INTERNATIONAL JOURNAL OF APPLIED METHODS IN ELECTRONICS AND COMPUTERS

IJAMEC

e-ISSN: 3023-4409

www.ijamec.org

International Open Access

Volume 11
Issue 04

December, 2023

Research Article

Feedforward Neural Network-Based Indoor Air Quality Detection System

***Cihat Ediz Akbaba* [a, *], *Gökay Dişken* [b]**

[a]*Adana Alparslan Türkeş Science and Technology University, Department of Electrical and Electronics Engineering, Adana, Türkiye*
[b]*Adana Alparslan Türkeş Science and Technology University, Department of Artificial Intelligence Engineering, Adana, Türkiye*

ARTICLE INFO

Article history:
Received 18 May 2023
Accepted 6 November 2023

Keywords:
Indoor air quality
Internet of things
Smart home systems
IoT-based air quality
Air pollution monitoring

ABSTRACT

Indoor air quality is crucial for the sustainability of human life quality. Therefore, improving indoor air quality is critical for enhancing life quality. In this study, an artificial intelligence-based indoor air quality monitoring system is designed. The system consists of two main parts, hardware and software. The hardware part includes a control card and various sensors. The software part includes a C-based IDE software and a feedforward network, a deep learning algorithm, for establishing the connection between the control card and the sensors. The temperature, humidity, and gas concentration values obtained from the sensors at certain intervals were fed to the feedforward network's input layer through the control card. The feedforward network consists of the input layer, hidden layer, and output layer, and the decision on whether the air quality is normal or not was made at the output. The system described in this study is intended to provide a real-time monitoring solution for indoor air quality. By using a feedforward neural network, the system is able to learn patterns in the sensor data and detect changes in air quality that may indicate a problem. The system can be customized to suit the needs of different environments and can be used in a variety of settings, including homes, offices, and public spaces. Ultimately, the goal of the system is to improve human health and well-being by ensuring that indoor air quality is at a safe and healthy level.

1. Introduction

One of the biggest problems of our time is air pollution, which ranks second in terms of deaths caused by non-communicable health problems [1]. Poor air quality indoors poses a significant danger to human health, causing millions of deaths annually as a result [2]. Heating, cooling, ventilation, and indoor climate control systems used indoors can significantly contribute to indoor air pollution, which can be up to five times worse than outdoor pollution [3]. Monitoring indoor air quality and creating IoT-based decision-making systems can provide information on activities such as cooking, heating, disinfectant use, and indoor ventilation, which can benefit people with respiratory and similar illnesses [4].

In this work, a Feedforward Neural Network-based indoor air quality detection system is designed to detect the air quality in indoor environments using an artificial intelligence algorithm. This system analyzes real-time information related to indoor air quality by utilizing data collected through various sensors. Such a system measures various air quality factors such as gas particles, temperature, humidity, and CO2 levels present in the air. The measured data is used to trained a neural network algorithm to create a series of analytical models that determine the air quality status. Furthermore, the system automatically alerts individuals when the air quality drops below a certain level, allowing them to be informed about air quality problems. In this way, individuals can minimize health problems by ensuring a healthier indoor air quality. In conclusion, the deep learning-based indoor air quality detection system is a technology that helps individuals live in a healthier indoor environment.

Based on the research, the optimal temperature range for a comfortable living space is 19-22 degrees Celsius, while the humidity level should fall within the range of 40% to 60% [5, 6]. The Air Quality Index (AQI) is an important

* Corresponding author. E-mail address: *edizakbaba@gmail.com*
DOI: 10.58190/ijamec.2023.64

parameter for detecting indoor air quality, and the level of indoor air quality according to AQI based on certain numerical values is shown in table 1 [7, 8]. As part of our study, we utilized the gas concentration values obtained from the MQ-135 sensor to calculate the Air Quality Index (AQI). In this calculation, values ranging from 0 to 100 in ppm were deemed to represent a satisfactory indoor level.

