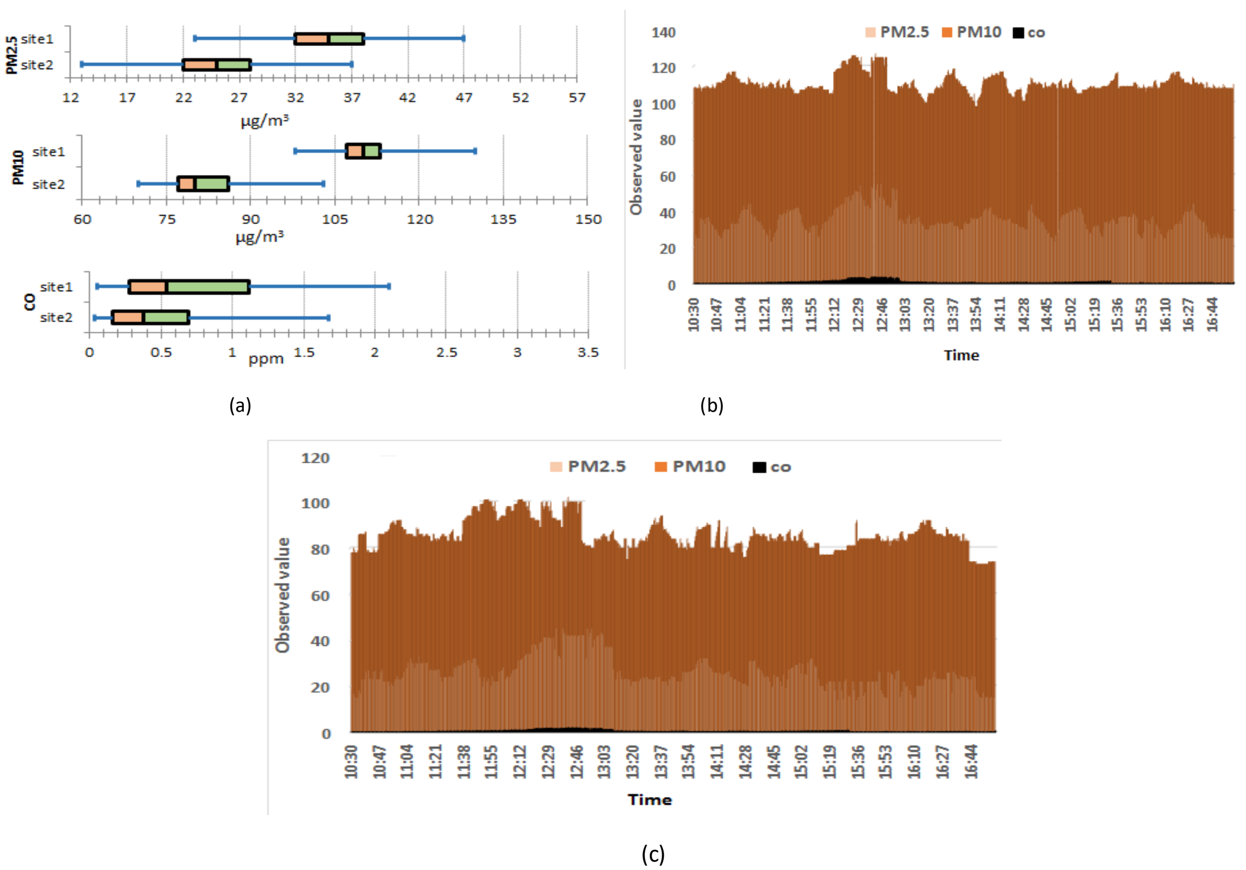


Fig. 6.(a) distribution comparison of observed Parameters (b) observation of PM2.5, PM10 and CO at rooftop during day time (site 1: outdoor) (c) observation of PM2.5, PM10 and CO at home during day time (site 2: indoor)

4.1.2. Quality of Service(QoS) and system performance under periodic transmission

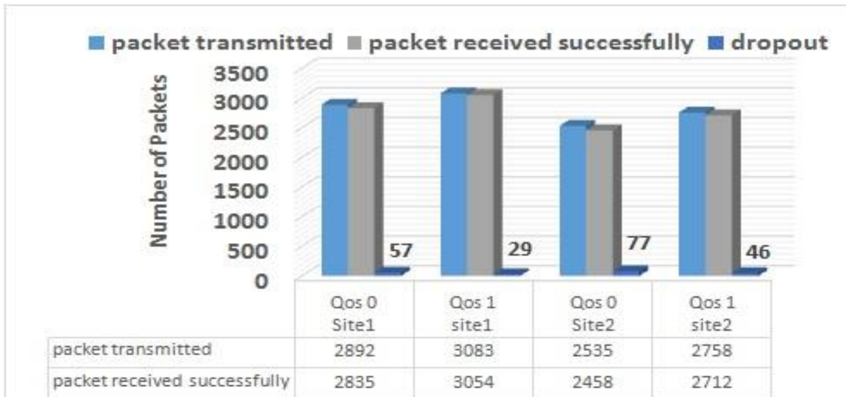


Fig. 7. System performance over an MQTT protocol: rooftop (site 1: Outdoor) and home (site 2: Indoor)

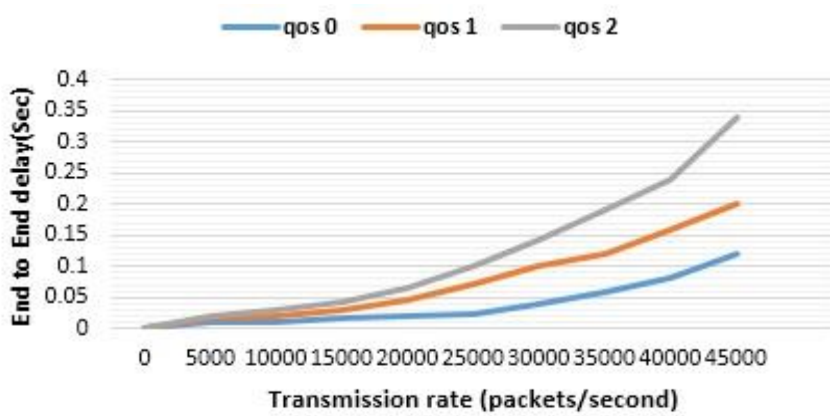
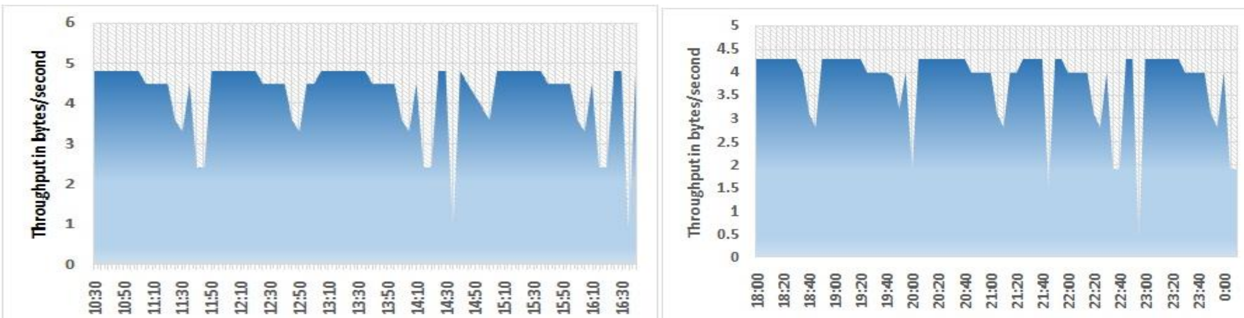


