IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 10, 2021, accepted March 18, 2021, date of publication March 22, 2021, date of current version April 2, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3067949

Multistage Ensemble Learning Model With Weighted Voting and Genetic Algorithm Optimization Strategy for Detecting Chronic Obstructive Pulmonary Disease

JOY DHAR
Department of Information Technology, Hatgobindapur M. C. High School, Bardhaman 713407, India
e-mail: joy.dhar@hatgobindapurschool.co.in

ABSTRACT Chronic Obstructive Pulmonary Disease (COPD) is a life-threatening lung ailment and a significant cause of morbidity and fatality globally. The early detection of COPD can provide timely proper medication and reduce the mortality rate. To obtain proper treatment and lessen the death rate, this study proposes a novel ensemble model: the Multistage Ensemble model (MSEN) with an optimized weighted voting technique to detect COPD early and help clinicians provide proper and timely medication. In this study, there are two pools of classifiers created in which four classifiers are placed in each pool. These two pools of classifiers are employed to form two weighted ensemble models based on a weighted voting strategy. This study combines those generated ensemble models using a weighted voting technique to form an MSEN model. The genetic algorithm is utilized to optimize the hyperparameters of each classifier in each pool. The weights of two generated ensemble models and each classifier are optimized using the grid search technique. This study employs the K-Nearest Neighbors approach to fill in the missing values, isolation forest to remove the outliers, and the LightGBM with Recursive Feature Elimination for feature selection. An evaluation of the suggested MSEN model is conducted on a real-world Exasens dataset to validate the suggested model's effectiveness which exhibits that the proposed model obtains better performance to detect COPD and provides superior performance than other machine learning models and existing benchmark techniques.

INDEX TERMS COPD detection, ensemble learning algorithm, genetic algorithm, LightGBM algorithm, machine learning, recursive feature elimination, voting classifier.

I. INTRODUCTION

Chronic Obstructive Pulmonary Disease (COPD) is a chronic respiratory life-threatening lung ailment, inducing breathing problems in sufferers because of airflow restrictions in the lungs [9], [13]. This disease severely warns each person's health and has a vast death rate worldwide. It is a growing illness, increasing gradually over time, while its indications are usually worse than other diseases. COPD's primary cause is the long-term risk of subjects to either smoking or other lung irritants, such as industrial dust, chemical fumes, or air pollution [9]. However, in rare incidents, a hereditary disease known as alpha-1 antitrypsin deficiency may further provide lung damage and COPD [9]. Based on the report by the global initiative for COPD (GOLD), this ailment is usually correlated with airway or alveolar irregularities created by notable appearance to deadly gases or particles [13]. As one of the most widespread lung illnesses globally, COPD continues a perfidious course with a usually long-lasting undiagnosed primary stage, influencing many people and causes significant economic pressure on healthcare systems [9]. COPD has become the third foremost reason for mortality worldwide, according to the report provided by the world health organization, and in 2020, it will fit one of China's leading respiratory ailments [13]. COPD's main symptoms are abnormal sputum (mucus) generation, chronic coughs, shortness of breath, chest tightness, wheezing [9]. Although a perfect medicine to reverse induced lung damages has still

The associate editor coordinating the review of this manuscript and approving it for publication was Juan A. Lara.

to be discovered, early analysis has proved to have a crucial task in the powerful management of COPD [9].

Among available analysis and discovery techniques, the spirometry pulmonary function test is the most fundamental, well-organized primary care approach for COPD diagnosis [9]. Along with this test, patients' lung capacity is estimated while breathing in-out cycles [9]. The morbidity and mortality associated with COPD are widely under-diagnosed because of the spirometry test's limited sensitivity in the range of 64.5–79.9% [9], [13], [14]. Therefore, a helpful machine learning approach with healthy clinical reliability is a crucial need for diagnosing, treating, and self-management of COPD [9]. However, machine learning (ML) is a valuable technique that can foretell medical conditions and enables caregivers to make medical decisions precisely. Among numerous ML classification approaches: Support Vector Machines (SVMs), Logistic Regression (LR), eXtreme Gradient Boosting (XGB), Gradient Boosting (GB), and principal component analysis are among the most well-known approaches employed for classifying the health disease-relevant data [9]. However, various feature selection and extraction techniques are also employed to classify various medical data.

In this regard, ML-based methodologies can perform complex computational processes to determine the diseases from the massive volume of data. Such ML-based models have recently assisted by lessening the healthcare field's possible errors and making precise early identification of Parkinson's disease, heart disease, Alzheimer's disease, cervical cancer, liver cancer, breast cancer, and several other diseases [16]. Hence, such ML-enabled approaches can help physicians in their decision-making regarding various kinds of health diseases, including COPD and alleviate the workload of the physicians and make a precise and timely treatment.

Each ML researcher's main target is to provide a stable and precise predictive model with the most desirable performances. Medical data exploration is the most significant issue because of its close connection to a life of individuals. However, such afore-mentioned ML classification models have obtained lower performances to detect any disease, including COPD.

To address this issue and provide the lowest error rate regarding diagnosis and treatment, we propose a novel ensemble model: multistage ensemble learning (MSEN) model based on the weighted voting strategy to detect COPD. Recently, various past research works have utilized various ensemble learning-based methodologies to predict several health diseases and solve the problems that occur after utilizing the state-of-art ML approaches.

However, according to our understanding, this is the first research to introduce a multistage ensemble model based on the weighted voting strategy for detecting COPD. In this study, there are eight classification models: eXtreme Gradient Boosting Machine (XGB), Extra Trees (ET), Random Forest (RF), Gradient Boosting (GB), Logistic Regression (LR), Support Vector Machine for Classification (SVC) and its variant NuSVC, and K-Nearest Neighbors (KNN), are employed to generate the suggested MSEN model using weighted voting strategy [9], [19]. In this regard, there are two pools of classifiers created in which four classifiers are placed in each pool. XGB, ET, RF, and GB classification models are placed in the first pool, and the remaining classifiers: LR, SVC, NUSVC, and KNN, are placed in the second pool. Those four classifiers of each pool are employed to generate two ensemble models individually using the weighted voting technique. However, the generated two ensemble models are further utilized to combine and generate a new MSEN model using a weighted voting strategy.

The primary contribution of this study is conducted by four folds that are described below.

1. Introduce a novel ensemble model: MSEN, to achieve the most reliable result. There are eight classifiers used in this study to generate two ensemble models using the weighted voting strategy. Thus, these two ensemble models are further employed to combine and introduce a new model: MSEN. The genetic algorithm is utilized to tune each classification model's hyper-parameters in each pool to enhance performance. However, the grid search strategy is utilized for optimizing the weight of each classifier in each pool. Moreover, it also optimizes the weight of each generated ensemble model.
2. K-nearest neighbors approach is utilized to fill the missing values to gain more reliable data in terms of quality and isolation forest to detect and remove the outliers from the given preprocessed dataset for providing better prediction accuracy.
3. The LightGBM algorithm is utilized to provide the importance score of each feature and sort them based on their score, and the recursive feature elimination approach is employed to choose the essential features from them and provide help to the proposed model by enhancing its performance.
4. An evaluation of the suggested model is conducted employing a real-world publicly available Exasens dataset. The performance comparison between the proposed model and the various ML algorithms demonstrates that the proposed model achieves a more reliable performance for detecting COPD. A comparative study with the current benchmark methods shows that the proposed model gains the most reliable performance to detect COPD compared to the previous benchmark models.

Therefore, this study's remainder is constructed in the following. Section 2 exhibits some of the relevant previous works and research gaps, while Section 3 elaborates the suggested model with the data utilized in our methodology. On the other hand, Section 4 illustrates the proposed model's experimental results on the chosen dataset and discusses. Section 5 outlines the conclusion and explains the future works regarding this study.

II. RELATED WORKS AND RESEARCH GAPS

ML and deep learning-based methods are growing speedily in the health sector and may better detect and diagnose disease. In this regard, to precisely classify COPD patients' severity, several researchers have examined numerous ML and deep learning models to help clinical decision-making.

A. TRADITIONAL ML-BASED METHODS

Regarding previous research works relevant to traditional ML models, Spathis and Vlamos used the RF classification algorithm to foretell COPD [1]. They used 132 clinical records with 22 features corresponding to unique patients. The authors gained a precision value of 97.7% after employing an RF classifier to predict COPD patients. Gawlitza *et al.* [2] utilized quantitative computed tomography and five partial ML models for predicting COPD after utilizing 75 patient records in which the KNN model, polynomial regression, and Gradient boosting obtained the lowest mean relative error: 16%. Halder *et al.* [3] utilized various ML algorithms: KNN, SVM, LR, DT, and discriminant analysis (DA) to predict COPD after employing 30 COPD-affected sufferers and 25 healthy controls' records. Such clinical records were gathered from the All India Institute of Medical Sciences, Raipur, Chhattisgarh [20, 21]. The authors obtained 100% accuracy by the SVM and LR classification algorithms.

On the other hand, While Fang *et al.* [4] presented an integrated model: direct search simulated annealing [4] approach with SVM to diagnose COPD using a knowledge graph after utilizing the COPD dataset. In this COPD dataset, there are 1200 samples available, of which 750 samples are belonging to COPD sufferers, and the remaining 450 samples are belonging to non-COPD sufferers. They utilized an adaptive feature subset selection approach [4] to select the most optimal features from the given input dataset. They obtained the performance in terms of accuracy value: 95.1% to diagnose COPD. Zheng *et al.* [5] developed a combined approach: serum metabolic and least-squares SVM (LS-SVM) biomarkers for classifying COPD. They gathered data from the First Affiliated Hospital of Wenzhou Medical University [5] after recruiting 54 COPD patients and 74 patients, excluding COPD [5]. The authors declared that the linear and polynomial least-squares support vector machine models obtained the performances in terms of accuracy values: 80.77% and 84.62%, and the AUC values: 0.87 and 0.90 for the diagnosing the COPD. While Peng *et al.* developed a C5.0 decision tree [6] classification model to foretell acute exacerbation COPD hospitalized sufferers' prognosis with objective clinical symptoms [19]. The medical reports exhibited that 410 hospitalized [19] acute exacerbation COPD subjects are gathered from the respiratory unit database of TAHSYU. [6], [19], and 28 features are chosen. This data is randomly split into a training set and a testing set. The accuracy obtained by the C5.0 decision tree classification model is 80.3% [19].

While Wang *et al.* [7] utilized five machine learning algorithms (RF, SVC, LR, KNN, and naïve Bayes) to build acute exacerbation COPD classification models [7], they collected 303 electronic medical records (EMRs) data from the China-Japan Friendship Hospital between February 2011 and March 2017 [7]. One hundred thirty-five records relevant to COPD subjects and 168 records related to non-COPD subjects are available from the China-Japan Friendship Hospital. The authors declared that the SVM model obtained the sensitivity value: 0.80, specificity value: 0.83, and AUC value: 0.90. Moll *et al.* [8] developed an ML mortality prediction model [8] for predicting the severity of COPD. They collected data from the COPDGene and ECLIPSE studies [8] in which they selected 30 clinical, spirometry, and imaging features [8]. The authors implemented random survival forests and Cox regression for selecting features [8].

In contrast, Soltani Zarrin *et al.* [9] utilized various machine learning models to obtain high classification accuracy for detecting COPD. The authors employed two samples with different sizes (80 samples and 239 samples) of the Exasens COPD dataset. In this regard, 80 samples contain dielectric properties and demographic information. On the other hand, 239 samples comprise only demographic features without using dielectric properties. In this research work, the authors concluded that the XGB model obtained an accuracy value of 91.25% after using 80 samples of the Exasens dataset. On the other hand, the authors also declared that the XGB model also achieved an accuracy value of 92.05% after utilizing 239 samples of the same dataset.

