Real-Time Non-Invasive Detection and Classification of Diabetes Using Modified Convolution Neural Network

S. Lekha, *Member, IEEE*, and Suchetha M., *Member, IEEE*

***Abstract*—Non-invasive diabetes prediction has been gaining prominence over the last decade. Among many human serums evaluated, human breath emerges as a promising option with acetone levels in breath exhibiting a good correlation to blood glucose levels. Such correlation establishes acetone as an acceptable biomarker for diabetes. The most common data analysis strategies to analyze the biomarkers in breath for disease detection use feature extraction and classification algorithms. However, snags such as computational cost and lack of optimal feature selection on application to real-time signals reduce the efficiency of such analysis. This paper explores the use of a one-dimensional (1-D) modified convolution neural network (CNN) algorithm that combines feature extraction and classification techniques. The approach proposed in this paper is found to significantly reduce the limitations associated with using these techniques individually and thereby improving the classifier's performance further. This paper proposes to apply a modified 1-D CNN on real-time breath signals obtained from an array of gas sensors. The experimentation and the performance of the system is carried out and evaluated.**

***Index Terms*—Acetone, breath, convolution neural network, diabetes, non-invasive.**

I. Introduction

MONITORING and maintaining blood glucose levels (BGL) within prescribed parameters hold significant importance in the globally recommended strategy for diabetes management and control. Invasive techniques that are conventionally associated with measurement of BGL expose the patients to the possibility of infections and also carry with them innumerable other risks, the most prominent being the risk of slow wound healing, a symptom often associated with this disease [1]. It has also been observed that a psychological fear of needles in the population delays timely detection and control of the disease. Addressing these issues, several new emerging researches have focused on the possibility of developing an accurate non-invasive diabetes predictive system suitable for clinical applications [2]. Exhaled human breath is composed of a multitude of volatile organic compounds (VOC) which act as promising bio-markers to many pathological disorders thereby differentiating diseased individuals from healthy subjects [3]. Among these VOCs, breath acetone ($(CH_3)_2CO$) concentration is found to remain elevated in diabetic patients. Consequently, breath analysis emerges as an effective non-invasive diagnostic tool to detect diabetes that can be repeated frequently without risks. Researches have reported that the normal acetone level in exhaled breath in a healthy human subject varies between 0.22 ppm–0.80 ppm. This level rises to 1.76 ppm–3.73 ppm in a subject diagnosed with type-2 diabetes and it soars to as high as 21 ppm in a type-1 diabetic cases [4]. Traditional techniques such as gas chromatography - mass spectroscopy [5], selected ion flow tube mass spectroscopy [6] and cavity ring down spectroscopy [7] used for breath analysis fail to meet the criteria required in clinical applications as they use complex and expensive laboratory equipments that require highly trained personnel to operate. Studies have shown that chemical sensors called e-nose sensors offer a portable, economical, user-friendly alternative which overcomes the shortcomings mentioned above. Metal oxide semi conductor devices, conducting polymers, quartz crystal resonators and MEMS sensors are the most prevalent e-nose sensors employed in identification of various gases [8]–[10]. In this paper, breath signals required for data analysis is measured using an array of metal oxide semiconductor (MOS) gas sensors.

The data thus obtained is further analyzed using various pattern recognition techniques. In order to quantitatively analyze the VOC concentration and enhance the sensitivity of the sensor, pattern recognition techniques which include feature extraction and classification have been employed in previous studies [11]. In these analyses, initially, features are extracted from the raw analog signals and appropriate classifiers are used to segregate the gas concentrations as normal and abnormal. Selection of a proper feature extraction technique is crucial as this aids in extracting robust information from the signals thus improving the efficiency of the classifier. In prior studies, feature extraction techniques such as principal component analysis (PCA) [12], linear discriminant analysis (LDA) [13], statistical features (such as standard deviation, mean and variance) [14] have been analyzed while classification algorithms such as decision tree [15], k-nearest neighborhood [16], artificial neural networks [17], perceptron [18], support vector machine [19] and other models are some of the commonly used algorithms. These feature extraction and classification systems have been found to be effective in diabetes detection applications. Ke Yan *et al.*, have employed principle component analysis to extract features and

Manuscript received April 19, 2017; revised August 7, 2017 and September 21, 2017; accepted September 22, 2017. Date of publication September 27, 2017; date of current version August 31, 2018. *(Corresponding author: Suchetha M.)*

The authors are with the School of Electronics Engineering, Vellore Institute of Technology, Chennai 600048, India (e-mail: s.lekha2014@vit.ac.in; suchetha.m@vit.ac.in).

Digital Object Identifier 10.1109/JBHI.2017.2757510

support vector machines with Gaussian kernel being used to classify the breath samples [20]. Literature also shows that transient features such as magnitude, difference, derivative, integral and time features were calculated and support vector regression algorithms were used to detect diabetes [21]. Hamdi Melih, *et al.* have classified variations in frequencies through a radial basis function neural network [22].

