Place-based Uncertainty Prediction using IoT Devices for a Smart Home Environment

Dr. Amr Jadi
Department of Computer Science and Information
College of Computer Science and Engineering, University of Ha'il, Ha'il, Saudi Arabia

***Abstract*—In this work, an uncertainty prediction method for the home environment is proposed using the IoT devices (sensors) for predicting uncertainties using place-based approach. A neural network (NN) based smart communication system was implemented to test the results obtained from place-based approach using the inputs from sensors linked with internet of things (IoT). In general, there are so many smart systems for home automation is available for alerting the owners using IoT, but they can communicate only after an accident happens. But it is always better to predict a hazard before it happens is very important for a safe home environment due to the presence of kids and pet animals at home in the absence of parents and guardians. Therefore, in this work, the uncertainty prediction component (UPC) using place-based approach helps to make suitable prediction decisions and plays a vital role to predict uncertain events at the smart home environment. A comparison of different classifiers like multi-layer perceptrons (MLP), Bayesian Networks (BN), Support Vector Machines (SVM), and Dynamic Time Warping (DTW) is made to understand the accuracy of the obtained results using the proposed approach. The results obtained in this method shows that place-based approach is providing far better results as compared to the global approach with respect to training and testing time as well. Almost a difference of 10 times is seen with respect to the computing times, which is a good improvement to predict uncertainties at a faster rate.**

***Keywords*—IoT; place-based approach; uncertainty prediction; MLP; SVM; BN; DTW**

I. INTRODUCTION

In recent times, the internet of things (IoT) playing a pivotal role in building a modern society, infrastructure and been instrumental towards the rapid growth of smart cities [1]. The trend of people migrating to urban cities for a better lifestyle made most of the governing authorities to allow improving the growth of smart cities. Therefore, in simple, it is very much evident that major activities by humans, machinery, and administration have a great tracking of life events. Similarly, the possibility of being ignorant/least bothered, as a human tendency towards different homely activities increases due to the excessive involvement of machine-made activities in life. The possibility of these machines to get failed or giving false output response cannot be denied. Especially when it is related to kids at home in the absence of parents/guardians the possibility of risk can be more terrifying and may sometimes lead to death as well. Therefore, it is always suggested to use disruptive technologies (such as artificial intelligence, blockchain, 3D printing, IoT, etc.) carefully especially when it is applied to a home environment [2]. Yes, today’s home environment is completely based on electronic gadgets/machinery and is capable of making things faster and easier for a comfortable human life. They are just one click away from making wonders with their features and quality services but a wrong click also may push you into uncertain situations as well. They need careful/skilful operators for better services and need a careful monitoring system for these device functionalities. Recently many business models are proposed to make secure and smart IoT based home environments by leading real-estate companies to enhance the comfort zone of human life.

A secured smart home was suggested by Yuan and Peng based on IoT by interfacing the web and smart phone applications for improving the user experience [3]. The authors provide a full solution to develop a smart home by using hardware design, intelligent controlling method, pervasive computing and virtual reality within their model. They also claimed to build a reduced power consumption and energy consumption model with a secured mechanism of the terminal gateway group. The health hazards using these smart homes and within the smart cities were addressed by Miori and Russo uses semantic knowledge representation using the web 3.0 [4]. The authors used specific ontology’s that consider the information from distributed environments. An integration approach based on IPV6 enabled service-oriented architecture (SOA) was proposed by Jung et al. to build automation and smart cities [5]. A proof of concept implementation and performance evaluation results is produced out of this work to build an advanced control scenario for the context of smart cities. Piyare used android based smart phones to develop a monitoring and home control system with low cost and flexible mode of operation [6]. Without using a dedicated server this method proposed a novel communication protocol for monitoring and controlling the home environment with all kinds of switching functionalities using mobile devices. The biggest disadvantage of IoT based systems towards achieving security in an IoT environment. A risk analysis was carried out towards using the smart home automation systems by Jacobsson et al. by involving the leading industrial actors [7]. The results from this research indicated reducing the risks by adding standard security features to the existing IoT architectures. The IoT services provided by Vivek and Sunil are in a secured way by using the Wi-Fi ZigBee gateway for home automation [8]. In this work, this gateway includes the user interaction capabilities with an efficient way of sending and receiving the instruction from different protocols and the graphical user interface (GUI) in this work allows the users to get interacted with the settings of the ambient environment. Later, Puri and Nayyar suggested a home automation technique using PIC microcontroller (PIC 16F877A), Bluetooth (HC-05) sensors and android based technology [9]. The Bluetooth sensors used in this method are claimed to be useful for long-range and energy-efficient wireless communications. However, it might not be more appropriate using the Bluetooth for the long-range applications in the urban city environment due to the involvement of heavy noise. The role of wireless sensor networks (WSN) plays a vital role in covering the long ranges and to avoid noise issues in several applications. A secured IoT based smart home automation system was proposed by Pirbhulal et al. used the WSN for operating different home appliances [10]. In this work, for providing suitable energy-efficient data encryption the authors used a triangle-based security algorithm (TBSA), which is based on an efficient key generation mechanism. Using this method, the secured data transmission was possible and the network could cover long ranges as well. An attempt made by Saha et al. to use the advanced IoT based remote system for health monitoring, home automation and an alarm system [11]. From the proposed method a patient will get an alarm to provide the prescribed medicine in the scheduled time by using an email or SMS. An interactive dual-mode IoT based smart home automation was proposed by Hamdan et al. can monitor and control most of the home appliances remotely [12]. These appliances are interfaced with a single chip microcontroller with an in-built wireless access point which establishes the communication with the home server. This system is scalable and can add or remove the devices connected from different rooms based on the demand/ priority.

In this work, neural network (NN) based methods are used to identify and implement for any kind of uncertainty prediction with the help of IoT devices for establishing a smart communication between the owners and prediction component. The place-based uncertainty prediction models are introduced here for improving efficiency, speed, and accuracy. In the next section, detailed information of neural networks functionality as an uncertainty prediction module is presented for the smart home environment.

II. Functioning of Neural Networks

NN works on biological genetic pressures applied to pre-wire forms of the natural neural networks [13]. The first layer is formed with the input nodes followed by, hidden layers and output layers. The nodes and layers of an artificial NN (ANN) represented for a nucleus and axon of a biological NN.

