Chapter 7

Cloud-Connected Intelligent Gas Sensor System for Qualitative Estimation of Blood Glucose Level Through Analysis of Exhaled Breath VOCs

Blood Glucose Level (BGL) monitoring is essential in diabetes, but traditional methods require invasive blood sampling. To overcome this, researchers have found that exhaled breath contains Volatile Organic Compounds (VOCs) that correlate with BGL. Electronic noses (e-noses) provide unique signature responses to VOCs, making them useful for non-invasive BGL estimation. In phase-I of this study, we have proposed a method for 'qualitative' estimation of BGL by analyzing e-nose responses to breath VOCs using an array of seven tin-oxide-based metal-oxide-semiconductor (MOX) gas sensor elements interfaced with a microcontroller which transmits gas sensor array responses to a cloud-connected remote data processing centre (RDPC). Three volunteers, with high, low, and normal BGL, respectively, exhaled on the sensor node for 10 minutes each, at a sampling rate of 10 samples per minute before and after breakfast for three days, producing a dataset of 1800 samples, which were analyzed using two-stage analysis space transformation method at the RDPC. First, the dataset was pre-processed using Standardized Principal Component Analysis (SPCA) space transformation method. Then, three classifiers, Decision Tree (DT), XGBoost, and multi-layer perceptron (MLP), were trained on the SPCA transformed dataset. The proposed system was tested using 30 unknown breath test samples of the volunteers. The best-performing system was a MLP trained in SPCA transformed dataset, associating each test sample correctly with the BGL of respective volunteer and achieved a mean squared error (MSE) of 4.42×10^{-5}. Our proposed method offers a rapid, low-cost, and convenient alternative for non-invasive 'qualitative' estimation of BGL in three levels viz. high, low and normal using e-noses.

7.1 Introduction

Around 463 million individuals worldwide are affected by the severe chronic metabolic condition known as diabetes mellitus (DM). Studies show that DM caused 4.2 million adult deaths in 2019, one death every eight seconds [188]. Type 1 (T1DM)

and type 2 (T2DM) occur from the death of β-cells resulting from insulin sensitivity [111], [189]. The first reports of the use of breath as a diagnostic tool date back to Hippocrates, who identified uncontrolled diabetes from the patient's mouth's acetone odor [190]. There are two approaches to diabetes detection: invasive and non-invasive methods [112].

Lekha et al. developed a non-invasive diabetic detection and classification technique with only two tin-oxide-based sensors (MQ 3 and MQ 5) and Convolutional Neural Network (CNN). In their experiment, they achieved accurate detection and classification of type I and type II grades of diabetes using samples gathered from 25 volunteers by taking only two MQ sensors and AI models [112]-[114]. Invasive techniques can carry a higher risk of complications, such as bleeding, infection, and organ damage. It can also psychologically affect patients, such as anxiety, fear, or stress. Non-invasive techniques generally refer to medical techniques that do not require inserting instruments or other invasive procedures to examine a patient. This method is cost-effective and reduced the risk of complications and anxiety. Typically, the exhaled breath of humans includes N2 (78%), O2 (16%), CO2 (4%), and other gases (2%) [191]. Acetone, Isoprene, Ethane, Ethanol, Methane, Nitric Oxide, Carbon Monoxide and Hydrogen gases have been identified in breath analysis as potential biomarkers to indicate high blood glucose levels (BGL) [113], [191], as shown in Table 7.1. Glucose monitoring instruments have recently been developed as wearable technology. These screening tools are often expensive and depend on blood contact and direct sampling [192]. The technology for precise non-invasive monitoring and prediction of BGL has not yet been established [193].

Table 7.1 VOCs Bioomarkers in Human Breath

Biomarkers	Range (ppm/ppb)
Acetone	0.1 – 1 ppm
Ethanol	0.1 – 10 ppm
Isoprene	0.1 – 1 ppm
Methane	1 – 10 ppm
CO	1 – 20 ppm
H_2S	< 1 ppm
Nitric Oxide	< 1 ppm
Ammonia	< 1 ppm
Pentane	<1 ppm
Ethane	0.1 – 2 ppb

In recent literature, the preponderance of breath biomarkers has been identified so far utilising spectrometry-based techniques, which are the industry standard exhaled breath analysis methods [115]-[116]. The most accurate method for identifying VOCs is gas chromatography (GC), Chromatography and Ion Mobility Spectroscopy (GC-IMS), Proton Transfer Reaction Mass Spectrometry (PTR-MS) and Selected Ion Flow Tube Mass Spectrometry (SIFT-MS) [111], [115]-[117] but these methods are complex, costly, time-consuming and require trained personnel. These high-end equipment use statical methods for analyte analysis. As a result, research on alternative sensing techniques is being carried out to develop a portable, small, non-invasive and user-friendly 'qualitative' BGL, among which e-noses are one of the most feasible tools. In phase-I of this experiment, we have established a prima-facie proof-of-concept to establish a correlation between the constituent components of the exhaled breath of humans and their BGLs categories as high, normal and low. Accordingly, by engaging only three volunteers, we analysed their exhaled breath and qualitatively estimated their BGL as high, normal and low. We implemented and tested this e-nose as CPS compatible intelligent gas sensing system (IGSS) using an IoT platform on the Amazon web services (AWS) cloud. This work reports that 'qualitative' estimation of BGL can be obtained using our proposed IGSS. The architecture of CPS compatible IGSS using an Cloud-IoT for Qualitative BGL estimation as shown in Figure 7.1.