Fig. 8. End to End delay against QoS level in simulation

Figure 7 shows the performance of the system at QoS level 0 and 1 for the duration of 12 hours. In the case of QoS level 0, packet dropout ratio has been recorded around 1.97% and for QoS, level 1 the packet dropout ratio has been recorded around 0.94 % at site1(indoor). To analyze effect of QoS level on end to end delay, we simulate the publisher environment using MQTT-JMeter(a tool from apache). MQTT plugin configured with JMeter, can serve to perform testing in which simulated clients register to the broker. Figure 8 shows the end to end delay co-relation with transmission rate under various QoS levels during the simulation. It is observed that the end to end delay will also increase whenever a packet is transmitted from a lower QoS level to a higher QoS level due to retransmission and acknowledgment overhead. So the selection of the QoS level becomes an essential criterion for mitigating end to end delay and packet loss ratio. As shown in Figure 7, the accuracy values recorded at site

1(outdoor) and site 2(indoor) are around 98% and 96% respectively. The accuracy values have been calculated considering the number of transmitted and received packets at indoor and outdoor sites. Accuracy and dropout parameters are representing the system performance in terms of reliable delivery of MQTT messages including sensing parameters. However, it cannot be interpreted as performance measurement or accuracy measurement of air quality pollution sensors. As shown in Figure 9(a) and 9(b), it is observed that the recorded average throughput of the sensing node is around 4.28 for site 1 and 4.6 for site 2 for the duration of 6 hours.



(a) (b)

Fig. 9. Throughput of the Sensing Unit (a) home (site 2: indoor) (b) rooftop (site 1: Outdoor)

4.1.3. Power consumption optimization scheme performance

The average power usage of node under the power optimization scheme discurssed in subsection 3.1.4 can be calculated as:

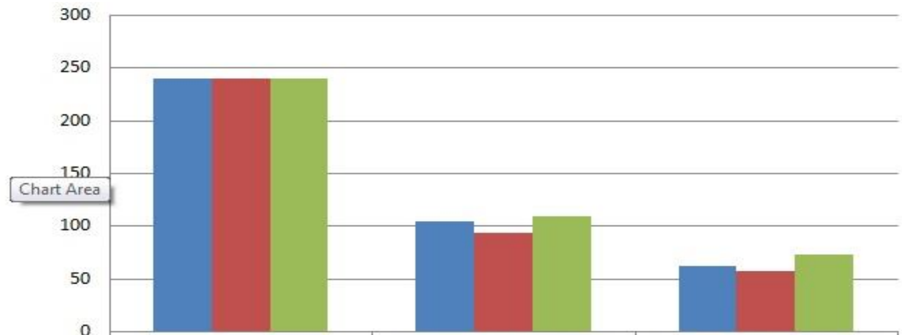
$$P = \left[\frac{I_{lsm} * T_{sleep} + I_{active} * T_{active}}{T_{si}} + I_{CO}\right] * V_{in} \tag{8}$$

where I_{lsm} is current drawn in light sleep mode by the smart node, T_{sleep} is time duration in sleep mode, I_{active} is current drawn in an active mode, T_{active} is time that includes data collection and stabilization time of sensors, T_{si} is the total time/sampling interval that includes node operation and sleep period, I_{co} is the current drawn by Carbon Monoxide sensor which is never at rest and Vin is the input voltage. A smart node will remain in a sleep mode for 60 seconds out of around 90 seconds sampling interval. The average power consumption of a smart node for the period of 90 seconds is 413 mW plus consumption of CO sensor (CO sensor is never at rest) by applying power optimization scheme which is around 900 mW plus consumption of CO sensor without power optimization. The battery lasting time is found to be extended as mentioned in below table during the experiments.

	Without Power	**Under Power Reduction**
Battery Life Time (2500	7 hrs 15 minutes	12 hrs 50 minutes

4.1.4 Event based transmission performance

Figure 10 shows the number of transmission under the employed scheme experimented for 6 hours a day. Number of transmission is counted considering number of subscripted MQTT messages available at the subscriber. Table indicates very less number of transmitted messages in event based transmission scheme. Reduced number of transmission results in power consumption reduction.



	Periodic	Event based transmission (with N=3, δ=0.05)	Event based transmission (with N=3, δ=0.10)
Number of transmissions (day one)	240	104	62
Number of transmissions (day Two)	240	93	57
Number of transmissions (day Three)	240	109	73

Fig. 10. Message transmission under event based transmission

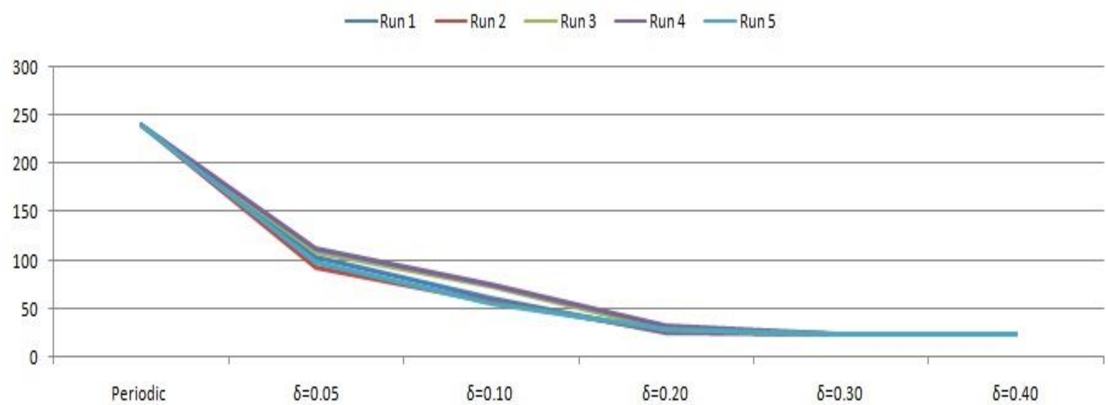


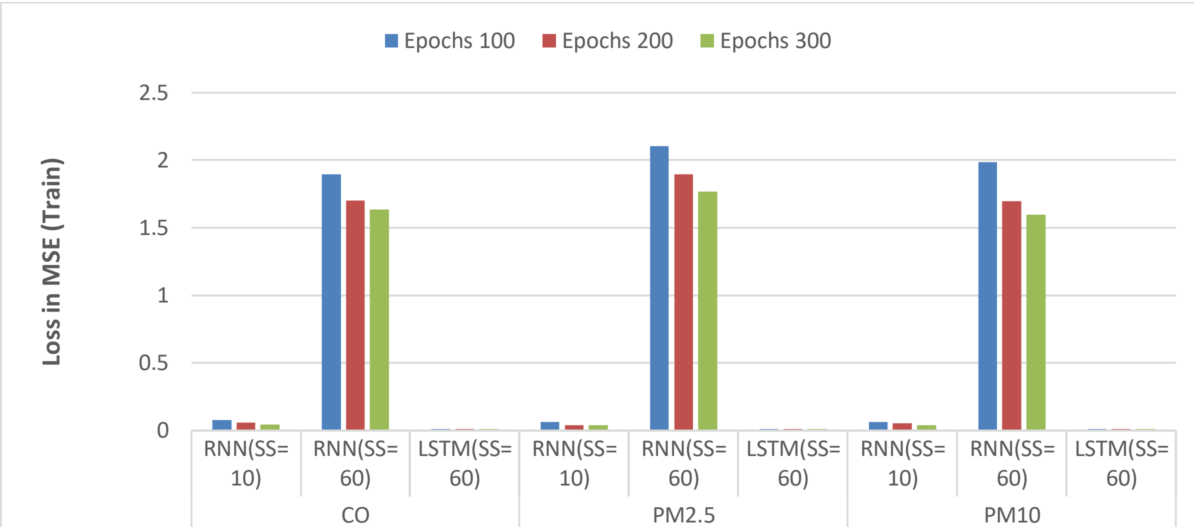
Fig. 11. Message transmission for N=3 and various values of delta

Employability of such scheme depends on tradeoff between the requirements of tolerance in micro level changes(sensitiveness) to be reported at server versus power consumption. Figure 11 shows the effect of delta value on number of transmissions over five different runs. Number of transmission becomes periodic transmission under the larger value of delta as shown in figure 11.