Table 1. AQI and health meaning

AQI	Air quality level
0-50	good
51-100	moderate
101-150	unhealthy for sensitive groups
151-200	unhealthy
201-300	very unhealthy
301-500	hazardous

2. Materials and methods

The general structure of the proposed system is shown in Figure 1. The system works as follows; The control card will receive the ambient temperature value from the DHT11 temperature and humidity sensor, and will receive the ambient gas concentration value from the MQ-135 sensor. If the temperature value is outside the predetermined range, the air conditioner will turn on, and if the gas concentration value is outside the predetermined range, the alarm system will turn on. The temperature, humidity, and gas concentration values are recorded in the database, and also transmitted to the neural network. Each neuron in the hidden layer receives inputs from the input layer, multiplies them by weights, and calculates the outputs by passing them through the ReLU function, which is an activation function. Finally, the output layer processes the outputs from the hidden layer to produce the final output of the network. Based on the output of the network, the audible warning system connected to the microcontroller output will alert the occupants of the environment about the normality of indoor air quality. The circuit diagram of the system is given in Figure 2.

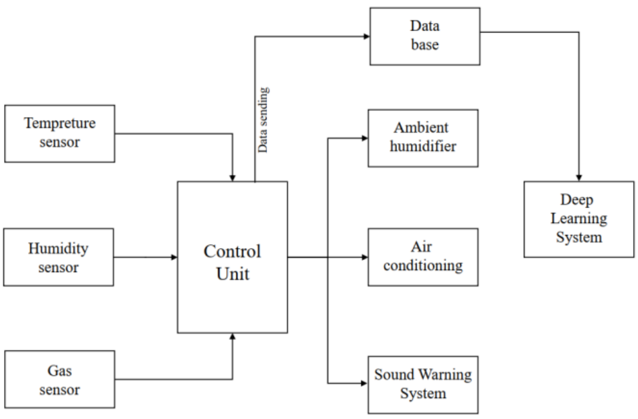

Figure 1. General structure of the system.

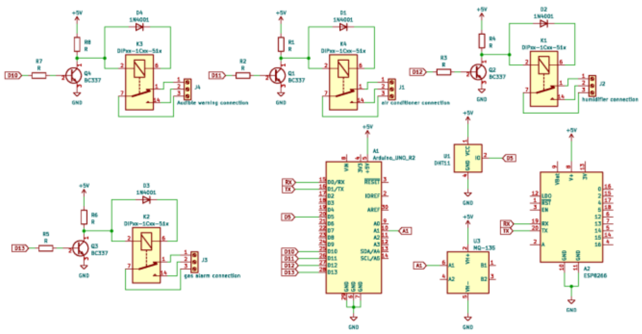

Figure 2. System electronic circuit diagram. The microcontroller card output is connected to four relays, each of which controls a different device. Specifically, K1 relay regulates the humidifier, K2 relay manages the gas alarm, K3 relay controls the audible warning, and K4 relay governs the air conditioner. According to the output of the neural network, the audible warning system linked to the K3 relay alerts users to whether the indoor air quality is normal or not.

The hardware components used in the system are shown in Figure 3. Figure 3a [9] displays the Arduino Uno control board, which is a low-cost and high-speed board that uses the Atmega328p microcontroller. It has 14 digital pins and 6 analog pins. The DHT11 sensor, shown in Figure 3b, is a low-cost and highly efficient sensor that measures temperature and humidity and can exchange data with microcontrollers such as Arduino and Raspberry Pi [10]. Figure 3b shows the pin structure of the DHT11 sensor [11]. Figure 3c [12] shows the ESP8266 wireless sensor module, which uses the TCP/IP protocol, has an internal antenna, and can monitor information from peripheral devices independently over the internet. The MQ-135 sensor is another low-cost sensor that provides both analog and digital output and can measure the concentration of gases such as NH3, NOx, alcohol vapor, benzene, smoke, and CO2 with a low response time [13]. To measure gas concentration in parts per million (PPM) using the MQ-135 sensor, analog pins should be used [14]. The analog information obtained using the ESP8266 wireless module will be sent to the database and used in the deep learning system for training with the Python programming language.