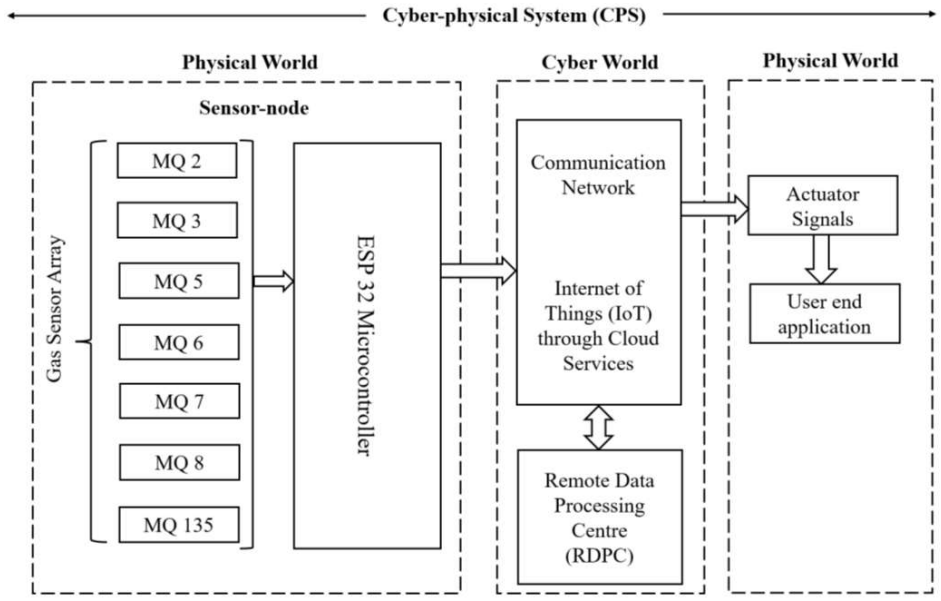

Figure 7.1 The architecture of CPS compatible IGSS using a Cloud-IoT for Qualitative BGL estimation

Further, in phase II of this study, we are working with 150+ volunteers over a period of one year in a clinical setup to develop an AI model for 'quantitative' estimation of BGL.

Several commercial devices (e.g., electronic nose or e-nose systems) are now available on the market that diagnose disease based on exhaled breath used to detect diabetes. Some of the popular are FOX 4000 (Alpha Mos, France), FAIMS breath analyser (Owlstone Medical, UK), Ketonix Bluetooth and USB non-invasive breath analyser (Ketonix AB, Sweden), Portable Breath Acetone Meter PBAM (Biosense™ Readout Health, USA), Cyranose Electronic Nose (Sensigent, USA) and Keyto Breath Sensor (Keyto, USA) [14]-[118]. In the direct method, the gas sample is exhaled directly into the device [119]. In contrast, in the indirect method, it is collected in specially designed bags, for example, Tedlar® bags [195], Teflon sealed bags with saliva and moist filter [196] and fluorinated ethylene propylene breath gas collection bags [197]. Tedlar® bags (Dupont de Nemours) made of polyvinyl fluoride are the most popular bags used in breath sampling studies [198]. Further, turner et al. have established a very close correlation between acetone present in exhaled human breath with the BGL present in the blood plasma. Their method can be further improvised as a gold standard for the measurement of non-invasive BGL estimation in a clinical environment [111].

An e-nose is a device that detects and identifies VOCs/gases/odors using an array of gas sensors. In the context of metal-oxide-semiconductor (MOX) gas sensors-based arrays, an e-nose can play a significant role in enhancing the performance and functionality of such gas sensor array [116]. These are cross-sensitive sensors and arrays. MOX gas sensor arrays generate complex and multidimensional data, which require sophisticated algorithms to process and analyse effectively. Artificial Intelligence (AI) algorithms, such as machine learning and deep learning, can efficiently analyse the data generated by the sensor array, identify patterns, and classify different classes of BGL with high accuracy. Pattern recognition algorithms can detect subtle differences in the response patterns of the gas sensor array and accurately classify different classes of BGL based on their unique patterns.

The transformation of healthcare from in-person consultation to telemedicine is possible through the development of Internet of Things (IoT) compliant cloud-based technologies. In this paper, we have developed a physical sensor node which is a simple

Cloud-Connected Intelligent Gas Sensor System (CC-IGSS) prototype for three types of BGL (high/normal/low) estimation. The proposed system can collect data from the VOCs present in human breaths at a remote processing station in real time through the cloud. This system is convenient, cost-effective, realistic and real-time monitoring at remote processing station by anyone. Our proposed method can be effectively used for real-time 'qualitative' BGL estimation of a patient as prima-facie tool in the household environment.

Fasting blood glucose level refers to the amount of glucose present in the blood after an individual has fasted for a certain period of time, typically 8-12 hours. Postprandial (PP) BGL refers to the amount of glucose in the blood after a meal. In healthy individuals, PP BGL typically rises after meals but returns to normal levels within two to three hours. It's important to note that consistently high PP BGL can indicate diabetes or other health conditions. The Considered Qualitative Levels of BGL are presented in Table 7.2.

Table 7.2 The Considered Qualitative Levels of BLOOD Glucose

Fasting	Postprandial (PP)	Qualitative BGL level
Less than 70 mg/dL	Less than 140 mg/dL	Low
70 mg/dL - 99 mg/dL (less or equal)	140 mg/dL-180 mg/dL	Normal
More than 100 mg/dL	More than 180 mg/dL	High

Accordingly, in this paper, we have proposed an IoT Compliant Cloud-Connected method using a sensor-node consisting of an array of seven tin-oxide-based gas sensing elements, connected to cloud for its analysis at a remote processing station using IGSS advanced data processing algorithms using AI. We have used light weighted Message Queuing Telemetry Transport (MQTT) IoT protocol to send captured data from the sensor array to the cloud. Further, we have received data on remote processing centre from DynamoDB through Razor Structured Query Language (SQL). This data is then analysed using two-stage highly advanced algorithms at remote data processing centre for high performance and estimate the BGL in three classes viz., low, normal and high.

We have captured the breath samples of three volunteers in real-time using a sensor node and recorded the sensor response at a remote processing station through cloud services. Later, we transformed the captured data into an SPCA analysis space transformation method and classify using Decision Tree (DT), XGBoost and Multi-layer

Perceptron (MLP). The basic schematic block diagram of the CC-IGSS is presented in Figure 7.2.