As mentioned earlier, though selection of suitable feature extraction and classification techniques play a crucial role in determining the accuracy of the outcomes, formulation of a common optimal feature extraction and classification technique has not been reported. Manual or handcrafted selection of features used in these studies may yield satisfactory accuracy in a particular set of data but may not be able to concede optimal performance in another. Also these data analysis techniques using different algorithms for feature extraction, selection and classification increases computational time and cost thereby rendering these techniques sub-optimal for use in real time application. An alternative approach using deep learning algorithms is proposed in this paper to effectively address these issues. Convolution Neural Network (CNN) is fast emerging as a popular pattern recognition technique [23], [24]. There has been widespread acceptance of this machine learning algorithm in 2D image recognition, video analysis, speech and language processing. As this neural network has exhibited promising results in optimizing feature extraction and reducing computational time, it has also been adapted for 1D signal data [25]. This paper proposes to identify and classify raw breath signals obtained from the MOS sensors by adapting a modified CNN algorithm thereby predicting diabetes.

In this paper, Section II discusses the specifications of the sensory system and describes the details of breath sample collection. Section III introduces the architecture of the proposed convolution neural network. Section IV outlines the details of the experimentation done for data collection, implementation of the modified CNN and finally evaluates the performance and the computational time of the classifier. The work has been concluded in Section V.

II. Hardware for Data Acquisition

This system describes the mechanism of detecting acetone gas concentration from breath samples as well as the data signals obtained from this analysis. For the purposes of obtaining data for this study, a sensory device consisting of an array of commercially available MOS sensors is employed. MQ series of MOS gas sensors are electro-chemical sensors with good affinity to VOC in breath. An array of MQ-3 and MQ-5 sensors manufactured by Hanwei Electronics is used here. These sensors contain a sensitive layer of tin dioxide (SnO_2) and when they are exposed to various acetone concentrations in exhaled breath, it is noticed that the conductivity of the sensors change. The change in conductivity is measured by calculating the variations of the output signals from the sensors. The standard operating conditions for these sensors include a heating voltage of 5 V and a circuit voltage of 5 V. Also a load resistance of 68K ohms is employed to increase the sensitivity of the sensors to the acetone gas molecule. The heating voltage supplies the standard working temperature for the sensors and the circuit voltage provides the detection voltage to the load resistance. These analog voltage signals are then acquired through an arduino uno board. Arduino

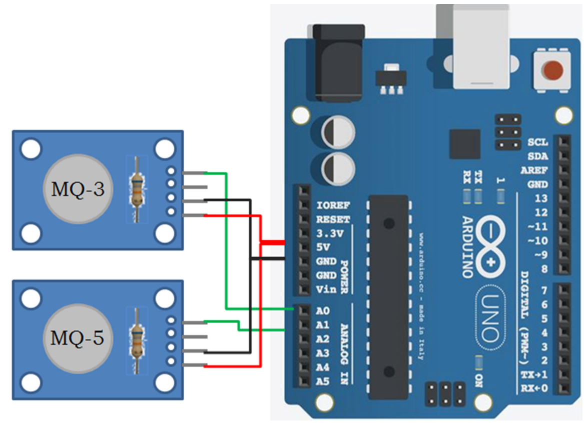

Fig. 1. Configuration of sensor array.

is an open source platform which consists of a programmable circuit board and an Integrated Development Environment platform. It is a microcontroller board (ATmega328P) which has a baud rate of about 9500. These MQ sensors output are given to the analog input port (pin number A0 and A1). The arduino uno is programmed through the python language and the sensors readings are collected for an interval of 1000 sec. The outline configuration of the sensor array is shown in Fig. 1.

In order to reduce the interference of external gases present in the air, these sensors are enclosed inside a gas chamber. In this analysis a non-static small volume gas chamber called the transpacer is used to enclose the sensory array. This gas chamber has a capacity of 218 ml in which one end of the chamber is enclosed with a disposable mouthpiece. A group of diabetic and healthy breath samples were collected after obtaining informed voluntary consent of the subjects. The subjects were asked to blow through the mouthpiece of the gas chamber and the reading from sensors collected for an interval of 1000 sec. The data signals thus acquired are processed further through the CNN algorithm.

III. Architecture of the Proposed Convolution Neural Network

CNN, a variation of a back-propagation neural network, replicates an animal visual cortex system which learns and recognizes specific features and patterns in a visual image. This algorithm achieves significant reduction in pre-processing data and eliminates the need for separate feature extraction techniques by automatically learning and extracting features from raw signal data. In this paper, the CNN model proposed contains an input layer, a few convolution and max-pooling layers, a fully connected MLP layer and finally an output layer. An overview of the architecture of the CNN model is explained in Fig. 2. In this model the raw data signals from the gas sensors are fed into the input layer. These raw signals are filtered by the convolution layer and sub-sampled by the max-pooling layers thus obtaining a reduced optimal feature set. These initial layers of the CNN architecture represent the feature extraction mechanism while the fully connected layers represent the classification mechanism. A detailed framework of the three layers, which define the structure of the neural network are explained in the forthcoming sections.