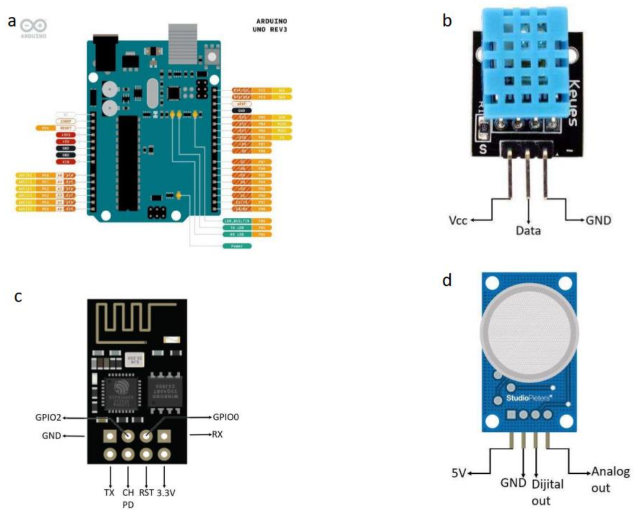

Figure 3. The electronic components used in the circuit are: a) Arduino UNO microcontroller, b) DHT11 temperature sensor, c) ESP8266 WiFi module, and d) MQ-135 gas sensor.

Technical details of the Arduino Uno control board, DHT11 sensor, ESP8266 WiFi module, and MQ-135 gas sensor are included in Table 2 [9, 15, 16].

Table 2. Technical details of the components

	Arduino Uno	DHT-11	ESP8266	MQ-135
Operating Voltage	5 V	3-5V	-	5V
Input Voltage	6-20 V	-	-	-
MCU	-	-	Xtensa	-
Digital I/O Pins	14	-	-	-
Detection / Measurement	-	-	-	Various Gasos
DC Current on I/O Pins	40 mA	-	-	-
SRAM		-	160 kB	-
DC Current on 3.3 V Pin	50 mA	-	-	-
Preheat Time	-	-	-	20 second
Flash Memory	21 KB			-
Frequency 80 MHz		-	Frequency 80 MHz	-
SRAM	2 KB	-	-	
EEPROM	1 KB		-	-
Analog output voltage	-	-	-	0-5V
Digital output voltage	-	-	-	0-5V (TTL Logic)
Clock Speak	16 MHz	-	-	-
Temperature Range	-	0-50 °C / ± 2 °C	-	-
Humidity Range	-	20-80% / ± 5%	-	-
Sampling Rate	-	1 Hz	-	-
Body Size	-	15.5x 12x 5.5mm	-	-
Max Current During	-	2.5 mA	-	-
Wi-Fi	-	-	802.11 b/g/n	-
Flash	-	-	SPI Flash, up to 16 MB	-
ADC	-	-	10 bit	-
Analog Input Pins	5	-	-	-
SPI/I2C/I2S/UART	-	-	2/1/2/2	-

The system software consists of two Arduino Integrated Development Environment (IDE) programs based on the C/C++ language and a Python programming language used for deep learning operations. The Arduino IDE program is the official software introduced by Arduino.cc for loading codes onto Arduino control boards, compiling, and editing them [17].