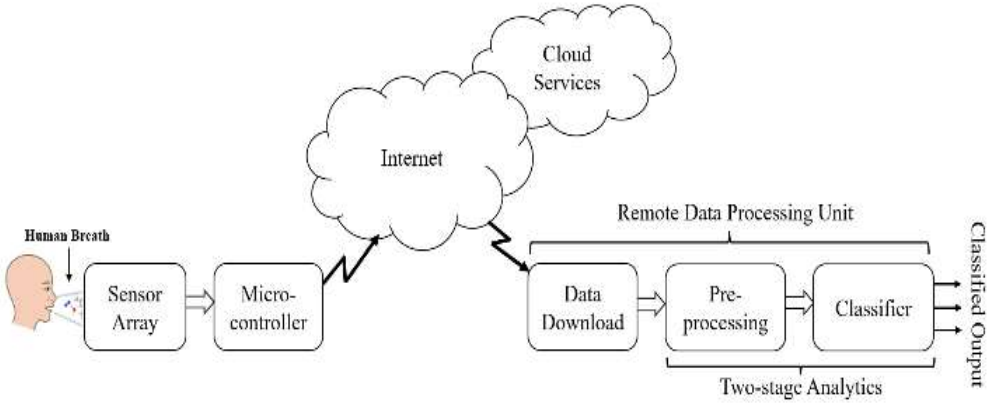

Figure 7.2 Schematic Block Diagram of the Cloud-Connected Intelligent Gas Sensor System (CC-IGSS).

The motivation and Contribution of this work are highlighted as follows:

- We have proposed an efficient method for qualitative estimation of BGL through exhaled human breath in real time.
- For the first time, a cloud-connected IoT protocol is used for real-time BGL estimation using e-noses.
- The proposed CC-IGSS has been designed using a two-stage analysis space transformation method to ensure the classifier models deliver high performance.

7.2 MATERIALS AND METHODS

We have verified our proposed method by designing and fabricating the proposed Intelligent Gas Sensor System (IGSS) for 'qualitative' Blood Glucose Level (BGL) estimation through the VOCs in human breath. Further details have given in upcoming subsections, as below:

7.2.1 The Design Concept of Cloud Connected IGSS

In the present study, our proposed system has three components correspond, as shown in Figure 7.3:

(i) Physical Gas Sensor System (or e-nose): We have used an array of seven tin-oxide gas sensors. Tin-oxide MOX-based gas sensors are cross-sensitive and non-selective and respond to multiple VOCs/gases/odors with different sensitivities when generates a unique signature pattern of the VOCs present in human breath.

(ii) Cloud-based data transfer stage to receive the volunteer's VOC responses remotely at the data processing station. The real time sensor node responses from the gas sensor node are sent to a cloud platform such as Amazon Web Services (AWS). A cloud offers various services: computation, storage, databases, application deployment, blockchain, robotics, AI platforms, and IoT. AWS IoT Core enables the secure connection and management of many devices at scale. It provides device management features such as device registration, authentication, and authorization. AWS IoT Analytics can preprocess and transform raw data from IoT devices, including filtering, enriching, and transforming data before it's stored in a data store for analysis. Amazon DynamoDB is a fast and flexible NoSQL database service provided by AWS. It is designed to deliver high performance, scalability, and availability for internet-scale applications. Razor SQL is a popular SQL editor and database management tool that supports connection with DynamoDB to store real-time data from the AWS cloud.

(iii) Computation: We have used two-stage space transformation methods to transform the captured dataset into the transformed domain using SPCA. Further, we have deployed DT, XGBoost and MLP for classification approaches to analyze the transformed dataset.

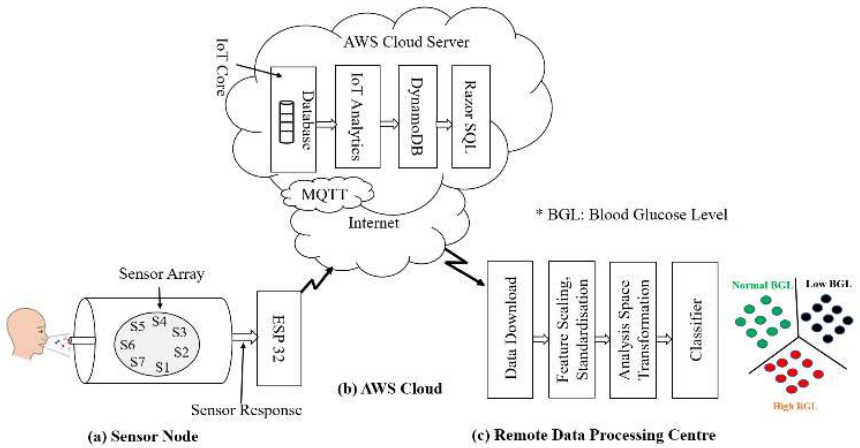

Figure 7. 3 (a)-(c) The proposed architecture of cloud-connected intelligent gas sensor system (CC-IGSS).

In this work, we have created a cutting-edge, intelligent system that uses the cloud to estimate BGL in real time. The samples were captured using a gas sensor node and sent & store on the cloud-connected Dynamo DB tables. The data's apparent clusters were revealed via the principal component analysis. An MLP was trained to classify unknown samples over the cloud in real time.

We have used seven different types of tin-oxide-MOX sensors (Table 7.2) and a microcontroller to create a sensor node. The sensor node is powered by 5V (DC). Details of the considered sensor's characteristics show cross-sensitivity over various gases in Table 7.3.

Table 7.3 Gas sensor characteristics showing cross-sensitivity over various gases

Gas Sensor	Selectivity	Detection Range
MQ 2	Alcohol, i-Butane, Hydrogen, Liquefied Petroleum Gas (LPG), Smoke, Methane, Propane	200–5000 ppm LPG and Propane, 300–5000 ppm Butane, 5000–20,000 ppm Methane, 300–5000 ppm Hydrogen, 100–2000 ppm Alcohol
MQ 3	Alcohol, Methane, Benzene, LPG, Carbon Monoxide, Hexane	25–500 ppm Alcohol
MQ 5	Alcohol, Carbon Monoxide, Hydrogen, LPG, Methane	200–10,000 ppm
MQ 6	Iso-Butane, Propane, LPG,	300–10,000 ppm
MQ 7	Carbon Monoxide	20 – 2000 ppm
MQ 8	Hydrogen	100–10,000 ppm
MQ 135	Alcohol, Ammonia, Benzene, Carbon Dioxide, Smoke, NOx	10–300 ppm Ammonia, 10–1000 ppm Benzene, 10–300 ppm Alcohol

The proposed approach is a non-invasive approach for three levels of BGL estimations. When operating this IGSS in cascaded form, we can identify BGL estimation through human breath in real-time from anywhere. The raw sensor array responses are analyzed remotely in the analysis space transformation domain using famous transformation models where all three clusters of low, normal and high BGL classes show separately.

