Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2022.0122113

Optimization of the Electronic Nose Sensor Array for Asthma Detection Based on Genetic Algorithm

DAVA AULIA[1], RIYANARTO SARNO[1], (Senior Member, IEEE), SHINTAMI CHUSNUL HIDAYATI[1], and MUHAMMAD RIVAI[2], (Member, IEEE)

[1]Department of Informatics, Institut Teknologi Sepuluh Nopember, Kampus ITS Sukolilo-Surabaya 60111, Surabaya, Indonesia
[2]Department of Electrical Engineering, Institut Teknologi Sepuluh Nopember, Kampus ITS Sukolilo-Surabaya 60111, Surabaya, Indonesia

Corresponding author: Riyanarto Sarno (riyanarto@if.its.ac.id)

This work was supported in part by Asian Development Bank under Higher Education for Technology Innovation (ADB HETI) Project; the Indonesian Ministry of Education and Culture under Penelitian Terapan Unggulan Perguruan Tinggi (PTUPT) Program; and Institut Teknologi Sepuluh Nopember (ITS) under project scheme of the Publication Writing and Intellectual Property Right (IPR) Incentive Penulisan Publikasi dan Hak Kekayaan Intelektual (PPHKI).

ABSTRACT The human body releases several gases and volatile organic compounds through exhaled breath. This compound can be used as markers of lung disease, including asthma. An electronic nose can play a role in determining a patient's condition. The main problem that often occurs is the selection of appropriate sensors based on their characteristics and performance in detecting various gases to provide an optimal system while still providing high accuracy. Genetic algorithms have a good advantage in applying feature selection problems that can effectively solve noise and collinearity problems through three main genetic operators: crossover, mutation, and selection. This study aims to apply this method to determine the optimal number of gas sensors in identifying healthy people and asthma suspects through an exhaled breath. Several classification methods are combined with selected gas sensor arrays to obtain an optimized electronic nose, including support vector machine (SVM), random forest (RF), extreme gradient boosting (XGBoost), artificial neural network (ANN), one-dimensional convolutional neural network (1D-CNN), long short-term memory (LSTM), gated recurrent unit (GRU), 1D CNN-LSTM, and 1D CNN-GRU. These machine-learning approaches are usually used for electronic nose systems as highly accurate classification methods depending on the parameters. The experimental results showed that the genetic algorithm produced five gas sensors that provided a certain sensor pattern on the exhaled breath from the asthma suspects. Meanwhile, the 1D-CNN model was chosen as a classification method for the asthma dataset with an accuracy of 96.6%, a precision of 96.1%, a recall of 95.5%, and an F1-score of 95.6%.

INDEX TERMS Asthma, diseases, electronic nose, genetic algorithm.

I. INTRODUCTION

ASTHMA is a type of chronic disease disorder that can make the lungs unable to function properly. This disease can make it difficult for people to breathe. Asthma can be caused by exposure to cigarette smoke, dust, and environmental chemical compounds [1] that enter the respiratory organs and attack the lungs. This disease is not contagious. However, it is common in children and adults and is increasing globally. Based on the World Health Organization (WHO) statistics in 2018, this disease affected more than 339 million people worldwide [2]. Around hundreds of thousands died in the same year, most in low- and middle-income countries. Among the characteristics of a person with this symptom are coughing, wheezing, chest tightness, and shortness of breath.

Asthma biomarkers can be identified in several ways, including a collection of blood cells, urine, sputum, and exhaled breath [3]. Several biomarkers involve body contact that makes the patient feel irritated. Hence, the analysis of exhaled breath can be an alternative to non-invasive methods. Generally, a person's breath contains chemical compounds like oxygen (O_2), carbon dioxide (CO_2), water vapor (H_2O), nitric oxide (NO), hydrogen (H_2), hydrogen sulfide (H_2S), and volatile organic compounds (VOCs) [4]. These compounds can be measured by a specific tool called gas chromatography. However, this requires a long time and is expensive. As a result, it is only used in laboratories. For this reason, an electronic nose can be offered as an alternative tool to measure and analyze chemical compounds' content in exhaled breath

[5]. The electronic nose has been widely applied in various fields, such as the classification of perfume [6], food quality [7], and diseases in the human body [8], [9].

The volatile compounds in exhaled breath require a different sensor array with a good response signal. For this reason, the sensor array in the electronic nose system needs to be optimized while still providing high accuracy. Sensor selection is a problem that often occurs in its application to provide optimal sensors at lower costs [10]. As a result, a sensor selection method is needed. This method can eliminate sensors with similar performance as well as sensors that do not have a significant effect on the classification level. Therefore, a system involving an optimal number of sensors can produce a stable and reliable intelligent system. The sensor selection method can be handled by using correlation coefficients and cluster analysis [11]. Correlation coefficients and distinguishing performance value are calculated to remove sensor redundancy. Then the independence of the sensors is obtained through cluster analysis, and the number of sensors is confirmed. However, this method lacks generalizability and is difficult to implement.

Genetic algorithm (GA) is a global searching optimization technique inspired by natural selection. The method is like the process of genetic evolution, where genetics most suitable to the environment can survive longer [12]. This algorithm has the advantage of being able to solve both simple and difficult problems that have many goals and constraints. It exhibits obviously exceptional performance for a variety of problems. GA keeps the benefits of both metaheuristic search algorithms and stochastic optimization techniques. Compared to other evolutionary algorithms, this combination allows the genetic algorithm to achieve global optima in relatively fewer generations [13]. GA seeks to identify the individual with the best fitness value from a search space containing numerous viable solutions. Genotypes represent these solutions, and the genotypes with the best fitness values may be more likely to be selected as parents for the upcoming generation [14]. Genetic algorithms have three main operators, including selection, crossover and mutation. Chromosomes with low cost have a high mating probability by selection to bias the population to congregate towards the best solution. GA employs crossover and mutation operators to achieve information exchange between individuals and local search and does not rely on gradient information in order to find optimal solutions globally [15]. In addition, the method does not require domain knowledge or information about the structure of the problem being optimized. Additionally, it works well even when the objective function is not smooth, a circumstance in which derivative methods cannot be applied [14]. As a result, the GA is the most widely used evolutionary algorithm procedure for parameter optimization, and it is known to be highly effective for feature selection and reducing the dimensionality of the problem space.

In this study, the GA method was applied to obtain the optimal number of sensors in the electronic nose system without reducing the system's performance in predicting healthy people and asthma suspects. This paper is organized as follows. Section 1 covers the background of this study. Section 2 provides the proposed method and experimental setup. Section 3 presents the experimental results and discussion. Section 4 summarizes the essential results and future research.

II. METHODS AND MATERIALS

This section describes the methods and materials used in this study, including subject descriptions, proposed research design, electronic nose system, pattern recognition methods, steps to determine the optimal model architecture, statistical performance measurements, and sensor selection procedures.

A. SUBJECTS AND PROPOSED RESEARCH DESIGN

The exhaled breath data were obtained from 60 volunteer subjects consisting of 30 healthy people and 30 asthma suspects [16]. The subjects were between 30 and 60 years old, had no acute or chronic disease, and were non-smokers. Various levels of asthma severity were obtained from the global initiative for asthma (GINA) and asthma control test (ACT) measurements [2]. The severity levels are divided into three classes, namely controlled asthma, partly-controlled asthma, and uncontrolled asthma, where each severity has different symptom characteristics. Controlled asthma patients have mild symptoms and do not require serious treatment. Partly-controlled asthma patients have several severe symptoms that limit their activities, thus requiring moderate treatment. Patients with uncontrolled asthma need serious treatment and medication.