4.2 Results and discussion of air quality prediction system

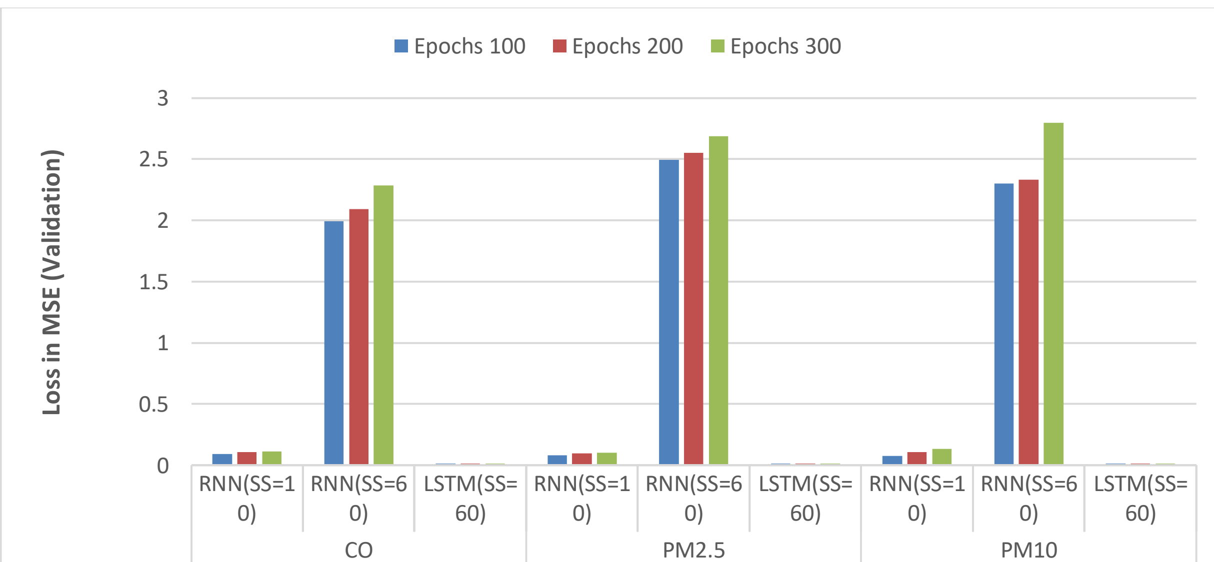
4.2.1 MSE comparison for training and validation data under proposed model

- Performance of LSTM Vs RNN



	CO RNN(SS=10)	CO RNN(SS=60)	CO LSTM(SS=60)	PM2.5 RNN(SS=10)	PM2.5 RNN(SS=60)	PM2.5 LSTM(SS=60)	PM10 RNN(SS=10)	PM10 RNN(SS=60)	PM10 LSTM(SS=60)
Epochs 100	0.0761	1.8967	0.0118	0.0643	2.1032	0.0104	0.0617	1.9873	0.0111
Epochs 200	0.0579	1.6993	0.0107	0.0408	1.8976	0.01	0.051	1.6973	0.0105
Epochs 300	0.0443	1.6341	0.0104	0.0392	1.7689	0.0089	0.0398	1.5961	0.0102

(a)

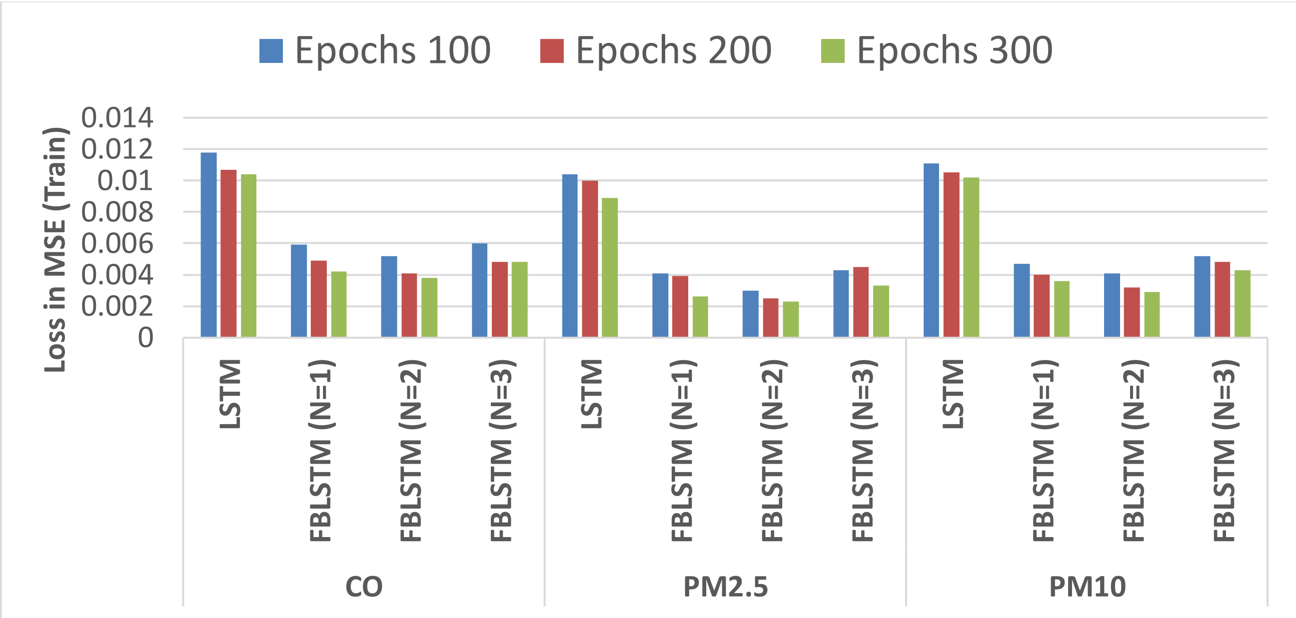


	CO RNN(SS=10)	CO RNN(SS=60)	CO LSTM(SS=60)	PM2.5 RNN(SS=10)	PM2.5 RNN(SS=60)	PM2.5 LSTM(SS=60)	PM10 RNN(SS=10)	PM10 RNN(SS=60)	PM10 LSTM(SS=60)
Epochs 100	0.0903	1.9942	0.014	0.0834	2.4926	0.0119	0.0782	2.3031	0.0135
Epochs 200	0.1079	2.0934	0.0129	0.0986	2.5503	0.013	0.1087	2.3314	0.0137
Epochs 300	0.1127	2.2832	0.0144	0.1023	2.6845	0.0147	0.1324	2.7941	0.0153

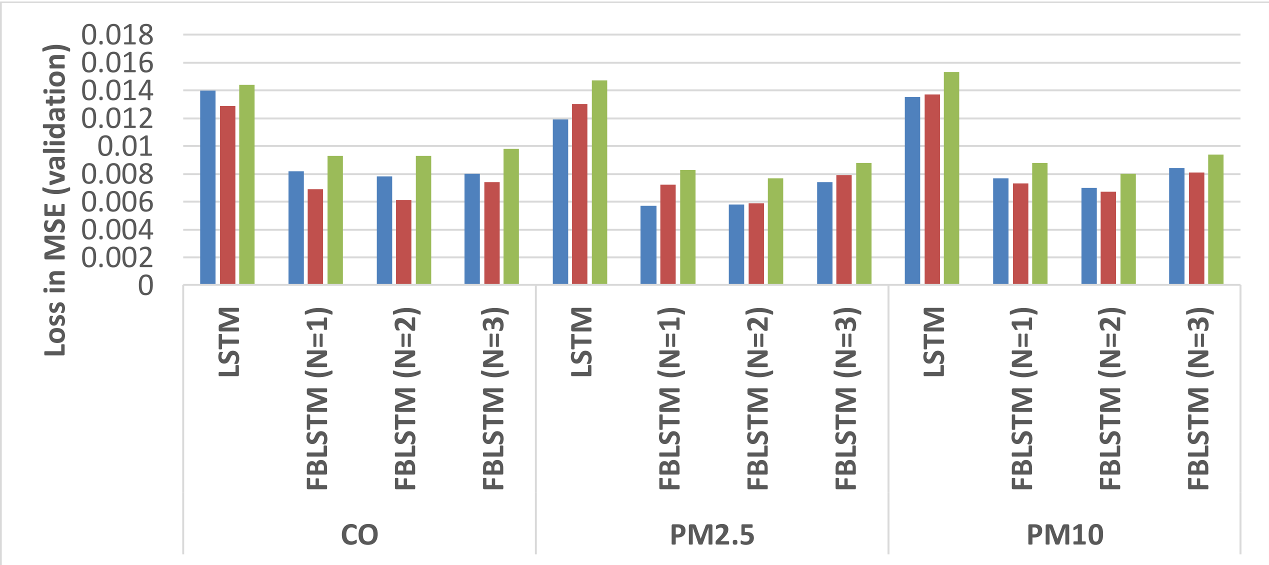
(b)