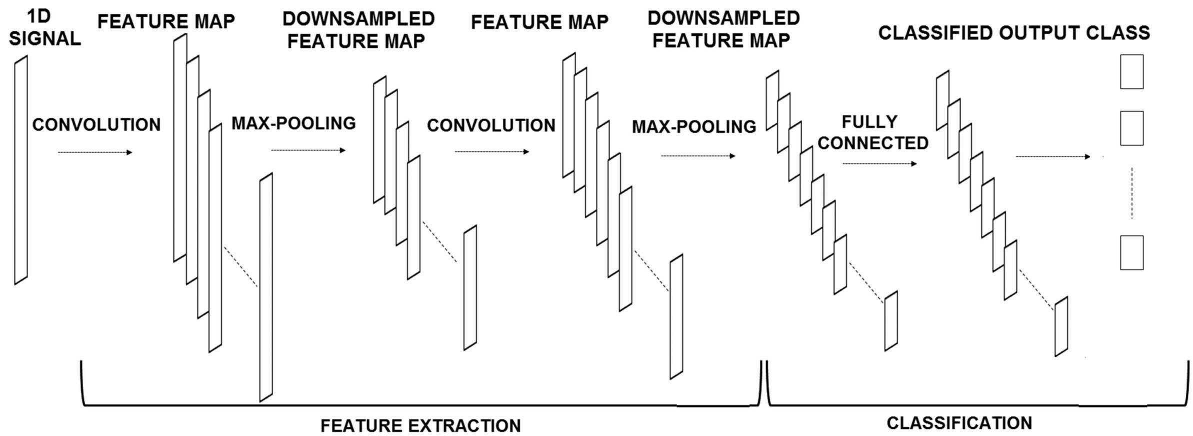

Fig. 2. Architecture of the convolution neural network.

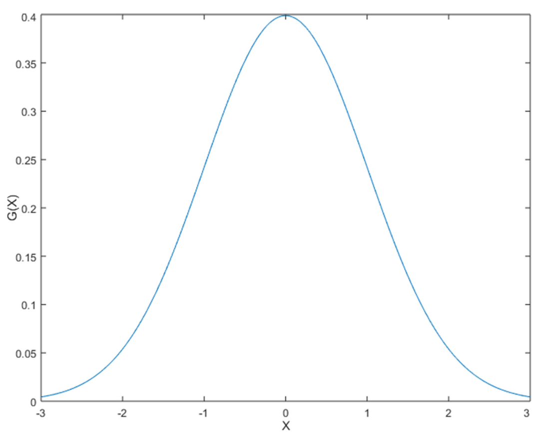

Fig. 3. Gaussian filter kernal.

Algorithm 1: Non-Overlapping Windowing Technique for Segmentation.

Procedure: Segmentation
Steps :
Initialize the window length as $\downarrow s$.
Initialize the total number of segments $numseg$ which is calculated by dividing the length of the data set by the window length.
Initialize the starting point of the segment $sp = 1$ and the end point $ep = sp + \downarrow s$.
for $j = 1$ to $numseg$ **do**
    $x_i^*[j] = x_i(sp : \downarrow s)$
    Update $sp = sp + numseg$
    Update $ep = sp + \downarrow s$.
end for
return x_i^*

A. Modified Convolution Layer With 1D Gaussian Kernel Filter

This convolution layer is designed to transform the input signals into a set of feature maps. To achieve this, the layer employs convolution operation which may be defined as the sum of the dot product of two functions (here, input signal and Gaussian kernel) after one function is reversed and shifted. These input signals are filtered using 1D kernel filter and the resultant convolved outputs produces the feature maps. In this analysis a Gaussian based 1D kernal filter as shown in Fig. 3 is used.

Considering s_i as input to the convolution layer of length m, k_i as the kernel convolution filter of length n and x is the output feature map of length $m + n - 1$. The equations below expresses the convolution operations to obtain the output feature map.

$$x_i(1) = s_i(1)k_i(1)$$
$$x_i(2) = s_i(1)k_i(2) + s_i(2)k_i(1)$$
$$x_i(3) = s_i(1)k_i(3) + s_i(2)k_i(2) + s_i(3)k_i(1)$$
$$.$$
$$.$$
$$x_i(n) = s_i(n)k_i(1) + s_i(n-1)k_i(2) + \ldots\ldots + s_i(0)k_i(n)$$

Hence on generalizing, the convoluted feature maps is obtained by,

$$x_i(n) = \sum_{r=-m}^{m} (s_i(r+1)k_i(n-r+1)) \tag{1}$$

B. Max-Pooling Layer for Reduced Feature Set

It is seen that after the convolution layer, the dimension of the feature map is large and there is a need to reduce its dimension. In this layer, the feature maps are down sampled by taking the maximum value of the map over a small window length. A non-overlapped windowing technique is used to segregate the feature sets. The proposed algorithm for feature set segmentation is detailed in Algorithm 1. Next, the maximum value within this window length is taken. This is achieved by,

$$X_i = \max(x_i^*) \tag{2}$$

where $\max()$ operation calculates the maximum value and X_i is the down-sampled feature map. This reduced feature maps are then given to the fully connected layer where the prediction is carried out.