This study proposed a research design shown in Figure 1. The first step was preparing all volunteers to provide test samples in 1L (liter) Tedlar bags, in which all subjects, both healthy and asthma suspects, had been diagnosed by a pulmonary specialist at the hospital. An electronic nose consisting of seven gas sensors, a chamber, and an electric pump has been developed and used as a test device. All electronic components are implemented on a printed circuit board (PCB) to reduce circuit and environmental noise. This instrument is equipped with a neural network pattern recognition algorithm. Various configurations were tested and evaluated to obtain the optimal neural network model. The standardization and feature extraction of sensor signals were used to represent the sample characteristics grouped into three datasets, including binary and multi-class. The genetic algorithm, equipped with a support vector machine (SVM) as a fitness function, used the binary dataset to determine the number of sensor groups consisting of different types of sensors. All sensor groups were tested in the three datasets using the optimal neural network model. Each dataset provided optimal sensors based on high accuracy values. The dominant sensors were selected based on the type of sensors that frequently appeared in each dataset. The final step was testing the dominant sensors on the three datasets using different classification models, including SVM, random forest (RF), extreme gradient boosting (XGBoost), artificial neural network (ANN), one-dimensional convolutional neural network

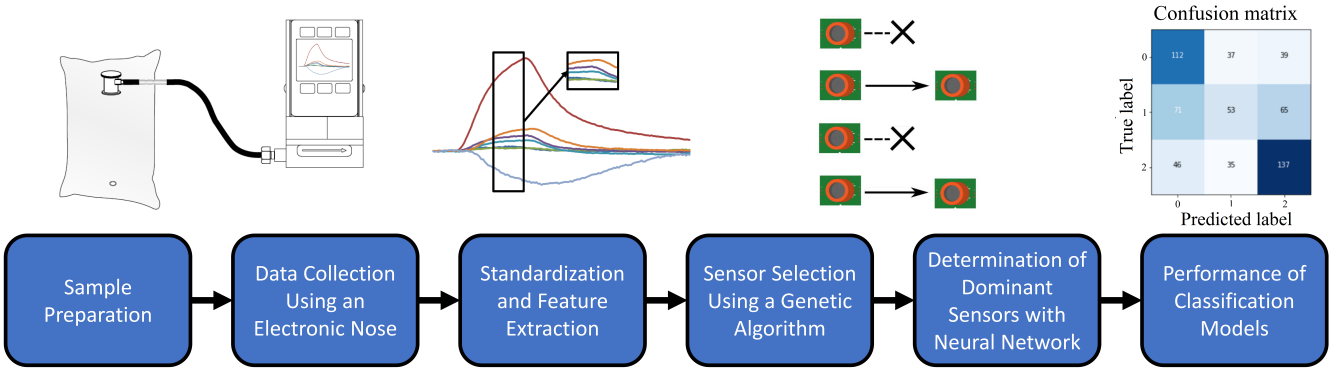

FIGURE 1. The proposed method.

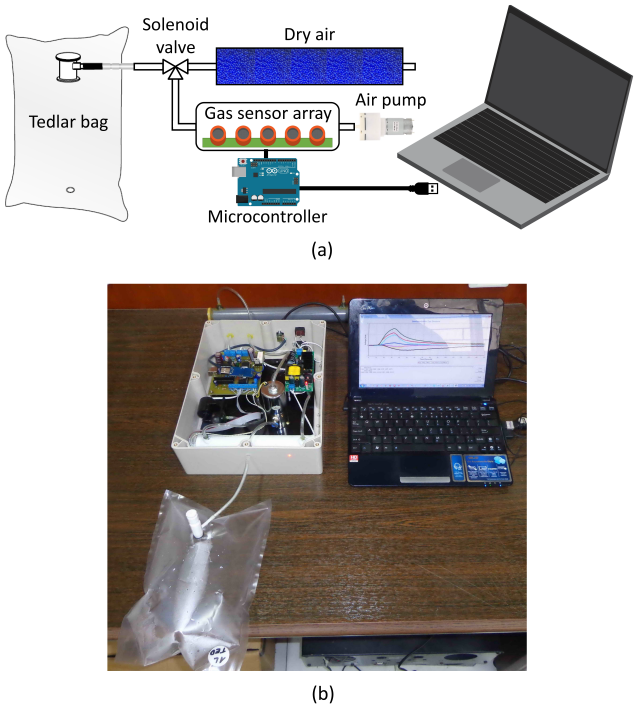

FIGURE 2. Electronic nose used in experiment: (a) block diagram, (b) realization.

(1D-CNN), long short-term memory (LSTM), gated recurrent unit (GRU), 1D CNN-LSTM, and 1D CNN-GRU. The model evaluation includes accuracy, precision, recall and F1-score.

B. THE ELECTRONIC NOSE

The electronic nose system used in this study consists of an array of gas sensors, a microcontroller, and a computer shown in Figure 2. In this case, a type of metal oxide gas sensor was used to record the exhaled breath. The advantages of using metal oxide semiconductors as gas sensing materials include mechanical robustness, low cost, simplicity of use, long lifetime, short response time, and high sensitivity to exhaled breath markers [17], [18]. The sensors are MQ-7, MQ-8, MQ-131, MQ-136, MQ-137, MQ-138, and TGS4161 shown in Table 1. Each sensor can respond to certain gases which are considered as biomarkers of lung disease. This is based on previous studies [8], [9], and characteristics related to the sensor's sensitivity and selectivity to the type of gas shown in their respective datasheets. The gas sensor array was allocated in the chamber with a volume of 240 mL to avoid ambient air to provide an accurate sensor response.

TABLE 1. Types and characteristics of gas sensors.

Sensor	Gas	Manufacturer
MQ-7	Carbon monoxide	Hanwei Electronics Co., Ltd
MQ-8	Hydrogen	
MQ-131	Ozone, Nitric oxide	
MQ-136	Hydrogen sulfide	
MQ-137	Ammonia	
MQ-138	Volatile organic compounds	
TGS4161	Carbon dioxide	Figaro Engineering Inc.

The sampling procedure began by flushing the sensor chamber for 15 seconds. Then, the sampling stage of the Tedlar bag was carried out for 40 seconds. The subsequent chamber flushing was carried out for 95 seconds, which can be used for the following sample. The output voltages of the gas sensors were converted by the analog to digital converter (ADC) on the microcontroller. Then, they were sent to the computer via universal serial bus (USB) communication stored in an excel file format. The datasets represented different response curves and were standardized using the StandardScaler method expressed as (1).

$$z = \frac{x - \mu}{\sigma} \tag{1}$$

where x is the sensor response, μ is the average value, and σ is the standard deviation.

C. CLASSIFICATION MODEL ALGORITHMS

Previous studies have applied the SVM model to several scenarios, including differentiating healthy people from asthma suspects and classifying asthma with different severity levels. The highest results were obtained with an accuracy value of

TABLE 2. SVM's kernel functions.

Type of kernel	Formula	Modified parameter
Linear	$f_k(x, x_i) = x \cdot x_i$	C
RBF	$f_k(x, x_i) = exp(-\gamma * \|x - x_i\|^2)$	C, γ
Sigmoid	$f_k(x, x_i) = tanh(\gamma * (x \cdot x_i) + coef)$	$C, \gamma, coef$
Polynomial	$f_k(x, x_i) = (\gamma * (x \cdot x_i) + coef)^d$	$C, \gamma, d, coef$

TABLE 3. SVM's parameter values range.

Parameter	Description
Kernel	Linear, RBF, sigmoid, and polynomial
C	1-100
γ	0.001-1
d	2-4
$coef$	0.0, 0.1

89.5%. However, it is still possible to improve these results by changing the model parameters. This method is known as hyperparameter optimization [19] by applying the grid search cross-validation (GridSearchCV) technique. The module is part of the scikit-learn, a software machine learning library for the Python programming language, which can help determine the best model parameters based on the highest accuracy. In this study, SVM, RF, GridSearchCV, and genetic algorithm were obtained from the scikit-learn library, while ANN, 1D-CNN, LSTM, and GRU were obtained from the Keras library.

1) Support Vector Machine (SVM)

SVM can be used for classification or regression in linear and non-linear cases based on the theory of statistical learning [20]. This model has a hyperplane function to separate classes. It also has some mathematical functions to solve certain calculations, namely the kernel [21]. This function can be linear, radial basis function (RBF), sigmoid, and polynomial, shown in Table 2. Each kernel has different parameters which can be modified, including the C function, gamma (γ), degree (d), and coefficient ($coef$).