Fig. 12 Comparison of MSE of LSTM and RNN for (a) training and (b) validation

- Performance of FBLSTM Vs LSTM



(a)



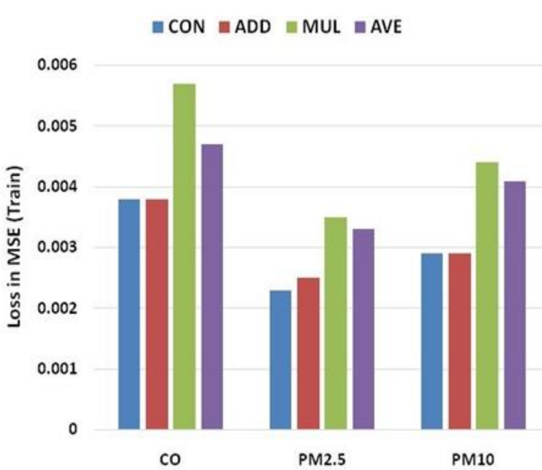
(b)

Fig. 13 Comparison of MSE of proposed model(FBLSTM) with LSTM for (a) training and (b) validation

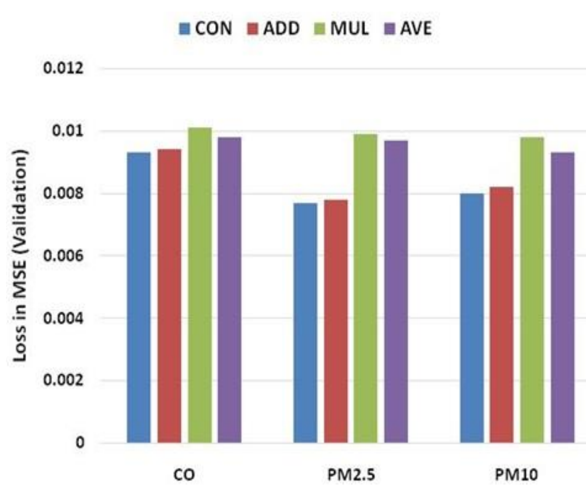
Implementation of the proposed FBLSTM model and experimentations are carried out using Keras 2.1.6 which uses Tensorflow in the back end. The model uses 60 units in each layer and tanh function is utilized as activation function. The input window or sequence size in each sample is set to 60. The ADAM algorithm is utilized in model for optimization which adapts the rate of learning based on the average of first as well second moments of the gradients. The batch size of 32 is used during the experiments. The MSE values listed are averaged values over six repeated runs. Experiments are conducted till 300 epochs and loss reported at the end of 100, 200 and 300 epochs are listed for both train and test(validation) data in the table.

In figure 12, results of RNN and simple LSTM are compared. RNN suffers from a vanishing gradient problem as the sequence sample size increases which also can be observed in the performance evaluation(figure 12). As the sequence size increases from 10 to 60 the loss value increases. LSTM outperforms RNN greatly specially for higher sequence sizes.The number of unidirectional layers in stacking in proposed model, to gain minimum loss for prediction of air pollutant time series data is decided through experiments. Figure 13 shows the loss in MSE(mean squared error) with various stacking options for the proposed FBLSTM model. Figure 13 shows the output of the experiments with the "CONCAT"(Con) function for merging the two layers (forward and backward), which is the default one in Keras. Mean Squared Error values highlighted in bold in the table show the minimum observed loss during experiments. The minimum value of loss in turn indicates the best accuracy for time series prediction. It can be observed that in the proposed FBLSTM model with the discussed parameter setting, minimum loss is observed with stacking of two hidden layers. Moving further by adding one more layer to the existing unidirectional stacking i.e three hidden layer, the performance starts degrading. The proposed model outperforms both the RNN and the simple LSTM layer.

- Performance comparison of various merging mode of FBLSTM



(a)

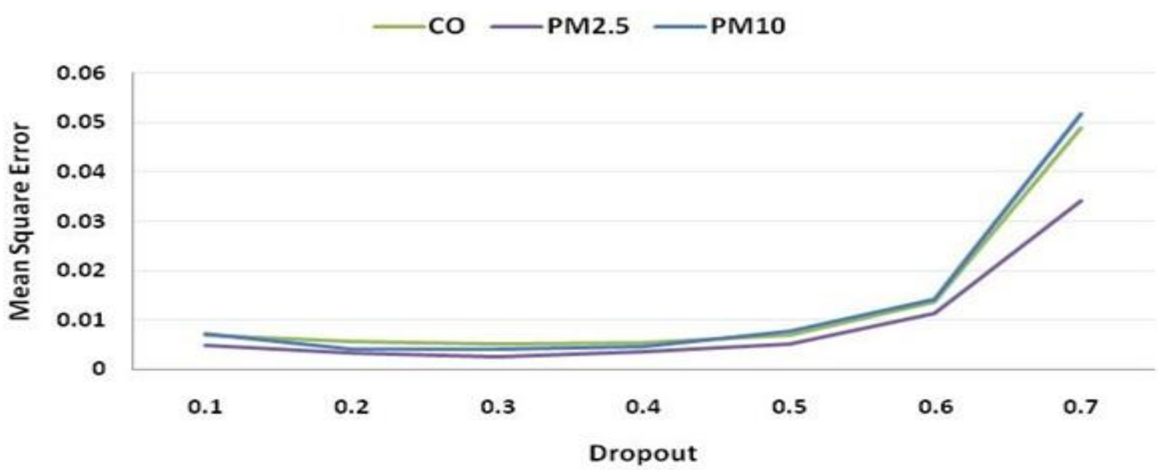


(b)

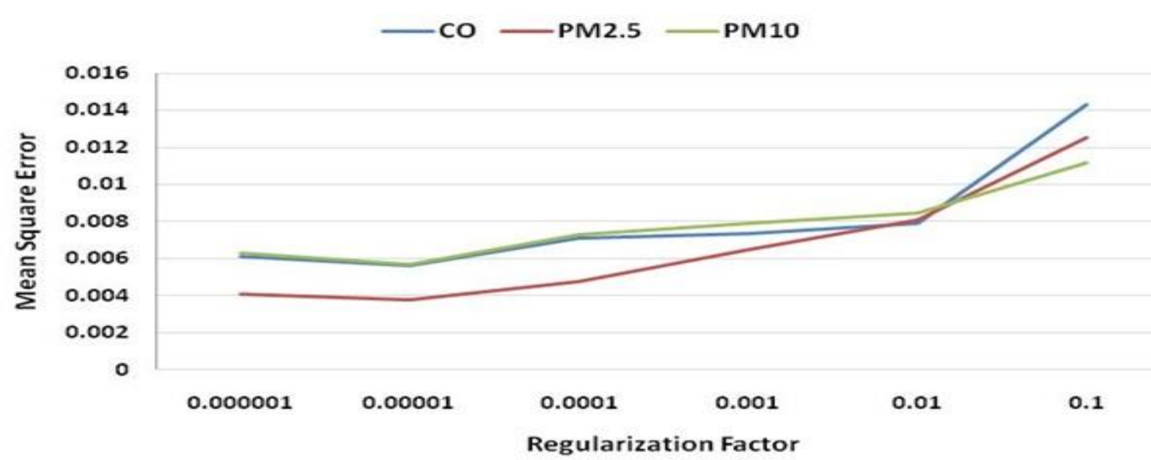
Fig. 14. MSE for merge function alternatives over: (a) training data (b) validation data