B. DEEP LEARNING-BASED METHODS

In comparison to ML models, there are various deep learning-based models developed in previous studies. In this regard, Nunavath *et al.* [10] exhibited a feed-forward neural network for distinguishing the COPD sufferers, and a long short-term memory to early foretell COPD exacerbations and the following triage [6]. The authors employed data extracted from the "United4Health" based on EU-funded-project while an entire home monitoring period predicts COPD exacerbations [20]. Xu *et al.* [11] built a full-group artificial neural network model to predict COPD after using full group datasets consisted of 18471 proper clinical records. Their developed model obtained an accuracy value: 86.45% and an F1 score value: 82.93%. Tang *et al.* [12] built four-layer deep learning [6] approach, which uses a specifically configured recurrent neural network [6], to managing temporal change in COPD progression [6]. The complexity of their approach directed to a flawed interpretation [6].

While Wang *et al.* [13] developed a COPD-enabled transfer learning [13], termed a balanced probability distribution (BPD) algorithm [13]. It integrated instance and feature-enabled transfers [13] to enhance the correctness of their approach. They applied the COPD dataset given by the Clinical Medical Science Data Center [13], and more than 360 varieties of features were employed. One thousand two hundred samples were then selected from the COPD dataset [13] from the partner healthcare system's electronic medical records. Hence, it incorporated 750 subjects affected

TABLE 1. Description of all utilized features and target variable of exasens dataset with relevant information.

Clinical features and the target variable	Features types	Data type	Range	Missing values in %
ID		Object	-	0
Min Imaginary Part	Dielectric properties	Numeric	-337.35 to -225	66.52
Avg Imaginary Part			-328.281 to -225	
Min Real Part			-626.86 to -44	
Avg Real Part			-473.929 to -44	
Gender	Demographic	Nominal	0 (for female) to 1 (for male)	0
Age		Numeric	17 to 93	0
Smoking		Nominal	1 to 3, where 1 for Non-smoker, 2 for Ex-smoker, and 3 for Active-smoker	0
Diagnosis	Target variable	Nominal	0 (for healthy controls) to 1 (for COPD)	0

by COPD and 450 subjects without COPD. The authors obtained an accuracy of 92.1% to classify the COPD subject and non-COPD subject. Zarrin *et al.* [14] presented a memristive neuromorphic platform for the on-chip recognition of the saliva samples of COPD subjects and healthy persons after utilizing the UCI repository-based publicly available Exasen dataset [14]. The authors employed 80 samples of the Exasens COPD dataset to perform their developed model. The performance of their developed model obtained 89% of the accuracy value for the detection of COPD. Du *et al.* [15] built a deep convolutional neural network [15] to evaluate three-dimensional lung airway trees from computer vision [15], thereby generating models to recognize COPD while utilizing 280 participants CT scan records collected between 2016 and 2018 from the Central hospital affiliated with Shenyang medical college [15]. The authors also utilized the Bayesian optimization approach to optimize the hyperparameters of the generated deep convolutional neural network [15]. Their generated model achieved the performance in terms of accuracy values: 88.6% and 86.4% using grey and binary snapshots.

After investigating the previous research works, it has been exposed that most of the studies involved in developing either deep learning-based methods or utilized traditional machine learning models to detect COPD. Such researchers did not focus on developing a new ensemble learning model, achieving the most desirable prediction accuracy for detecting COPD. Hence, this study fills this research gap.

There are no previous research works available which already developed an ensemble learning approach for detecting COPD, according to our knowledge. Therefore, this paper introduces a multistage ensemble learning model through which to achieve the most reliable and desirable outcomes and outperforms previous benchmark strategies for detecting COPD.

III. MATERIALS AND METHODS

In this section, the materials and the method utilized in our suggested research are examined. Additionally, we also examined the Exasens dataset employed to perform in our proposed model.

A. EXASENS DATASET

In this paper, this study employs an open access Exasens dataset [9], which is available in the UCI ML repository for implementing our proposed model. The researchers utilize eight features in this dataset to precisely classify and recognize COPD patients' saliva samples and healthy people [9]. There are 239 samples collected as demographic information for detecting COPD in which dielectric characterizations were performed on 80 samples out of the available 239 samples [9] because of the biosensor's limited life-cycle [9]. However, in this study, for highlighting the vital function of demographic attributes for detecting COPD, analyses are conducted on 239 samples of this dataset with dielectric properties. In this study, two groups of saliva samples, such as 160 samples for healthy controls and 79 samples for COPD sufferers [9], are used for investigating the performance of our proposed ensemble model. This dataset is elaborated in detail in Table 1, in which 33.26% of whole data are missing in the Exasens dataset.

B. METHODOLOGY

In this section, various stages of the recommended approach are exhibited in Figure 1. Such stages are the feature engineering stage (fill in missing values, outlier detection and removal, data normalization, and LightGBM-RFE for feature selection), the multistage ensemble model based on the weighted voting technique, and the model evaluation stage. The central components of each stage are briefly elaborated on as follows.

1) FEATURE ENGINEERING STAGE

In this stage, several sub-stages are performed in this study to enhance the performance of the suggested ensemble model. Such sub-stages are given below.

a: FILL IN MISSING DATA

In this study, four attributes have missing values, which are exhibited in Table 1. In this regard, to fill the missing data in the Exasens dataset, this study applied the k-nearest neighbor approach [18]. It can retain the original input data distribution by [18] choosing a proper K (where K = 5, in this study) value. The nearest neighbor's method delivers it feasible to determine the missing values based on various closest

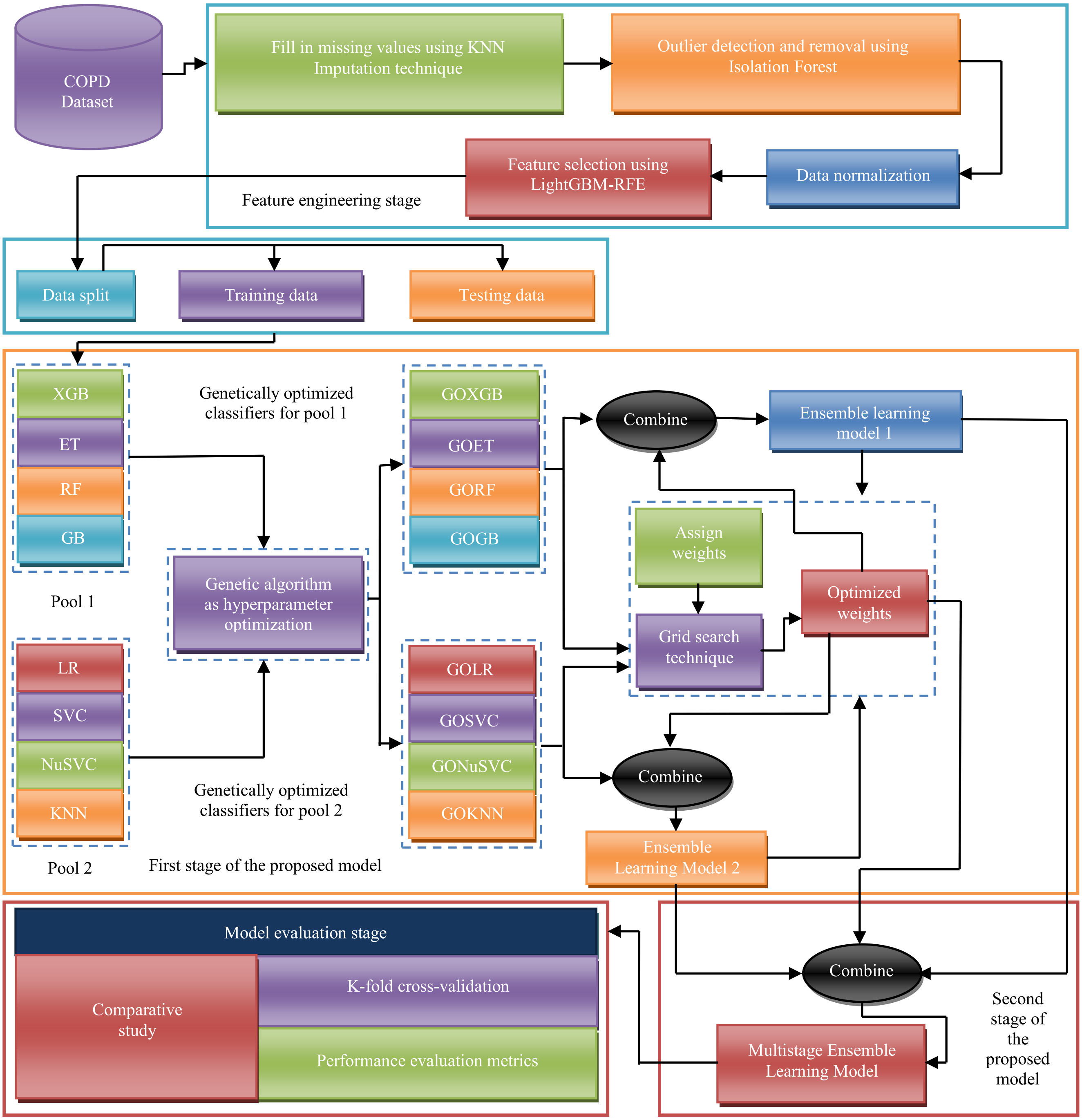


FIGURE 1. Block diagram of the proposed Multistage ensemble model for detecting COPD.

samples [18]. It is a more reliable, precise estimation than the traditional approach: mean, median, and mode, to fill in the missing values. Achieving more reliable quality data directs the development of a helpful classification model.

b: OUTLIER DETECTION AND REMOVAL

The existence of outliers is one of the common reasons to degrade the performance of any ML model. So, it is necessary to detect and remove the outliers from the given input dataset. However, in this study, Figure 2 exhibits the outliers that are available in the Exasens dataset. In this regard, this study employs an Isolation Forest (IF) to remove the outliers from the input dataset. According to Liu *et al.*, the IF is based on the principles in which outliers are the minority and have abnormal behavior on variables, compared to typical cases [27]. Hence, given a decision tree whose sole purpose is to identify a specific data point, fewer dataset splits should be required for isolating an outlier than for isolating a common

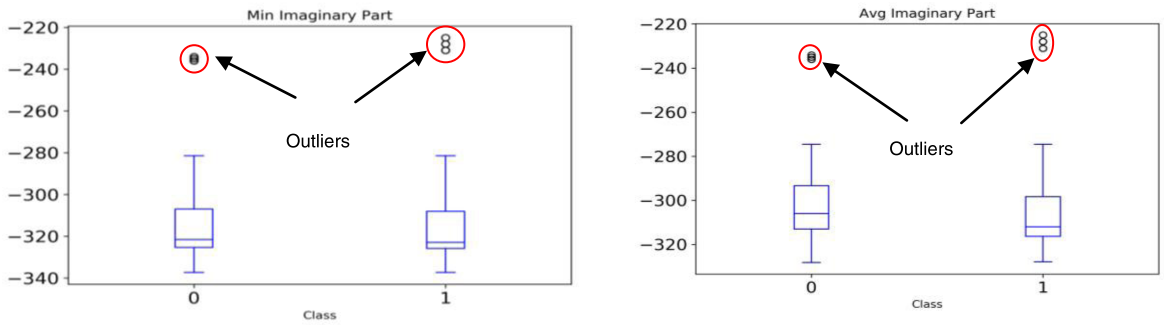


FIGURE 2. Outliers of the features corresponding to target classes.

data point [27]. The parameters used in IF are also exhibited in Table 2 to remove outliers from the given data.