C. Fully Connected Layer

The final layer of the CNN is the fully connected multi-layer perceptron (MLP) which gives the classified output. As discussed in previous sections, the convolution and the max-pooling layers are used to extract features form the raw input signal while the fully connected MLP layer is used for prediction and classification. This layer works on a principle similar to the traditional neural network. It maps the features extracted from the previous layer to their appropriate class. An activation function called the sigmod function is used to classify these extracted features and to predict the output. The decision function is described as,

$$y_i = sig\left(\sum_{i=1}^{N} X_i w_i + b_i\right) \tag{3}$$

where the output y_i represents the predicted class, X_i is the input feature map, w_i and b_i are the shared weights and bias and $sig()$ represents the sigmoid activation function. Considering h as the input argument, this activation function can be given as,

$$sig(h) = \frac{1}{1+e^{-h}} \tag{4}$$

This activation function ranges in values between $[-1, 1]$. Similar to the artificial neural network, this layer is trained by back-propagation with stochastic gradient descent in order to reduce the difference between the estimated output and the desired output at the fully-connected layer.

1) Training the CNN With Modified Back-Propagation Algorithm: The CNN algorithm classifies data through supervised training. This is achieved by adopting back-propagation and stochastic gradient descent methods. Initially, during the forward propagation, the weights and bias for the neural network are defined randomly which results in increasing the mean square of error of classification. In this algorithm the mean square error is calculated by,

$$e = \sum(d_i - y_i) \tag{5}$$

where y_i is the estimated output and d_i is the desired output. This error is minimized by employing the gradient decent algorithm. In this algorithm the weights and the bias which are initially defined randomly are updated in an iterative manner thus reducing the error of classification. In order to reduce this error, the partial derivative of the error with respect to the weights and the bias is calculated. In the traditional back-propagation algorithm [25], [26], multiplying and dividing the partial differentiation of the errors with ∂y_i the expression is given by,

$$\frac{\partial e}{\partial w_i} = \frac{\partial e}{\partial y_i}\frac{\partial y_i}{\partial w_i} = \frac{\partial e}{\partial y_i} y_i \quad \text{where} \quad \frac{\partial y_i}{\partial w_i} = y_i \tag{6}$$

hence,

$$\frac{\partial e}{\partial w_i} = \Delta e y_i \tag{7}$$

and,

$$\frac{\partial e}{\partial b} = \frac{\partial e}{\partial y_i}\frac{\partial y_i}{\partial b} = \frac{\partial e}{\partial y_i} \tag{8}$$

hence,

$$\frac{\partial e}{\partial b} = \Delta e \tag{9}$$

Algorithm 2: Training the Modified CNN algorithm.

Procedure: CNN-Training
Steps :
1: Initialize the weights defined by w_i and the bias b_i randomly.
2: Initialize the total number of iterations I.
3: **for** $i = 1$ to I **do**
4: Forward Propagation: Calculate the outputs at each neuron where the output is y_i
5: Backward Propagation: Compute the mean square errors given by Δe which is derived from the convolution operation.
6: Update the weights and bias.
7: **end for**
8: **return** Δe

In the CNN algorithm, the above mentioned equations are modified and the errors calculation is expressed by,

$$\frac{\partial e}{\partial w_i} = conv(y_i, \Delta e) \quad \text{and} \quad \frac{\partial e}{\partial b} = \sum \Delta e \tag{10}$$

where $conv()$ is the 1D convolution operation. Finally the weight and bias is updated by the equations,

$$w_i(t+1) = w_i(t) - \epsilon\frac{\partial e}{\partial w_i} \text{ and } b_i(t+1) = b_i(t) - \epsilon\frac{\partial e}{\partial b_i} \tag{11}$$

where ϵ is the learning factor and t is the present iterative state. The CNN iteratively repeats the forward and backward propagation's till a minimum error is achieved. The Algorithm 2 explains the steps followed to train the convolution neural network.

2) Testing of the CNN: From the sensor data signals acquired, a few signals are used to train the CNN network the rest are used to test the classifier. On training the CNN network the optimal weights and bias are calculated to accurately classify the data as healthy, type-1 diabetes and type-2 diabetes. During the testing of the classifier, initially the features using the convolution and max-pooling layers are calculated and then given to the fully connected MLP layer. These optimal weights and bias calculated are then applied to the feature set in the decision function discussed above to predict the class. It is observed that more precise accuracy levels are obtained.