Table 3 describes the values for each SVM parameter used in this study. The C function controls the classification error by trading in the correct training data based on the decision function's margin size. With the bigger C value, the decision function tries to separate the data classes by forming a tiny margin on the decision line, leading the model to overfit. However, the smaller C value results in a large margin, making some data in certain classes inseparable. This causes the model to become underfitting. Gamma is a parameter by which SVM considers the points when it makes the decision boundaries. High gamma only covers the data close to the boundary line. On the other hand, low gamma can consider distant data. SVM tries to cover all data using this decision line. With high gamma, the line will form a curve approaching each data class. This can provide better results. However, in many cases, this can cause the model to become overfitting. For this reason, the gamma value was set to less than one. In some cases, it can be difficult to determine the value of C and gamma parameters, which depend on the distribution of the data involved. The degree controls flexibility to the decision boundaries. The coefficient adjusts the independent term in the kernel function. In this experiment, the degree and coefficient values were set within a small range close to the default.

TABLE 4. RF's parameter values range.

Parameter	Description
Criterion	Gini, Entropy
Max depth	None
Max features	sqrt
Min samples leaf	1-3
Min samples split	1-5
N estimators (trees)	1-200

2) Random Forest (RF)

RF is an ensemble model which builds a set of decision trees to implement better performance in any classification or regression task [22]. The calculation is performed to evaluate the results of all decision trees and provide one output through voting results, known as the majority voting technique. This model has several parameters that can be modified, such as the number of trees, depth of trees, number of samples, feature types, and others [23]. The main parameter is the impurity criterion, a function to measure the split quality of the decision tree, namely gini and entropy. The gini calculation can be expressed by (2).

$$Gini = 1 - \sum_{i=1}^{N} (p_i)^2 \tag{2}$$

where p_i is the proportion of i^{th} class labels and N is the number of class labels. The gini index can determine the optimal splitter to build a pure decision tree. Table 4 describes the values for each RF parameter used in this study. The maximum depth was set to "none". Deeper trees can provide more data information. Likewise, more trees may allow better results. Other parameters were set close to the default.

3) eXtreme Gradient Boosting (XGBoost)

XGBoost has a similar architecture to RF. However, the model can provide better accuracy by employing gradient boosting [24]. This function is used as an optimization process to minimize a loss function or error value using a gradient descent algorithm. The additional parameters are learning rate, gamma (γ) [24], and others described in the eXtreme Gradient Boosting (XGBoost) library as an open-source software library that provides a regularizing gradient boosting for Python programming language. The entropy calculation can be expressed as (3).

$$Entropy(S) = -\sum_{i=1}^{N} p_i log_2(p_i) \tag{3}$$

TABLE 5. XGBoost's parameter values range.

Parameter	Description
Criterion	Gini, Entropy
Gamma	0.1-1
Max depth	None
Min child weight	1-4
Learning rate	0.01, 0.05, 0.1, 0.15, 0.2
N estimators (trees)	1-200

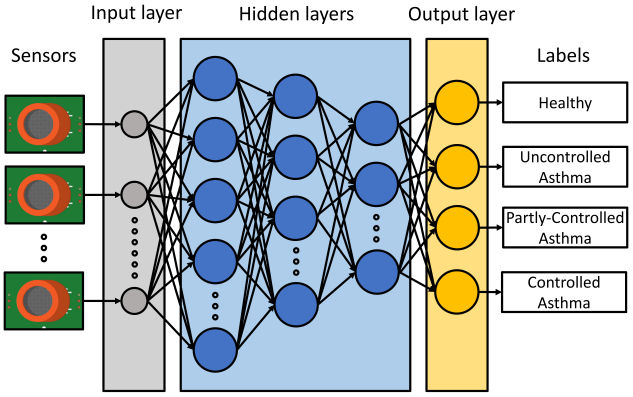

FIGURE 3. The proposed ANN architecture.

where S is a set of all instances in the dataset. Table 5 shows the XGBoost parameters used in this study.

A higher learning rate can make the model calculate faster, but most of the training data are not involved, which may not be accurately predicted. Gamma determines the minimum loss reduction to form a new tree split. The minimum value of the instance weight is required for each child. The minimum weight is applied to form a new node in a tree. These two parameters were set close to the default.

4) Artificial Neural Network (ANN)

ANN is a non-linear model that mimics biological neurons. This network consists of input, hidden, and output layers. The term deep neural network (DNN) emerges by applying many hidden layers between input and output [25]. The proposed ANN architecture is shown in Figure 3. This architecture involves two different methods, namely feed-forward and back-propagation [26]. The feed-forward acts as a one-way computation applied at the testing phase. As a result, this method cannot fix the weight value. Meanwhile, the back-propagation fixes the weight by returning the value to the previous layer repeatedly. This method is implemented in the training phase and usually as a fine-tuning of neural networks.

In this study, variations of the ANN architecture were applied to obtain a model with more optimal performance. This architecture consists of a number of hidden layers and their neurons described in Table 6. All neurons in this hidden layer are fully connected and implement the ReLu activation function. ReLu has a gradient of 1 for positive and 0 for negative, which is a better solution for vanishing gradients [27]. On the other hand, neurons in the output layer involve

TABLE 6. Configuration of ANN models.

Model	Number of hidden layers	Hidden layers with the neurons
ANN 1	1	[5]
ANN 2		[20]
ANN 3		[35]
ANN 4		[50]
ANN 5	2	[50] [5]
ANN 6		[50] [20]
ANN 7		[50] [35]
ANN 8		[50] [50]
ANN 9	3	[50] [50] [5]
ANN 10		[50] [50] [20]
ANN 11		[50] [50] [35]
ANN 12		[50] [50] [50]
ANN 13	4	[50] [50] [50] [5]
ANN 14		[50] [50] [50] [20]
ANN 15		[50] [50] [50] [35]
ANN 16		[50] [50] [50] [50]

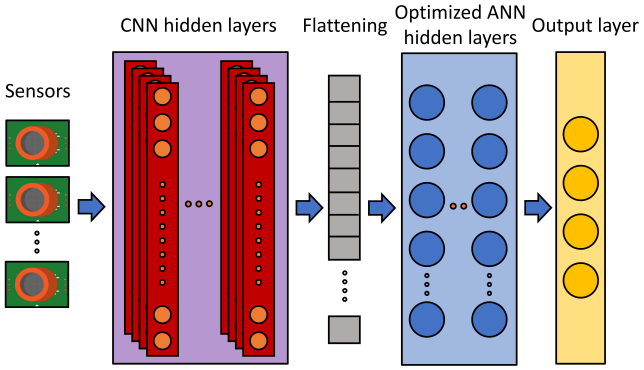

FIGURE 4. The proposed 1D-CNN architecture.

Sigmoid for the binary-class and Softmax for the multi-class. In the training phase, the number of batch sizes was set to 5, and the epoch was set to 250.

5) One-Dimensional Convolutional Neural Network (1D-CNN)

CNN is a type of classification model that has a convolutional layer consisting of a set of filters or kernels. It is used as a parameter for training image datasets in two or three-dimensional formats [28]. In addition, CNN also serves as a classification for the field of signal processing, which is named 1D-CNN [9]. This method has been implemented to classify diabetes [9].

In our study, 1D-CNN was applied because it has a more efficient architecture. The proposed 1D-CNN architecture is illustrated in Figure 4. This study determines the optimal architecture in the hidden layer of CNN. Table 7 shows varying architectural specifications for the 1D-CNN model. Each convolution layer involves several different filter sizes, including 16, 32, and 64. The use of the max pooling layer (MP) was also considered in this experiment. The kernel size was set to 2, and the ReLu activation function was implemented in each layer. Then, the output of the CNN hidden layers was flattened to form a one-dimensional vector and

TABLE 7. Configuration of 1D-CNN models.