There are four merging functions available in Keras. Concat(Con) is the default merging option where the output of the respective cell state from forward and backward layers are simply concatenated together. Mul and Add are the merge modes where such outputs are multiplied or added respectively. Ave is the merge function where the average of the corresponding outputs is taken. We utilized the optimum FBLSTM model setup which has two hidden layers and experimented with all possible four merge modes. As shown in Figure 14 the Con function performs best over train as well as test data and achieves minimum loss function. Add function also perform near equal to the Con function while Mul function has the highest loss amongst all four.

4.2.2 Performance improvement by overfitting resolution for validation data

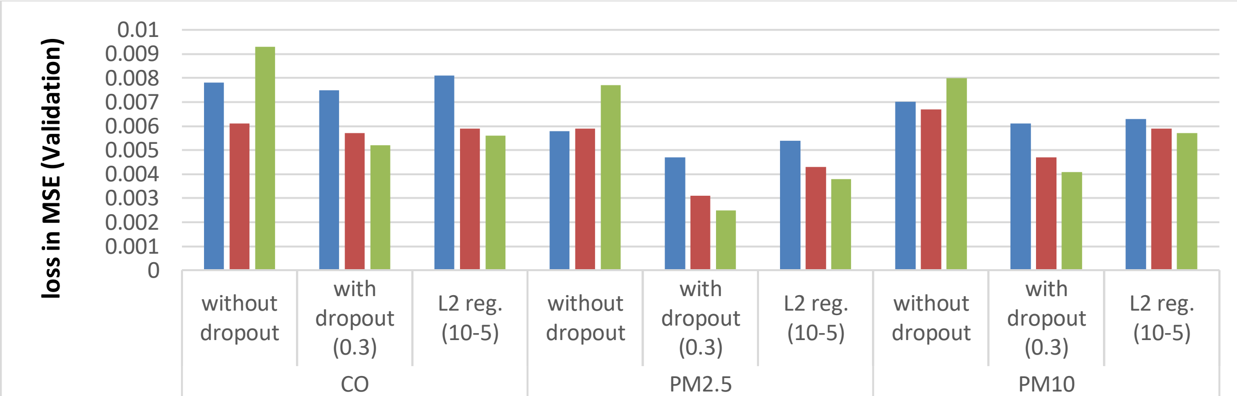


(a)



(b)

Fig. 15. Performance of the model for: (a) various values of dropout parameter under dropout technique (b) various values of lambda or regularization factor under L2 regularization



	CO without dropout	CO with dropout (0.3)	CO L2 reg. (10-5)	PM2.5 without dropout	PM2.5 with dropout (0.3)	PM2.5 L2 reg. (10-5)	PM10 without dropout	PM10 with dropout (0.3)	PM10 L2 reg. (10-5)
Epoch 100	0.0078	0.0075	0.0081	0.0058	0.0047	0.0054	0.007	0.0061	0.0063
Epoch 200	0.0061	0.0057	0.0059	0.0059	0.0031	0.0043	0.0067	0.0047	0.0059
Epoch 300	0.0093	0.0052	0.0056	0.0077	0.0025	0.0038	0.008	0.0041	0.0057

Fig. 16. MSE comparison of proposed model(FBLSTM) under regularization techniques for validation data

Keras supports the implementation of the dropout by using dropout layers. Dropout layers are added in between hidden layers. Input and recurrent edges or connections to LSTM units are excluded with set probability from activation and updates of weights during training of a network. The addition of dropout layers results in simulating a large number of networks with a very dynamic network structure in parallel. Also due to dropout, a neural network can never rely on any input node because every node has probability to be removed. So, the network cannot assign any high weight to a specific feature. The probability of such dropout is again hyperparameter which is needed to be chosen with experiments. Figure 15(a) shows all experimented values of dropout, plotted against the MSE loss recorded for particular dropout value. Applied dropout value can vary in the range of 0 to 1. It can be seen from figure 15(a) that minimum loss is recorded for 0.3 value of dropout and after 0.5 dropout value, there is a sudden increase in the loss function. Dropout value 0.3 in Keras is representing 30 percent probability of node removal during training.

Another technique of regularization is using weight decay known as L2 regularization. The neural network always tries to minimize the cost function by adjustment of weights and biases. For L2 regularization a component is added which penalizes the large weights. The component is added to the cost function. The addition of the component drives the overall weight matrix values down, which in turn reduces the activation function influence. Due to that comparatively less complex activation function may be fit to observations, which helps in reducing overfitting. Figure 15(b) shows all experimented values of lambda plotted against the MSE loss recorded for particular lambda value (regularization factor). It can be seen from the figure that the minimum loss recorded for 10^{-5} value of lambda. Figure 16 shows loss function MSE recorded for three timeseries data of pollution parameters at the end 100, 200 and 300 epochs for validation data. The table compares the MSE values for 0.3 dropout and 10^{-5} for lambda value which are found to be providing optimum results during respective regularization experiments. The Results show that dropout based regularization outperforms weight decay(L2) regularization and more suitable to our model. Dropout regularization can achieve stable conversation along with loss in MSE up to 0.0052, 0.0025 and 0.0041 for CO, PM 2.5 and PM 10 respectively.

4.2.3 Performance with self-attention mechanism adoption

Model (timestep)	Week1						Week2					
	CO	Rate	PM2.5	Rate	PM10	Rate	CO	Rate	PM2.5	Rate	PM10	Rate
FBLSTM(T_X)	0.0051	-	0.0025	-	0.0041	-	0.0039	-	0.0032	-	0.0045	-
FBLSTM + Attention(T_X)	0.0047	-	0.0023	-	0.0036	-	0.0038	-	0.0031	-	0.0038	-
FBLSTM($4T_X$)	0.0055	7.84	0.0028	12.00	0.0045	9.76	0.0044	12.82	0.0035	9.38	0.0051	13.33
FBLSTM + Attention ($4T_X$)	0.0049	4.26	0.0025	8.70	0.0039	8.33	0.004	5.26	0.0033	6.45	0.0041	7.89
FBLSTM ($8T_X$)	0.0073	32.73	0.0044	57.14	0.0066	46.67	0.0061	38.64	0.0058	65.71	0.0076	49.02
FBLSTM + Attention ($8T_X$)	0.0053	8.16	0.0029	16.00	0.0044	12.82	0.0043	7.50	0.0037	12.12	0.0047	14.63
FBLSTM ($12T_X$)	0.011	50.68	0.0083	88.64	0.0118	78.79	0.0097	59.02	0.0103	77.59	0.0125	64.47
FBLSTM + Attention (12_X)	0.006	13.21	0.0034	17.24	0.0052	18.18	0.0051	18.60	0.0044	18.92	0.0052	10.64

Table 2. MSE comparison of FBLSTM with attention and without attention after 300 epochs for various time horizons

	Input Window size	CO	Rate	PM2.5	Rate	PM10	Rate
FBLSTM	60	0.0055	-	0.0035	-	0.0046	-
FBLSTM + Attention		0.0036	-	0.0021	-	0.0034	-
FBLSTM	80	0.0084	52.73	0.0057	62.86	0.0074	60.87
FBLSTM + Attention		0.0045	25.00	0.0029	38.10	0.0045	32.35
FBLSTM	120	0.0223	165.48	0.0191	235.09	0.0208	181.08
FBLSTM + Attention		0.0086	91.11	0.0061	110.34	0.0091	102.22