7.2.2 The Prototype Design

The prototype consists of an array of seven-element tin-oxide MOX-based gas sensor elements (MQ-2, MQ-3, MQ-5, MQ-6, MQ-7, MQ-8 and MQ-135), which generates real-time signature patterns of the VOCs of human breath. The sensor node is placed into a simplistic gas chamber (food-graded bottle) for VOC collection. Inside the gas chamber, all sensors are fitted in a circular order on all sides of the chamber, and wire connections are made with the microcontroller. The electrical characteristics of sensors and devices used in this IGSS are shown in Table 7.4.

Table 7.4 Electrical characteristics of the components as used in the prototype

Components	Input Voltage	Power Ratings
ESP 32	5V	130mA
ESP 32 GPIO pins	3.3V	40mA
MQ sensor	5V	150mA

It consists of a microcontroller interfaced with sensors. A basic communication protocol was set up between the microcontrollers and the computer to send data generated during the experiment and synchronize with the sensor node and end with data processing at the remote processing center. The PCB circuit diagram of IGSS for BGL Estimation has been shown in Figure 7.4.

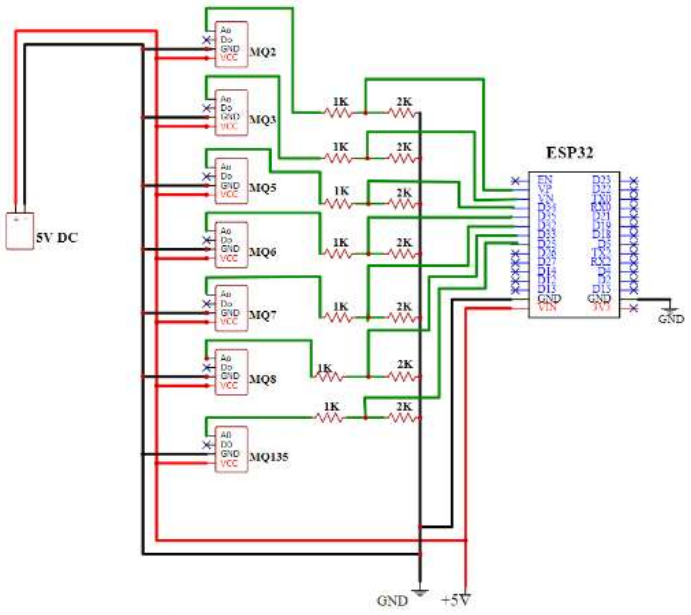

Figure 7.4 PCB circuit layout of IGSS for BGL estimation.

We have connected seven tin-oxide MOX gas sensors with a microcontroller to capture the breath sample and send it to the cloud. Inside the gas chamber, all sensors are fitted in a circular order on all sides of the chamber, and wire connections are made with the microcontroller. The physical view of fabricated IGSS for BGL estimation has been shown in Figure 7.5. The microcontroller has an in-built Wi-Fi module, and it can connect to the internet and access the cloud. To ensure correct measurements, gas sensors must be preheated for around 15 minutes for the readings to become steady. As a result, after powering on the gas sensors, we did not take any readings for the first 15 minutes while allowing fresh air to enter. After this, we obtained readings from three volunteers one-by-one for 10 minutes regularly through a mouth space bottle (sensor chamber). We have captured two times breath samples (a) before breakfast (in the empty stomach situation) and (b) after breakfast (after 2 hours of breakfast).

Figure 7.5 (a) Placement of sensors, (b) Sensor node connection with cloud, and (c) Physical test procedure.

The sampling time for breath sample collection was 10 minutes, and the sampling rate was ten samples per minute. Captured samples from the gas sensor node had been transmitted to the cloud. The captured data was stored in DynamoDB in a real-time situation. We have captured 100 samples in each condition (before breakfast and after breakfast in each condition) per person in 3 days. In this experiment, we collected a total of 1800 breath samples (900 samples before and 900 samples after breakfast) from three

volunteers. The mouth space bottle was opened for around 30 minutes after each experiment to allow the air/gases/biomarkers out and ensure that the prior breath biomarkers had the least possible impact on the sensor array.

7.2.3 The Experiment

In phase-I, we conducted the experiment in a clinical setup and measured the BGL of volunteers using a portable glucometer (model: Accu-chek active, CE0088, Mannheim Germany). We captured three volunteers' breath samples under two conditions, (a) before breakfast, and (b) after two hours of breakfast. All three volunteers (1, 2 and 3) have taken the same breakfast; the breakfast scheduled time has been mentioned in Table 5(b). Volunteer-III is a diabetic patient and has taken medicine (prescribed by the doctor) before breakfast. Volunteer-I and Volunteer-II have not taken any medicine or liquid before breath sampling. We have taken two times breath samples of all subjects, i.e., before and after breakfast (two hours after breakfast). The experiment details have been given in Tables 7.5(A) - (B) and 7.6.

Sample collection details:

A. Before Breakfast (BBF)

Table 7.5 (a). Breath sampling records before breakfast

Volunteer	Preheat time	Sampling Time	Flush Time
1	6:15 AM –6:30 AM	6:31 AM –6:40 AM	6:41 AM –7:10 AM
2	7:21 AM –7:35 AM	7:36 AM –7:45 AM	7:46 AM –8:15 AM
3	8:31 AM –8:45 AM	8:46 AM –8:55 AM	8:56 AM –9:25 AM

B. After Breakfast (ABF)

Table 7.5 (b). Breath sampling records after breakfast

Volunteer	Breakfast Time	Preheat Time	Sampling Time	Flush Time
1	8:00 AM	9:45 AM –10:00 AM	10:01 AM – 10:10 AM	10:11 AM – 10:40 AM
2	9:00 AM	10:50 AM – 11:05 AM	11:06 AM – 11:15 AM	11:16 AM – 11:45 AM
3	10:00 AM	12:00 – 12:15 Noon	12:16 – 12:25 Noon	12:26 – 12:55 Noon

The sensor node has been integrated into the food-graded bottles with AWS IoT core to capture breath samples from human breath from anywhere. The data acquisition node, controlled by a microcontroller, must be connected to the internet, specifically to the cloud, in a remote processing station.