Model	Number of convolutional layers	Hidden layers with filters and max pooling
1D-CNN 1	1	[16]
1D-CNN 2		[32]
1D-CNN 3		[64]
1D-CNN 4		[16] [MP]
1D-CNN 5		[32] [MP]
1D-CNN 6		[64] [MP]
1D-CNN 7	2	[16] [16]
1D-CNN 8		[32] [32]
1D-CNN 9		[64] [64]
1D-CNN 10		[16] [16] [MP]
1D-CNN 11		[32] [32] [MP]
1D-CNN 12		[64] [64] [MP]

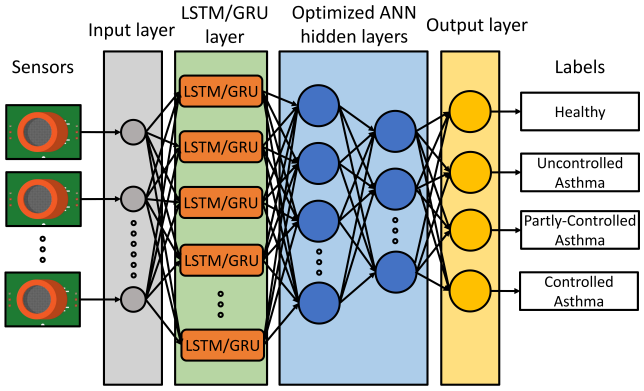

FIGURE 5. The proposed LSTM/GRU architecture.

forwarded to the optimal ANN hidden layers. The output layer is implemented to provide predictive results based on the classes.

6) Long Short-Term Memory Network (LSTM)

LSTM is a modified recurrent neural network (RNN) model which can improve the ability to predict information based on long periods [29]. LSTM can store and delete old data that are no longer relevant. With several gates in the architecture, this model can store and combine the collection of processed information. RNN and LSTM are any neural networks that contain one or more iterative (or cyclic) connections. Therefore, this model can solve the problem of vanishing gradients. LSTM is used to classify big data, which is usually in the form of text, language, and data sequence such as sensor signals, electrocardiogram (ECG), and electroencephalography (EEG) [30].

The proposed architecture of LSTM is shown in Figure 5. This study implemented a single layer of LSTM with varying numbers of memory cell units, including 16, 32, 64, and 128 described in Table 8. The ReLu activation function was also involved in this layer. The results of the LSTM layer were forwarded to the optimal ANN layers. The number of neurons in the output layer were adjusted to the number of classes in each dataset.

TABLE 8. Configuration of LSTM and GRU models.

Model	Hidden layer with the memory cell units
LSTM/GRU 1	[16]
LSTM/GRU 2	[32]
LSTM/GRU 3	[64]
LSTM/GRU 4	[128]

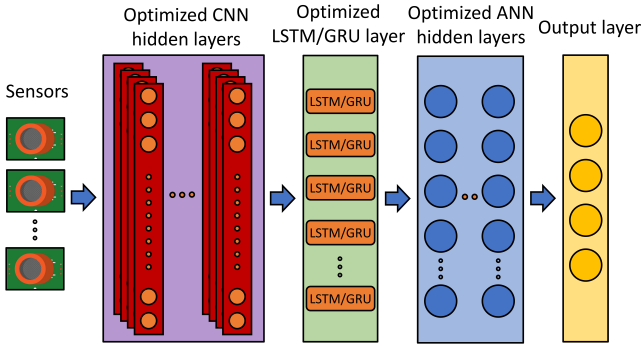

FIGURE 6. The proposed 1D CNN-LSTM/GRU architecture.

7) Gated Recurrent Unit (GRU)

GRU is the newer variant of RNN, built with a simpler architecture than LSTM by removing the output gate. This algorithm uses a single hidden state that combines the forget and input gates into one update gate [31]. The operations and gates within the GRU are also different. To solve the RNN's problems, GRU combines two gate operating mechanisms named update and reset gate. For this reason, the simpler architecture makes GRU use less memory and is faster than LSTM.

The proposed GRU architecture is similar to Figure 5, replacing the LSTM layer with a GRU layer and varying numbers of memory cell units, including 16, 32, 64, and 128, described in Table 8. The GRU layer was applied with ReLu activation function. The results from the GRU layer are forwarded to the optimal classification layer.

8) 1D CNN-LSTM/GRU

1D CNN-LSTM is a combination of 1D-CNN and LSTM which can improve the ability of deep neural networks modeling [32]. This method has been widely used because it has several advantages, including increased performance and sequential high dimensional input. This model has been used to detect epileptic seizures based on EEG signal analysis [33]. Whereas for 1D CNN-GRU has also been implemented to analyze potential health medications and generic molecules [34]. The 1D CNN-LSTM/GRU architecture proposed in our study is depicted in Figure 6. This model consists of the optimal hidden layers of 1D-CNN, LSTM/GRU, and ANN.

D. STATISTICAL PERFORMANCE MEASUREMENT

Performance measurement on the model requires a method based on statistical calculations [26]. The model can predict and classify new data based on ground truth values.

In this study, the ground truth consisted of a collection of medical record data labeled with healthy people and asthma suspects. Confusion matrix, which is a method for measuring performance in model classification algorithms, was applied in this study. This method consists of four important elements, which can provide four different values, including accuracy, precision, recall, and f1-score. These variables are true positive (TP), true negative (TN), false positive (FP), and false negative (FN), which can determine the quality of the classification model with the following conditions:

- True positive (TP) - the number of data points are correctly classified to the healthy class.
- True negative (TN) - the number of data points are correctly classified to the asthmatic class.
- False positive (FP) - the number of healthy data points are wrongly classified to the asthmatic class.
- False negative (FN) - the number of asthmatic data points are wrongly classified to the healthy class.

Several calculations were applied in this study and are explained as follows.

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} \tag{4}$$

Accuracy (4) is the ratio of correct predictions to the total test data. Models with high accuracy can predict new data correctly.

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

Precision (5) is the ability to identify relevant data. High precision and accuracy can represent a better model.

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

Recall (6) is the ability to find all relevant cases in a dataset. Recall can be explained by the success of the model in retrieving information.

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{7}$$

F1-Score (7) is an evaluation metric to determine the performance of the model accuracy against the dataset. F1-score is a calculation that involves a combination of precision and recall values.

E. GENETIC ALGORITHM FOR SENSOR SELECTION

Genetic algorithm generates solutions for optimization based on genetic evolution, such as mutation, crossover, and selection [12]–[15]. In this study, a genetic algorithm was applied to evaluate and eliminate gas sensors based on their fitness values. In this method, SVM was involved as a fitness function. The technique for determining the SVM model parameter is illustrated in Figure 7. This stage used a binary dataset that were not standardized. All sensors represent the dimensions of features and were involved in this process. The

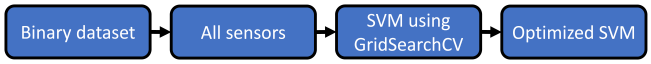

FIGURE 7. SVM as a GA's fitness function.

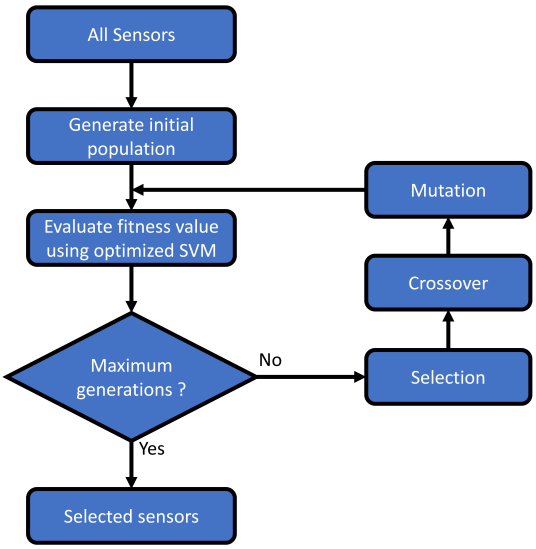

FIGURE 8. Genetic algorithm procedure.

number and types of candidate parameters needed were based on Table 3.