A. Biological Features of Neural Networks

In a biological NN, a neuron will be surrounded with a hair-like (thin) element (i.e. dendrites), which enables the active form of a neuron. They work as input terminals for different sources with certain threshold values. These neurons will burst when the summation of all input signals reachs to maximum levels and the resultant outputs will be carried forward by the axons. These axons will be thicker in size and potentially long as compared to the dendrites, which influence remote neurons that are linked with thousands of other neurons as well. In an artificial NN, the hidden layers will be functioning like the intermediate layers, which receive the input layers and combines them based on the weights of the edges. The calculated output is emitted to the outputs by the subsequent layers, which is considered as a predicted attribute. The NN based systems can be very useful and are very good/faster risk prediction systems in most of the real-time applications such as the healthcare industry, hospitality businesses, stock market detections, etc. [14]. On the other side, the same NN systems can be used as a group for multiple parameter prediction operations such as blood pressure, sugar levels, and heartbeats in the case of the hospital environment. For example, in a home environment the room temperature, humidity, pressure level, smog, windows and door positions, water levels in a tank, electronic switch positions (ON/OFF), status of electronic accessories, automated machines, etc. must be operated within their ideal/defined status levels. These ideal/defined values will be trained to the hidden layers of the neural networks.

B. How a Neural Network Works?

The role of NN is to receive the inputs from the sensor devices that are working based on the fundamentals of internet of things (IoT) and process them along with the ideal values and trained values of the hidden layers of a NN and produce the resultant outputs at output nodes. The output values from the NN system will generate the vector outputs matrix. The conversion of these matrix values will be taking place internally to provide some of the numerical values at the output. These numerical values will be compared by the checker component with various ideal values stored in the database for predicting the uncertainty in the home environment.

To perform the internal operations within the different layers of the NN back propagation algorithm was used to determine the resultant output from the given input signals and the trained values in the hidden layers. In the back-propagation algorithm, the data will be trained to the hidden layers as per the required/expected outputs. The main reason to use this algorithm is due to its nature of being a supervised learning algorithm, which uses multi-layer perceptions for changing the weights of the hidden layers based on the adjustments needed to obtain at the output nodes [13]. This algorithm uses computed output error values to rectify the weights in the backward direction, whereas forward propagation is used to retrieve the total error in this method. Neurons will be activated at the time of forwarding propagation using a sigmoid activation function as given below:

$$f(x) = \frac{1}{\left(1+exp^{-input}\right)} \tag{1}$$

The back-propagation algorithm works with four following steps:

- Using the input patterns, forward propagation is performed to calculate the error output.
- Weight values of the weight matrix will be modified based on the resultant values obtained by using Eq. 1.
- Repeat step 1.

- Process of the algorithm finishes once the output patterns are matched with the target values/patterns.

C. Uncertainty Prediction Component

The role of uncertainty prediction component (UPC) in this work ensures to obtain the information (i.e. ideal values of different parameters from the database) and to compare with the values updated at the input nodes received from sensors by converting them to input vector-matrix form as shown in Fig. 1. The hidden layer weights and random values will be defined by using the back-propagation algorithm and by using some predefined values respectively. The input vector-matrix is compared with the hidden layer weights and the resultant output vector-matrix will be formed to supply the output to the checker component.

D. Functioning of Uncertainty Prediction Component

There are different parameters considered in this work to obtain the sensor data to be analyzed and it will be in the form of analog data. The data collected from the home environment must be converted to a digital and floating form in terms of '0' and '1' because the neural network-based hidden layers will understand the inputs given in digital form only. The weight of the neurons will be equal to '0' or '1' based on the resultant processed values of inputs at nodes and the hidden layer values (i.e. trained values). The vector data set will be converted as a set of a matrix at the input layers from the obtained analog input equivalent values.

The uncertainty prediction component using NN helps to improve the speed of prediction towards any kind of uncertainty in the home environment. The problem detected by the sensing devices can be processed and identification of the risk using the proposed UPC improved the efficiency of the system. Accurate results can be obtained easily using the UPC. In the next section, the detailed architecture of the proposed method for the smart home environment is explained with suitable design components used for IoT. In simple, this method can be called as artificial intelligence (AI) based architecture uses machine learning concepts (such as multi-layer perceptrons (MLP), support vector machines (SVM), Bayesian networks (BN), and dynamic time warping (DWT), etc.) also. These techniques help in computing the inputs of different forms observed by the sensors in the absence of the human helps to predict the uncertainties.

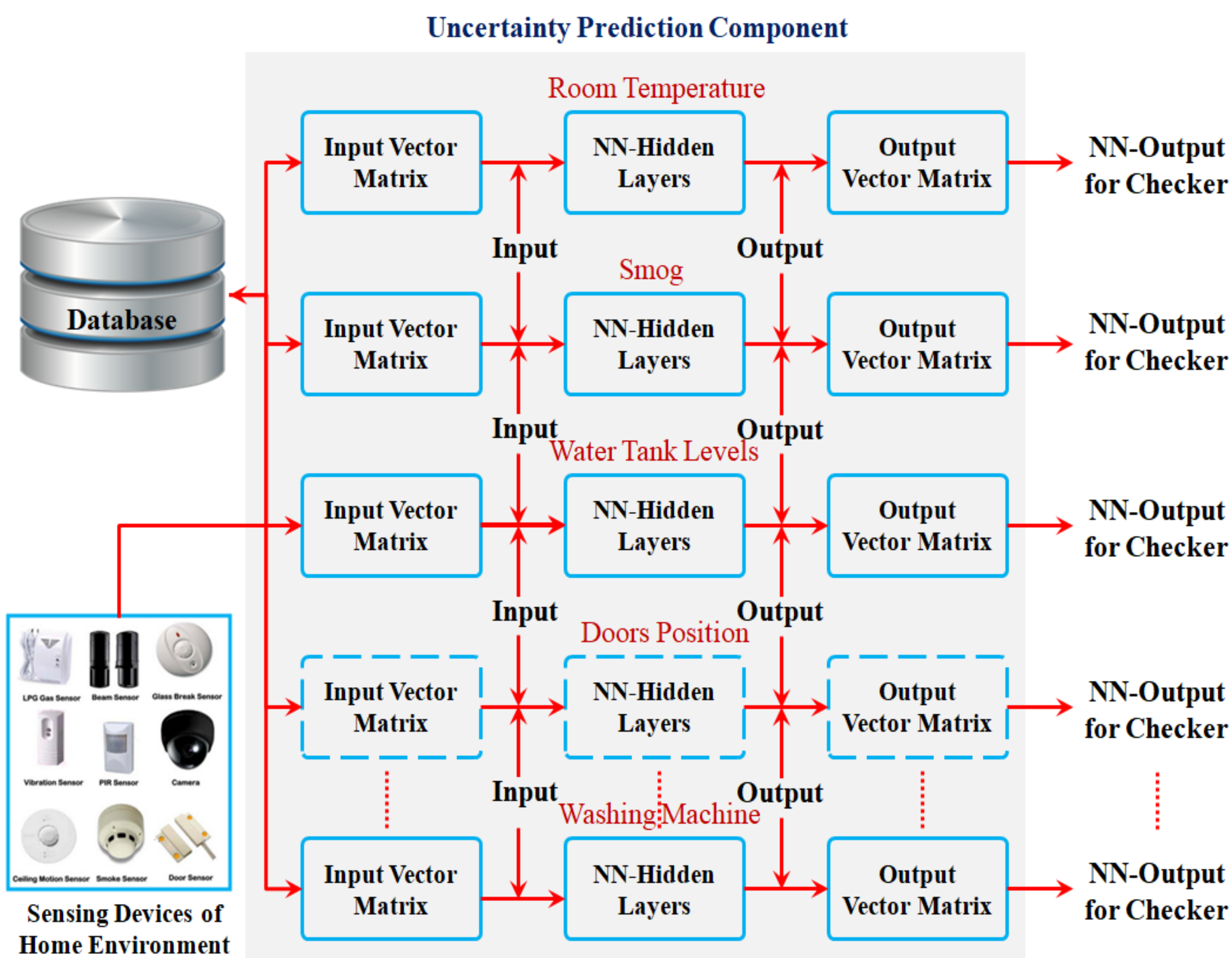

Fig. 1. Different Layers of Neural Network Parameters.