In this experiment, the sensor node was first heated up for 15 minutes while keeping the gas chamber closed, and the steady state of the sensor responses was achieved. Consequently, the gas chamber inlet opened for the next 10 minutes, and the breath samples of the considered volunteer were captured at a remote data processing center (RDPC). Then the breath exposure was stopped, and for the next 30 minutes, the gas chamber was purged with fresh ambient air to ensure that the sensor responses returned to the baseline conditions. The same procedure was repeated to capture the breath samples of the other volunteers, as considered (BBF & ABF).

Accordingly, each experimental phase continues for 55 minutes and raw sensor responses are captured and repeated in both situations. Therefore, the experiment took taken total of 990 minutes (55 minutes × 2 times × 3 Volunteers × 3 days). Throughout the experiment, all sensor responses return back to the baseline responses, and no sensor poisoning takes place. During this period, a total of 1800 samples were captured at the sampling rate of 10 samples per minute. Further details of the dataset and the samples collected are given in Table 7.6.

In regard of the samples belong to the three classes (before and after breakfast each) of BGL. The dataset contains 600 samples for low BGL, 600 samples for normal BGL, and 600 samples for high BGL in both situations. The captured dataset was then segregated into training and testing datasets. The training dataset consisted of 590 × 3 = 1970 samples and 10 × 3 = 30 samples consisted for testing purposes. The testing dataset was separated from the training or validation of the classifiers at any stage.

7.2.4 Conceptual Background of Data Preprocessing and Classifiers

These captured breath samples have been pre-processed by the analysis space transformation method SPCA [49]. We have utilized analysis space transformation methods for transforming raw data into space transformation for clear and compact visualization and better classification performance. It is shown that a classifier performs

much better on transformed data than raw sensor responses. Further, we analyzed these transformed datasets using DT, XGBoost, and MLP classifiers.

7.2.4.1 Data Preprocessing

Sensor data can be very high-dimensional and noisy, making it difficult to identify relevant features for analysis. Analysis space transformations can also help reduce the data's dimensionality, which can simplify subsequent pre-processing. It has been observed that space-transformed data has much better performance and clear 3D scatter cluster visualization. The illustrative diagram of analysis space transformation for the 3D scatter plot is shown in Figure 7.6.

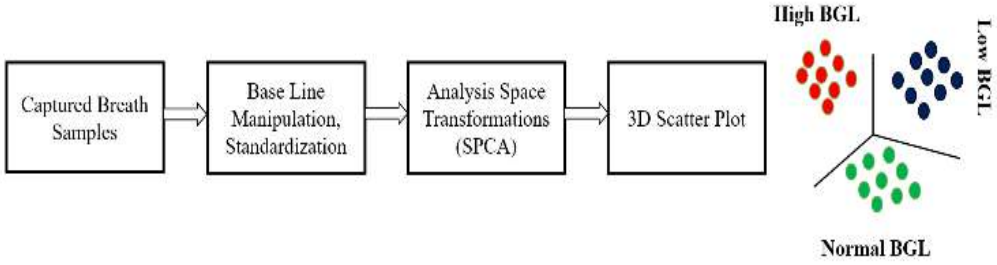

Figure 7.6 Analysis of Space Transformation for the 3D Scatter Plot.

Table 7.6 Distribution of Breath Samples

Volunteer	BGL (mg/dL)		Capture Breath Samples and Data Sampling Time (minute)				
	BBF	ABF	BBF	Sample Time	ABF	Sample Time	BBF
Low BGL	84	95	300	10	300	10	300
Normal BGL	113	135	300	10	300	10	300
High BGL	168	210	300	10	300	10	300
Total			900	30	900	30	900
* BBF: Before Breakfast, ABF: After Breakfast, BGL: Blood Glucose Level							

- Calculate the mean of each variable in the dataset:

$$x'_j = (1/n) \sum_{i=1}^{n} x_{ij} \quad (72)$$

- Standardize the variables by subtracting the mean from each observation and dividing by the standard deviation:

$$x_{ij} = \left(x_{ij} - x'_j\right)/s_j \tag{73}$$

- Calculate the covariance matrix of the standardized variables:

$$S = 1/(n-1)\sum_{i=1}^{n}(z_i - z')(z_i - z')^T \tag{74}$$

- Calculate the eigenvectors and eigenvalues of the covariance matrix:

$$Sv = \lambda v \tag{75}$$

- Sort the eigenvalues in decreasing order, and select the top eigenvectors corresponding to the largest eigenvalues. These eigenvectors form the new set of variables called principal components. Transform the original variables into the new set of principal components:

$$y_i = z_i V \tag{76}$$

7.2.4.2 Classifiers

Accordingly, the raw sensor responses were first transformed into SPCA domain. This is a very effective method used for feature extraction and dimensionality reduction [49]. For the performance enhancement of the IGSS, we have used SPCA as the method for feature extraction. We have all seven principal components (PCs) for the training and testing of the classifier used in the IGSS, without any information loss and the first three PCs for the 3D scatter plot.

Once we have obtained, the SPCA transformed version of the raw sensor responses, consisting of the 1800 samples vectors with 7 element sample vectors. The transformed dataset was then segregated again into two parts, i.e., the training and testing dataset consisting of 1790 and 30 samples in the SPCA transformed domain, respectively. Further, we have used various popular classifiers such as DT, XGB and MLP. Further details of these classifiers can be found in the literature [49], [199] -[201]. Among these popular classifiers, MLP based classifier outperforms the other types of classifiers. The schematic of the classifier design using a two-stage analysis of space transformation is

shown in Figure 7.7. We have implemented three classifiers DT, XGBoost, and MLP classifier. In this experiment, XGBoost is less performer, where DT and MLP have classified all separate classes with 100% accurate classification.

Figure 7.7 Classifiers design using two-stage analysis of space transformation.

DT: A decision tree classifier is a machine-learning algorithm for classification tasks. It is a tree-like model where each node represents a feature or attribute, and each branch represents a decision rule. The tree is constructed by recursively partitioning the data into subsets based on the most informative features until a stopping criterion is met, such as the purity of the subsets. To classify a new instance, the algorithm starts at the root node and evaluates the corresponding feature. Based on the decision rule, it moves down the tree to the next node until it reaches a leaf node, representing a class label. In this case, the criterion used is "entropy," which measures the impurity of the nodes in the decision tree.