The sampling proportion of the dataset was divided using the stratified k-fold cross-validation by dividing the data of each class with the same proportion. This method consists of two parameters, namely random state and shuffle. In this case, the random state was set to "none" and the shuffle was assigned the value of "False" or "True". The results were obtained by comparing the output from the use of these scenarios. The GridSearchCV method helps determine the best parameters based on the highest accuracy value. The result of this process is the optimal SVM parameter as an estimator of the genetic algorithm. Figure 8 describes the sensor selection procedure using the genetic algorithm.

The first step involved all sensors as candidates. This procedure also used the binary dataset. The number of population variables was determined to form many candidate solutions.

TABLE 9. Genetic algorithm structure.

Parameter type	Parameter Values
Population size	100
Function of fitness	Optimized SVM
Number of generations	50
Probability of crossover	0.7
Independent probability for each attribute to be exchanged	0.1
Independent probability for each attribute to be mutated	0.05
Probability of mutation	0.001
Tournament size	3

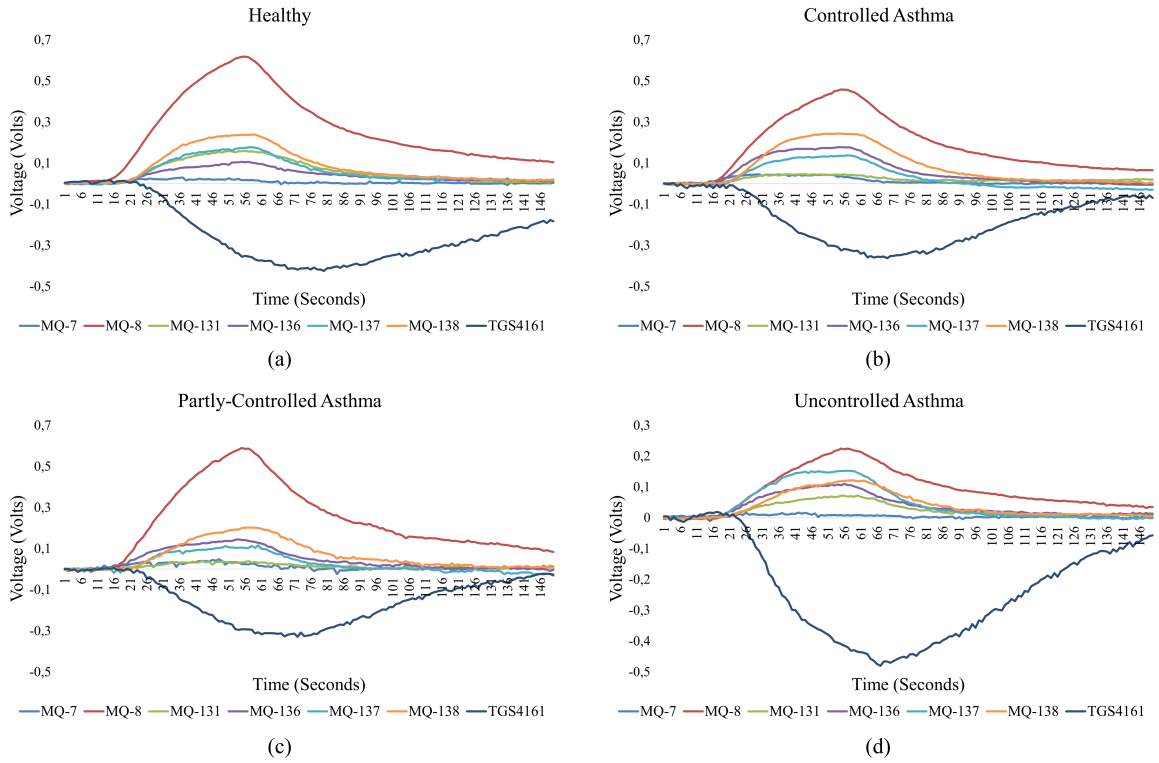

FIGURE 9. The gas sensor responses to exhaled breath of (a) healthy, (b) controlled asthma, (c) partly-controlled asthma, and (d) uncontrolled asthma.

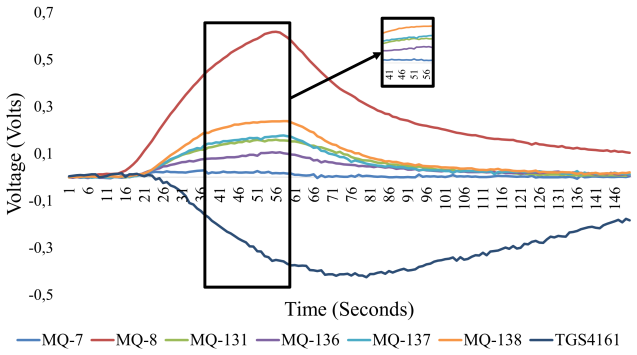

FIGURE 10. The sampling points representing the dataset.

Each candidate has a group of chromosome genes that can be mutated and changed. The fitness function evaluates all candidates to form optimal solutions based on this problem. This generation is also called iteration, which is used to improve the value of fitness through selection, crossover, and mutations in one cycle. Then, the process stopped until the last generation. The sensors were selected based on this genetic process, where the sensor with the smallest value was eliminated. Table 9 shows the genetic algorithm parameters.

The population was set to 100. A large population can allow the best solution. However, this process requires a longer time. The fitness function used the optimal SVM model. The number of generations was 50, which means this method processed 50 iterations. The crossover probability was set to 0.7. Large values can potentially produce new solutions by combining the features of both parents. Meanwhile, the crossover does not occur at low values. The new solution will be the same as the parents. The mutation probability was set to 0.001. Low values ensure that only a small portion of the population mutates in each generation [35]. The other parameters correspond to the default values.

III. EXPERIMENTAL RESULTS

In this section, the dataset, sensor selection, dominant sensor determination, and optimization of classification models are explained.

A. DATASET

All subject data were compiled based on the analysis of medical records. The asthma category involved three different types of severity, including controlled asthma, partly-controlled asthma, and uncontrolled asthma. All these categories were labeled based on ACT scores. The highest score was 25, representing controlled asthma suspects. The moderate score with a range of 20-24 was classified as partly-controlled asthma suspects. The lowest score under 20 was categorized as uncontrolled asthma suspects.

Each type of asthma consisted of 10 subjects. For this reason, all datasets were categorized into binary and multi-class classes. The binary class consisted of healthy and asthma datasets. The multi-class consisted of healthy and asthma with

TABLE 10. The number of subjects to provide the dataset.

Class	Total subjects	Total data
Healthy	30	180
Controlled Asthma	10	60
Partly-Controlled Asthma	10	60
Uncontrolled Asthma	10	60
All	60	360

TABLE 11. SVM evaluation for GA's fitness function.

Kernel	C	γ	coef	d	Shuffle	Accuracy (%)
Linear	67	-	-	-	False	88.6
RBF	67	0.185	-	-	False	87.8
Sigmoid	95	0.086	0.1	-	False	85.3
Polynomial	39	0.99	0.0	2	False	88.8
Linear	61	-	-	-	True	94.2
RBF	55	0.842	-	-	True	95.0
Sigmoid	65	0.076	0.0	-	True	93.3
Polynomial	77	0.86	0.1	2	True	93.9

different severity levels datasets.

Examples of gas sensor responses for each class are shown in Figure 9. Each category has a specific sensor response curve. In this study, the dataset was represented by several sampling points as a feature of each curve shown in the Figure 10. The data were taken between the 45th and 55th seconds where each curve did not overlap. This period is extracted into six samples (i.e., 45th, 47th, 49th, 51st, 53th and 55th seconds) representing the measurement data. The total data can be seen in Table 10. This varied data is needed to avoid overfitting problems in machine learning algorithms [36].

B. SENSOR SELECTION USING GA

This study used the stratified k-fold cross-validation (k = 5), which explains the proportion of data divided into five parts with a percentage of 80% for training and 20% for testing. The shuffle function was also applied, which was very useful for the model to study the proportion of data that were divided randomly. This was used to evaluate the best SVM model. Table 11 shows the results of the SVM evaluation with various parameters.