IV. Results and Discussion

This section explains the details on the experimental data collected from the sensory unit, the details of the implementation of the 1-D CNN algorithm, the performance evaluation of the proposed system and compares the computational cost of the CNN algorithm with other data processing algorithms.

A. Collection of Breath Signals Using the Test Setup

As explained in Section III, the gas concentrations from the breath are acquired through the MOS sensors. Each subject is initially asked to hold their breath and then blow into the mouthpiece of the transpace (gas chamber). The analog voltage of the sensors (MQ-3 and MQ-5) is found to increase and these signals are acquired by the arduino board. In order to store these signal value for further data classification the arduino is

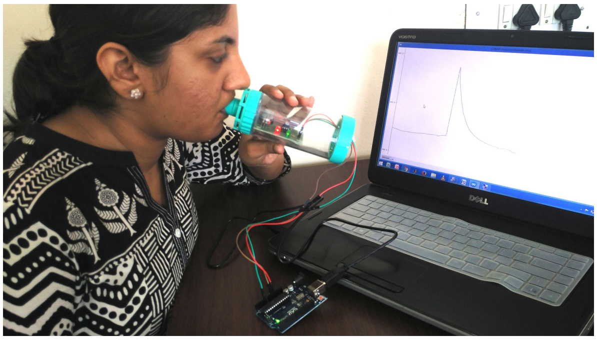

Fig. 4. Test setup and experimentation.

TABLE I
COLLECTED SUBJECT DATABASE COMPOSITION

Collected Data	Patient Characteristics		
Type	Healthy	Type 2	Type 1
No	11	9	5
Male/Female	6/5	7/2	3/2
Age	20–38	35–50	22–29
BMI	20–26.5	23–29	25–27
Range of HbA1c	4–5.8%	6.2–8%	6.8–8.5%
Duration of Diabetes	–	2–6years	4–9years
Range of Acetone Values Obtained(ppm)	0.24–0.71	1.89–4.29	9.2–14

programmed and the signals variations are recorded for 1000 seconds. The entire framework of the data acquisition system is shown in Fig. 4. These data are then given to the CNN for data classification. For the purpose of our study, a total of 25 signals containing a few healthy, type 1 diabetic and type 2 diabetic samples were collected. These reading were collected with the consent of the subject volunteers. This research was carried out following the principles of the Declaration of Helsinki. The Table I shows the detailed summary of the data collected for this analysis.

B. Implementation of the Proposed 1D CNN

These raw signals acquired from the data acquisition system are then given to the CNN classifier which is programmed through the Matlab environment. Feature extraction is achieved by convoluting the input with a kernel filter weight and then down-sampling through max-pooling. In this analysis a Gaussian based 1-D kernel filter is used. In our analysis, the CNN architecture is designed with 4 convolution and max-pooling layers and 2 MLP layers as the output layers to achieve maximum efficiency during the training of the real-time signal output collected from patients. After the signals propagate through these convolution and max-pooling layers, the feature maps extracted are of a reduced dimension. These features are further classified by training them using the fully connected MLP layers with back-propagation. Initially, the weights and bias of each neuron are set randomly and the errors are calculated. In our analysis there are three output classes and hence the output layer is set to size three. The number of neurons in the hidden layer is set as 20 and this gives the optimal performance according to the previous literature [25], [27]. As motioned in Section IV, in order to reduce the mean-square error the weight and bias values are updated. This process is done iteratively and

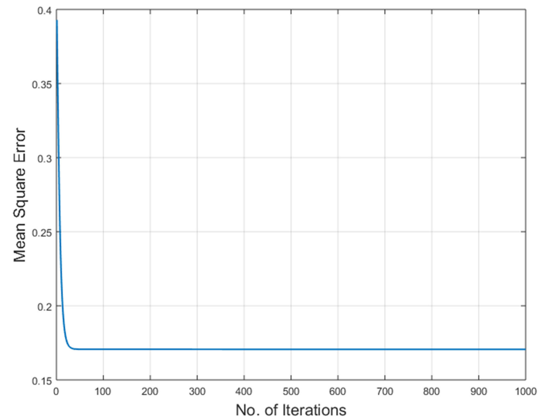

Fig. 5. Mean square error plot.

after each iteration, it is found that the mean-square error (MSE) reduces. Here in the proposed 1D CNN, the maximum number of back propagation iterations was set to 1000 with 0.001 as the learning factor. Initially the input feature sets are of a dimension of 1×1000. These signals are convoluted with the kernal filer and are downscaled by a scalar factor of 4. Finally the feature set is reduced down to the size of 1×4. Among the data samples collected from the sensory unit a total of 15 samples were used to train the proposed 1D CNN algorithm and the rest were used to test the classifier. The Fig. 5 shows the MSE plot and it is observed that the error is reduced after each iteration.