TABLE 2. Hyperparameters and their values assigned to remove outliers using isolation forest.

List of Hyperparameters	Values
Maximum number of samples	Auto
Number of estimators	1000
Contamination	0.3

c: DATA NORMALIZATION

After completing the above-stated sub-stages, a data normalization approach is implemented. During data normalization, the data is reorganized to use the data for further analysis [18]. Data normalization's primary objective is to assort the data and eliminate redundant data, which might seem inside the dataset [18]. The standard normalization approaches are z-score, min-max, and many more [18]. The most reliable normalization strategy relies on the data to be normalized [18]. So, we utilized the min-max scalar approach for normalizing Exasens data, and data is rescaled to the range between 0 and 1 [18].

d: LIGHT GRADIENT BOOSTING MACHINE WITH RECURSIVE FEATURE ELIMINATION ALGORITHM FOR FEATURE SELECTION

The unrelated existence of features is one of the main reasons for the overfitting of a classifier. Thus feature selection should be performed before starting to train the classification model. The reason is that the feature selection process can enhance the classification methodology's performance, leading to faster and more cost-effective methods. Hence, this study utilizes a LightGBM [26] with the recursive feature elimination (LightGBM-RFE) algorithm, which comprises two approaches: LightGBM and recursive feature elimination (RFE) utilized to choose the essential attributes and assist in enhancing the performance of this proposed ensemble model. This study uses the Exasens dataset features, supplied as an input to the LightGBM-RFE algorithm (a wrapper-based feature selection approach), after performing the above-stated sub-stages on the given input dataset.

In implementing the LightGBM-RFE approach in this proposed research, it can select the essential feature subsets after removing unimportant features and diminishing the overfitting from the given input set of features (A), where $A = \{a_1$ $a_2, \ldots, a_m\}$, where m exhibits the number of input attributes relating to the input dataset D and select the essential features subset X; $X = \{x_1, x_2, \ldots, x_n\}$, where n represents the number of most essential features selected from A. Hence, $X \subset A$ then such vital features X passed to the proposed ensemble model for further processing.

Microsoft researchers introduced the LightGBM algorithm [26] that is primarily implemented for classification. However, the LightGBM algorithm can select features to rely on its feature importance value. This algorithm is implemented to sort given features rely on their feature importance score. In this regard, the LightGBM algorithm is utilized to obtain a notable score of each attribute and assign weights to those attributes. Later, the weighted sum of each feature's scores in all boost trees is employed to gain the ultimate significant value. Next, these features are sorted rely on their ultimate score. Hence, the importance of each feature is estimated using the following equation.

$$y_i^P = \sum\nolimits_{q=1}^{Q} f_q(a_i) \tag{1}$$

where $f_q(a_i) \in$ importance score of the ith feature vector on the qth tree, input dataset D, where $D = \{(a_{i,1}, y_i), (a_{i,2}, y_i), \ldots, (a_{i,m}, y_i)\}$, and element $(a_{i,m}, y_i)$ signifies that the mth feature vector's label is y_i.

The objective function of the LightGBM algorithm is expressed by the equation as follows:

$$\mathrm{Obj}(t) = \sum\nolimits_i l(y_i, y_i^P) + \sum\nolimits_q \Omega(f_q) + c \tag{2}$$

where $l(y_i, y_i^P)$ exhibits the loss among the actual value y_i and the foretold value y_i^P, which makes the difference between the LightGBM and the rest of the Gradient boosting decision tree in terms of computational performance speedup and method workability, $\sum_q \Omega(f_q)$ refers to the regular function, which indicates the complexity of the approach, and c represents an extra parameter that restricts overfitting and tunes the tree's depth.

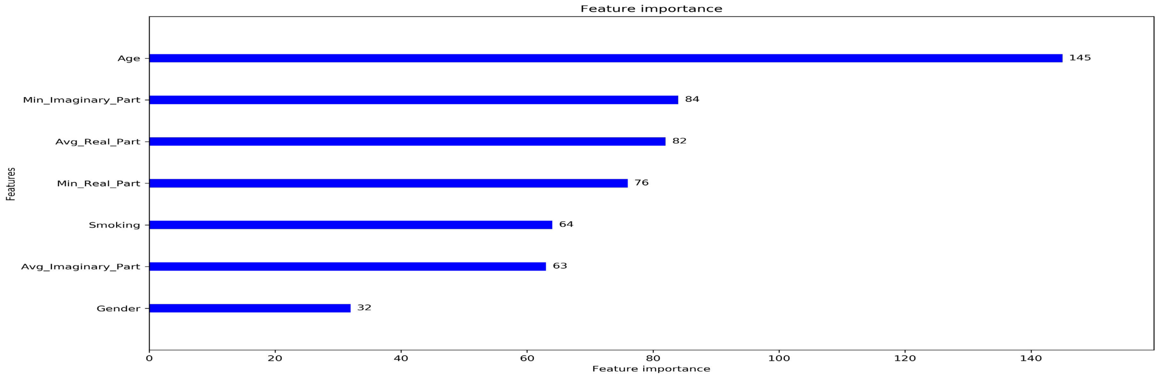


FIGURE 3. **Importance score of each feature of COPD dataset in descending order.**

After gaining each input feature's A importance score, then sorted such features based on their importance, and after then, recursive feature elimination can eliminate the least essential features from the given input feature set. These processes iterate N intervals until the needed number of features X is achieved.

In this study, each feature's importance score by the LightGBM algorithm is exhibited in sorted (descending) order in Figure 3.

2) PROPOSED MULTISTAGE ENSEMBLE LEARNING MODEL BASED ON WEIGHTED VOTING TECHNIQUE WITH THE GENETIC ALGORITHM OPTIMIZATION STRATEGY

In this section, this study introduces a novel multistage ensemble model (MSEN), which incorporates optimized weights through a grid search strategy for each classifier for detecting COPD. In the first stage for generating the proposed ensemble model, eight classifiers are employed to develop two ensemble learning models with the help of the weighted voting technique. The hyperparameters of each classifier are optimized by the genetic algorithm (GA). The grid search strategy optimizes the weights of each classifier. In the second stage for generating the MSEN model, those above-said generated ensemble models are combined to generate a new multistage ensemble model after utilizing a weighted voting strategy.

Furthermore, the grid search technique also tunes the weights of each above-stated generated ensemble model to enhance the prediction. The process of developing a new ensemble model, MSEN, based on weighted voting technique was divided into three main procedures: (1) base model generation, selection, and optimization, (2) two ensemble learning models generations, and (3) a new multistage ensemble learning model generation. Such procedures mentioned above and other relevant techniques are elaborated as follows. Figure 4 exhibits the flowchart of the proposed multistage ensemble model in analyzing Exasens data for classifying the disease.

a: BASE CLASSIFIERS

In this section, eight different classification models: XGB, ET, RF, GB, LR, SVC, NuSVC, and KNN, are used in this study to generate a new multistage ensemble model. In respect to this matter, firstly, there are two pools of classifiers created in this study to generate two different ensemble learning models with the help of a weighted voting technique in which four classifiers: XGB, ET, RF, and GB, are placed into the first pool and utilize as the first pool of classifiers to form a weighted voting-based ensemble learning model. On the other hand, the other four classification models: LR, SVC, NuSVC, and KNN, are placed into the second pool and employed as a second pool of classifiers to generate another ensemble learning model.

b: GENETICALLY OPTIMIZED BASE CLASSIFIERS

The hyperparameters are one of the primary causes to enhance the performance of a classification model. Thus, optimal hyperparameters are beneficial for improving the accuracy of a classification model. So, generating the optimal hyperparameters should be performed before utilizing the classification model. Hence, in this study, each base classifier's hyperparameters are optimized using a genetic algorithm (GA). Generally, GA is the most well-known metaheuristic methodology in which the chromosomes with the fittest continuation ability [25] and suitability to the environment are further possibly to persist and proceed on their potentials to coming generations [23], [25]. The following generations will further acquire their parents' features and involve more suitable and unsuitable chromosomes [23]. The more suitable chromosomes will be further able to survive and have fitted offspring [23], [25]. However, unsuitable chromosomes will progressively vanish [25]. The fittest suitable chromosomes will be recognized as the global optimum after performing several generations [23], [25].

GA is implemented as the hyperparameter optimization methodology in which each chromosome represents as a hyperparameter. In each evaluation, the decimal value of each chromosome is the hyperparameter's real input value. Each chromosome has several genes and each of which is a binary number [23]. The crossover and mutation operators are delivered on the genes of this individual [23], [25]. The population comprises every probable value within the initialized individual; on the other hand, the fitness function