E. The Interaction between the Input Layers

The interaction between different layers is carried out through the hidden layers by using the weights of neurons in terms of '0' and '1'. The UPC provides a continuous assessment between different inputs and hidden layer values using the Sigmoid function and is given by

$$Sigmoid\ (x) = \frac{1}{(1+e^{-x})} \tag{2}$$

Firing rules are very flexible to calculate the timing of firing neurons based on the input patterns. However, it is very important to note that the firing rules applied at input and output stages are different; and the generalization of neurons will be considered at both the ends as they are sensible for random patterns applied at the time of training the hidden layers. Such a flexible property of the firing rules is implemented by using the Hamming distance technique [13]. This flexibility helps UPC to work efficiently for finding multiple and complex natured uncertainty accurately.

This system will help the house owners and the builders to create a peaceful environment in the society by avoiding possible uncertain events in their colonies by predicting the unexpected accidents.

F. Datasets of different Activities in Smart Home Environment

In this work, a dataset is created with daily living activities in smart homes, with well-adapted set of actions for a particular constraint. Five goals have been established which need to be met while recording the dataset.

Goal-1: Record the live home environment data and save it to a hard disk.

Goal-2: Establish and classify the realistic routines of the owners from general public (i.e. visitors).

Goal-3: Use a long-time scale for recording the data.

Goal-4: Ensure to connect more number of smart home sensors to more number of appliances and objects for obtaining more accuracy with the end results.

Goal-5: Now label all four primary contexts with accurate dimension throughout the experiment.

At the time of data collection phase three issues may influence seriously as follows: i) some of the appliances are not used by the owners and may use only few parts of the apartment which are more familiar; ii) some of the common activities performed at home may not be followed by everyone due to different cultural and traditional reasons (for example: five times prayer followed by majority of population in Saudi Arabia, but people visiting from different countries may not follow the same pattern); and iii) some of the activity classes may have few instances during data collection phase, and may influence the training neural network with appropriate information may become a challenging task.

III. Proposed Architecture for Smart Home Environment

In the proposed architecture, there are three main components to address: i) Sensing Devices, ii) Runtime Monitoring Component, and iii) Communication System as shown in Fig. 2. All these components are connected by using various embedded networking devices and IoT devices for establishing a communication interface between internal devices and end-users by using a middleware processing included with neural network systems.

1) Sensing devices: The sensing devices are connected with different types of sensors and their interfacing devices using embedded systems and electronic circuitry. Most of them are operated using microprocessors for different types of real-time applications based on sensors and their controlling operations.

The sensors involved in the smart homes/devices are considered to be the eyes and ears of that environment, which help to inform the owners about any kind of uncertain situation. These devices help to avoid major accidents and reduce the level of damages at home environments [15]. Automated devices control the appliances, events, and activities based on the instructions given by the microprocessors/microcontrollers. They help to monitor and react to different changes in the environment in the absence of human beings.

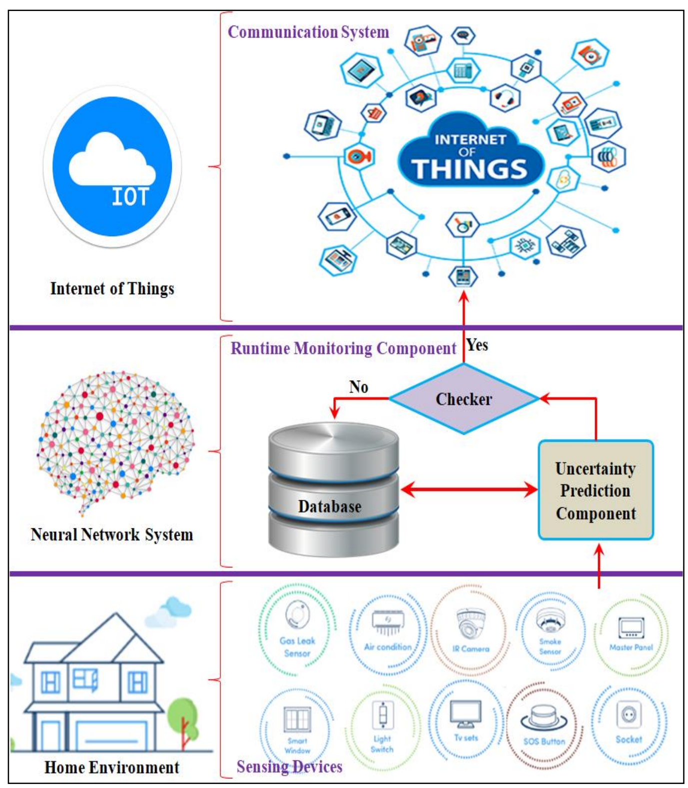

Fig. 2. Proposed NN based Architecture for Smart Home Environment.

2) Runtime monitoring component: The runtime monitoring component in this work includes a database, and checker component along with the uncertainty prediction component, which is completely functioning based on the neural networks-based system. Apart from these activities, the runtime monitoring component also involved with the mitigation process, which helps to assess different scenarios and identify the seriousness of an event taking place in the home environment.

a) Database: It helps to interact with different types of input and output devices to accumulate and store the information. Based on the demand/instructions of the microprocessor, the database helps different components to provide the desired information or store the information by establishing two-way communications. In this work, the database stores the ideal values of different parameters (such as room temperatures, humidity, door positions, motion, water tank levels, gas leakages, etc.) and provide the same to the uncertainty prediction component in the form of a input vector matrix to compare with the latest values at the input nodes of the neural network from the sensing devices. The database design in this work used MySQL application due to simplicity and faster operation as compared to ORACLE or SQL based database designs.

b) Checker: This will check the executed output results of the UPC and ideal values that are stored in the database to identify the environment of the house regularly. If the obtained results are within the controlled range of the ideal values means there is no problem in the home environment. The checker will send an alert signal to the next level (i.e. communication system using IoT devices) if any kind of big difference is observed with the values of UPC and the ideal values stored in the database.