Table 3. Analog values obtained from sensors on Day 1, Day 2 and Day 3 for 5 time points

	Time point-1	Time point-2	Time point-3	Time point-4	Time point-5
Day 1 Temp. (℃)	34	29	22	18	9
Day 1-Humidity(%)	63	45	36	29	22
Day 1-Air quality(ppm)	184	140	120	97	71
Day 2-Temp. (℃)	32	30	22	19	10
Day 2-Humidity(%)	62	44	37	32	25
Day 2-Air quality(ppm)	190	170	135	120	76
Day 3-Temp. (℃)	32	30	35	36	12
Day 3-Humidity(%)	60	48	38	40	30
Day 3-Air quality(ppm)	160	150	130	122	125
Day 4-Temp. (℃)	38	32	36	25	12
Day 4-Humidity(%)	60	62	38	35	36
Day 4-Air quality(ppm)	140	156	128	100	85
Day 5-Temp. (℃)	36	32	30	25	18
Day 5-Humidity(%)	62	60	48	45	44
Day 5-Air quality(ppm)	185	136	125	102	85

We used temperature, humidity, and gas ratio values as input variables for the feedforward model that was generated when inputs were sent to the system for 5 days and 5 different time points. These three input variables contained 5 different values for each time point (Time point-1, Time point-2, Time point-3, Time point-4, Time point-5).

To create our model, we first determined the input layer and hidden layers. We set the hidden layers as a layer with 10 neurons. Additionally, we created one neuron for the output layer because our output will only be 0 or 1.

We normalized each input value separately. Therefore, we needed to add 5 normalization layers for each input variable.

As a result, our model will be as follows:

- Input layer: 15 input variables
- Hidden layer: A 10-neuron ReLU activation function
- Normalization layers: 5 normalization layers for each input variable
- Output layer: A 1-neuron sigmoid activation function

To train this model, we used pre-determined data. We matched our training data with a label that indicated whether it was normal or abnormal for each time point. In this way, our model was able to learn from this data and produce accurate results when tested with new data.

We trained a feedforward neural network with ReLU activation function to predict the indoor air quality based on these input values. The network was trained to output 0 if the input values were within the normal range and 1 if any of the input values were outside of the normal range.

The use of the ReLU function as an activation function in deep learning models for education has become increasingly popular. ReLU (Rectified Linear Unit) is a type of activation function commonly used in artificial neural networks. ReLU outputs the same value as input if the input is greater than or equal to zero, but returns zero if the input is negative. This property of ReLU makes it a widely used activation function in artificial neural networks.

The mathematical expression of ReLU is given by the following equation:

$$f(x) = \max(0, x)$$

In this equation, x represents the input value of the ReLU function, and the expression max(0, x) returns x if x is greater than or equal to zero, and returns 0 if x is negative. The use of ReLU in artificial neural networks has many advantages, including fast computation, simple structure, and good results [18].

The normal range values used for the input variables were 19-22°C for temperature, 60% for humidity, and 0-100 ppm for gas ratio. After training the network with the input values, we determined that the indoor air quality was not normal based on the predicted output values.

In Figures 4,5 and 6, the temperature, humidity and air quality values measured at 5 different time points for 5 consecutive days are shown in graphical form.

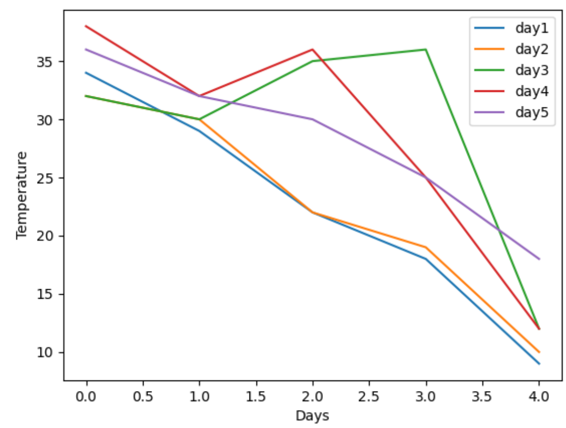

Figure 4. Temperature values measured for 5 different time points across 5 consecutive days.