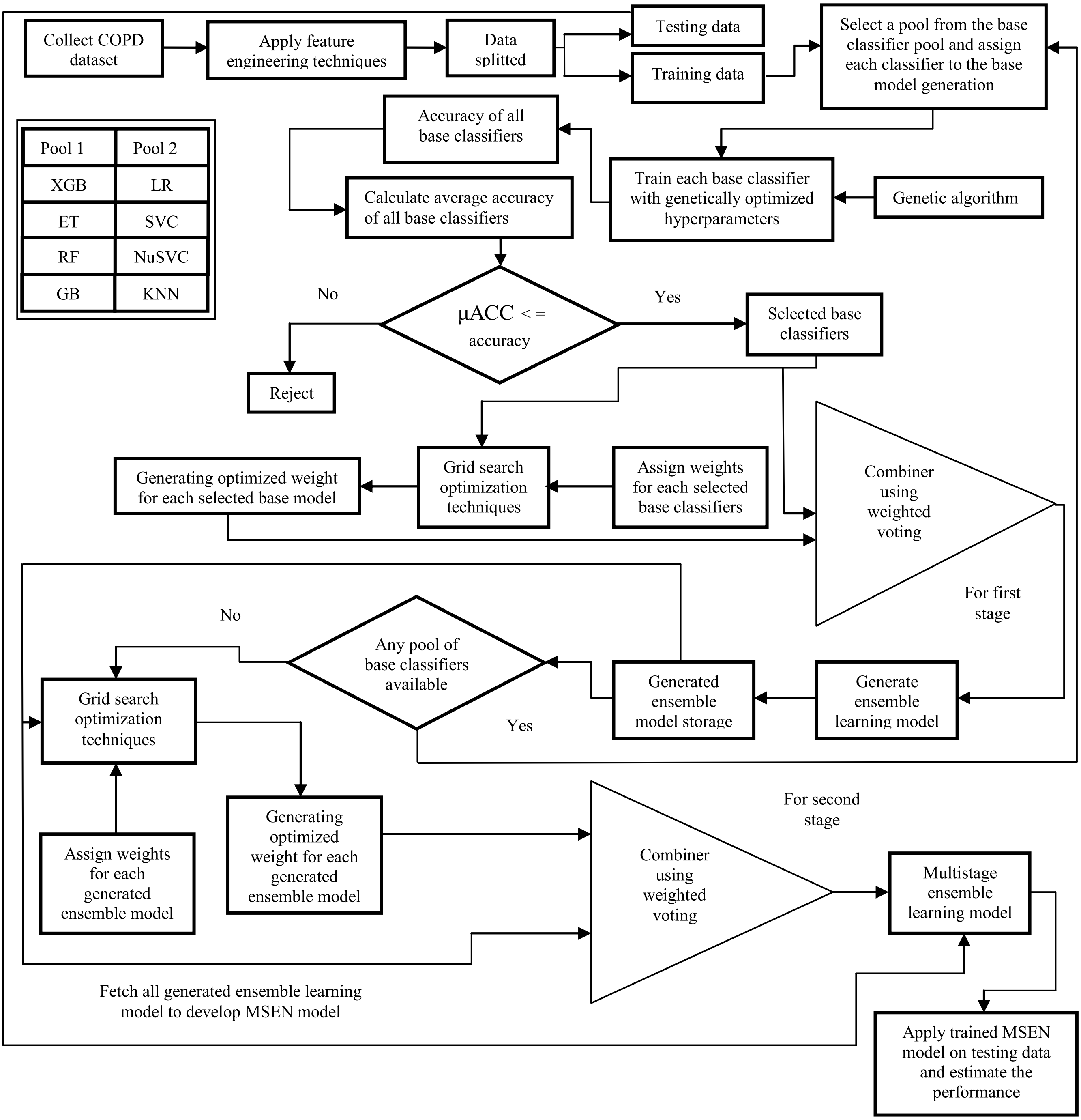

Collect COPD dataset
Apply feature engineering techniques
Data splitted
Testing data
Training data
Select a pool from the base classifier pool and assign each classifier to the base model generation
Pool 1
Pool 2
XGB
LR
ET
SVC
RF
NuSVC
GB
KNN
Accuracy of all base classifiers
Calculate average accuracy of all base classifiers
Train each base classifier with genetically optimized hyperparameters
Genetic algorithm
No
µACC <= accuracy
Yes
Selected base classifiers
Reject
Generating optimized weight for each selected base model
Grid search optimization techniques
Assign weights for each selected base classifiers
Combiner using weighted voting
For first stage
No
Any pool of base classifiers available
Yes
Generated ensemble model storage
Generate ensemble learning model
Grid search optimization techniques
Generating optimized weight for each generated ensemble model
Assign weights for each generated ensemble model
Combiner using weighted voting
For second stage
Multistage ensemble learning model
Fetch all generated ensemble learning model to develop MSEN model
Apply trained MSEN model on testing data and estimate the performance

FIGURE 4. The flowchart of the proposed multistage ensemble learning model.

(AUC assessment metric) indicates the hyperparameters' assessment metrics [23], [25]. While the hyper-parameter values are randomly initialized, such hyper-parameters frequently do not include the best hyperparameter values; various GA-based operators, namely selection, crossover, and mutation, must be conducted to recognize the best optimal hyperparameters based on the fittest suitable individuals.

The selection operator selects those individuals with the best-fitted function values. Individuals with best-fitted function values tend to move to the following generation with a higher possibility. They produce new individuals with the best aspects of parents to hold the population size fixed. Selecting the individual assures that each generation's significant aspect can be moved to the next generations [23]. The crossover operator is then utilized to produce new individuals by swapping the proportion of genes in various individuals [23]. After then, the mutation or variation operator is too employed for generating new individuals by randomly modifying one or more than one individual's genes [23]. The crossover and variation processes allow the next generations to have several properties and diminish the possibility of losing helpful information. The time complexity for generating the best individuals as the best optimal hyperparameters by the genetic algorithm is $O(n^2)$. Table 3 outlines the parameters that are utilized to perform the genetic algorithm as the hyperparameter optimization methodology. Algorithm 1 exhibits the procedure to perform GA for hyperparameter optimization purposes. Furthermore, Table 4 demonstrates the hyperparameters list used by each above-said classification model for optimization purposes using GA.

TABLE 3. List of parameters used to perform the GA.

SI No.	List of parameters	values
1	Params	List of hyperparameter values of each classification model
2	Population size	100
3	Number of generations	100
4	Crossover probability (gene_crossover_prob)	0.95
5	Mutation probability (gene_mutation_prob)	0.05

c: BASE MODEL GENERATION, SELECTION, AND OPTIMIZATION OF WEIGHTS

In this section, generating the base model is the primary procedure to build a new ensemble model. In this regard, the genetically optimized base classification models are utilized for training and willing to combine to form an ensemble classification model. However, base classifier selection is an essential component for developing a new ensemble classifier model because a satisfactory accuracy of any classification model is necessary to build any new ensemble learning model. If any classifier's accuracy becomes low or any classifier obtains unsatisfactory performance, then the performance of the newly generated ensemble model becomes degrading. In MSEN, each base classifier is selected based on the average of all classification models' accuracy values. However, the weight of each selected base classifier is optimized through a grid search strategy. In this study, there are two pools of classifiers (PoCs) created in Algorithm 6 that exhibits to fetch only one pool of classifiers at each ith iteration from the list of PoCs for generating the base model.

On the other hand, Algorithm 2 exhibits the procedure for generating the base model. Algorithm 2, which utilizes such four classifiers of the selected pool with the genetically optimized hyperparameters for those classifiers of the selected pool exhibited in Algorithm 1, are utilized for training and generating an ensemble learning model. While Algorithm 3 exhibits the procedure to select the base model and provide help for generating an ensemble learning model with higher accuracy. Algorithm 3 exhibits the results in the base model that is used in the process of combining the models.

In this section, the training dataset D_{Train} is assigned as (X_m, Y_m) with m training samples as input, where m represents the total number of instances in D_{Train}. On the other hand, the testing dataset D_{Test} is assigned as (X_n, Y_n) with n testing samples, where n exhibits the total number of instances in D_{Test} from the class set $Y = (y_1, y_2, \ldots, y_r)$, where r represents the number of classes available in D_{Train} and D_{Test}. In the stage of base model generation in Algorithm 2, it utilizes Algorithm 1 for optimizing the hyperparameters of each base classifier to enhance the performance of the suggested model and generate each base model with genetically optimized hyperparameters.

In respect of selecting the base classifier in Algorithm 3, the generated base model accepted the probability value set $P = (P_1, P_2, \ldots, P_n)$ and the predicted result set $R = (R_1, R_2, \ldots, R_n)$ that was derived from the probability of classifying the testing sample set (D_{Test}). Then estimate the average of all utilized base classifiers' accuracy value (μACC) after estimating the accuracy value of each base classifier. However, the accuracy (ACC) value is equal to the numbers of all accurate predictions split by the total number of test data instances. The average of all classifiers' accuracy value is further employed to select the base model with the following equation's help.

For each base classifier:

$$\mu ACC \leq ACC \tag{3}$$

In the above equation, if an μACC is equal to or less than the actual ACC of each base classifier, then such classifier model(s) are selected for generating an ensemble model; otherwise, rejected.

Algorithm 4 exhibits the procedure for generating weight for each selected base classifier model using the grid search optimization technique. In this regard, assign the weight of each selected base classifier model after utilizing c^c numbers of combinations, where c represents the number of the selected base classifiers, then implement grid search optimization strategy for tuning the weight of each selected base classifier and generate a most reliable optimized weight for each selected base classifier model.

For example, we assume that there are two Classifier models available and the weight of each classifier model is assigned for optimization purpose as [[0, 0], [0, 1], [1, 0], [1, 1]]. Hence, there are generating four combinations (that means 2^2) available for assigning the weight of two classifier models.

d: COMBINING STRATEGY FOR THE GENERATION OF A NEW ENSEMBLE MODEL

Algorithm 5 demonstrates the combination strategy of each selected base classifier model with the help of the weighted voting technique to generate an ensemble learning model. In this algorithm, the optimized weight of each selected base

TABLE 4. Hyperparameters utilized in each classifier for optimization using genetic algorithm.

Sl No.	Classifier Name	Hyperparameters with a brief description	Value
1	XGB	Boosting purpose (booster)	['dart', 'gbtree', 'gblinear']
		Learning rate to remove overfitting (learning_rate)	[0.1 - 0.99]
		Gamma value for minimum loss reduction (gamma)	[0 - 100]
		Minimum child weight (min_child_weight)	[0 – 100]
		Maximum number of the depth of the tree (max_dept)	[2 - 512]
		Maximum delta step (max_delta_step)	[1 – 100]
		Subsample ratio to remove overfitting (subsample)	[0.0 - 1.0]
		Sampling method to utilize for sampling the training examples (sampling_method)	['uniform', 'gradient_based']
		Columns' subsample ratio when developing each tree (colsample_bytree)	[0.0 - 1.0]
		Columns' subsample ratio for each level (colsample_bylevel)	[0.0 - 1.0]
		Columns' subsample ratio for each node (colsample_bynode)	[0.0 - 1.0]
		L1 regularization term (reg_alpha)	[0 – 10]
		L2 regularization term (Reg_ lambda)	[1 - 10]
		tree_method	['exact', 'approx', 'hist']
		scale_pos_weight for controlling the balance of positive and negative class (scale_pos_weight)	[0.0 - 1.0]
		Controls a way new nodes are added to the tree using Grow policy (grow_policy)	['depthwise', 'lossguide']
		Maximum number for leaves (max_leaves)	[0 - 2048]
		Maximum number of discrete beans (max_bin)	[2 – 1024]
		Type of sample (sample_type)	['uniform', 'weighted']
		Type of normalized approach (normalize_type)	['tree', 'forest']
		Number of estimators (n_estimators)	[1 - 1000]
2 and 3	ET and RF	bootstrap	[True]
		Maximum number of the depth of the tree (max_dept)	[2 – 512]
		The minimum weighted fraction to be at a leaf node (min_weight_fraction_leaf)	[0.0 – 0.5]
		The maximum number of features (max_features)	['auto', 'sqrt']
		The minimum number of samples required to be at a leaf node (min_samples_leaf)	[1 – 10]
		Minimum number of samples for each split (min_samples_split)	[2 – 10]
		Number of estimators (n_estimators)	[1 - 1000]
		Criterion	['gini', 'entropy']
		Minimum impurity decrease (min_impurity_decrease)	[0.0 – 10.0]
		The maximum leaf nodes (max_leaf_nodes)	[2 – 2048]
		Out-of-bag score (oob_score)	[True]
		Weights associated with classes (class_weight)	['balanced', 'balanced_subsample']
		Minimal Cost-Complexity Pruning (ccp_alpha)	[0.0 – 1.0]
		random_state	[1 – 100]
4	GB	Loss	['deviance', 'exponential']
		Learning rate to remove overfitting (learning_rate)	[0.1 – 0.99]
		subsample	[0.1 – 1.0]
		Maximum number of the depth of the tree (max_depth)	[2 – 512]
		The minimum weighted fraction to be at a leaf node (min_weight_fraction_leaf)	[0.0 - 0.5]
		The minimum number of samples needed to be at a leaf node (min_samples_leaf)	[1 – 10]
		minimum number of samples for each split (min_samples_split)	[2 – 10]
		Number of estimators (n_estimators)	[100 – 1000]
5	LR	Penalization (penalty)	['l1']
		The inverse of regularization strength (C)	[0.1 – 1.0]
		intercept_scaling	[1 – 10]
		Solver	['liblinear', 'saga']
		Maximum number of iteration (max_iter)	[1 – 1000]
		l1_ratio	[0 – 1]
6	SVC	probability	[True]
		gamma	['scale', 'auto']
		Kernel functions (kernel)	['linear', 'poly', 'rbf', 'sigmoid']
		decision_function_shape	['ovo', 'ovr']
		Maximum number of iteration (max_iter)	[1 – 700]
		Cost (C)	[0.1 – 10.0]
		random_state	[1 – 100]
7	NuSVC	probability	[True]
		Gamma	['linear', 'poly', 'rbf']
		Kernel functions (kernel)	['scale', 'auto']
		decision_function_shape	['ovo', 'ovr']
		Maximum number of iteration (max_iter)	[1 – 700]
		To control the number of support vectors (Nu)	[0.1 – 0.5]
		random_state	[1 – 100]
8	KNN	Weights	['uniform', 'distance']
		Number of neighbors (n_neighbors)	[2 – 140]
		Leaf size	[10 – 100]
		Algorithm	['ball_tree', 'kd_tree', 'brute']
		P	[1 – 5]
		Metric	['Minkowski']