Entropy measures the randomness of the target variable's distribution in a node, and the algorithm minimises it. The splitter is the strategy to split a node in a decision tree. The two options are "best" and "random." In this case, the "best" splitter is used, which means the algorithm will try to find the best split among all possible features. The maximum depth is the maximum number of levels allowed in the decision tree. In this case, the maximum depth is set to 3, which means that the decision tree can have, at most, three levels of nodes. The maximum features are the maximum number of features the algorithm considers when looking for the best split. In this case, the value "auto" is used, which means that the algorithm will consider all the features. The architecture diagram of a decision tree is presented in Figure 7.8.

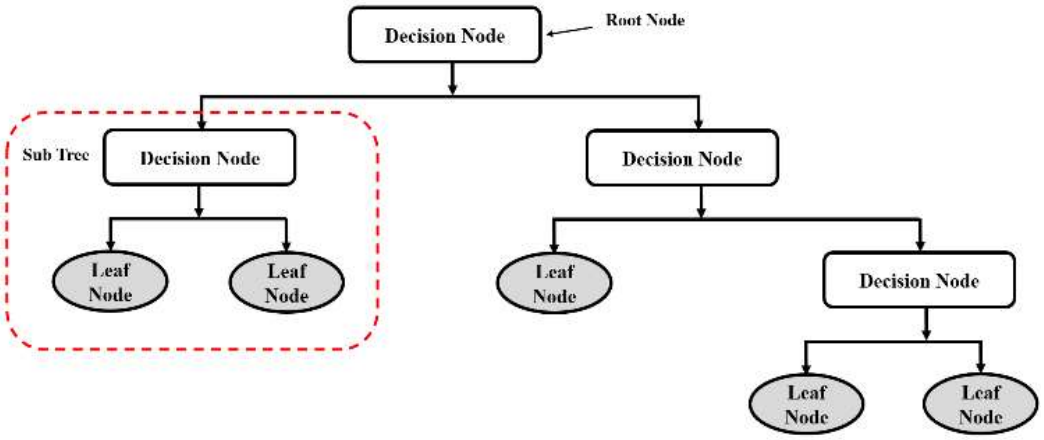

Figure 7.8 The architecture of the Decision Tree Classifier.

XGBoost: The XGBoost classifier is a specific implementation of XGBoost that is used for classification tasks. It uses decision trees as its weak models and combines their predictions through gradient boosting. The algorithm works by iteratively adding decision trees to the model, with each new tree correcting the errors of the previous ones. The process continues until a maximum number of trees or a minimum improvement in accuracy is reached. The learning rate is the step size at which the algorithm learns from the mistakes of each iteration. A lower learning rate can lead to more accurate results, but it also makes the algorithm slower to converge. In this case, the learning rate used is 0.1, which is a moderate value.

The number of estimators is the number of decision trees to be created by the algorithm. Increasing the number of estimators can lead to better performance but can also lead to overfitting. The maximum depth of a decision tree is the maximum number of levels it can have. A deeper tree can learn more complex relationships between features but can also lead to overfitting. In this case, the maximum depth is set to 4, which is a moderate value. The minimum sum of instance weight needed in a child node. In other words, it is the minimum number of instances required in each child node. This parameter controls overfitting by setting it to a higher value. In this case, the value used is 6, which is a moderate value. Gamma is a parameter used to control regularization, which penalizes the model for creating new nodes in the tree. A higher gamma value means more regularization, which can help prevent overfitting. Regularization is a method to prevent overfitting by adding a penalty term to the loss function. The reg_alpha parameter

controls the L1 regularization term. The number of threads is the number of CPU cores to use for parallel processing. In this case, 4 threads are used. The architecture of the XGBoost classifier is presented in Figure 7.9.

MLP: An MLP classifier is a type of artificial neural network that is commonly used for supervised learning tasks, such as classification and regression. The MLP comprises multiple layers of nodes (also known as neurons) that are connected by weighted edges. The first layer is the input layer, which receives the input data. The last layer is the output layer, which produces the network's output. In between the input and output layers, there are one or more hidden layers, which help the network to learn and extract useful features from the input data.

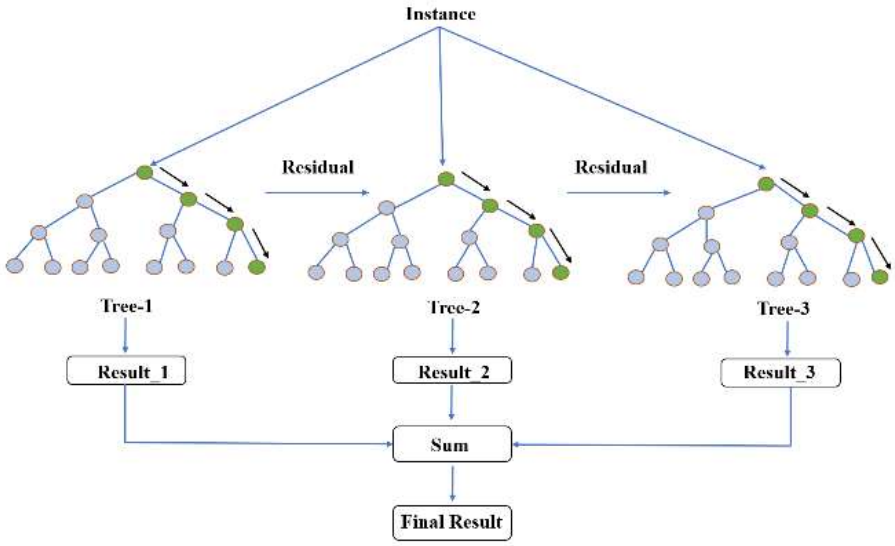

Figure 7.9 Architecture of XGBoost Classifier.

Each node in the MLP computes a weighted sum of its inputs and applies a non-linear activation function to produce its output. The weights and biases of the MLP are adjusted during training using an optimization algorithm such as backpropagation, which minimizes the difference between the predicted outputs and the true labels. The MLP model consists of one or more hidden layers, and this parameter specifies the number of nodes in each hidden layer. In this case, there is one hidden layer with 11 nodes.