3) Communication system: The communication in this work is established using IoT based system in three ways, i.e. between the targeted objects (various sensors), web servers and the Internet, and different types of social objects (such as mobiles, laptops, internet-based accessories, etc.) [16].

a) Tagged Objects: This category includes RFID, low power sensors, monitoring devices, etc. from the sensing devices to observe the status of the home environment for temperature, water levels, etc. as mentioned above in Fig. 5.

b) Web Servers and Internet: There are so many other technologies for short-range communication using ZigBee, Wi-Fi, etc. [17]. However, they are not capable to communicate by using the low power sensing devices in the IP networks. Therefore, web-based services are needed to get integrated with these sensor nodes for communicating longer distances.

c) Social Objects: Most of the time, data received from such types of monitoring devices will be huge and the relevant executed information will increase exponentially. Therefore, the role of social objects, clouds, etc. plays a vital role in storing the data and using the same effect at the appropriate locations/scenarios. On the other hand, various security issues also can be addressed easily by using such type of social objects, as the services providers will take care of encryption, authentication, and are more cost-effective for the implementation.

To implement this architecture for predicting the uncertainties using NN based runtime monitoring component needs to follow certain rules as discussed in Section-4 and Section-5.

IV. Uncertainty Prediction using the Proposed Method

The list of events taking place using the proposed method are given below as shown in Fig. 3 in a flowchart and some of the highlights are listed below:

- The ideal condition of every device/accessory must be defined by the user in the home environment.
- Based on the inputs the database will help the uncertainty prediction module to generate the appropriate training values for the hidden layers.
- The sensor(s) will try to identify the changes at the home environment regularly and will transfer the same information to the next level, i.e. for UPC and to the runtime monitoring component.
- In the UPC, the weights of the hidden layers will be comparing the input vector matrix and provide the resultant output for the checker component.
- The checker component will compare the ideal values of the database for different specific parameters with the predicted values.
- For any kind of huge gap between the ideal values and the predicted values, an alert will be passed through the IoT section, communicating the owner about the uncertainty.

A. *Uncertain Parameters of a Home Environment*

In this work, the definition of certain and uncertain conditions of the home environment are listed as shown in Table I. As per the guidelines of the US Environmental protection agency, the pollution levels of smog containing CO of 15.5 ppm to 30.4 ppm and NO_2 of 0.65 ppm to 1.24 ppm [18].

In a certain condition, the people living in the home will feel comfortable and it will be safe from any kind of harmful situations. Whereas, in case of uncertain conditions are very much uncomfortable due to the changes in conditions or any kind of unfortunate events such as weather changes, short circuits, etc. For example, the sudden changes in weather increase the humidity in the room and kids might find it uncomfortable at home in the absence of parents.

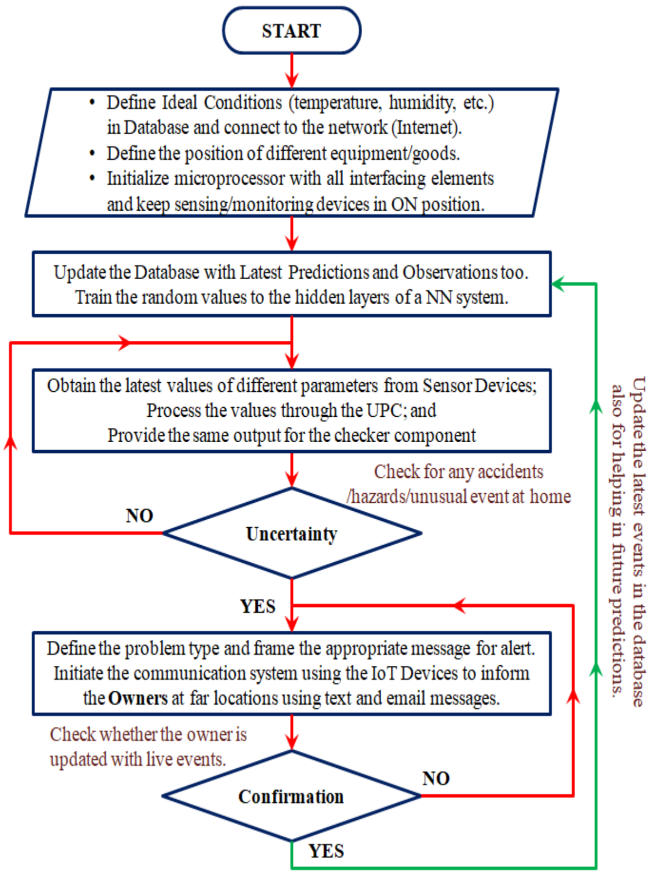

Fig. 3. Flowchart Shows the Events of the Smart Home Environment for Uncertain Events.

TABLE. I. DIFFERENT PARAMETERS WITH THEIR NORMAL AND UNCERTAIN CONDITIONS

Parameters	Certain	Uncertain
Temperature	17 to 35 ^{0}C	Below/Above 35 ^{0}C
Smog	Less than the uncertainty range.	15.5-30.4 ppm for CO & 0.65-1.24 ppm for NO_2
Humidity	30 to 50 percent	Beyond or below the range
Water Levels	Must be around 85%	More than 85%

Similarly, in a smart home, considering human activity recognition (HAR) [19] helps to identify the regular activities performed by the owners and to predict the uncertainties. Some of the common practices followed by the owners at smart home are listed as possible activity classes and are grouped by places in the dataset as follows:

Places	Activity Classes
Entrance	Enter, Exit, Cleaning
Kitchen	Prepare, Cook, Dish Washing, Cleaning
Living Room	Eat, Watch TV, Using Computer, Cleaning
Toilet	Using, Cleaning
Staircase	Going Up, Going Down, Cleaning
Bathroom	Use Sink, Toilet, Shower, Cleaning
Office	Using Computer, Watch TV, Cleaning
Bedroom	Dressing, Read, Nap, Cleaning

It is very complicated task to consider an occupant following accurate timings on daily basis and when it comes to multi-owners it will become a serious and complex issue to analyse the number of instances using HAR. Berlemont *et al.* suggested a method to deal with the problems related with gestures and actions using Siamese NN (SNN), which represents two identical ANNs runs simultaneously on two different types of instances [20]. It is also important to note that the data types in the dataset will change based on the type of sensing device used for a particular activity. Some of the data may be in binary form (such as door closed or opened, i.e. assuming Close with '0' and Open with '1'). Similarly, data can be an integer, real number, and could be categorical too. The neural network may need to learn the number of instances and their frequency during working days and weekends/holidays. For such complicated information the network must be well trained with maximum activities and their instances. For example, using toilet may have few instances but using staircase and computing at home may have more instances during a weekend/holiday as compared to any working day.