Algorithm 1 Genetic Algorithm

```
Input:
Training dataset: $D_{Train} = (X_m, Y_m)$ with m training samples
and Testing dataset: $D_{Test} = (X_n, Y_n)$ with n testing samples;
hyperparameters of the employed classifier;
Number of generation: ng;
Crossover probability, mutation probability, and tournament size: = 3;
Procedure
1.  Generate initial population randomly, Pop, where Pop $\epsilon$ entire search space;
2.  Encode Pop as individuals, where genes $\epsilon$ hyperparameter values;
3.  Estimate an AUC of the utilized classifier as a fitness function for each individual: F(individual) = max AUC/100;
4.  Perform selection operation using 'Roulette wheel;'
5.  Perform crossover operation using 'Single point crossover;'
6.  Perform mutation operation using 'Flip bit mutation;'
7.  Generate new population $Pop_{new}$;
8.  Compute the AUC of each individual in $Pop_{new}$;
9.  if AUC(individual in $Pop_{new}$) > AUC(individual in Pop):
10.     /* Replace the individuals of Pop with the fitter individuals of $Pop_{new}$ based on AUC value */
11.     AUC(individuals in Pop) ← AUC(individuals in $Pop_{new}$)
12. else: return to step 4;
13. if ng to process:
14.     return to step 3;
15. else:
16.     return optimal hyperparameters;
```

classifier is utilized to generate a better ensemble learning model. The following equation is utilized to perform as the weighted voting enabled combining strategy to form a new ensemble learning model, which is shown in the following.

$$EN(X_m) = \sum\nolimits_{e=1}^{E} \omega_c^{\varphi} EN_e(X_m) \tag{4}$$

where ω_c^{φ} represents the optimized weight of each selected base classifier model, $EN(X_m)$ represents a newly generated ensemble model based on weighted voting technique, and $EN_e(X_m)$ represents each selected base classifier model.

e: GENERATION OF A NEW MULTISTAGE ENSEMBLE LEARNING MODEL

This section describes generating a new multistage ensemble learning model (MSEN) using a weighted voting strategy. The proposed MSEN model combines two generated ensemble learning models depends on the weighted voting technique. The weight of each generated ensemble model is optimized through a grid search optimization strategy. In this regard, Algorithm 6 demonstrates generating a new multistage ensemble model based on the weighted voting technique.

Algorithm 2 Base Model Generation

```
Input:
Training dataset: $D_{Train} = (X_m, Y_m)$ with m training samples
and Testing dataset: $D_{Test} = (X_n, Y_n)$ with n testing samples;
Procedure
for each classifier from T, where T = 1 to t: /* T = the number of classifiers */
    $\theta_T$ = Call genetic algorithm for hyperparameters optimization in Algorithm 1;
    Generate base models with optimized hyperparameters: $M_b(\theta_T) = \{m_1(\theta_1), m_2(\theta_2), \ldots, m_t(\theta_t)\}$;
Return $M_b(\theta_T)$;
```

Algorithm 3 Base Model Selection

```
Input:
Training dataset: $D_{Train} = (X_m, Y_m)$ with m training samples
and Testing dataset: $D_{Test} = (X_n, Y_n)$ with n testing samples;
Procedure
Call base model generation in Algorithm 2;
Generate the probability value $V_i$ of each $M_b(\theta_T)$ of the test sample:
    $V_i = \{v_1, v_2, \ldots, v_n\}$;
Generate the prediction $R_i$ of each $M_b(\theta_T)$ of the test sample:
    $R_i = \{r_1, r_2, \ldots, r_n\}$;
Estimate accuracy, ACC of each $M_b(\theta_T)$;
Estimate average of all classifiers' accuracy: $\mu ACC = \frac{1}{T}\sum_{t=1}^{T} ACC_t$, where $ACC_t \in$ the accuracy of each base classifier;
for each classifier t from T:
    if $\mu ACC \leq ACC_t$:
        Select the base classifier;
    else: rejected;
end for;
return selected base classifiers;
```

In the first stage of the proposed ensemble model, it creates two pools of classifiers (PoCs), which acts as an input in Algorithm 6 and utilizes to generate two ensemble learning models simultaneously using a weighted voting strategy. In Algorithm 6, only one pool of classifiers is fetched from the PoCs at each ith iteration and generates an ensemble learning model. This process continues until the PoCs available. In this algorithm, generate j number of ensemble learning models, which are further used to combine and formed an MSEN model after utilizing the algorithm of ensemble learning model generation. In this study, the value of j is assigned as 2.

In the second stage of the proposed model, it assigns the weight of each generated ensemble model with j^j combinations similar to assigning the weight of each selected base classifier model shown in Algorithm 4, where j represents the number of the generated ensemble model. Next, utilize the

Algorithm 4 Optimized Weight Generation

```
Input:
Training dataset: $D_{Train} = (X_m, Y_m)$ with m training samples and Testing dataset: $D_{Test} = (X_n, Y_n)$ with n testing samples;
Procedure
    Call base model selection in Algorithm 3;
    Generating optimized weights:
        (a) Assign a weight of each selected base classifier model from 1 to c: /∗ c represents the number of selected base classifiers ∗/
            $\omega = \{c^c\}$; such that $0 < c < $ number of selected classifier model, where $\omega \in$ the weight of each selected base classifier;
        (b) Apply the grid search optimization strategy for optimizing the weight $\omega$ of each selected base classifier model.
        (c) Generate the most suitable optimized weight for each base classifier:
    $\omega^{(\varphi)} = \{\omega_1^{\varphi}, \omega_2^{\varphi}, \ldots, \omega_c^{\varphi}\}$; where $\varphi \in$ applied grid search technique for optimizing each weight of each selected base model;
    Return $\omega^{(\varphi)}$;
```

Algorithm 5 A New Ensemble Model Generation

```
Input:
Training dataset: $D_{Train} = (X_m, Y_m)$ with m training samples and Testing dataset: $D_{Test} = (X_n, Y_n)$ with n testing samples;
Procedure
    Call base model generation in Algorithm 2;
    Call base model selection in Algorithm 3;
    Call weight generation through optimization in Algorithm 4;
    Utilize the weighted voting strategy to form an ensemble model:
        for each selected base classifiers from 1 to c:
            $EN(X_m) = \sum_{e=1}^{E} \omega_c^{\varphi} EN_e(X_m)$;
        end for;
    Output a new ensemble model
```

Algorithm 6 Multistage Ensemble Learning Model Generation

```
Input:
Training dataset: $D_{Train} = (X_m, Y_m)$ with m training samples and Testing dataset: $D_{Test} = (X_n, Y_n)$ with n testing samples;
Pool 1 = [‘XGB’, ‘ET’, ‘RF’, ‘GB’]; /∗ First pool ∗/
Pool 2 = [‘LR’, ‘SVC’, ’NuSVC’, ‘KNN’]; /∗ Second pool ∗/
PoCs = [Pool 1, Pool 2]; /∗ Two pools of classifiers ∗/
j = 2 /∗ number of PoCs ∗/
Procedure
  /∗ First stage for generating the proposed ensemble learning model ∗/
    for each iteration from 1 to j: /∗ for each ensemble model development up to j ∗/
        for each ith iteration: /∗ where i = 1 to 2 ∗/
            Select PoCs[i]; /∗ Select one pool of classifiers in each iteration ∗/
            $EN_j(X_m)$ = call ensemble learning model generation in Algorithm 5.
            i = i + 1;
            j = j + 1;
        end inner for;
    end outer for;
  /∗ Second stage for generating the proposed ensemble learning model ∗/
    Generating optimized weights for each ensemble model $EN_j(X_m)$:
        (a) Assign a weight of each $EN_j(X_m)$ from 1 to j: /∗ j represents the number of generated ensemble model ∗/
            $\omega^{EN} = \{j^j\}$; such that $0 < j < 2$, where $\omega^{EN} \in$ weight of each $EN_j(X_m)$;
        (b) Apply the grid search optimization strategy for optimizing the weight $\omega^{EN}$ of each ensemble $EN_j(X_m)$ model;
        (c) Generate the most suitable optimized weight:
            $\omega^{EN(\varphi)} = \{\omega_1^{EN(\varphi)}, \ldots, \omega_j^{EN(\varphi)}\}$; where $\varphi \in$ applied grid search technique for optimizing each weight of each ensemble model;
    Utilize the weighted voting strategy to form a multistage ensemble model:
    for each $EN_j(X_m)$ from 1 to j:
        $MSEN(X_m) = \sum_{e=1}^{E} \omega_j^{EN(\varphi)} EN_{j,e}(X_m)$,
        where $EN_{j,e}(X_m) \in$ each generated ensemble model;
    end for;
    Output: Generate a new multistage ensemble model $MSEN(X_m)$;
```

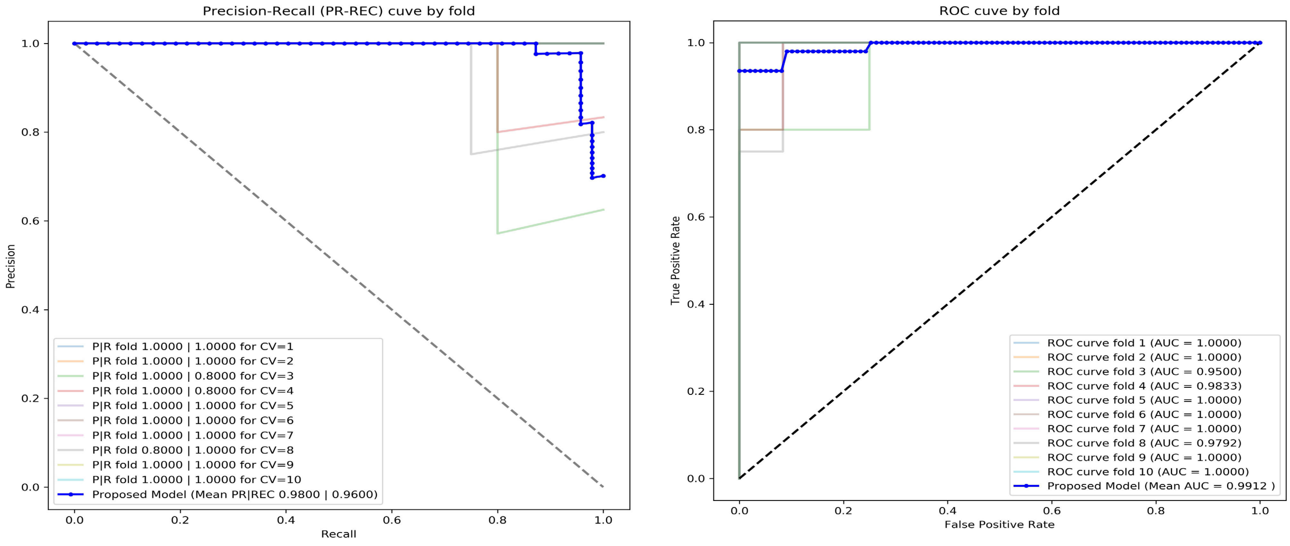
grid search optimization technique to optimize the weight of each above-said generated ensemble learning model and generate the most optimal weight for each generated ensemble learning model. Finally, a weighted voting enabled combination strategy is used in both generated ensemble learning models and combined to build a new multistage ensemble learning model for detecting COPD.