The activation function is used to introduce non-linearity into the MLP model. The Relu activation function is used, which stands for the rectified linear unit. It is a

popular activation function that sets negative values to zero and keeps positive values unchanged. The solver is the optimization algorithm used to train the MLP model. The "adam" solver is used, which is a popular stochastic gradient descent (SGD) optimization algorithm. The batch size is the number of training examples used in each iteration of the training process. A smaller batch size can lead to more noise in the updates, but it can also speed up the training process. In this case, the batch size used is 100.

The learning rate determines the step size at which the algorithm learns from the mistakes of each iteration. The "adaptive" learning rate is used, which means that the learning rate is adjusted based on the progress of the training process. The maximum number of iterations is the number of times the MLP model is trained on the data. In this case, the maximum number of iterations used is 100. CV stands for cross-validation, and it is used to evaluate the performance of the MLP model. In this case, a 5-fold cross-validation is used, which means that the data is split into 5 equal parts, and the algorithm is trained on 4 parts and tested on the fifth part. This process is repeated 5 times, with each part being used as the test set once. The schematic diagram of the proposed SPCA transformation process and the MLP classifier is presented in Figure 7.10 (a) – (b).

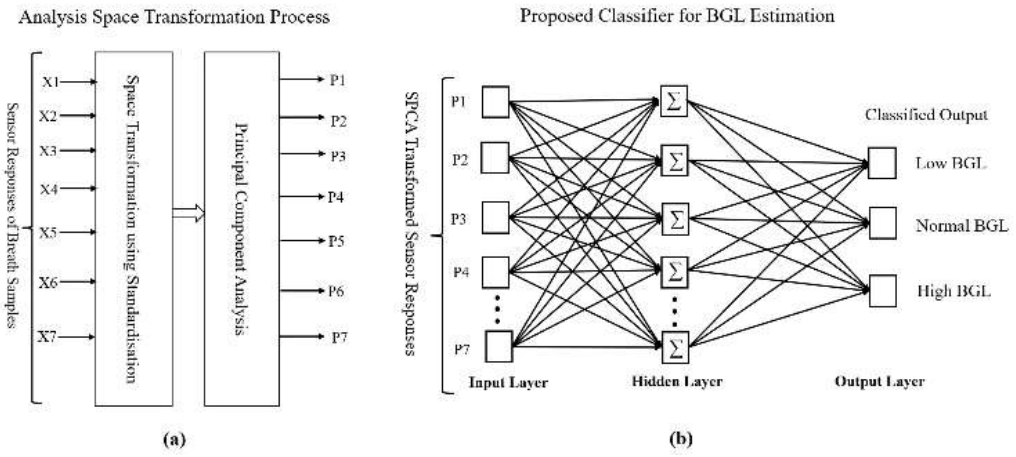

Figure 7.10 Schematic diagram of the proposed SPCA transformation process and the MLP classifier.

The performance parameters taken while designing the classifiers: Decision Tree, XGBoost and MLP, have been presented in Table 7.7.

Table 7.7 Hyperparameter Tunning on DT, XGB and MLP

Classifier	Hyperparameter
DT	Criterion: entropy, splitter: best, max depth:3, max features: auto, cv=5
XGBoost	Learning_rate:0.1, n_estimators:1000, max_depth:4, min_child_weight:6, gamma=0, reg_alpha:0.005, nthread:4, cv=5
MLP	Hidden layer sizes =11, activation function: Relu, solver: adam, batch size:100, learning rate: adaptive, max iteration: 100, cv=5

7.3 RESULS AND DISCUSSION

The proposed work has been carried out using Python 3.9.0 software running on the remote processing station's computer. The IGSS prototype has been cloud-connected and interfaced with the computer using VSCode.

7.3.1 Sensor Response Patterns of Breath Samples

As can be seen in Figure 7.11 (a), the blue star (*) marks are showing high BGL, which can observe as they are forming a distinct cluster. Further, the plus (+) green dots show normal BGL, and the red triangle (^) sign shows low BGL of the volunteers. These are shown in separate clusters. Some clusters are shown overlapping because, in 3D space, they are separate. However, the 2D image presentation looks to be overlapped. However, they are not looking overlapping, as shown in Figure 7.11 (a).

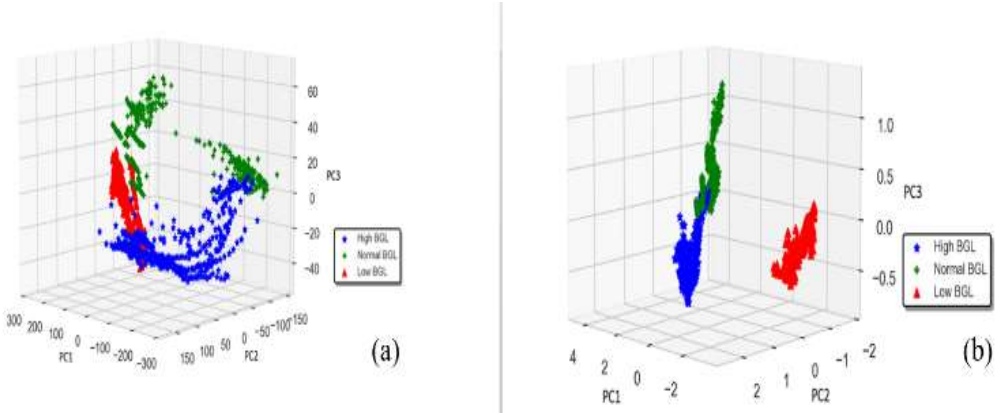

Figure 7.11 (a) 3D Scatter Plot of PCA transformed raw dataset, (b) 3D Scatter Plot of SPCA dataset for BGL Estimation.

We have used the first three PCs of SPCA transformed data for a 3D scatter plot, as shown in Figure 7.11 (b). However, while designing the classifier, we have considered all seven PCs of the SPCA transformation to ensure that 100% of the information is utilised.