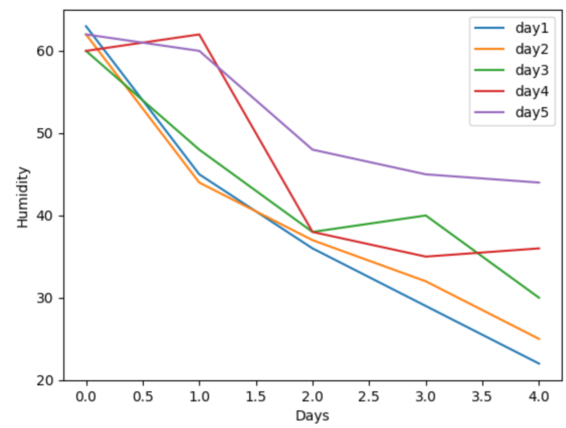

Figure 5. Humidity values measured for 5 different time points across 5 consecutive days.

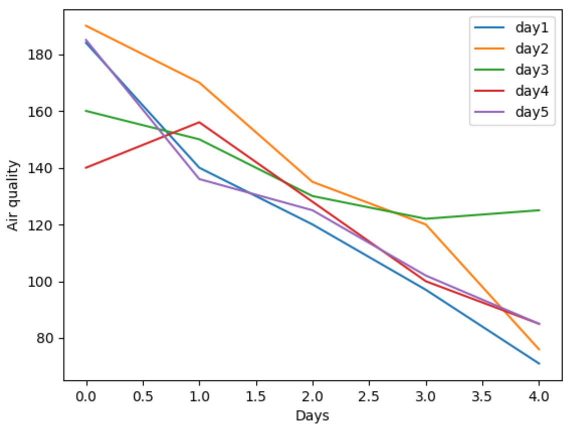

Figure 6. Air quality values measured for 5 different time points across 5 consecutive days.

3. Results

In this study, a feedforward neural network was developed and trained to predict indoor air quality based on the values of temperature, humidity, and gas ratio. The network was trained with 15 input values, and the output was set to 1 if the values were outside of the normal range, and 0 if they were within the normal range. The normal values for temperature, humidity, and gas ratio were defined as 19-22, 40-60%, and 0-100 ppm, respectively.

Analog values obtained from temperature, humidity, and gas sensors were used as input to train the neural network and the ReLU function was used as the activation function. The values obtained from the sensors for five consecutive days were used for testing the network.

The results of the test showed that the indoor air quality was not normal according to the input values, as the network outputted a 0 for all input values as shown in Figure 7. Therefore, it can be concluded that the indoor air quality was within the normal range and there was no need for any corrective actions.

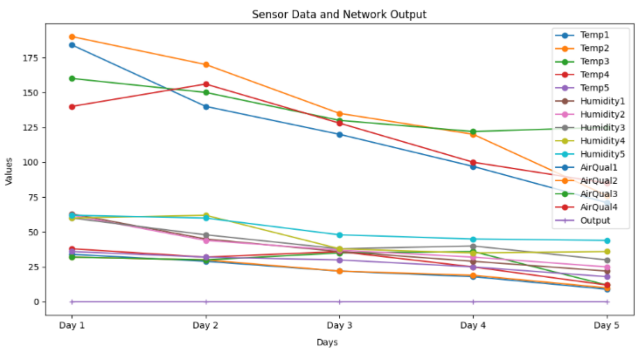

Figure 7. Analog input values and neural network output values taken from sensors for 25 different time points in 5 different days.

4. Discussion

The development of a feedforward neural network can be a useful tool in predicting indoor air quality based on the values of temperature, humidity, and gas ratio. The results of this study show that such a network can be trained to accurately predict indoor air quality based on these parameters. The use of microcontroller control cards and sensors can provide real-time data on indoor air quality, which can be used to train and test the network. In addition, the network can be continuously updated with new data to improve its accuracy and performance. In conclusion, the use of a feedforward neural network-based indoor air quality detection system can be an effective approach to predicting indoor air quality, and it can be a valuable tool for monitoring and improving indoor air quality in various settings such as homes, offices, and hospitals.