3) MODEL EVALUATION USING EVALUATION METRICS AND CROSS-VALIDATION TECHNIQUE

To assess the performance of the suggested ensemble model, we have utilized five performance assessment metrics: AUC, precision, recall, F1-measure, accuracy, in which four metrics are estimated using a confusion matrix.

1. **Accuracy:** It is computed by equation (5).

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$



PR-REC curve

ROC curve

FIGURE 5. PR-REC and ROC curves of the proposed MSEN model for the COPD dataset after utilizing a 10-fold CV technique.

2. **Precision (PR):** It is estimated by equation (6).

$$PR = \frac{TP}{TP + FP} \tag{6}$$

3. **Recall (REC):** It is determined by equation (7).

$$REC = \frac{TP}{TP + FN} \tag{7}$$

4. **F1-measure:** It is achieved from precision and recall metrics. The following equation exhibits it:

$$\text{F1-measure} = 2 \times \frac{PR \times REC}{PR + REC} \tag{8}$$

5. **AUC:** The area under the Receiver Operating Characteristic (ROC) curve is employed to estimate the AUC metric [24].

In this study, the stratified 10-fold cross-validation [15] (CV) technique is implemented to guarantee that dependent class frequencies are almost maintained in each training and validation fold [18]. In this technique, the test and train datasets are formed by randomly choosing the Exasens data individually for each class during keeping the proportions among classes [18]. In this regard, the CV is conducted on the whole available dataset.

IV. EXPERIMENTAL RESULTS AND DISCUSS

The resulting performance accomplished by the suggested ensemble model conducted on the Exasens dataset using a 10-fold CV technique is shown in Table 5. In this regard, the proposed MSEN model obtains the mean precision value: 0.9800, recall value: 0.9600, F1-measure value: 0.9667, AUC value: 0.9912, and accuracy value: 0.9820, respectively, exhibited in Table 6 for the detection of COPD, after utilizing a 10-fold CV technique. In this study, the ROC and precision-recall curves of the suggested model are exhibited in Figure 5. Such curves are used to demonstrate the proposed ensemble model's performance much better and also give reality and provide a perfect view regarding the proposed model's performance after using the 10-fold CV technique.

TABLE 5. The performance achieved by the suggested MSEN model using a 10-fold CV strategy.

Fold Number	Precision (PR)	Recall (REC)	AUC
1	1.0	1.0	1.0
2	1.0	1.0	1.0
3	1.0	0.8	0.9500
4	1.0	0.8	0.9833
5	1.0	1.0	1.0
6	1.0	1.0	1.0
7	1.0	1.0	1.0
8	0.8	1.0	0.9792
9	1.0	1.0	1.0
10	1.0	1.0	1.0
Mean	**0.9800**	**0.9600**	**0.9912**

TABLE 6. The performance achieved by the proposed MSEN model after using different performance evaluation metrics.

Accuracy	Precision (PR)	Recall (REC)	F1-measure	AUC
0.9820	0.9800	0.9600	0.9667	0.9912

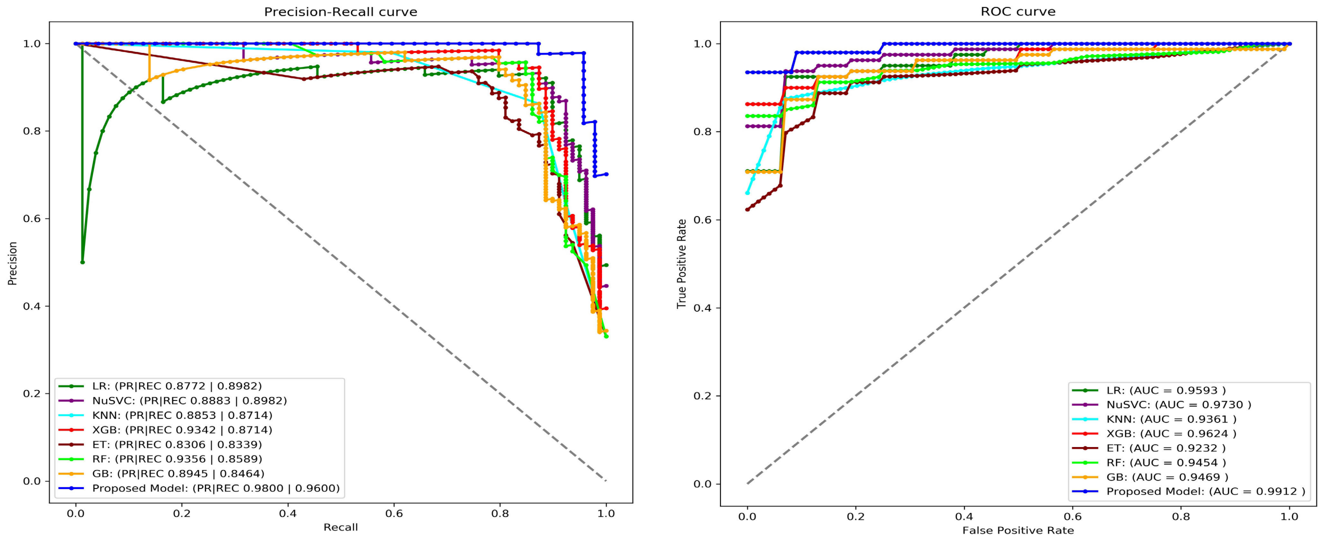
However, the proposed model achieves such above-declared results after selecting six features from the given input feature set of the Exasens dataset after utilizing the LightGBM-RFE wrapper-based feature selection algorithm [22]. In this study, the LightGBM algorithm emphasizes each feature and sorts in descending order based on their feature importance score. In this regard, the importance of each feature is exhibited in Figure 3. In the next step, the RFE approach is employed to

TABLE 7. Genetically optimized value of each hyperparameter for each classifier.

Sl No.	Classifier	Optimized value of each hyperparameter
1	XGB	'gbtree', 0.99, 3, 2, 242, 58, 1.0, 'uniform', 1.0, 0.0, 0.0, 5, 7, 'hist', 1.0, 'lossguide', 1155, 214, 'weighted', 'forest', 209
2	ET	True, 414, 0.0, 'auto', 1, 2, 407, 'gini', 1900, 0.0, True, 'balanced', 0.0, 100
3	RF	True, 329, 0.0, 'sqrt', 1, 2, 409, 'gini', 0.0, 1351, True, 'balanced', 0.0, 1
4	GB	'exponential', 0.99, 0.1, 375, 0.0, 1, 10, 630
5	LR	'l1', 1.0, 10, 'saga', 18, 1
6	SVC	True, 'scale', 'rbf', 'ovr', 30, 0.1, 58
7	NuSVC	True, 'rbf', 'auto', 'ovr', 31, 0.1, 39
8	KNN	'uniform', 18, 25, 'ball_tree', 2, 'minkowski'

TABLE 8. The performance comparison of the suggested MSEN model with various ML models.

Model	Accuracy	Precision (PR)	Recall (REC)	F1-measure	AUC
LR	0.9207	0.8772	0.8982	0.8840	0.9593
NuSVC	0.9248	0.8883	0.8982	0.8882	0.9730
KNN	0.9121	0.8853	0.8714	0.8698	0.9361
XGB	0.9330	0.9342	0.8714	0.8952	0.9624
ET	0.8830	0.8306	0.8339	0.8233	0.9232
RF	0.9288	0.9356	0.8589	0.8884	0.9454
GB	0.9121	0.8945	0.8464	0.8617	0.9469
Proposed MSEN model	**0.9820**	**0.9800**	**0.9600**	**0.9667**	**0.9912**



PR-REC curve

ROC curve

FIGURE 6. Performance comparison of the suggested MSEN model with various ML models.

choose the most significant features and eliminate unessential features from the input feature set.

Table 7 exhibits the optimized hyperparameters and the optimized values of each classification model. Table 7 comprises all the descriptions about optimized hyperparameters of all utilized classifiers involve in the suggested ensemble model.

A. PERFORMANCE COMPARISON WITH MACHINE LEARNING ALGORITHM

In this section, the performance comparison of the suggested ensemble model with various machine learning (ML) models: XGB, ET, RF, GB, LR, NuSVC, and KNN after utilizing the 10-fold CV approach are demonstrated in Table 8 and Figures 6 and 7. In this regard, the suggested model achieves the highest performance in terms of the precision value: 0.9800, recall value: 0.9600, F1-measure value: 0.9667, AUC value: 0.9912, and accuracy value: 0.9820, respectively, for detecting COPD. In this study, LR and NuSVC models obtain the second-best recall value: 0.8982, while the XGB model obtains the second-best accuracy value: 0.9330 and F1-measure value: 0.8952 after utilizing the Exasens dataset. In comparison, the RF model gains the second-best precision value of 0.9356; on the other hand, the NuSVC model also

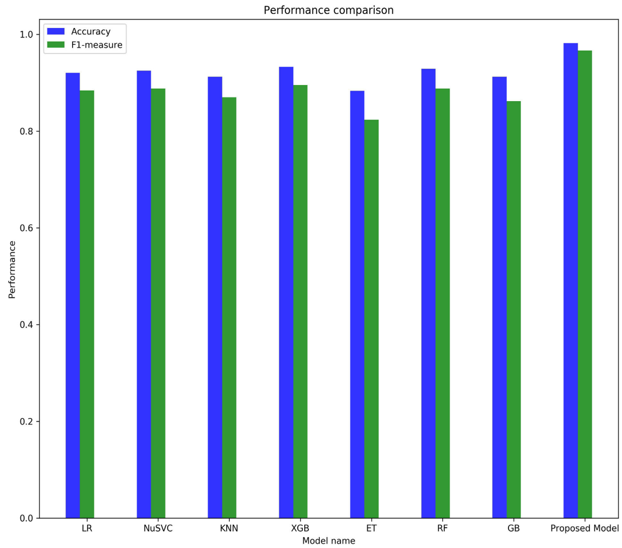


FIGURE 7. Performance comparison of the suggested MSEN model with various ML models.

obtains the second-best AUC value: 0.9730 after utilizing the Exasens dataset.

Hence, this study exhibits that the suggested model improves the performance than the LR, NuSVC, XGB, and RF models in terms of the precision values: 10.28%, 9.17%, 4.58%, and 4.44%, recall values: 6.18%, 6.18%, 8.86%, and 10.11%, F1-measure values: 8.27%, 7.85%, 7.15%, and 7.83%, AUC values: 3.19%, 1.82%, 2.88%, and 4.58%, and accuracy values: 6.13%, 5.72%, 4.9%, and 5.32%, respectively after utilizing Exasens dataset.

B. PERFORMANCE COMPARISON WITH GENETICALLY OPTIMIZED MACHINE LEARNING ALGORITHM

In this section, the performance comparison of the suggested ensemble model with various genetically optimized ML models: genetically optimized LR (GOLR), genetically optimized SVC (GOSVC), genetically optimized NuSVC (GONuSVC), genetically optimized KNN (GOKNN), genetically optimized XGB (GOXGB), genetically optimized ET (GOET), genetically optimized RF (GORF), and genetically optimized GB (GOGB), after utilizing the 10-fold CV approach, are demonstrated in Table 9 and Figures 8 and 9. In this regard, the suggested model achieves the highest performance in terms of the precision value: 0.9800, recall value: 0.9600, F1-measure value: 0.9667, AUC value: 0.9912, and accuracy value: 0.9820, respectively, for detecting COPD. In this study, GOSVC model obtains second-best accuracy value: 0.9542, recall value: 0.9250, and F1-measure value: 0.9313 after utilizing Exasens dataset. On the other hand, the GOXGB model obtains the second-best precision value: 0.9564 and AUC value: 0.9740 after utilizing the Exasens dataset. Hence, this study exhibits that the suggested model improves the performance compared to the GOSVC and GOXGB models in terms of the precision values: 3.61% and 2.36%, recall values: 3.5% and 7.61%, F1-measure values: 3.54% and 5.38%, AUC values: 1.99% and 1.72%, and accuracy values: 2.78% and 3.65%, respectively after utilizing Exasens dataset.