Based on these results, SVM with RBF kernel implementing the shuffled data was chosen as a fitness function for the genetic algorithm with an average accuracy of 95.0%. The following step was the sensor selection using the genetic algorithm. The stratified k-fold cross-validation with the shuffled data was also applied to the binary class dataset. Table 12 shows the number of sensors consisting of selected gas sensor types. However, this method provides a sensor group consisting of various gas sensors in each evaluation. This uncertainty will increase as the number of sensors decreases. For this reason, several evaluations were needed based on the highest fitness value to ensure consistent sensor compositions.

C. THE OPTIMAL ANN MODEL DETERMINATION

The performance of ANN depends on the selection of appropriate parameters. Therefore, it is necessary to find the optimal architectural model. An evaluation was carried out from the simplest to the more complex ANN architecture to obtain the appropriate parameters described in Table 6. For example, the simplest ANN architecture had one hidden layer containing five neurons besides the input and output layers. The more complex architecture had a higher number of hidden layers and their neurons. For this purpose, a standardized binary class dataset was employed in each ANN architecture to determine its performance, whose results are shown in Table 13. In this case, all sensors were involved in this evaluation process. The results indicated that ANN 7 and ANN 8 had the highest accuracy of 99.0%. As a result, the first architecture was chosen because it has a smaller number of neurons.

TABLE 12. Selected sensors using the GA method.

No	Number of sensors	Sensors
1	7	MQ-7, MQ-8, MQ-131, MQ-136, MQ-137, MQ-138, TGS4161
2	6	MQ-7, MQ-8, MQ-136, MQ-137, MQ-138, TGS4161
3	5	MQ-8, MQ-136, MQ-137, MQ-138, TGS4161
4	4	MQ-8, MQ-131, MQ-138, TGS4161
5	3	MQ-8, MQ-131, TGS4161
6	2	MQ-131, TGS4161

TABLE 13. Performance evaluation of each ANN model of binary class for seven sensors.

Model	Accuracy (%)	Model	Accuracy (%)
ANN 1	95.0	ANN 9	98.5
ANN 2	94.0	ANN 10	94.0
ANN 3	97.5	ANN 11	98.0
ANN 4	96.5	ANN 12	97.5
ANN 5	95.5	ANN 13	97.0
ANN 6	98.5	ANN 14	97.5
ANN 7	99.0	ANN 15	98.0
ANN 8	99.0	ANN 16	98.5

D. DOMINANT SENSOR DETERMINATION

The optimal neural network model is used to evaluate the sensors that often appear in each dataset for all sensor groups selected by GA. Table 14 summarizes the performance of ANN on all datasets in each sensor group. In the binary class, the group of five sensors (i.e., MQ-8, MQ-136, MQ-137, MQ-138, and TGS4161) had the highest average accuracy of 98.2%. Meanwhile, for the three-class and four-class datasets, a group of seven sensors (i.e., MQ-7, MQ-8, MQ-131, MQ-136, MQ-137, MQ-138, and TGS4161) had the highest accuracy of 98.8%, and 95.0%, respectively. The results indicated that the types of gas sensors that often appeared in each dataset were MQ-8, MQ-136, MQ-137, MQ-138, and TGS4161.

E. OPTIMIZATION OF CLASSIFICATION MODELS

After the five dominant sensors were determined, it is necessary to evaluate them with several classification models to obtain a reliable system. This study involved SVM, RF, XG-

TABLE 14. The ANN performance for determining the dominant sensors.

The number of sensors	Accuracy in each category (%)		
	Two classes	Three classes	Four classes
7	98.2	98.8	95.0
6	98.2	93.6	92.6
5	98.2	95.0	92.8
4	95.2	87.4	90.6
3	90.6	78.8	85.6
2	90.0	51.2	69.2

TABLE 15. Performance of 1D-CNN, LSTM and GRU models using the dominant sensors.

Model	Accuracy (%)	Model	Accuracy (%)
1D-CNN 1	99.5	1D-CNN 7	98.0
1D-CNN 2	97.7	1D-CNN 8	99.0
1D-CNN 3	99.0	1D-CNN 9	97.5
1D-CNN 4	97.2	1D-CNN 10	94.5
1D-CNN 5	97.2	1D-CNN 11	94.5
1D-CNN 6	98.2	1D-CNN 12	94.2
LSTM 1	99.0	GRU 1	96.4
LSTM 2	98.5	GRU 2	96.6
LSTM 3	99.2	GRU 3	98.6
LSTM 4	98.2	GRU 4	98.6

Boost, ANN, 1D-CNN, LSTM, GRU, 1D CNN-LSTM, and 1D CNN-GRU models with various parameters. To obtain the best performance for each configuration, all models were tested in the binary class with the dominant sensors.

The optimal SVM configuration was according to Table 3. SVM with the parameters, namely RBF kernel, C = 11, and Gamma = 0.706 gave the best performance for this model. The best RF configuration was based on Table 4, using the parameters, namely gini criteria (i.e., max depth = none; max feature = sqrt; min sample leaf = 1; min sample split = 2; estimator (tree) = 84).

XGboost, with the parameters described in Table 5, gave the best performance (gamma = 0.4, learning rate = 0.1, min child weight = 1, estimator = 52). Meanwhile, the performance of the 1D-CNN, LSTM/GRU models was determined based on the architectures described in Table 7, and Table 8, respectively. The fully connected layer in all three models applied the optimal ANN architecture (ANN 7). Table 15 presents the accuracy of each architecture of the three models. Among the candidates, 1D-CNN 1 leads with an accuracy score of 99.5%. In contrast, LSTM 3 has the highest percentage of 99.2%. GRU 3 and GRU 4 provide the same accuracy value of 98.6%. For this reason, GRU 3 was chosen because it has fewer memory cell units. In addition, the 1D CNN-LSTM and 1D CNN-GRU models, each consisting of three optimal architectures, are also involved in the evaluation purposes. The nine selected models were tested with the entire asthma dataset.

Table 16 shows the evaluation results of nine models for all asthma datasets. In the binary class, 1D-CNN demonstrated superior performance with an accuracy of 99.7%. However, all models tended to provide higher average accuracy compared to other datasets due to the smaller number of classes. In the three classes, the six models had accuracy above 95%,

TABLE 16. Evaluations of prediction performance for each model.

Model	Average (%)			
	Accuracy	Precision	Recall	F1-Score
Two classes				
SVM	98.3	98.3	98.3	98.3
RF	96.3	96.3	96.3	96.3
XGBoost	95.6	96.0	95.6	95.6
ANN	98.2	98.2	98.2	98.2
1D-CNN	99.7	99.7	99.7	99.7
LSTM	98.5	98.5	98.5	98.5
1D CNN-LSTM	98.2	98.2	98.2	98.2
GRU	98.1	98.1	98.1	98.1
1D CNN-GRU	97.4	97.5	97.4	97.4
Three classes				
SVM	95.5	95.5	95.5	95.5
RF	83.5	84.0	83.5	83.5
XGBoost	83.0	84.5	83.0	83.5
ANN	95.0	95.4	95.0	95.0
1D-CNN	96.0	96.5	96.0	96.0
LSTM	95.2	95.5	95.2	95.2
1D CNN-LSTM	96.2	96.2	96.2	96.0
GRU	95.5	95.7	95.5	94.7
1D CNN-GRU	94.8	94.9	94.8	94.8
Four classes				
SVM	92.5	91.5	88.5	89.0
RF	93.0	93.5	90.5	91.5
XGBoost	92.5	94.0	89.0	91.0
ANN	92.8	90.6	91.2	90.8
1D-CNN	94.0	92.2	90.7	91.2
LSTM	95.0	93.2	92.7	92.7
1D CNN-LSTM	93.7	91.2	91.7	91.2
GRU	93.3	91.8	90.3	90.7
1D CNN-GRU	93.3	93.4	90.2	91.1

TABLE 17. Testing accuracy and training time for 1D-CNN, LSTM and their combined models.