7.3.2 Performance of Classifier for Classifying the BGL Classes

As described in Section 7.3.2, We have considered three types of classifiers viz., DT, XGBoost and MLP classifiers. The network architecture of XGBoost and Decision tree are very complex and computationally more intensive. MLP is a simpler and high performer than XGBoost and DT, so MLP is the best classifier. DT and XGBoost have got 97% accuracy, i.e., 29 out of 30 samples, but MLP classified all samples correctly and got 100% accuracy. Mean Squared Error (MSE) has been used for performance evaluation between actual and predicted values of BGL classes. As described in Table 7.2, we have taken five test samples of each volunteer during the fasting and five samples during the PP stage, i.e., two hours after breakfast. Considering that for any individual, the numerical value of BGL may vary within the ranges as specified, our classifier places respective BGL suitably. This explains the flatness of the levels shown in the Figure 7.12. The classification performance of the IGSS trained and tested in the SPCA domain has been depicted in Figure 7.12. MSE performance has been shown in Table 7.6.

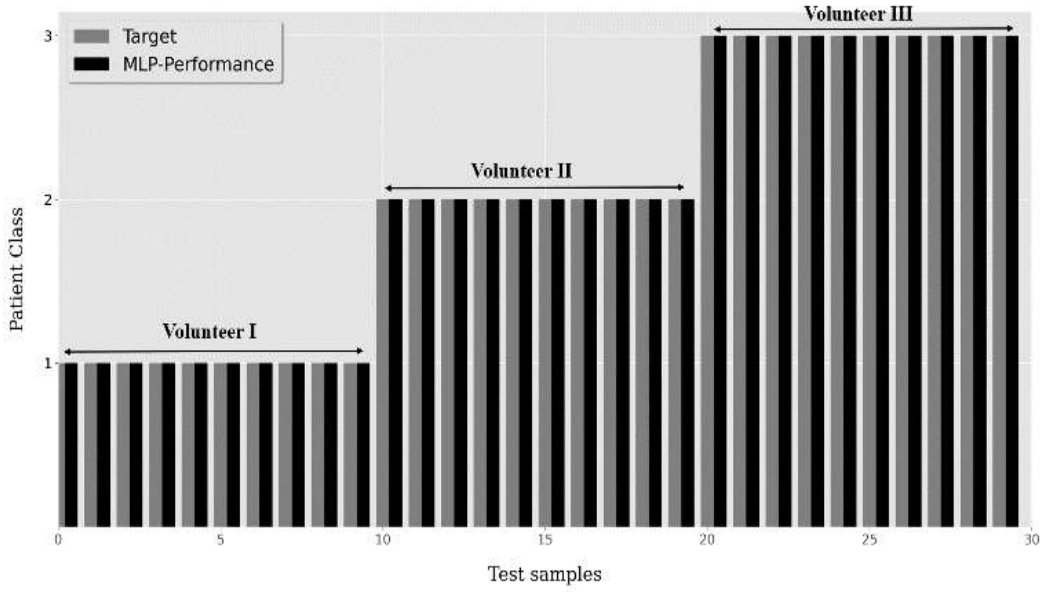

Figure 7.12 Performance of MLP classifier.

For clear and correct visualization, the confusion matrix of the MLP classifier has been shown in Figure 7.12, which shows the 'all correct' classification of 30 unknown samples taken from the testing samples, not used in training or validation data in the SPCA transformation domain. The Confusion matrix has been shown in Figure 7.13.

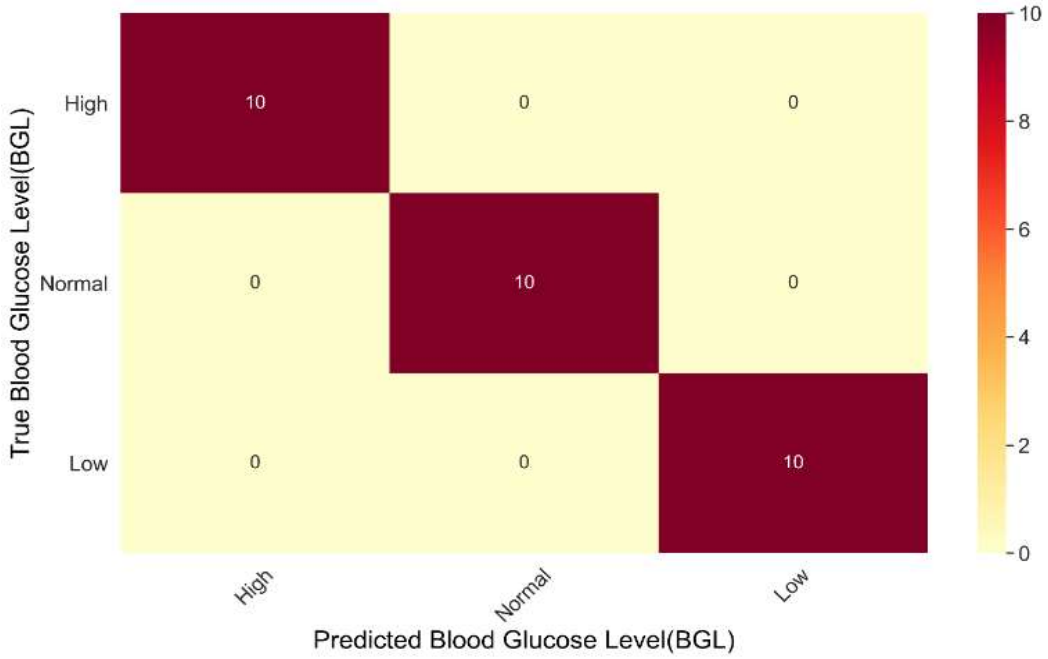

Figure 7.13 Confusion Matrix of SPCA MLP classifier.

The minimum MSE of raw responses is 3.38×10^{-4}, the maximum MSE is 4.11×10^{-3}, and the average MSE is 1.66×10^{-3}. On the other hand, the minimum MSE of the proposed method is 4.42×10^{-5}, the maximum MSE is 6.67×10^{-5}, and the average MSE is 5.79×10^{-5}. MSE of raw responses and transformed dataset of all three samples (C1, C2 and C3) have been shown in Table 7.8.

Table 7.8 MSE of Raw Response and Proposed Method

SAMPLE	RAW RESPONSE	PROPOSED METHOD
C1	4.11×10^{-3}	6.28×10^{-5}
C2	5.5×10^{-4}	4.42×10^{-5}
C3	3.38×10^{-4}	6.67×10^{-5}