TABLE 9. The Performance comparison of the suggested MSEN model with various genetically optimized ML models.

Model	Accuracy	Precision (PR)	Recall (REC)	F1-measure	AUC
GOLR	0.9371	0.9328	0.8839	0.9012	0.9649
GOSVC	0.9542	0.9439	0.9250	0.9313	0.9713
GONuSVC	0.9292	0.8906	0.9125	0.8936	0.9730
GOKNN	0.9413	0.9514	0.8714	0.9057	0.9701
GOXGB	0.9455	0.9564	0.8839	0.9129	0.9740
GOET	0.9290	0.9107	0.8857	0.8940	0.9632
GORF	0.9413	0.9453	0.8839	0.9078	0.9718
GOGB	0.9500	0.9453	0.9125	0.9245	0.9719
Proposed MSEN model	**0.9820**	**0.9800**	**0.9600**	**0.9667**	**0.9912**

C. PERFORMANCE COMPARISON WITH GENETIC ALGORITHM BASED WRAPPER ENABLED FEATURE SELECTION APPROACH WITH MACHINE LEARNING ALGORITHM

In this section, the performance comparison of the suggested ensemble model with GA-enabled wrapper-based feature selection approach with various ML classifiers after utilizing the 10-fold CV approach is exhibited in Table 10 and Figure 10. In this regard, there are several ML classifiers utilized, such as GA with LR (GA-LR), GA with SVC (GA-SVC), GA with NuSVC (GA-NuSVC), GA with KNN (GA-KNN), GA with XGB (GA-XGB), GA with ET (GA-ET), GA with RF (GA-RF), and GA with GB (GA-GB), to compare the performance with the suggested MSEN model. The main reason is to select GA due to its flexibility while working as a wrapper approach. The suggested model achieves the highest performance in terms of the precision value: 0.9800, recall value: 0.9600, F1-measure value: 0.9667, AUC value: 0.9912, and accuracy value: 0.9820, respectively, for detecting COPD. In this study, GA-LR model obtains second-best accuracy value: 0.9413, F1-measure value: 0.9071, precision value: 0.9439, recall value: 0.8839, and AUC value: 0.9732 after utilizing Exasens dataset. On the other hand, the GA-XGB model obtains the lowest accuracy value: 0.9159, F1-measure value:0.8684, and the GA-KNN model obtains the lowest precision value: 0.9014 and AUC value: 0.9456, and the GA-GB model obtains the least recall value: 0.8446 after utilizing the Exasens dataset. Hence, this study exhibits that the suggested model improves the performance compared to the above-specified GA-enabled feature selection with various ML classifiers after utilizing the Exasens dataset.

D. PERFORMANCE COMPARISON WITH STATE-OF-ART BENCHMARK TECHNIQUE

In this study, Table 11 exhibits the performance comparison of our proposed ensemble model with the available benchmark techniques used to develop automated detecting COPD using the publicly available Exasens dataset. A significant

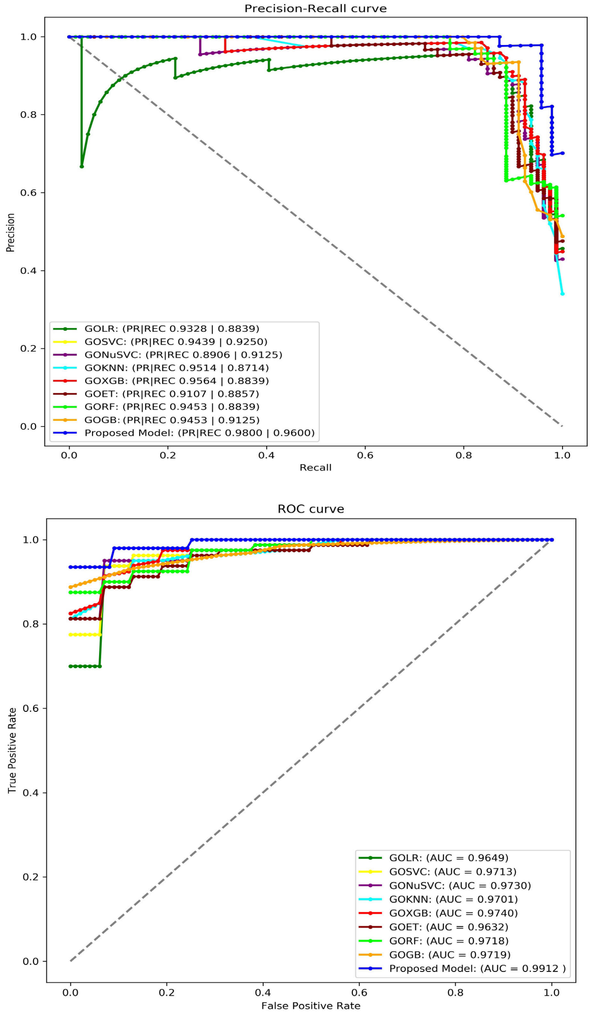


FIGURE 8. Performance comparison of the suggested MSEN model with various genetically optimized ML models.

TABLE 10. The Performance comparison of the suggested MSEN model with genetic algorithm-based feature selection with various ML models.

Model	Accuracy	Precision (PR)	Recall (REC)	F1-measure	AUC
GA-LR	0.9413	0.9439	0.8839	0.9071	0.9732
GA-SVC	0.9330	0.9231	0.8839	0.8960	0.9569
GA-NuSVC	0.9288	0.8902	0.8839	0.8902	0.9607
GA-KNN	0.9205	0.9014	0.8714	0.8769	0.9456
GA-XGB	0.9159	0.9048	0.8482	0.8684	0.9578
GA-ET	0.9161	0.9036	0.8464	0.8688	0.9375
GA-RF	0.9328	0.9389	0.8607	0.8918	0.9582
GA-GB	0.9199	0.9038	0.8446	0.8705	0.9491
Proposed MSEN model	**0.9820**	**0.9800**	**0.9600**	**0.9667**	**0.9912**

observation is that our proposed model achieved higher accuracy than the previously reported literature using the Exasens dataset.

Regarding the performance comparison, Table 11 exhibits that the suggested ensemble approach achieved the most desirable outcomes than the previous benchmark techniques. Our proposed multistage ensemble model has yielded the most reliable and desirable performance for the detection of COPD. The proposed MSEN model is the first work suggested to detect COPD. In [9], the authors obtained good performance with two different data preparation types (80 samples with dielectric properties and demographic

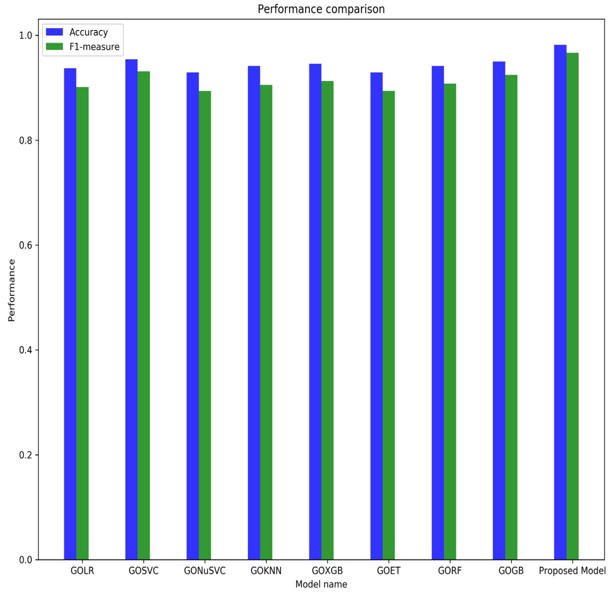


FIGURE 9. Performance comparison of the suggested MSEN model with various genetically optimized ML models.

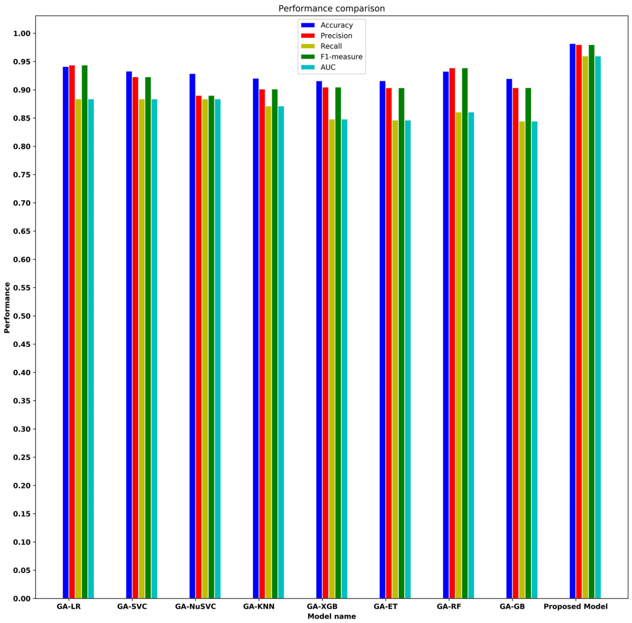


FIGURE 10. Performance comparison between the suggested model and GA based feature selection with various ML models.

TABLE 11. The performance comparison of the suggested ensemble model with the state-of-art benchmark techniques for detecting COPD using Exasens.

Study	Method utilized	Accuracy
Zarrin et al. [9]	XGBoost	91.25% (for 80 samples) and 92.05% (for 239 samples)
Zarrin et al. [14]	Memristive neuromorphic chip	89%

information and 239 samples with demographic information). However, the suggested ensemble model achieved the most desirable performance, with 239 samples containing dielectric properties and demographic information. In [14], the authors obtained lower performance compared to our proposed model. Hence, it may not be a desirable solution for real-world applications.

According to our knowledge, this is the first research to utilize the weighted voting-based multistage ensemble learning model, which combines two ensemble learning models using the weighted voting technique with genetic algorithm and grid search optimization based hyperparameter optimization techniques for intelligent detection of COPD.

The advantages of the suggested ensemble model are elaborated below:

1) This proposed model achieved the most reliable and desirable accuracy.
2) Employed different types of classification models to gain reliable outcomes.
3) The MSEN model utilized the genetic algorithm to optimize each classifier's hyperparameters to enhance the proposed model's performance.
4) The proposed model used LightGBM-RFE as a wrapper-based feature selection algorithm to select the most significant feature subset from the input feature set to obtain a better result.
5) The MSEN model utilized the K-nearest neighbors model to fill the missing values to achieve a better quality of data.
6) Employed Isolation Forest to remove the outliers from the input dataset to achieve a higher quality of data.

However, this proposed model also has limited disadvantages that are discussed below:

1) Requires a long time to train the model.
2) The proposed ensemble approach is complex.
3) We require more datasets for generating a more durable and precise approach. In this regard, we intend to validate our model with more datasets.

The proposed multistage ensemble model (MSEN) is exhibited in Figures 1 and 4, which comprises eight classification models in two pools (four classifiers in each pool) in which four classifiers of each pool utilized to generate two ensemble learning models with the help of the weighted voting technique. The proposed model combines those newly generated ensemble models to build a new ensemble model after utilizing a weighted voting strategy. However, the GA optimizes each classifier's hyperparameters exhibited in Tables 4 and 7 to enhance the suggested MSEN model's performance. The grid search strategy optimizes each classifier's weights and the newly generated ensemble model to generate a better prediction. However, the LightGBM algorithm exhibited the importance of each feature and sorted them in descending order, exhibited in Figure 3. Next, the RFE algorithm selects the most critical features and removes the irrelevant features from the given input feature set; hence, improve the performance of the suggested ensemble learning approach.