Model	Accuracy (%)	Training Time (s)
1D-CNN	96.6	15.4
LSTM	96.2	47.9
1D CNN-LSTM	96.0	48.5

including SVM, ANN, 1D-CNN, LSTM, 1D CNN-LSTM and GRU.

On the other hand, in the four classes, 1D-CNN, LSTM, and 1D CNN-LSTM dominated the accuracy scores. This explains that the architecture of 1D-CNN and LSTM and their combinations provided a more reliable level of classification. However, 1D-CNN was preferred because of its simpler construction with shorter training time as described in Table 17. LSTM involved more complex neurons which include the forget gate. Therefore, the 1D-CNN model was used to describe the confusion matrix for all dataset classes presented in Figure 11.

IV. CONCLUSION

This study applied the electronic nose method to classify healthy people and asthma suspects with different levels of severity. The electronic nose system involved a sensor array consisting of seven gas sensors. Meanwhile, the genetic algorithm was used to determine the optimal number of sensors that involved the SVM model as its fitness function. Several

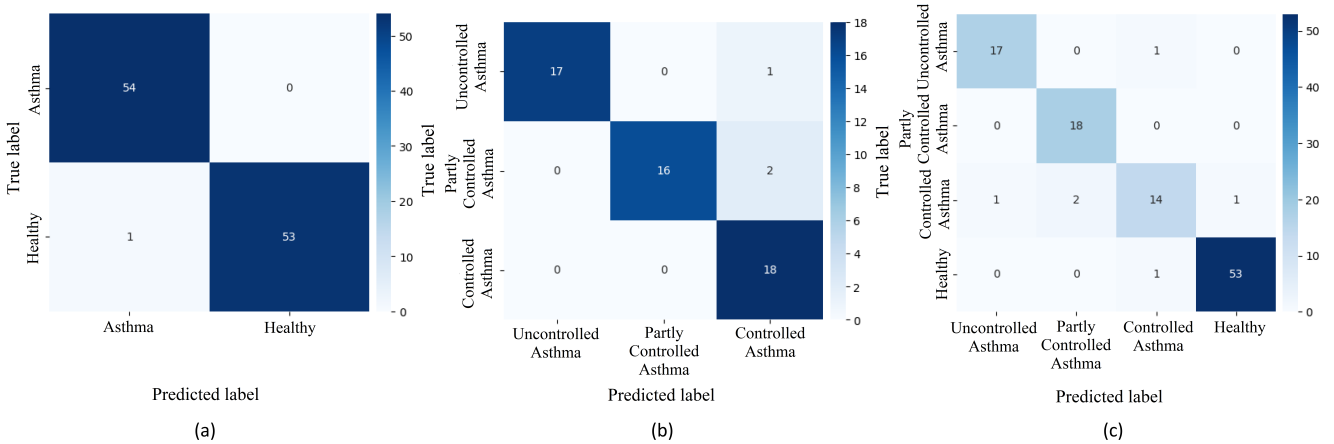

FIGURE 11. **The 1D-CNN prediction for (a) binary class, (b) three classes, and (c) four classes.**

classification models were used to evaluate the performance of the selected sensors. The experimental results showed that the genetic algorithm could generate five gas sensors, namely MQ-8, MQ-136, MQ-137, MQ-138, and TGS4161, producing a specific sensor pattern to the exhaled breath of the asthma suspects. Meanwhile, for classification purposes, the 1D-CNN, LSTM, and 1D CNN-LSTM models had an accuracy of at least 96%. However, the 1D-CNN model was preferred as a classifier for asthma datasets with a high level of accuracy of 96.6%, a precision of 96.1%, a recall of 95.5%, an F1-score of 95.6% and the shortest training time. This prototype still has a large size, including a sensor chamber and computer. For future research, we will reduce the size of the sensor chamber to accelerate the response of the gas sensors. In addition, the classification model will be implemented on a single board computer to provide a compact and portable electronic nose system.

ACKNOWLEDGMENT

The authors would like to thank Hari Agus Sujono from the Department of Electrical Engineering, Institut Teknologi Adhi Tama Surabaya and Muhammad Amin from the Department of Pulmonology and Respiratory Medicine, Airlangga University, Surabaya, who provided the dataset of asthma patients from Dr. Soetomo General Hospital, Surabaya, Indonesia.

REFERENCES

[1] T. Mattila, T. Santonen, H. R. Andersen, A. Katsonouri, T. Szigeti, M. Uhl, W. Wąsowicz, R. Lange, B. Bocca, F. Ruggieri, M. Kolossa-Gehring, D. A. Sarigiannis, and H. Tolonen, "Scoping review - The association between asthma and environmental chemicals," *Int. J. Environ. Res. Public Health*, vol. 18, no. 1323, pp. 1–13, 2021.

[2] T. Mulugeta, T. Ayele, G. Zeleke, and G. Tesfay, "Asthma control and its predictors in Ethiopia: Systematic review and meta-analysis," *PLoS One*, vol. 17, no. 1, pp. 1-16, 2022.

[3] A. Tiotiu, "Biomarkers in asthma: state of the art," *Asthma Res. Pract.*, vol. 4, no. 10, pp. 1-10, 2018.

[4] H. Kiss, Z. Örlős, Á. Gellért, Z. Megyesfalvi, A. Mikáczó, A. Sárközi, A. Vaskó, Z. Miklós, and I. Horváth, "Exhaled biomarkers for point-of-care diagnosis: Recent advances and new challenges in breathomics," *Micromachines*, vol. 14, no. 391, pp. 1–29, 2023.

[5] L. F. de Oliveira, C. Mallafré-Muro, J. Giner, L. Perea, O. Sibila, A. Pardo, and S. Marco, "Breath analysis using electronic nose and gas chromatography-mass spectrometry: A pilot study on bronchial infections in bronchiectasis," *Clin. Chim. Acta*, vol. 526, pp. 6–13, 2022.

[6] E. Tsuchitani, M. Nomura, M. Ota, E. Osada, N. Akiyama, Y. Kanegae, T. Iwamoto, R. Yamaoka, and Y. Manome, "Recording the fragrance of 15 types of medicinal herbs and comparing them by similarity using the electronic nose FF-2A," *Chemosensors*, vol. 10, no. 20, pp. 1–13, 2022.

[7] R. Sarno, K. Triyana, S. I. Sabilla, D. R. Wijaya, D. Sunaryono, and C. Fatichah, "Detecting pork adulteration in beef for halal authentication using an optimized electronic nose system," *IEEE Access*, vol. 8, pp. 221700–221711, 2020.

[8] H. Hendrick, R. Hidayat, G.-J. Horng, and Z.-H. Wang, "Non-invasive method for tuberculosis exhaled breath classification using electronic nose," *IEEE Sens. J.*, vol. 21, no. 9, pp. 11184–11191, 2021.

[9] Misbah, M. Rivai, F. Kurniawan, Z. Muchidin, and D. Aulia, "Identification of diabetes through urine using gas sensor and convolutional neural network," *Int. J. Intell. Eng. Syst.*, vol. 15, no. 1, pp. 520–529, 2022.

[10] P. Borowik, L. Adamowicz, R. Tarakowski, K. Siwek, and T. Grzywacz, "Odor detection using an e-nose with a reduced sensor array," *Sensors*, vol. 20, no. 3542, pp. 1–20, 2020.

[11] J. Wang, C. Zhang, M. Chang, W. He, X. Lu, S. Fei, and G. Lu, "Optimization of electronic nose sensor array for tea aroma detecting based on correlation coefficient and cluster analysis," *Chemosensors*, vol. 9, no. 266, pp. 1–20, 2021.

[12] M. Rostami, K. Berahmand, and S. Forouzandeh, "A novel community detection based genetic algorithm for feature selection," *J. Big Data*, vol. 8, no. 2, pp. 1–27, 2021.