Table 3. MSE comparison of FBLSTM with attention and without attention after 300 epochs for various input windows

Self attention mechanism applied and tested during the experiments. Self attention layer is kept as the last layer in the model(FBLSTM) shown in figure 2. The output of self attention layer is given as input to the final dense layer for prediction. To understand effect on loss function and improvement to existing model, we analyze self attention mechanism with two dimensions, time horizon as well input window size or input lag. While increasing time horizon, the sequence size is kept of the same length. By keeping same sequence size and increase in time horizon, employed recorded parameter samples in training realizes more fluctuations compared to small time horizon. Table 2 compares the loss function value (mean squared error) obtained for the FBLSTM (with two hidden layer), without attention and with attention mechanism for the three air quality parameters. The table shows the effect on MSE value with the increase in time horizon. First two rows in the table depict the MSE value for T_x (the basic time step in the input sequence) which is 90 seconds. The time horizon increment further is obtained by aggregating the recorded value for the basic timestep, i.e $4T_x$ horizon is the aggregated value over 360 seconds and so on. To keep the total sample size and samples in each sequence same with the extension of time horizon (for each horizon), more training data(samples) are required. So training data (recorded observations of air quality parameters) covers data of one complete week. The table shows the results obtain over data of two such weeks. The rate column in the table shows the percentage of increase in MSE value with the increase of time horizon from the previous one. It can be seen from the table that with the extension of time horizon, the rate of increase in MSE (compared to the previous horizon) remains small for the model with attention mechanism. The high rate of increase in MSE represents rapid reduction in prediction performance with the extension in horizon. It can be seen that initially there is not much difference between performance of two models but with higher time horizon, model with attention mechanism performs substantially well compared to model without attention. Table 3 shows the performance of the two model with increase in input window size over recorded air quality parameters observations of a single day. The MSE value and rate of increase in MSE is listed for input widow size of 60, 80 and 120. The table indicates that the model with attention mechanism realizes lower MSE and slow rate of increase in MSE for higher input window size. Thus the table depicts self-attention mechanism provides better performance for longer sequences.

5. CONCLUSIONS AND A ROAD-MAP

5.1 CONCLUSIONS

In the proposed research work, we have proposed a cost efficient economic IoT based air quality monitoring system to measure a variety of air pollutants for indoor and outdoor environments. We have proposed standardized layer architecture(framework) and addressed the issues of complex architecture (no standardization) design in IoT. The proposed architecture supports features like no relay node requirement or network setup dependency, authentication of publisher at broker during registration phase and physical level integrity check while reading data from sensors. The architecture uses cloud broker which make it scalable. The architecture uses topic based hierarchy for storage of published messages at broker that enables support for subscribers with different platform i.e python script, android etc. Also due to such topic based hierarchy site specific data distribution to the relavant subscribers can be handled easily. The proposed system is also tested at a variety of quality of service levels at two sites. The goal of QoS level demonstration and testing is to justify the data delivery accuracy of the system which is very critical for real time air quality monitoring. The obtained results have recorded accuracy of 98% at site 1(outdoor) and 96% at site 2(indoor), accuracy in terms of reliable delivery. A web and mobile interface has been designed to display the measured parameters in near real-time. The system also reduces power consumption by switching a smart sensing node into five different modes. With the given power consumption reduction scheme the battery lasting life can be increased to 1.7% of the battery lasting period without any such scheme. Moreover, a novel event based transmission scheme is implemented to reduce number of transmission. However exact power saving in terms of absolute power unit is not represented, rather reduction in transmission of MQTT messages under the employed scheme is represented.

Reliable and precise prediction of air pollution or air quality parameters is of great importance. In the presented work, LSTM based deep learning framework is also proposed for the prediction of air quality parameters. The framework includes forward and backward LSTM based model for learning the influence of other observations on current prediction in two directions. The proposed model is using bidirectional learning with unidirectional further stacking to improve the performance. Optimum performance is achieved with two hidden layers and CONCAT merging function (out of four merging function alternatives available in Keras) in the proposed model and the model outperforms the simple RNN and LSTM based model. The proposed work is also solving the overfitting issue by applying L2 regularization and dropout methods. Best fit for validation is found with 0.3 dropout value and 10^{-5}

lambda value under dropout and L2 regularization methods respectively. The dropout method performs better and achieves lower value of MSE with good convergence compared to L2 regularization. Finally, self-attention is applied as the last layer in the model and the effect is assessed on two dimensions, for various extended time horizons (T_x,$4T_x$,$8T_x$ and $12T_x$) and with varying input windows (60,80 and 120). The results show significant improvement in the performance with both dimensions. Future work of the current study includes an extension of attention application and model capability in learning the influence of temperature and humidity (recorded at same timesteps) in the prediction of air quality parameters.

5.2 ROAD MAP FOR FUTURE WORK

We have represented QoS analysis which can assist the proposed IoT based air quality monitoring system in the implementation of an adaptive algorithm in future. The adaptive algorithm can facilitate auto switching between varieties of QoS levels as and when required. The auto-switching process highly depends on the parameter such as critical pollution duration. In prediction of air quality parameters, we have not applied any mechanism to assess the influence of temperature and humidity recorded over the prediction of particulate matter or carbon monoxide. future work of the current study includes an extension of attention application and model capability in learning the influence of temperature and humidity (recorded at same timesteps) in the prediction of air quality parameters.

6. REFERENCES

[1] A. Zanella, N. Bui, A. Castellani, L. Vangelista and M. Zorzi, "Internet of Things for Smart Cities" in IEEE Internet of Things Journal, vol. 1, no. 1, pp. 22-32, Feb. 2014.

[2] World Health Assembly, 69, "Health and the environment: a draft road map for an enhanced global response to the adverse health effects of air pollution: report by the Secretariat", 2015[Online]. Available: World Health Organization https://apps.who.int/iris/handle/10665/252673.

[3] S. Gulia, I. Khanna, K. Shukla., M. Khare, "Ambient air pollutant monitoring and analysis protocol for low and middle-income countries: An element of comprehensive urban air quality management framework", in Atmospheric Environment, vol. 222, pp. 1-25, 2020.

[4] Lancet Planet Health, "The impact of air pollution on deaths, disease burden, and life expectancy across the states of India: The Global Burden of Disease Study", 2017[Online]. Available: http://dx.doi.org/10.1016/ S2542-5196(18)30261-4

[5] WHO, "Air quality guidelines for particulate matter, ozone, nitrogen dioxide, and sulfur dioxide. Global update", 2005 [Online]. Available :http://whqlibdoc.who.int/hq/2006/WHO_SDE_PHE_OEH_06.02_eng.pdf

[6] Hu. R., X. XY., Xu. SK., Y. Wang, M. Jiang, L. R. Wen, W. Lai, and L. Guan, "PM2.5 exposure elicits oxidative stress responses and mitochondrial apoptosis pathway activation in HaCaT keratinocytes" in Chin Med J, vol. 130, pp. 2205–2214, 2017.

[7] M. Piao., H. YJ., K. Shilnikova, J. JW., "Particular matter 2.5 damages skin cells by inducing oxidative stress subcellular organelle dysfunction and apoptosis" in Arch. Toxicol, vol. 92, 2077–2091, 2018.

[8] L. Atzori A. Iera G. Morabito, "The internet of things: A survey" in Computer Networks, vol. 54, no. 15, pp. 2787-2805, 2010

[9] Dockery Douglas W. et al., "An association between air pollution and mortality in six US cities" in New England journal of medicine, vol. 329, no. 24 , pp. 1753-1759, 1993.