C. Leave-One Out Cross Validation of the Classifier

The robustness of a classifier model can be analyzed through the means of a statistical technique called cross-validation. This technique is often used to evaluate the classifiers generalization ability, which validates the performance of the classifier on previously unseen data. The main idea of this technique is to split a given data set into sub-samples which shall be considered for training and testing of the classifier respectively. This training data set is considered for estimating the free parameters and the predictive effectiveness of the fitted model of the classifier. In our analysis, we have employed the leave-one out cross validation method. This leave-one out cross validation technique is evaluated over the convoluted data set and the misclassification rate and mean square error of the classifier is calculated. It was observed that a misclassification rate of 0.0714 and a mean square error of 0.1436 were achieved. These are relatively low and confirming the efficiency of the classifiers.

D. Performance Evaluation of the Classifier

The performance of the classifier in this study is evaluated by calculating the Accuracy (ACC), Sensitivity (SE) and the Specificity (SP). These parameters are estimated by,

$$\mathrm{ACC} = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \tag{12}$$

$$\mathrm{SE} = \frac{T_N}{T_P + F_N} \tag{13}$$

$$\mathrm{SP} = \frac{T_N}{F_P + T_N} \tag{14}$$

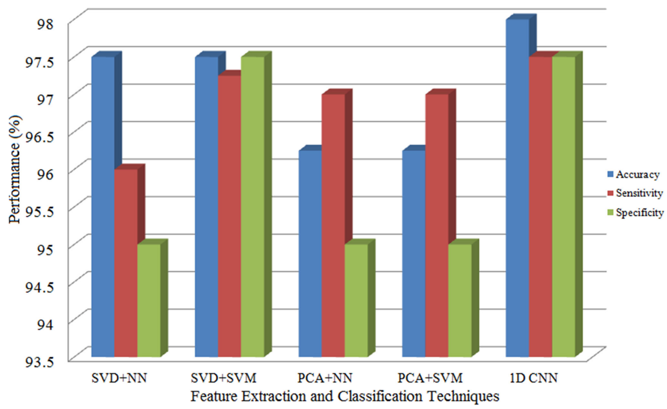

Fig. 6. Performance evaluation of the classifier.

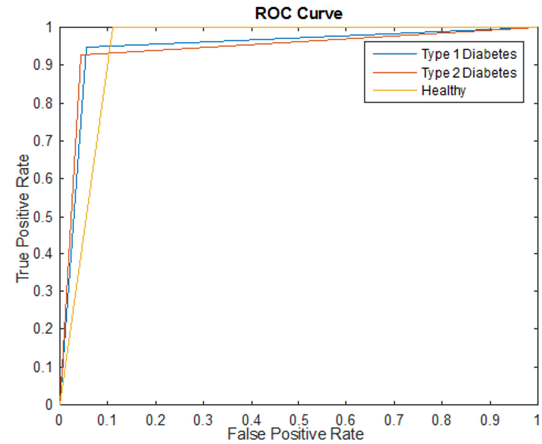

Fig. 7. ROC curve.

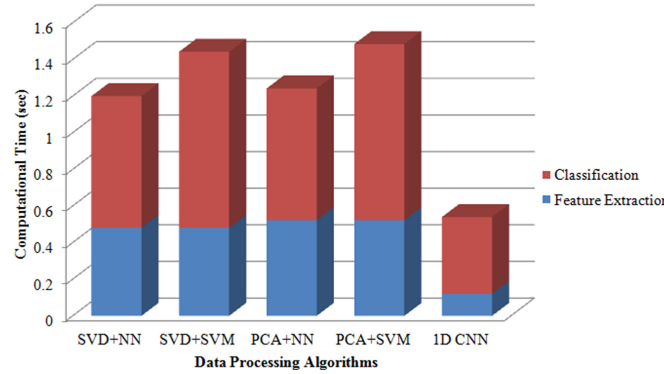

Fig. 8. Comparison of the computational time.

where T_P is the true positive value, T_N is the true negative value, F_P is the false positive value while F_N is the false negative value [28]. In order to evaluate the performance of the 1D CNN classifier these parameters are compared with other feature extraction and classification techniques. In this analysis two other feature extraction techniques are considered. The Singular Value Decomposition algorithm (SVD) [29] and the Principle Component Analysis (PCA) techniques are applied to the raw input data signal. Both these techniques are feature extraction and reduction techniques. The SVD technique decomposes the input signal to their orthogonal eigen vectors and values. It is a commonly used matrix decomposition technique. The PCA transforms the input signal values to a group of linear uncorrelated values which are termed principle components. These feature extraction techniques are applied to the input signals collected from the sensory device as mentioned in Section III. The analysis is also carried out using two other classifiers which are the Support Vector Machine (SVM) and the Neural Network (NN) with back propagation to classify the above calculated features. The feature obtained through PCA and SVD are separately applied to the SVM and NN based classifier and the performance parameters (ACC, SE and SP) are calculated for the same. Similar to the 1D CNN classifier, a portion of the data has been trained and the rest of the data samples have been used to test the classifier. The Fig. 6 shows the performance evaluation of all the classifiers. The Receiver Operating Characteristics (ROC) curve is also derived for the given classifier so as to validate the performance of the classifier. The Fig. 7 shows the ROC curve with the area under the curves as 0.9659, 0.9625 and 0.9644 for type 1, type 2 and healthy respectively. The ROC curve shows a good classification performance for the developed convolution neural network.