[13] N. Ghatasheh, I. Altaharwa, and K. Aldebei, "Modified genetic algorithm for feature selection and hyper parameter optimization: Case of XGBoost in spam prediction," *IEEE Access*, vol. 10, pp. 84365–84383, 2022.

[14] T. Liu, W. Zhang, P. McLean, M. Ueland, S. L. Forbes, and S. W. Su, "Electronic nose-based odor classification using genetic algorithms and fuzzy support vector machines," *Int. J. Fuzzy Syst.*, vol. 20, pp. 1309–1320, 2018.

[15] D. Yang, Z. Yu, H. Yuan, and Y. Cui, "An improved genetic algorithm and its application in neural network adversarial attack," *PLoS One*, vol. 17, no. 5, pp. 1–17, 2022.

[16] H. A. Sujono, M. Rivai, and M. Amin, "Asthma identification using gas sensors and support vector machine," *Telkomnika*, vol. 16, no. 4, pp. 1468–1480, 2018.

[17] F. Vajhadin, M. Mazloum-Ardakani, and A. Amini, "Metal oxide-based gas sensors for the detection of exhaled breath markers," *Med. Devices Sensors*, vol. 4, pp. 1–11, 2021.

[18] D. R. Wijaya, R. Sarno, and E. Zulaika, "Gas concentration analysis of resistive gas sensor array," in *2016 International Symposium on Electronics and Smart Devices (ISESD)*, 2016, pp. 337–342.

[19] K. E. Hoque and H. Aljamaan, "Impact of hyperparameter tuning on machine learning models in stock price forecasting," *IEEE Access*, vol. 9, pp. 163815–163830, 2021.
[20] T. Y. Wen and S. A. M. Aris, "Hybrid approach of EEG stress level classification using k-means clustering and support vector machine," *IEEE Access*, vol. 10, pp. 18370–18379, 2022.
[21] J. Yang, Z. Wu, K. Peng, P. N. Okolo, W. Zhang, H. Zhao, and J. Sun, "Parameter selection of Gaussian kernel SVM based on local density of training set," *Inverse Probl. Sci. Eng.*, vol. 29, no. 4, pp. 536–548, 2021.
[22] M. Sheykhmousa, M. Mahdianpari, H. Ghanbari, F. Mohammadimanesh, P. Ghamisi, and S. Homayouni, "Support vector machine versus random forest for remote sensing image classification: A meta-analysis and systematic review," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 13, pp. 6308–6325, 2020.
[23] F. Hidayat and T. M. S. Astsauri, "Applied random forest for parameter sensitivity of low salinity water injection (LSWI) implementation on carbonate reservoir," *Alexandria Eng. J.*, vol. 61, no. 3, pp. 2408–2417, 2022.
[24] M. Chen, Q. Liu, S. Chen, Y. Liu, C.-H. Zhang, and R. Liu, "XGBoost-based algorithm interpretation and application on post-fault transient stability status prediction of power system," *IEEE Access*, vol. 7, pp. 13149–13158, 2019.
[25] Malikhah, R. Sarno, S. Inoue, M. S. H. Ardani, D. P. Purbawa, S. I. Sabilla, K. R. Sungkono, C. Fatichah, D. Sunaryono, A. Bakhtiar, Libriansyah, C. R. S. Prakoeswa, D. Tinduh, and Y. Hernaningsih, "Detection of infectious respiratory disease through sweat from axillary using an e-nose with stacked deep neural network," *IEEE Access*, vol. 10, pp. 51285–51298, 2022.
[26] M. Rivai, F. Budiman, D. Purwanto, M. S. A. Al Baid, Tukadi, and D. Aulia, "Discrimination of durian ripeness level using gas sensors and neural network," in *Procedia Computer Science, Elsevier B.V.*, 2022, pp. 677–684.
[27] Y. Yu, K. Adu, N. Tashi, P. Anokye, X. Wang, and M. A. Ayidzoe, "RMAF: Relu-memristor-like activation function for deep learning," *IEEE Access*, vol. 8, pp. 72727–72741, 2020.
[28] R. Agyeman, M. Rafiq, H. K. Shin, B. Rinner, and G. S. Choi, "Optimizing spatiotemporal feature learning in 3D convolutional neural networks with pooling blocks," *IEEE Access*, vol. 9, pp. 70797–70805, 2021.
[29] R. A. Priyantina and R. Sarno, "Sentiment analysis of hotel reviews using latent dirichlet allocation, semantic similarity and LSTM," *Int. J. Intell. Eng. Syst.*, vol. 12, no. 4, pp. 142–155, 2019.
[30] K. Yamamoto and T. Ohtsuki, "Non-contact heartbeat detection by heartbeat signal reconstruction based on spectrogram analysis with convolutional LSTM," *IEEE Access*, vol. 8, pp. 123603–123613, 2020.
[31] K. E. ArunKumar, D. V. Kalaga, C. M. S. Kumar, M. Kawaji, and T. M. Brenza, "Comparative analysis of gated recurrent units (GRU), long short-term memory (LSTM) cells, autoregressive integrated moving average (ARIMA), seasonal autoregressive integrated moving average (SARIMA) for forecasting COVID-19 trends," *Alexandria Eng. J.*, vol. 61, no. 10, pp. 7585–7603, 2022.
[32] Y. A. M. Alsumaidaee, C. T. Yaw, S. P. Koh, S. K. Tiong, C. P. Chen, T. Yusaf, A. N. Abdalla, K. Ali, and A. A. Raj, "Detection of corona faults in switchgear by using 1D-CNN, LSTM, and 1D-CNN-LSTM methods," *Sensors*, vol. 23, no. 3108, pp. 1–19, 2023.
[33] G. Xu, T. Ren, Y. Chen, and W. Che, "A one-dimensional CNN-LSTM model for epileptic seizure recognition using EEG signal analysis," *Front. Neurosci.*, vol. 14, no. 578126, pp. 1–9, 2020.
[34] D. O. Oyewola, E. G. Dada, O. Emebo, and O. O. Oluwagbemi, "Using deep 1D convolutional grated recurrent unit neural network to optimize quantum molecular properties and predict intramolecular coupling constants of molecules of potential health medications and other generic molecules," *Appl. Sci.*, vol. 12, no. 7228, pp. 1–19, 2022.
[35] S. N. Langazane and A. K. Saha, "Effects of particle swarm optimization and genetic algorithm control parameters on overcurrent relay selectivity and speed," *IEEE Access*, vol. 10, pp. 4550–4567, 2022.
[36] K. Alomar, H. I. Aysel, and X. Cai, "Data augmentation in classification and segmentation: A survey and new strategies," *J. Imaging*, vol. 9, no. 46, pp. 1–26, 2023.

DAVA AULIA is currently pursuing a Master's degree in Informatics Department, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia. He is interested in the fields of artificial intelligence, deep learning, internet of things (IoT), embedded systems, and gas sensors.

RIYANARTO SARNO (Senior Member, IEEE) received a Doctorate (Ph.D.) degree in 1992. He is currently a Professor of the Department of Informatics, Institut Teknologi Sepuluh Nopember (ITS). He is interested in research projects on machine learning, internet of things, knowledge engineering, enterprise computing, and information management. He has researched process mining for five years and is an author of more than five books and over 300 scientific articles, making him the world top 2% scientist in 2020 by Standford University.

SHINTAMI CHUSNUL HIDAYATI received a B.S. degree in informatics from Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia, and M.S. and Ph.D. degrees in computer science and information engineering from the National Taiwan University of Science and Technology, Taipei, Taiwan. She is a lecturer in the Department of Informatics, Institut Teknologi Sepuluh Nopember (ITS), Surabaya, Indonesia. Before joining ITS, she was a postdoctoral research associate at the Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan, from 2017 to 2019. Her current research interests include machine learning and data mining and their applications to multimedia analysis, information retrieval, and computer vision.

MUHAMMAD RIVAI (Member, IEEE) received a Doctorate degree in 2006. He is currently a lecturer in the Department of Electrical Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia. His research interests include sensors, chemical sensors, gas sensors, electronics, robotics, and artificial intelligence applications.