V. DISCUSSIONS

Place-based activity recognition is popularly known approach with primary context dimensions included with identity, time, place and activity. These four dimensions are strongly inter-related with each other. For example, an occupant sleeping in the bed room at the night time considers all four activities: occupant (identity), sleeping (activity), bed room (place), and night time (time) covering all four activities. Consider all sensors in the smart home are represented by $S = \{S_1,\ S_2, \dots\ S_n\}$ and all the activity classes are represented by $\boldsymbol{\mathcal{A}}$ and current instances with a, which is a subset of $a \in \boldsymbol{\mathcal{A}}$. Combining all these may be represented by *global approaches*. Now considering i representing a particular place (i.e. a *local approach*) and set of activity classes are represented by $\boldsymbol{\mathcal{A}}^{(i)}$. Let's say $S^{(i)} = \left\{S_1^{(i)},\ S_2^{(i)}, \dots\ S_{n^i}^{(i)}\right\}$ representing the complete set of sensors at i^{th} place of the smart home. Any local approach at the i^{th} place belongs to a classifier needs to get recognized and the current instance at any point is $a \in \boldsymbol{\mathcal{A}}^{(i)} \cup \{none\}$ with the combined data produced by $S^{(i)}$. Here, $none$ is a dummy class and it is very much needed to assign a meaningful label for an instance unless it is a part of $S^{(i)}$. An activity recognition approach is proposed in this work by learning local models for each instance, place and active classes using the following steps: First, use the localization algorithm to locate an occupant at i^{th} place; and then to recognize the occupant activity at i^{th} place using the local model. Now by considering set of all places of a home by $\mathcal{P}$ and the cardinal of this set is represented by $|\mathcal{P}|$, respective set of decisions are computed for $\left\{\Delta^{(1)}, \Delta^{(2)} \dots \Delta^{(\mathcal{P})}\right\}$.

here,

$$\Delta^{(i)} = \left\{\delta_{1,1}^{(i)}, \dots, \delta_{1,|A^{(1)}|}^{(i)}, \dots, \delta_{|\mathcal{P}|,1}^{(i)}, \dots, \delta_{|\mathcal{P}|,\boldsymbol{\mathcal{A}}^{(|\mathcal{P}|)}}^{(i)}, \delta_{none}^{(i)}\right\} \qquad (3)$$

and $\delta_{k,j}^{(i)} \in [0,1]$ represents degree of membership with the decision of a classifier at i^{th} place of j^{th} activity class and k^{th}

place. In the proposed approach the local models will learn to recognize different activity classes which occurs at respective places, i.e. if $i \neq k$ gives $\delta_{k,j}^{(i)} = 0$. Here the NN needs to make a decision and the decision fusion step is to compute set of fused decision ($\bar{\Delta}$) is given by

$$\bar{\Delta} = \left\{\bar{\delta}_1^{(i)}, \ldots, \bar{\delta}_{|A^{(1)}|}^{(1)}, \ldots, \bar{\delta}_1^{(|\mathcal{P}|)}, \ldots, \bar{\delta}_{\mathcal{A}^{(|\mathcal{P}|)}}^{(|\mathcal{P}|)}, \bar{\delta}_{none}\right\} \tag{4}$$

Now from each set of decisions $\{\Delta^{(1)}, \ldots, \Delta^{(\mathcal{P})}\}$ each place is considered from a smart home and from $\bar{\Delta}$ a conclusion is made to identify the class of a particular instance at a^{th} activity from p^{th} place when a class is having maximum decisions in $\bar{\Delta}$. Therefore,

$$\bar{\delta}_a^{(p)} = \max\left(\begin{matrix} max \\ j,k \end{matrix}\ \bar{\delta}_j^{(k)}, \bar{\delta}_{none}\right) \tag{5}$$

From the above Eq. (5), a given instance will be related to only one activity when the similar activities are occurring throughout the home simultaneously. By using maximum values of $\bar{\Delta}$, modification of decision process for multiple activity labels is possible. The proposed place-based activity recognition scheme is shown in Fig. 4.

There is a requirement of expert knowledge for setting up entire process in a smart home, and need to define the membership of each sensor with respective place is essential. Sensors are generally installed closely with each other and the possibility of discovering same places by the sensors is more. To tackle/address this problem, clustering methods are implemented. The same data collected by nearest sensors helps to find correlated data. It is necessary to standardize the sensor data in a way to have 0 mean and a variance of 1. The training dataset is given by considering mean as $\bar{s}$ and standard deviation of outputted values the sensor with σ as

$$s_t' = \frac{s_t - \bar{s}}{\sigma} \tag{6}$$

Also considering the noise parameters in the collected data, it is essential to ensure a noise reduction process is being carried out. A basic filtering method is used to control the noise and it is controlled by $\beta\ \epsilon\ [0,1]$ is given by

$$s_t' = \beta s_t + (1 - \beta) s_{t-1} \tag{7}$$

Here maximum filtering process is considered when $\beta = 0$; minimum filtering also considered when $\beta = 1$ when $s_t' = s_t$ for any value of t. The basic method applied over raw sensor data is shown in Fig. 5 with the resultant filtered output, where the original signal shape is conserved by softening all the noisy oscillations.

A. *Classical Classifiers*

There are several classifiers which can be used for predicting the uncertainties in a smart home environment. These classifiers included with multi-layer perceptrons (MLP) of ANN, Bayesian networks (BN) used for probabilistic graphical modelling, support vector machines (SVM) are used for different types of kernel methods, dynamic time warping (DTW) is used for measuring the geometric similarities, and hidden Markov models (HMM).

MLP uses the feed forward techniques of ANN with no loops and in this method the output is connected with the input of another neuron at the next stage or layer. By considering $n_i^{(j)}$ as i^{th} neuron of j layer with the inputs from a set of outputs neurons of N_{j-1} from previous layers as $\left\{y_i^{(j-1)}, \ldots, y_{N_{j-1}}^{(j-1)}\right\}$gives an output say $y_i^{(j)}$. This output can be computed by using the following equation:

$$y_i^{(j)} = \varphi\left(b_i^{(j)} + \sum_{k=1}^{N_{j-1}} w_{k,i}^{(j-1)}, y_k^{(j-1)}\right) \tag{8}$$