We can see the performance achieved by the proposed MSEN model using a 10-fold CV strategy in Figure 5. In Figures 6 and 7, we can see the proposed model's performance comparison with various ML models. In this regard, the proposed model achieved the best performances.

On the other hand, a performance comparison of the suggested model with various genetically optimized ML models exhibited in Figures 8 and 9. The suggested ensemble model achieved the best performance compared to all used genetically optimized models. In respect to performance comparison, the performance comparison between the suggested model and GA-based feature selection with various ML models exhibited in Figure 10. In this regard, the suggested model achieved the best reliable performance than the GA-based feature selection with the above-said various ML models. While Table 11 exhibits the performances determined by the suggested model and the previous benchmark techniques. The best performance is achieved by our proposed MSEN model achieved after comparing it with available benchmark strategies. Hence, it is validated that the proposed multistage ensemble model obtained the most reliable and the best classification performance.

V. CONCLUSION AND FUTURE WORK

This study introduces a novel multistage ensemble model based on the weighted voting technique for the early detection of COPD and helps clinicians to provide proper and timely medication and, therefore, to save lots of human lives. In this study, there are eight classifiers in two pools employed in which four classifiers of each pool are utilized to generate two ensemble learning models. These two generated ensemble models are further combined with the weighted voting technique's help to generate this proposed model. However, each classifier's hyperparameters and weight are optimized by the genetic algorithm and grid search optimization technique. Moreover, the grid search strategy also optimizes the weight of each generated ensemble model. The LightGBM-RFE algorithm selects the most significant features from the input feature set. The performed investigations strongly validate the effectiveness of the suggested multistage ensemble model using the Exasens dataset. The primary conclusions of this study are summarized below:

1) The suggested model is perfectly tailored to detect COPD and obtaining the most desirable performances in terms of precision value: 0.9800, recall value: 0.9600, F1-measure value: 0.9667, AUC value: 0.9912, accuracy value: 0.9820, respectively.
2) The suggested model outperforms the different machine learning algorithms and various genetically optimized ML models and achieves the most desirable performances.
3) The suggested model outperforms two recent state-of-art benchmark techniques and verifies as the best model for detecting COPD.

However, this proposed MSEN approach can still be improved by appending more new classification models (boosting) or remodeling the existing model. Perhaps combining various neural networks may also enhance the performance for classification. In this regard, the proposed model's future work is to add any of the above-said methods and enhance the diagnostic performance in detecting COPD and improve our proposed model's performance.

REFERENCES

[1] D. Spathis and P. Vlamos, ''Diagnosing asthma and chronic obstructive pulmonary disease with machine learning,'' *Health Informat. J.*, vol. 25, no. 3, pp. 811–827, Sep. 2019, doi: 10.1177/1460458217723169.

[2] J. Gawlitza, T. Sturm, K. Spohrer, T. Henzler, I. Akin, S. Schönberg, M. Borggrefe, H. Haubenreisser, and F. Trinkmann, ''Predicting pulmonary function testing from quantified computed tomography using machine learning algorithms in patients with COPD,'' *Diagnostics*, vol. 9, no. 1, p. 33, Mar. 2019, doi: 10.3390/diagnostics9010033.

[3] N. S. Haider, B. K. Singh, R. Periyasamy, and A. K. Behera, ''Respiratory sound based classification of chronic obstructive pulmonary disease: A risk stratification approach in machine learning paradigm,'' *J. Med. Syst.*, vol. 43, no. 8, p. 255, Aug. 2019, doi: 10.1007/s10916-019-1388-0.

[4] Y. Fang, H. Wang, L. Wang, R. Di, and Y. Song, ''Diagnosis of COPD based on a knowledge graph and integrated model,'' *IEEE Access*, vol. 7, pp. 46004–46013, 2019, doi: 10.1109/access.2019.2909069.

[5] H. Zheng, Y. Hu, L. Dong, Q. Shu, M. Zhu, Y. Li, C. Chen, H. Gao, and L. Yang, ''Predictive diagnosis of chronic obstructive pulmonary disease using serum metabolic biomarkers and least-squares support vector machine,'' *J. Clin. Lab. Anal.*, vol. 35, no. 2, Feb. 2021, Art. no. e23641, doi: 10.1002/jcla.23641.

[6] J. Peng, C. Chen, M. Zhou, X. Xie, Y. Zhou, and C.-H. Luo, ''A machine-learning approach to forecast aggravation risk in patients with acute exacerbation of chronic obstructive pulmonary disease with clinical indicators,'' *Sci. Rep.*, vol. 10, no. 1, p. 3118, Dec. 2020, doi: 10.1038/s41598-020-60042-1.

[7] C. Wang, X. Chen, L. Du, Q. Zhan, T. Yang, and Z. Fang, ''Comparison of machine learning algorithms for the identification of acute exacerbations in chronic obstructive pulmonary disease,'' *Comput. Methods Programs Biomed.*, vol. 188, May 2020, Art. no. 105267, doi: 10.1016/j.cmpb.2019.105267.

[8] M. Moll *et al.*, ''Machine learning and prediction of all-cause mortality in COPD,'' *Chest*, vol. 158, no. 3, pp. 952–964, Sep. 2020, doi: 10.1016/j.chest.2020.02.079.

[9] P. Soltani Zarrin, N. Roeckendorf, and C. Wenger, ''*In-vitro* classification of saliva samples of COPD patients and healthy controls using machine learning tools,'' *IEEE Access*, vol. 8, pp. 168053–168060, 2020, doi: 10.1109/ACCESS.2020.3023971.

[10] V. Nunavath, M. Goodwin, J. T. Fidje, and C. E. Moe, ''Deep neural networks for prediction of exacerbations of patients with chronic obstructive pulmonary disease,'' in *Engineering Applications of Neural Networks*. 2018, pp. 217–228, doi: 10.1007/978-3-319-98204-5_18.

[11] Q. Xu, W. Tang, F. Teng, W. Peng, Y. Zhang, W. Li, C. Wen, and J. Guo, ''Intelligent syndrome differentiation of traditional chinese medicine by ANN: A case study of chronic obstructive pulmonary disease,'' *IEEE Access*, vol. 7, pp. 76167–76175, 2019, doi: 10.1109/ACCESS.2019.2921318.

[12] C. Tang, J. M. Plasek, H. Zhang, M.-J. Kang, H. Sheng, Y. Xiong, D. W. Bates, and L. Zhou, ''A temporal visualization of chronic obstructive pulmonary disease progression using deep learning and unstructured clinical notes,'' *BMC Med. Informat. Decis. Making*, vol. 19, no. S8, pp. 1–9, Dec. 2019, doi: 10.1186/s12911-019-0984-8.

[13] Q. Wang, H. Wang, L. Wang, and F. Yu, ''Diagnosis of chronic obstructive pulmonary disease based on transfer learning,'' *IEEE Access*, vol. 8, pp. 47370–47383, 2020, doi: 10.1109/ACCESS.2020.2979218.

[14] P. S. Zarrin, F. Zahari, M. K. Mahadevaiah, E. Perez, H. Kohlstedt, and C. Wenger, ''Neuromorphic on-chip recognition of saliva samples of COPD and healthy controls using memristive devices,'' *Sci. Rep.*, vol. 10, no. 1, Dec. 2020, Art. no. 19742, doi: 10.1038/s41598-020-76823-7.

[15] R. Du, S. Qi, J. Feng, S. Xia, Y. Kang, W. Qian, and Y.-D. Yao, ''Identification of COPD from multi-view snapshots of 3D lung airway tree via deep CNN,'' *IEEE Access*, vol. 8, pp. 38907–38919, 2020, doi: 10.1109/access.2020.2974617.

[16] (May 1, 2011). *Juvenile Digital Degenerative: Topics by WorldWideScience.org*. WorldWideScience. [Online]. Available: https://worldwidescience.org/topicpages/j/juvenile+digital+degenerative.html

[17] M. Abdar and V. Makarenkov, ''CWV-BANN-SVM ensemble learning classifier for an accurate diagnosis of breast cancer,'' *Measurement*, vol. 146, pp. 557–570, Nov. 2019, doi: 10.1016/j.measurement.2019.05.022.

[18] W. Książek, M. Hammad, P. Pławiak, U. R. Acharya, and R. Tadeusiewicz, ''Development of novel ensemble model using stacking learning and evolutionary computation techniques for automated hepatocellular carcinoma detection,'' *Biocybern. Biomed. Eng.*, vol. 40, no. 4, pp. 1512–1524, Oct. 2020, doi: 10.1016/j.bbe.2020.08.007.

[19] (Nov. 17, 2020). *Predictors of Outpatients' No-Show: Big Data Analytics Using Apache Spark*. Research Square. [Online]. Available: https://www.researchsquare.com/article/rs-33216/v3

[20] (Jul. 27, 2018). *Deep Neural Networks for Prediction of Exacerbations of Patients With Chronic Obstructive Pulmonary Disease*. SpringerLink. [Online]. Available: https://link.springer.com/chapter/10.1007%2F978-3-319-98204-5_18

[21] N. Haider, B. Singh, R. Periyasamy, and A. Beher. (2019). *Respiratory Sound Based Classification of Chronic Obstructive Pulmonary Disease: A Risk Stratification Approach in Machine Learning Paradigm*. SpringerLink. [Online]. Available: https://link.springer.com/article/10.1007/s10916-019-1388-0

[22] G. Ansari, S. Gupta, and N. Singhal, *Natural Language Processing in Online Reviews* (Advances in Business Information Systems and Analytics). 2021, pp. 40–64, doi: 10.4018/978-1-7998-4240-8.ch003.

[23] L. Yang and A. Shami, ''On hyperparameter optimization of machine learning algorithms: Theory and practice,'' *Neurocomputing*, vol. 415, pp. 295–316, Nov. 2020, doi: 10.1016/j.neucom.2020.07.061.

[24] J. Dhar and A. K. Jodder, ''An effective recommendation system to forecast the best educational program using machine learning classification algorithms,'' *Ingénierie des Systèmes d Inf.*, vol. 25, no. 5, pp. 559–568, Nov. 2020, doi: 10.18280/isi.250502.

[25] (Jul. 30, 2020). *On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice*. DeepAI. [Online]. Available: https://deepai.org/publication/on-hyperparameter-optimization-of-machine-learning-algorithms-theory-and-practice

[26] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, ''LightGBM: A highly efficient gradient boosting decision tree,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 3146–3154.

[27] T. Cuijpers. (2021). *Teun Cuijpers, Author at ExtendedCognition*. ExtendedCognition. [Online]. Available: https://www.extended-cognition.com/author/teuncuijpers/

JOY DHAR was qualified in NTA UGC-NET for an Assistant Professor of Computer Science and Application Department awarded by the University Grants Commission and National Testing Agency. He is currently working as a Vocational Trainer with Hatgobindapur M. C. High School and an Independent Researcher on machine learning and deep learning-based methodology. His research interests include supervised and unsupervised machine learning, Bayesian optimization and its variants, deep learning, and genetic algorithm. He was also qualified at the Graduate Aptitude Test in Engineering (GATE) awarded by the Ministry of Human Resource Development (MHRD). He is currently a Reviewer of *Journal of Intelligent and Fuzzy Systems*.