[10]C. Arden Pope III and Douglas W. Dockery, "Health effects of fine particulate air pollution: lines that connect" in Journal of the air & waste management, vol. 56, no. 6, pp. 709-742, 2006

[11]Künzli, Nino, et al., "Public-health impact of outdoor and traffic-related air pollution: a European assessment" in The Lancet, vol. 356, no. 9232, pp. 795-801, 2000.

[12]Chang, Yu-Ching et al. "Risk Factors and Outcome Analysis in Children with Carbon Monoxide Poisoning" in Pediatrics & Neonatology, Vol. 58, no. 2, pp. 171 – 177, 2017.

[13]SikaryAsit Kumar, Dixit Sumit, Murty Om P, "Fatal carbon monoxide poisoning: A lesson from a retrospective study at All India Institute of Medical Sciences, New Delhi" in Journal of Family Medicine and Primary Care, Vol. 6, no. 4, pp. 791-794, 2017.

[14]Sharnil Pandya, , Ketan Kotecha, K., Mohammed Awais,, Saeed Akbarzadeh, Prosanta Gope, S C. Mukhopadhyay, Chen W., "Smart Home Anti-Theft System: A Novel Approach for Near Real-Time Monitoring and Smart Home Security for Wellness Protocol" in Appl. Syst. Innov. vol. 1, no. 4, 2018.

[15] Hemant Ghayvat, Awais M., Pandya S., Ren H., Akbarzadeh S., S Chandra Mukhopadhyay, Chen, C., Gope, P., Chouhan, A., Chen, W. "Smart Aging System: Uncovering the Hidden Wellness Parameter for Well-Being Monitoring and Anomaly Detection" in Sensors, vol. 19, no.4, article 766, 2019.

[16]H. Ghayvat and S. Pandya, "Wellness Sensor Network for modeling Activity of Daily Livings – Proposal and Off-Line Preliminary Analysis", 4th International Conference on Computing Communication and Automation (ICCCA), 2018, pp. 1-5.

[17]W. D. Patel, S. Pandya, B. Koyuncu, B. Ramani, S. Bhaskar and H. Ghayvat, "NXTGeUH: LoRaWAN based NEXT Generation Ubiquitous Healthcare System for Vital Signs Monitoring & Falls Detection", 2018 IEEE Punecon, Pune, India, 2018, pp. 1-8.
[18]Sheng-Tun Li, Shih-Wei Chou and Jeng-Jong Pan, "Multi-resolution spatio-temporal data mining for the study of air pollutant regionalization", Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, Maui, HI, USA, 2000, pp. 7
[19]Reisinger, G.Fraser, P. Johnston, R. McKenzie, and W.A. Matthews, “Slow-scanning DOAS system for urban air pollution monitoring”, Proceedings of the XVIII Quadrennial Ozone Symposium, 1996, pp. 12-21.
[20]M. Richards, M. Ghanem, M. Osmond, Y. Guo, J. Hassard,“Grid-based analysis of air pollution data” in ecological modeling, vol. 194, no. 1-3, pp. 274–286, 2006.
[21]Q. Abu Al-Haija, H. Al-Qadeeb, A. Al-Lwaimi. 2013. “Case Study: Monitoring of AIR quality in King Faisal University using a microcontroller and WSN” in Procedia Computer Science, vol. 21, pp. 517-521, 2013.
[22]N. Kularatna and B. H. Sudantha, "An Environmental Air Pollution Monitoring System Based on the IEEE 1451 Standard for Low Cost Requirements” in IEEE Sensors Journal, vol. 8, no. 4, pp. 415-422, April 2008.
[23] A. R. Al-Ali, I. Zualkernan and F. Aloul, "A Mobile GPRS-Sensors Array for Air Pollution Monitoring" in IEEE Sensors Journal, vol. 10, no. 10, pp. 1666-1671, Oct. 2010.
[24]Srinivas Devarakonda, ParveenSevusu, Hongzhang Liu, Ruilin Liu, LiviuIftode, BadriNath, “Real-time Air Quality Monitoring Through Mobile Sensing in Metropolitan Areas”, Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing, 2013.
[25]Atlan, Tracking Air Pollution in Delhi, 2014[Online]. Available: https://blog.socialcops.com/engineering/tracking-air-pollution-in-delhi, Accessed: 12 December, 2019.
[26]Sherin Abraham, Xinrong L, “A Cost-Effective Wireless Sensor Network System for Indoor Air Quality Monitoring Applications” in Procedia Computer Science, vol. 34, pp. 165-171, 2014.
[27]A. Kumar and G. P. Hancke, ”Energy efficient environment monitoring system based on the IEEE 802.15.4 standard for low cost requirements” in IEEE Sensors J., vol. 14, no. 8, pp. 2557-2566, 2014.
[28]S. Ferdoush and X. Li, “Wireless Sensor Network System Design Using Raspberry Pi and Arduino for Environmental Monitoring Applications” in Procedia Computer Science, vol. 34, pp. 103–110, 2014.
[29]M. Bacco, F. Delmastro, E. Ferro and A. Gotta, “Environmental Monitoring for Smart Cities” in IEEE Sensors Journal, vol. 17, no. 23, pp. 7767-7774, 2017.
[30]L. Morawska, P. Thai, X. Liu, A. Asumadu-Sakyia, G. Ayoko, A. Bartonova; A. Bedini, F. Chai; B. Christensen, M. Dunbabin, “Applications of low-cost sensing technologies for air quality monitoring and exposure assessment: how far have they gone?” in Environ. Int., vol. 116, pp. 286-299, 2018.
[31]D. K. Sharma, M. Rajput, S. A. Akbar and A. Kumar, “Development of embedded system for carbon nano tube (CNT) based Ammonia (NH3) gas sensor”, Annual IEEE India Conference (INDICON), 2015, pp. 1-4.
[32]Tiwari A., Sadistap S., Mahajan S.K., “Development of Environment Monitoring System Using Internet of Things” in Ambient Communications and Computer Systems. AISC, vol. 696, pp. 403-412, 2018.
[33]G. Marques, I. M. Pires, N. Miranda, R. Pitarma, “Air Quality Monitoring Using Assistive Robots for Ambient Assisted Living and Enhanced Living Environments through Internet of Things” in Electronics, vol. 8, no. 12, article. 1375, 2019.