E. Computational Complexity

One of the main advantage of the CNN classifier is that this classifier reduces the computational cost. In this section, the system's computational time is compared with the other feature extraction and classification algorithms. In CNN, the average execution time taken to extract features from the raw signal is about 0.1203 sec. This computational time is very less compared to the time taken to extract features through SVD technique (0.4803 sec) and PCA (0.5226 sec). Considering the classification of the CNN system through back-propagation with a total of 1000 iteration, the maximum time for training is about 0.4194 sec. The total time to train the SVM and the Neural network was about 0.9617 sec and 0.7194 sec respectively. The total computational time taken by the CNN algorithm is about 0.5397 sec. The Fig. 8 compares the execution time all the algorithms and it is observed that the execution time of the CNN algorithm is significantly low.

One limitation in the proposed approach is that minor interference in the response obtained from the MOS sensors due to the effect of moisture present in breath has not been considered. Furthermore, in the present study, we have examined only a limited number of patient data samples. We plan to overcome these shortcomings by implementing compensation circuits to account for the moisture content in breath samples and also increase the patient data sets to cover a wider population group in future.

V. Conclusion

A non-invasive method of detecting diabetes has been discussed and analyzed in this paper. The analysis involves the implementation of a one dimension convolution neural network to classify raw signals obtained from an array of MOS sensor's. The CNN algorithm convolutes the raw data signal with a kernal based filter to produce a set of feature maps. The dimension of these feature maps are reduced by using a maximization operation over the data samples and thereby down-sampling the features. These reduced feature maps are then applied to a fully connected classifier which follows a similar mechanism as the multilayer perceptron based neural network. It is observed that this algorithm reduces the computational cost and the need for optimal feature section techniques. In this paper, the raw breath signals have been successfully classified using the proposed 1D CNN algorithm and the mean square error as well as other performance measurement parameters have been computed. These performance measures are also compared with other feature extraction and classification techniques. It has been found that the algorithm substantially reduces the mean square errors and optimizes the overall performance of the classifier. Our future work will focus on implementing the proposed algorithm with suitable enhancements from breath signals obtained from other gas sensors and to include more patient data samples.

References

[1] K. G. M. M. Alberti and P. Z. Zimmet, "Definition, diagnosis and classification of diabetes mellitus and its complications. Part 1: diagnosis and classification of diabetes mellitus provisional report of a WHO consultation.," *Diabetic Med.*, vol. 15, no. 7, pp. 539–553, 1998.

[2] P. Makaram, D. Owens, and J. Aceros, "Trends in nanomaterial-based non-invasive diabetes sensing technologies," *Diagnostics*, vol. 4, no. 2, pp. 27–46, 2014.

[3] W. Miekisch, J. K. Schubert, and G. F.. Noeldge-Schomburg, "Diagnostic potential of breath analysis-focus on volatile organic compounds," *Clin. Chim. Acta*, vol. 347, no. 1, pp. 25–39, 2004,

[4] K. Yan, D. Zhang, D. Wu, H. Wei, and G. Lu, "Design of a breath analysis system for diabetes screening and blood glucose level prediction," *IEEE Trans. Biomed. Eng.*, vol. 61, no. 11, pp. 2787–2795, Nov. 2014.

[5] C. Deng, J. Zhang, X. Yu, W. Zhang, and X. Zhang, "Determination of acetone in human breath by gas chromatography-mass spectrometry and solid-phase microextraction with on-fiber derivatization," *J. Chromatography B Anal. Technol. Biomed. Life Sci.*, vol. 810, no. 2, pp. 269–275, 2004.

[6] K. T. Moorhead *et al.*, "Classifying algorithms for SIFT-MS technology and medical diagnosis," *Comput. Methods Programs Biomed.*, vol. 89, no. 3, pp. 226–238, 2008.

[7] C. Wang and A. B. Surampudi, "An acetone breath analyzer using cavity ringdown spectroscopy: An initial test with human subjects under various situations," *Meas. Sci. Technol.*, vol. 19, no. 10, pp. 105–604, 2008.

[8] D. Guo, D. Zhang, N. Li, L. Zhang, and J. Yang, "A novel breath analysis system based on electronic olfaction," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 11, pp. 2753–2763, Nov. 2010.