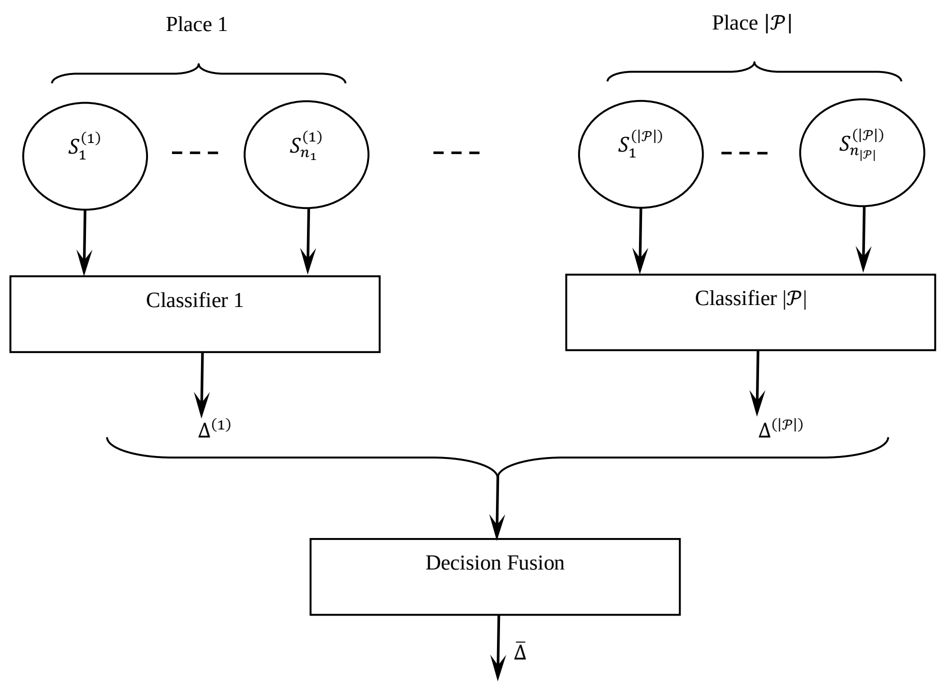

Fig. 4. Shows the Place-based Activity Recognition Scheme.

here, $b_i^{(j)}$ is the bias neuron, $w_{k,i}^{(j-1)}$ is the weight between two neurons (i.e. between k and $j-1$; and i and j), and finally activation function of neuron is represented by φ, which is equal to sigmoid function as listed in Eq. 2.

SVMs are the generalization of linear classifiers with two classes. For a vector of the input data $x = (x_1, \dots, x_N)$ a constructing linear classifier is not possible with the help of vector weights $w = (w_1, \dots, w_N)$. Therefore, the output of SVMs can be given by

$$y = wx^T + w_N \tag{9}$$

If the value of $y \geq 1$ it is classified in class 1, and if $y \leq -1$ then it is classified in class 2.

BN is considered as a directed acyclic graph representing joint probability distribution of different variables forming as a set. For a five variable inputs of $\boldsymbol{x} = (x_1, \dots, x_5)$ the BN is represented by different conditional dependencies as shown below:

$$p(x_1)p(x_2|x_1)p(x_3|x_1)p(x_4)p(x_5|x_2, x_3, x_4) \tag{10}$$

For the purpose of activity classification, any one variable of BN represents the activity class and remaining variables represents the obtained sensor data. For example, in the following Eq. 10 activity class is represented by x_1 and sensor data is represented by x_2, x_3, x_4, x_5.

$$\underset{x_1}{argmax}\ p(x_1|x_2, x_3, x_4, x_5) \tag{11}$$

There are many approaches existing to learn from the trained data for the BN structures and conditional probabilities of BN. There are some complex learning problems such as structure learning as compared to distribution learning.

In case of DTW, values of two time-dependent sequences are compared in a geometric approach. For example, assume values of two sensors as x_t and $y_{t'}$ with a time steps of $t \in [\![1, T]\!]$ and $t' \in [\![1, T']\!]$. To compare the sensor values it is necessary to conduct cost measurement with $c: [\![1, T]\!] \times [\![1, T']\!] \rightarrow [0, +\infty]$. Generally, for *similar* values of x_t and $y_{t'}$ output of cost function will be smaller values; and for *dissimilar* values of x_t and $y_{t'}$ output of cost function will have larger values. The warping path is defined as a sequence $p = (p_1, \dots, p_n)$ where the value of $p_i = (t_i, t_{i'}) \in [\![1, T]\!] \times [\![1, T']\!]$. This warping path need to consider and respect two conditions: boundary conditions ($p_1 = (1,1)\ and\ p_n = (T, T')$); and step size conditions $(\forall_i [\![1, n-1]\!], p_{i+1} - p_i \in \{(1,0), (0,1), (1,1)\})$. The advantage of boundary conditions is to ensure both start and end points are aligned together. Whereas step size conditions enforces all elements from both the sequences are part of warping path for at least once with none of the duplicate pairs in p. The later step also enforces monotonicity of the path (i.e. $t_i + t_{i+1}$ and $t'_i \leq t'_{i+1}$. Finally, the cost c_p of warping path p is given by

$$c_p(x, y) = \sum_{i=1}^{n} c\left(x_{t_i}, y_{t'_i}\right) \tag{12}$$

Therefore, DTW distance between two sequences is representing the overall cost of optimal warping path, which is smallest of all possible costs among different warping paths.

B. Experiment Results

At the time of processing data missing values are replaced with cubic spline interpolation and a low-pass filter (with $\beta = 0.1$) was used for noise reduction. Similarly, care was taken to ensure the value of mean is '0' and standard deviation of the sensors is equal to 1. Sensor values as input vectors are converted to feature vectors by resampling each instance with 20 time steps. WEKA library was used to implement the MLPs, SVMs, and BNs in this experiment [21].

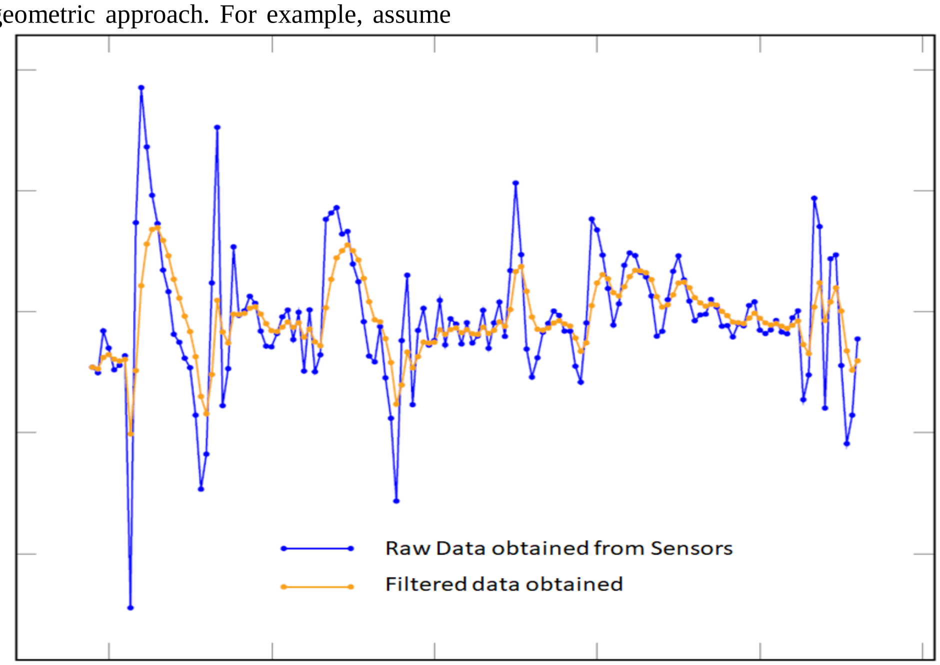

Fig. 5. Shows the Filtered Data with β = 0.3 after Compiling the Same with Raw Data Obtained from the Sensors.