[34]Hong-di He, Min Li, Wei-li Wang, Zhan-yong Wang, Yu Xue, "Prediction of PM2.5 concentration based on the similarity in air quality monitoring network" in Building and Environment, vol. 137, pp. 11-17, 2018.
[35]S. Dhingra; R. B. Madda; A. H. Gandomi; R. Patan, M. Daneshmand, "Internet of Things Mobile–Air Pollution Monitoring System (IoT-Mobair)" in Internet of Things Journal, vol. 6, no. 3, pp. 5577-5584, 2019.
[36]J. Huang et al., "A Crowdsource-Based Sensing System for Monitoring Fine-Grained Air Quality in Urban Environments" in IEEE Internet of Things Journal, vol. 6, no. 2, pp. 3240-3247, 2019.
[37]S. Sun, X. Zheng, J. Villalba-Díez, J. Ordieres-Meré, "Indoor Air-Quality Data-Monitoring System: Long-Term Monitoring Benefits" in Sensors, vol. 19, no. 19, article. 4157, 2019.
[38]CPCB,2019. National Air Quality Monitoring Programme. Available: https://cpcb.nic.in/about-namp/, Accessed: 12 December, 2019.
[39]Hong-di He, Min Li, Wei-li Wang, Zhan-yong Wang, and Yu Xue, "Prediction of PM2.5 concentration based on the similarity in air quality monitoring network" in Building and Environment, vol. 137, pp. 11-17, 2018.
[40]Y. Zhang, Y. He, and J. Zhu, "Research on forecasting problem based on multiple linear regression model PM2.5" in J. Anhui Sci. Technol. Univ., vol. 30, no. 3, pp. 92-97, 2016.
[41]K. R. Baker and K. M. Foley, "A nonlinear regression model estimating single source concentrations of primary and secondarily formed PM2:5," in Atmos. Environ., vol. 45, no. 22, pp. 3758-3767, 2011.
[42]Z. Wang and Z. Long, "PM2.5 prediction based on neural network", Proc. 11th Int. Conf. Intell. Comput. Technol. Automat. (ICICTA), Changsha, China, 2018, pp. 44-47.
[43]R. Chuentawat and Y. Kan-Ngan, "The comparison of PM2:5 forecastingmethods in the form of multivariate and univariate time series based on support vector machine and genetic algorithm", Proc. 15th Int. Conf.Electr. Eng./Electron., Comput., Telecommun. Inf. Technol. (ECTI-CON), Chiang Rai, Thailand, 2018, pp. 572-575.
[44]G. Hoek, R. Beelen, K. Hoogh, D. Vienneau, J. Gulliver, P. Fischer and D. Briggs, "A review of land-use regression models to assess spatial variation of outdoor air pollution" in Atmos. Environ., vol. 42, pp. 7561–7578, 2008.
[45]V. G. Kaburlasos, I. N. Athanasiadis, and P. A. Mitkas, "Fuzzy lattice reasoning (FLR) classifier and its application for ambient ozone estimation" in International Journal of Approximate Reasoning, vol. 45, no. 1, pp. 152-188, 2007.
[46]S. Zhang, X. Li, Y. Li, and J. Mei, "Prediction of Urban PM2.5 Concentration Based on Wavelet Neural Network", 2018 Chinese Control and Decision Conference (CCDC), Shenyang, 2018, pp. 5514-5519.
[47]G. Salman, Y. Heryadi, E. Abdurahman, and W. Suparta, "Single layer & multi-layer long short-term memory (LSTM) model with intermediate variables for weather forecasting" in Procedia Comput. Sci., vol. 135, pp. 89-98, 2018.
[48]Y. Tsai, Y. Zeng, and Y. Chang, "Air pollution forecasting using RNN with LSTM", Proc. IEEE 16th Int. Conf. Dependable, Autonomic Secure Comput., 16th Int. Conf. Pervasive Intell. Com-put., 4th Int. Conf. Big Data Intell. Comput. Cyber Sci. Technol. Congr., Athens, Greece, 2018, pp. 1074-1079.
[49]W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network" in IEEE Trans. Smart Grid, vol. 10, no. 1, pp. 841-851, 2017.
[50]J. Zhao, F. Deng, Y. Cai, and J. Chen, "Long short-term memory - Fully connected (LSTM-FC) neural network for PM2.5 concentration prediction" in Chemosphere, vol. 220, pp. 486-492, 2019.
[51]S. Kim, J. M. Lee, J. Lee, and J. Seo, "Deep-dust: Predicting concentrations of fine dust in Seoul using LSTM" in ArXiv, vol. abs/1901.10106, 2019.

[52]D. Qin, J. Yu, G. Zou, R. Yong, Q. Zhao, and B. Zhang, “A Novel Combined Prediction Scheme Based on CNN and LSTM for Urban PM2.5 Concentration” in IEEE Access, vol. 7, pp. 20050-20059, 2019.

[53]T. Xayasouk, H. Lee, and G. Lee, “Air Pollution Prediction Using Long Short-Term Memory (LSTM) and Deep Autoencoder (DAE) Models,” in Sustainability, vol. 12, no. 6, pp. 2570, 2020.

[54]Q. Tao, F. Liu, Y. Li, and D. Sidorov, “Air Pollution Forecasting Using a Deep Learning Model Based on 1D Convnets and Bidiectional GRU,” in IEEE Access, vol. 7, pp. 76690-76698, 2019.

[55]I. Verma, R. Ahuja, H. Meisheri, and L. Dey, “Air Pollutant Severity Prediction Using Bi Directional LSTM Network”, IEEE/WIC/ACM International Conference on Web Intelligence (WI), Santiago, 2018, pp. 651-654.

[56]Tong, W., Li, L., Zhou, X. et al., “Deep learning PM2.5 concentrations with bidirectional LSTM RNN” in Air Qual Atmos Health, vol. 12, pp. 411–423, 2019.

[57]Marek Miskowicz, “Send-On-Delta Concept: An Event-Based Data Reporting Strategy” in Sensors, vol. 6, no. 1, pp. 49-63,2006.

[58]V. Nguyen, Y. Suh, “Networked Estimation with an Area-Triggered Transmission Method” in Sensors, vol. 8, no. 2, pp. 897–909, 2008.

[59]Y. Suh, “Send-On-Delta Sensor Data Transmission with A Linear Predictor” in Sensors, vol. 7, no. 4, pp. 537–547, 2007.

[60]M. Schuster and K. K. Paliwal, “Bidirectional recurrent neural networks” in IEEE Transactions on Signal Processing, vol. 45, no. 11, pp. 2673–2681, 1997.

[61]A. Graves and J. Schmidhuber, “Framewise phoneme classification with bidirectional lstm and other neural network architectures,” in Neural Networks, vol. 18, no. 5, pp. 602–610, 2005.

[62]A. Graves, N. Jaitly, and A.-r. Mohamed, “Hybrid speech recognition with deep bidirectional lstm”, Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on. IEEE, 2013, pp. 273–278.

[63]Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, no. 7553, pp. 436–444, 2015.

[64]Hermans, Michiel and Schrauwen, Benjamin,“Training and Analysing Deep Recurrent Neural Networks “, in Advances in Neural Information Processing Systems 26, PP. 190-198, 2013. url = {http://papers.nips.cc/paper/5166-training-and-analysing-deep-recurrent-neural-networks.pdf}}

[65]Pascanu, R, Gulcehre, C, Cho, K &Bengio, Y, “How to construct deep recurrent neural networks”,in Proceedings of the Second International Conference on Learning Representations , ICLR 2014, Banff, Canada.

[66]Jakob Aungiers., https://www.altumintelligence.com/articles/a/Time-Series-Prediction-Using-LSTM-Deep-Neural-Networks. [accessed: 12-02-2020]

[67]H. Choi, K. Cho and Y. Bengio, “Fine-grained attention mechanism for neural machine translation'' in Neuro computing, vol. 284, pp. 171_176, Apr. 2018.

[68]Dzmitry Bahdanau, Kyung Hyun Cho, Yoshua Bengio, “Neural Machine Translation by Jointly Learning to Align and Translate”, Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015), San Diego, CA,USA, May 2015, PP. 7–9.

[69]W. Li, D. Guo, and X. Fang, ``Multimodal architecture for video captioningwith memory networks and an attention mechanism'' in Pattern Recognit. Lett., vol. 105, pp. 23_29, Apr. 2018.

7. PUBLICATIONS

[1] Virendra Barot, Viral Kapadia,“Towards Building a Scalable IoT based System for Carbon Monoxide Monitoring and Forecasting”, *International Journal of Advanced Science and Technology*, vol. 29, no. 3, pp. 5583 - 5590, March 2020.

[Publisher: Science and Engineering Research Support Society, Scopus Indexed, SJR : 0.11]

[2] Virendra Barot, Viral Kapadia, Sharnil Pandya,“QoS enabled IoT based Low Cost Air Quality Monitoring System with Power Consumption Optimization”, Cybernetics and Information Technologies, vol. 20, no. 2, p.p 122-140, June 2020.

[Publisher: Bulgarian Academy of Sciences, ACM digital library, Scopus, Web of Science(ESCI) Indexed, SJR 2019 : 0.31, Quartile 2019 : Q2, Source normalized Impact Factor 2019 : 0.699, Cite Score 2019: 2.1]

[3] V. Barot and V. Kapadia, "Air Quality Monitoring Systems using IoT: A Review," *2020 International Conference on Computational Performance Evaluation (ComPE)*, Shillong, India, 2020, pp. 226-231.

[DBLP, Scopus Indexed, proceeding is included and available in IEEE Xplore digital library]