[9] H. M. Saraoglu, A. O. Selvi, M. A. Ebeoglu, and C. Tasaltin, "Electronic nose system based on quartz crystal microbalance sensor for blood glucose and HbA1c levels from exhaled breath odor," *IEEE Sensors J.*, vol. 13, no. 11, pp. 4229–4235, Nov. 2013.

[10] K. Yan and D. Zhang, "A novel breath analysis system for diabetes diagnosis," in *Proc.Int. Conf. Comput. Healthcare*, 2012, pp. 166–170.

[11] K. T. Moorhead *et al.*, "Classifying algorithms for SIFT-MS technology and medical diagnosis," *Comput. Methods Programs Biomed.*, vol. 89, no. 3, pp. 226–238, 2008.

[12] S. Bedoui, R. Faleh, H. Samet, and A. Kachouri, "Electronic nose system and principal component analysis technique for gases identification," in *Proc. 10th Int. Multi-Conf. Syst. Signals Devices*, 2013, pp. 1–6.

[13] A. Loutfi, S. Coradeschi, G. K. Mani, P. Shankar, and J. B. B. Rayappan, "Electronic noses for food quality: A review," *J. Food Eng.*, vol. 144. pp. 103–111, 2015.

[14] K. Yan and D. Zhang, "Feature selection and analysis on correlated gas sensor data with recursive feature elimination," *Sensors Actuators, B, Chem.*, vol. 212, pp. 353–363, 2015.

[15] J. H. Cho and P. U. Kurup, "Decision tree approach for classification and dimensionality reduction of electronic nose data," *Sensors Actuators, B, Chem.*, vol. 160, no. 1, pp. 542–548, 2011.

[16] R. Gutierrez-Osuna and H. T. Nagle, "A method for evaluating data-preprocessing techniques for odor classification with an array of gas sensors," *IEEE Trans. Syst. Man, Cybern. Part B Cybern.*, vol. 29, no. 5, pp. 626–632, Oct. 1999.

[17] A. K. Srivastava, "Detection of volatile organic compounds (VOCs) using SnO2 gas-sensor array and artificial neural network," *Sensors Actuators, B, Chem.*, vol. 96, no. 12, pp. 24–37, 2003.

[18] D. Dutta, S. Ghosh, M. Narjinary, N. Bhattacharyya, and R. Bandyopadhyay, "Tin oxide based gas sensor array in electronic nose to monitor aroma of black tea," *Sens. Lett.*, vol. 14, no. 4, pp. 396–401, 2016.

[19] S. Bedoui, H. Samet, M. Samet, and A. Kachouri, "Gases identification with support vector machines technique (SVMs)," in *Proc. 1st Int. Conf. Adv. Technol. Signal Image Process.*, 2014, pp. 271–276.

[20] K. Yan, D. Zhang, D. Wu, H. Wei, and G. Lu, "Design of a breath analysis system for diabetes screening and blood glucose level prediction," *IEEE Trans. Biomed. Eng.*, vol. 61, no. 11, pp. 2787–2795, Nov. 2014.

[21] K. Yan and D. Zhang, "Blood glucose prediction by breath analysis system with feature selection and model fusion," in *Proc. 36th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2014, pp. 6406–6409.

[22] H. M. Saraoglu, A. O. Selvi, M. A. Ebeoglu, and C. Tasaltin, "Electronic nose system based on quartz crystal microbalance sensor for blood glucose and HbA1c levels from exhaled breath odor," *IEEE Sensors J.*, vol. 13, no. 11, pp. 4229–4235, Nov. 2013.

[23] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguist*, pp. 655–665, 2014.

[24] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Process. Lett.*, vol. 24, no. 3, pp. 279–283, Mar. 2017.

[25] S. Kiranyaz, T. Ince, and M. Gabbouj, "Real-time patient-specific ECG classification by 1-D convolutional neural networks," *IEEE Trans. Biomed. Eng.*, vol. 63, no. 3, pp. 664–675, Mar. 2016.

[26] D. Shi and Y. Gao, "A new method for identifying electromagnetic radiation sources using backpropagation neural network," *IEEE Trans. Electromagn. Compat.*, vol. 55, no. 5, pp. 842–848, Oct. 2013.

[27] T. Ince, S. Kiranyaz, L. Eren, M. Askar, and M. Gabbouj, "Real-time motor fault detection by 1-D convolutional neural networks," *IEEE Trans. Ind. Electron.*, vol. 63, no. 11, pp. 7067–7075, Nov. 2016.

[28] A. R. Fekr, M. Janidarmian, K. Radecka, and Z. Zilic, "Respiration disorders classification with informative features for m-health applications," *IEEE J. Biomed. Health Informat.*, vol. 20, no. 3, pp. 733–747, May 2016.

[29] J. A. Foster, J. G. McWhirter, M. R. Davies, and J. A. Chambers, "An algorithm for calculating the QR and singular value decompositions of polynomial matrices," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1263–1274, Mar. 2010.