References

[1] M. Neira, A. Prüss-Ustün, and P. Mudu, "Reduce air pollution to beat NCDs: from recognition to action," *Lancet*, vol. 392, no. 10154, pp. 1178-1179, 2018.

[2] V. V. Tran, D. Park, and Y.-C. Lee, "Indoor air pollution, related human diseases, and recent trends in the control and improvement of indoor air quality," *International journal of environmental research public health*, vol. 17, no. 8, p. 2927, 2020.

[3] J. Esquiagola, M. Manini, A. Aikawa, L. Yoshioka, and M. Zuffo, "SPIRI: Low Power IoT Solution for Monitoring Indoor Air Quality," in *IoTBDS*, 2018, pp. 285-290.

[4] D. Wall, P. McCullagh, I. Cleland, and R. Bond, "Development of an Internet of Things solution to monitor and analyse indoor air quality," *Internet of Things*, vol. 14, p. 100392, 2021.

[5] L. Dietz, P. F. Horve, D. A. Coil, M. Fretz, J. A. Eisen, and K. Van Den Wymelenberg, "2019 novel coronavirus (COVID-19) pandemic: built environment considerations to reduce transmission," *Msystems*, vol. 5, no. 2, pp. e00245-20, 2020.

[6] C. S. Hayles, M. Huddleston, P. Chinowsky, and J. Helman, "Summertime impacts of climate change on dwellings in Wales, UK," *Building Environment*, vol. 219, p. 109185, 2022.

[7] S. U. Rani, S. Rajarajeswari, J. G. Jaimon, and R. Ravichandran, "Real-Time Air Quality Monitoring System Using MQ-135 and Thingsboard," *Journal of critical reviews*, vol. 7, no. 18, pp. 4107-4115, 2020.

[8] M. Zhou, A. M. Abdulghani, M. A. Imran, and Q. H. Abbasi, "Internet of things (IoT) enabled smart indoor air quality monitoring system," in *Proceedings of the 2020 international conference on computing, networks and internet of things*, 2020, pp. 89-93.

[9] A. Sarkar, "Design of automatic hand sanitizer with temperature sensing," *International journal of innovative science research technology*, vol. 5, no. 5, pp. 1269-1275, 2020.

[10] F. M. Sharmila, P. Suryaganesh, M. Abishek, and U. Benny, "IoT based smart window using sensor Dht11," in *2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)*, 2019, pp. 782-784: IEEE.

[11] R. Ganiyu, O. Arulogun, and O. Okediran, "Development of a wireless sensor network for monitoring environmental condition on a farmland," *International Journal of Applied Information Systems (IJAIS)*, vol. 7, no. 3, pp. 28-34, 2014.

[12] Available: https://components101.com

[13] S. Kumar and A. Jasuja, "Air quality monitoring system based on IoT using Raspberry Pi," in *2017 International conference on computing, communication and automation (ICCCA)*, 2017, pp. 1341-1346: IEEE.

[14] F. N. Abbas, M. I. M. Saadoon, Z. K. Abdalrdha, and E. N. J. I. J. E. R. T. Abud, "Capable of gas sensor MQ-135 to monitor the air quality with Arduino Uno," vol. 13, no. 10, pp. 2955-2959, 2020.

[15] D. A. Aziz, "Webserver based smart monitoring system using ESP8266 node MCU module," *International Journal of Scientific Engineering Research*, vol. 9, no. 6, pp. 801-808, 2018.

[16] B. Sarwar, I. S. Bajwa, S. Ramzan, B. Ramzan, and M. Kausar, "Design and application of fuzzy logic based fire monitoring and warning systems for smart buildings," *Symmetry*, vol. 10, no. 11, p. 615, 2018.

[17] M. Fezari and A. Al Dahoud, "Integrated development environment "IDE" for Arduino," *WSN applications*, pp. 1-12, 2018.

[18] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 807-814.