Activity recognition performance of this work is compared with place-based approach with the global approach by using different types of classifier types and decision fusion methods. It is also important to note that most of the classes are not frequent in any smart home environment and the frequency may vary from class to class. Weighted F_1 score is used to compare the performances at different levels, rather comparing only the accuracy part of it. Performance measure helps to provide equal weights for all classes, irrespective of the total number of instances they consists off. For example, consider a set with activity instances (χ_a) with true label a for all activity classes $a \in \boldsymbol{\mathcal{A}}$, the F_1 score is calculated as shown below:

$$F_1 = \sum_{a \in \mathcal{A}} \frac{2}{|\chi_a|} \cdot \frac{Precision(a).Recall\ (a)}{Precision(a)+Recall\ (a)} \tag{13}$$

here, the Precision (a) is a ratio of number of correctly classified instances as a to the total number of classified instances as a. Similarly, Recall (a) is the ratio of number of correctly classified instances as a to the total number of instances of class a.

In the opportunity dataset there are about 80 classes of actions are labelled in this work and an additional class with label 'None' is placed considering a case when no action is performed. In this place-based approach the sensors are completely based on the locations of action classes as mentioned in the dataset in Section 4. In this work, five places (as shown in Fig. 6) are identified for experimental environment where recording of Opportunity dataset is possible: in the Kitchen, Tables, Bed Rooms, Exits, and TV's. All the sensors and action classes are distributed among these 5 places based on the number of classes. For example, in a kitchen there are 10 drawers (Open and Close), Dishwasher (ON and OFF), Lights (ON and OFF), Fridge (Open and Close), Stove (ON and OFF), and Windows (Open and Close) gives a total 30 action classes. Similarly, for three bedrooms there are 18 action classes by considering light, fan and air condition (AC) as ON/OFF. On the table, it is considered to have 12 action classed with the in-built sensors to some of the objects placed on a table. There are four TV's with 8 action classes and six exit (E_1 to E_6) gives a total of 12 action classes.

The evaluation process in this approach used 10-fold random cross-validation. The five sets of instances were used to train the neural network. The performance of this approach was evaluated by using the test set which never existed before the training the hidden layers. For selecting parameterization of the classifiers/decision fusion validation set was used. Evaluation of validation set took at first place with the learned models and this helps to avoid any kind of bias at the time of evaluation phase. F1 scores of five different places using the four classifiers are given in Table II from the Opportunity dataset.

Fig. 6. Shows the Five Places where the Opportunity Dataset was Recorded from different Sensors from the Smart House Environment: Table, Bedrooms, Kitchen, Exits and TV's.

From the Table II, it can be seen that the performances are depending on different types of classifiers. Performance using DTW shows poor results as compared BN at the place Kitchen. The observations also show a huge gap between some of the classification performances at some places. Overall recognizing actions from Kitchen seem more complicated as compared to other classifiers. Now in a process of comparing place-based and global approaches, all the places modelled by using same classifier as compared to the global approach in which all classifier types were used for considering a global model action from Opportunity dataset. From the results it can be seen that the place-based approach is far better than the global approach as shown in Table III.

The gap between these two approaches is not significant statistically because the F1 score deviation of the classifiers has a big gap when it is observed in MLP and DTW from the Table III. Also note that the standard deviation of all classifiers is small for place-based approach as compared with global approach, which is due to decision fusion step that tends to go for average out of the overall results.

In this work, anticipation is also made to calculate the computing times (see Table IV) as a benefit of place-based approach. The time taken to run a training phase and activity recognition is observed to be faster in place-based approach when compared with global approach. To calculate the computational times, it is necessary to use a high-end system configuration consisting of high frequency of operation and huge RAM. Here, the computing times are considered for only three classifiers (MLP, SVM, and BN; and ignored DTW since it is very slow as compared to other three classifier types as seen from in Table II) along with global approach over the Opportunity dataset. Whereas in case of decision fusion step it was being ignored due to very slow computing times and can be neglected as compared to any computing times.

TABLE. II. F1 SCORES FOR DIFFERENT CLASSIFIERS FROM FIVE PLACES IN OPPORTUNITY DATASET

	Place				
Classifier	**Kitchen**	**Table**	**Exits**	**Bedrooms**	**TV's**
MLP	**94.08%±1.67%**[1]	**98.57%±0.43%**[1]	99.12%±0.43%[1]	**96.62%±1.43%**[1]	**99.02%±1.03%**[1]
SVM	93.87%±1.23%[2]	98.47%±1.63%[2]	**99.34%±0.36%**[2]	95.71%±1.53%[2]	98.87%±1.73%[2]
BN	91.54%±1.29%	98.37%±0.43%	98.91%±0.53%	94.92%±1.23%	98.59%±1.68%
DTW	84.67%±1.13%	98.23%±0.37%	98.76%±0.93%	94.33%±1.58%	98.42%±1.36%
Parameters	[1]100 hidden neurons, 120 epochs, 0.2 learning rate, and 0.1 momentum [2]$C = 1000$, $\gamma = 0.01$				

TABLE. III. F1 SCORES FOR DIFFERENT CLASSIFIERS USING GLOBAL APPROACH AND/OR PLACE-BASED APPROACHES FROM THE OPPORTUNITY DATASET

	Classifier			
Approach	**MLP**	**SVM**	**BN**	**DTW**
Global	90.11%±1.57%[1]	90.12%±1.03%[2]	**90.72%±1.43%**[1]	79.62%±2.43%[3]
Place-based	**93.42%±1.36%**[3]	**91.23%±1.25%**[4]	89.36%±1.83%[5]	**84.33%±1.38%**[6]
Parameters used for global approach [1]100 hidden neurons, 120 epochs, 0.2 learning rate, and 0.1 momentum [2]$C = 1000$, $\gamma = 0.003$ ***Decision fusion used for place-based approach*** [3]SVM stacking using $C = 100$, $\gamma = 0.01$ [4]MLPstacking using 120 hidden neurons, 120 epochs, 0.2 learning rate, and 0.1 momentum [5]MLPstacking using 60 hidden neurons, 120 epochs, 0.2 learning rate, and 0.1 momentum [6]SVM stacking using $C = 20$, $\gamma = 0.05$				

TABLE. IV. AVERAGE COMPUTING TIMES FOR DIFFERENT CLASSIFIERS FROM FIVE PLACES IN OPPORTUNITY DATASET ALONG WITH GLOBAL APPROACH

		Place					
Classifier	**Phase**	**Kitchen**	**Table**	**Exits**	**Bedrooms**	**TV's**	**Global**
MLP	Training Test	956.36±159.32 13.68±1.32	714.11±110.05 14.68±1.67	689.85±19.92 11.24±1.54	547.96±95.69 10.98±1.27	521.63±58.25 9.58±1.25	15,586.36±1599.3 26.98±1.68
SVM	Training Test	29.65±0.96 7.58±0.08	24.15±0.32 9.69±0.18	19.64±0.32 7. 81±0.29	18.06±0.72 9.58±0.42	16.38±0.32 8.09±0.95	39.68±0.75 28.18±0.58
BN	Training Test	21.37±0.39 9.68±0.17	15.61±0.85 8.81±0.45	14.08±0.38 7.31±0.21	11.58±0.24 6.85±0.37	9.24±0.75 6.28±0.11	29.67±0.62 13.18±0.01
DTW	Training Test	0 4215.81±283.72	0 3512.90±224.29	0 3213.61±191.76	0 2973.58±147.57	0 2513.84±1381.48	0 7214.06±389.24
Computing Time in *Seconds*							

Considering the proposed place-based approach executed with multi-core computer processor, training phase can be parallelized. That means all places can be computed simultaneously by assuming decision fusion takes negligible time. Due to this approach the overall computing times of proposed place-based approach is shorter as compared with the global approach for both training and testing phases. For example, considering the MLP for any place as compared with the global approach the training time and testing times are shorter in place-based approach (as 15,586.36 seconds is bigger than any values of MLP classifier).

VI. Conclusions

In this work, a unique method to predict the uncertain events at the home environment is proposed using a neural network-based system. As compared to the earlier home automation systems, the present method can detect the possible uncertain situations and hazards in an efficient way to improve the safety measures for the people living in the home from any kind of serious life and property damage. There is a lot of improvement in monitoring the events and activities using the sensors and monitoring devices effectively adding the risk prediction component. Both of them working simultaneously and the UPC component will try to judge and mitigate the events based on the previous data also in the absence of the immediate present data when there is a problem with sensors, such as power failures, technical glitches, etc. The place-based approach results proved to be very much useful for the prediction of uncertainties in the smart home environment. Four classifiers were used to examine the performance of proposed UPC and MLP proved to be more accurate in terms or predicting uncertainties. Place-based approach delivered better results as compared with the global approach and the training and testing times required by both the approaches also shown similar results.

Acknowledgment

The authors gratefully acknowledge the support and facilities provided by the Management and Department of Computer Science and Information, University of Hail, Hail, Saudi Arabia.

References

[1] E. Park, A. del Pobil, and S. Kwon. "The role of internet of things (IoT) in smart cities: Technology roadmap-oriented approaches." Sustainability 10, no. 5 (2018): 1388.

[2] A. Reyna, C. Martín, J. Chen, E. Soler, and M. Díaz. "On blockchain and its integration with IoT. Challenges and opportunities." Future Generation Computer Systems 88 (2018): 173-190.

[3] X. Yuan, and S. Peng. "A research on secure smart home based on the internet of things." In 2012 IEEE International Conference on Information Science and Technology, pp. 737-740. IEEE, 2012.

[4] V. Miori, and D. Russo. "Anticipating health hazards through an ontology-based, IoT domotic environment." In 2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, pp. 745-750. IEEE, 2012.

[5] M. Jung, J. Weidinger, W. Kastner, and A. Olivieri. "Building automation and smart cities: An integration approach based on a service-oriented architecture." In 2013 27th International Conference on Advanced Information Networking and Applications Workshops, pp. 1361-1367. IEEE, 2013.

[6] R. Piyare. "Internet of things: ubiquitous home control and monitoring system using android based smart phone." International journal of Internet of Things 2, no. 1 (2013): 5-11.

[7] A. Jacobsson, M. Boldt, and B. Carlsson. "On the risk exposure of smart home automation systems." In 2014 International Conference on Future Internet of Things and Cloud, pp. 183-190. IEEE, 2014.

[8] G. V. Vivek, and M. P. Sunil. "Enabling IOT services using WIFI-ZigBee gateway for a home automation system." In 2015 IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), pp. 77-80. IEEE, 2015.

[9] V. Puri, and A. Nayyar. "Real time smart home automation based on PIC microcontroller, Bluetooth and Android technology." In 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), pp. 1478-1484. IEEE, 2016.

[10] S. Pirbhulal, H. Zhang, M. E Alahi, H. Ghayvat, S. Mukhopadhyay, Y. T. Zhang, and Wanqing Wu. "A novel secure IoT-based smart home automation system using a wireless sensor network." Sensors17, no. 1 (2017): 69.

[11] J. Saha, A. K. Saha, A. Chatterjee, S. Agrawal, A. Saha, A. Kar, and H. N. Saha. "Advanced IOT based combined remote health monitoring, home automation and alarm system." In 2018 IEEE 8th annual computing and communication workshop and conference (CCWC), pp. 602-606. IEEE, 2018.

[12] O. Hamdan, H. Shanableh, I. Zaki, A. R. Al-Ali, and T. Shanableh. "IoT-based interactive dual mode smart home automation." In 2019 IEEE International Conference on Consumer Electronics (ICCE), pp. 1-2. IEEE, 2019.

[13] A. Jadi. "An Early Warning System for Risk Management." Ph.D. Thesis. Software Technology Research Laboratory, DeMontfort University, 2013.

[14] A. Jadi, H. Zedan, and T. Alghamdi. "Risk management based early warning system for healthcare industry." In 2013 International Conference on Computer Medical Applications (ICCMA), pp. 1-6. IEEE, 2013.

[15] M. Yerragolla, K. Pallela, and I. P. Gera. "Intelligent security system for residential and industrial automation." In 2016 IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics Engineering (UPCON), pp. 229-234. IEEE, 2016.

[16] A. Singh, D. Kumar, and J. Hötzel. "IoT Based information and communication system for enhancing underground mines safety and productivity: Genesis, taxonomy and open issues." Ad Hoc Networks 78 (2018): 115-129.

[17] S. Al-Sarawi, M. Anbar, K. Alieyan, and M. Alzubaidi. "Internet of Things (IoT) communication protocols." In 2017 8th International conference on information technology (ICIT), pp. 685-690. IEEE, 2017.

[18] T. Fitz-Simons. Guideline for reporting of daily air quality: Air Quality Index (AQI). No. PB-99-169237/XAB; EPA-454/R-99/010. Environmental Protection Agency, Office of Air Quality Planning and Standards, Research Triangle Park, NC (United States), 1999.

[19] J. Cumin. "Recognizing and predicting activities in smart homes." PhD diss., 2018.

[20] S. Berlemont, G. Lefebvre, S. Duffner, and C. Garcia. "Class-balanced siamese neural networks." Neurocomputing 273 (2018): 47-56.

[21] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. "The WEKA data mining software: an update." ACM SIGKDD explorations newsletter 11, no. 1 (2009): 10-18.